MSc Project Report

16th September 2019

John H Booth (13133420)

BBK_BUCI058D7_1819 - MSc Data Science Project

MSc in Data Science

Department of Computer Science and Information Systems, Birkbeck College, University of London, 2019

Identifying Feature Importance in Pediatric Post-mortem Outcome with Machine Learning Models

Academic Declaration

I have read and understood the sections of plagiarism in the College Policy on assessment offences and confirm that the work is my own, with the work of others clearly acknowledged. I give my permission to submit my report to the plagiarism testing database that the College is using and test it using plagiarism detection software, search engines or meta-searching software.

Acknowledgements

I would like to thank Nigel Martin of Birkbeck College for his support and guidance during the process of producing this project and Professor Neil Sebire of Great Ormond Street Hospital (GOSH) and the Institute of Child Health for the clinical direction and insights. I am grateful to my colleagues in the GOSH Digital Research Environment for their continual encouragement. Finally I want to thank my wife, Christine, for her invaluable and inexhaustible support during the complete course.

Contents

Table of Figures

1 Abstract

Post-mortems are complex procedures that utilise a significant amount of hospital resources, yet despite this, cause of death is only determined in 45% of cases. The event itself can be very traumatic for the parents of the child, yet is essential for providing further clinical understanding of the patient's cause of death. Given this, there is an imperative to extract the greatest possible value from the data. Here, we investigated whether machine learning could be used to derive novel insights from the prediction of post-mortem outcomes.

A post-mortem database containing 7000 records across 300 variables was analysed and categorised into stage of examination (external and internal). The outcome of the examination was summarised as either 'cause of death determined' or 'not determined'. From these summarised data, cases were filtered by children aged <= 2 years, resulting in a dataset of 3,100 post-mortems.

Following this, decision tree, random forest, and gradient boosting machine models were iteratively built for each stage of the post-mortem examination and compared using their predictive accuracy metrics.

The naïve decision tree model using external examination data had a predictive performance of 68%. Model performance notably increased when trained on internal examination data. At each stage of the examination, a core set of data items, of which the final set included age, BMI, and heart weight were highlighted using model feature importance as key variables for determining post-mortem outcome. The use of increasingly complex modelling techniques was able to boost the predictive performance of the model by as much as 10%.

This project clearly shows the value of collecting clinical procedural data which can then be modelled using machine learning techniques to inform clinical practice. With more time, further modelling, including unsupervised clustering could be undertaken to derive further insights.

Supervisor:

Nigel Martin (nigel@dcs.bbk.ac.uk)

Department of Computer Science and Information Systems, Birkbeck College, University of London, London, UK.

2 Introduction

2.1 Background

Great Ormond Street Hospital for Children NHS Trust (GOSH) is the country's leading centre for treating sick children. With the UCL Great Ormond Street Institute of Child Health, GOSH is the largest centre for paediatric research outside the US.

Specialist Paediatric Pathologists perform perinatal, infant and childhood post-mortems including hospital referrals, forensic cases and those on behalf of Her Majesty's Coroner.

The Pathology Department has established a research database containing details of all post-mortems performed between 1996 and 2017. The database was originally used specifically for research into Sudden Unexpected Death in Infancy (SUDI). Since then it has been utilised for a number of other projects investigating SUDI, stillbirths and various aspects of paediatric autopsy procedure.

Currently the database holds 7000 records, each record representing an individual post-mortem. Up to 300 items of data can be defined for each post-mortem. These items of data are primarily used to record the first four stages of the post-mortem; the external examination followed by the internal examination split into the individual organs, grouped by bodily system, examined at both the macro and histological level.

The purpose of this project is to use data science analytic models, using the data recorded as features, to develop operational strategies that can be applied to paediatric post-mortems to prioritise which data is required to achieve the target of specifying the cause of death (COD).

2.2 Literary Review

A summary of the full literary review presented in the project proposal.

2.2.1 Paediatric Post-mortems

Paediatric post-mortems have their own specific issues as explained on the Royal College of Pathologists [*RCPath Contributors (2018)*]:

"Paediatric and perinatal pathology is concerned with identification of disease in the fetus, infant and child. It is age-specific rather than organ-specific and includes investigation of that organ unique to the fetus, the placenta. The spectrum of disease in this age range is very different from that seen in adults and the interaction of congenital malformation and growth of the child interact to produce unique pathology."

The Lullaby Trust, a charity that supports parents who have suffered the sudden loss of a child support research in this field gives a detailed breakdown of the different categories or presentations of post-mortems [*Lullaby Trust (2018)*]:

- TOP: Termination of pregnancy so the patient has not reach full term less than 24 weeks.
- Still birth: 24 weeks to full term.
- SUDI: Patients less than one year old.
- SUDC: Patients over 1 year.

2.2.2 Data Extraction

The Entity Attribute Value (EAV) model would be a good schema to use to extract the data for analytics. The advantages of using the EAV model for healthcare data are outlined by [*Löper, D., et al, 2013*] and the efficiency of storing data as described in [*Dinu, et al. (2007)]* .A clear example of the flexibility of the EAV model for health care data is given in [*Borodin, et al. (2015)].*

2.2.3 Data Wrangling

Although the majority of data held on a post-mortem is categorical a significant number of data is numeric data, lengths and weights. The importance of these values in determining cause of death is detailed by [*Horn,et.al., 2004*.].

Even within the main presentations of post-mortems described above these values can vary considerably. The approach of using growth charts in post-mortem analysis is described in [*Pryce, J.W., et.al. 2014].*

2.2.4 Analytics

The base analytic technique for this project will be Decision Trees with cross validation. Decision tree methodology is a commonly used data mining method for establishing classification systems based on multiple covariates or for developing prediction algorithms for a target variable [*Song, et al.2015*]. The key advantage of the Decision Tree technique is it simplifies complex relationships between input variables and target variables by dividing original input variables into significant subgroups, thus making the model easier to understand and interpret. [*Song, et al.2015*].

The main disadvantage of the technique is that using a single tree a model will suffer from low variance and high bias [*Analytics Vidhya Contributors (2016)*]. To combat this situation the project will consider ensemble methods which look to combine different techniques to better balance variance versus bias *[Abolfazl R, 2018*.].

The first technique to be considered will be Random Forest where the training data is split into a number of different sets and a tree is calculated for each set and the results combined *[Abolfazl R, 2018*.].

Gradient boosting is another technique that looks to decrease bias. Gradient boosting is a technique that looks to combine parameters that give a low prediction accuracy to produce a higher prediction accuracy [*Prashant G 2017*].

2.3 Report outline

The back bone of the project was developing a pipeline from the originating Post-mortem research database to the project results utilising different development strategies at various stages; this pipeline is laid out in detail in the methods chapter of the report.

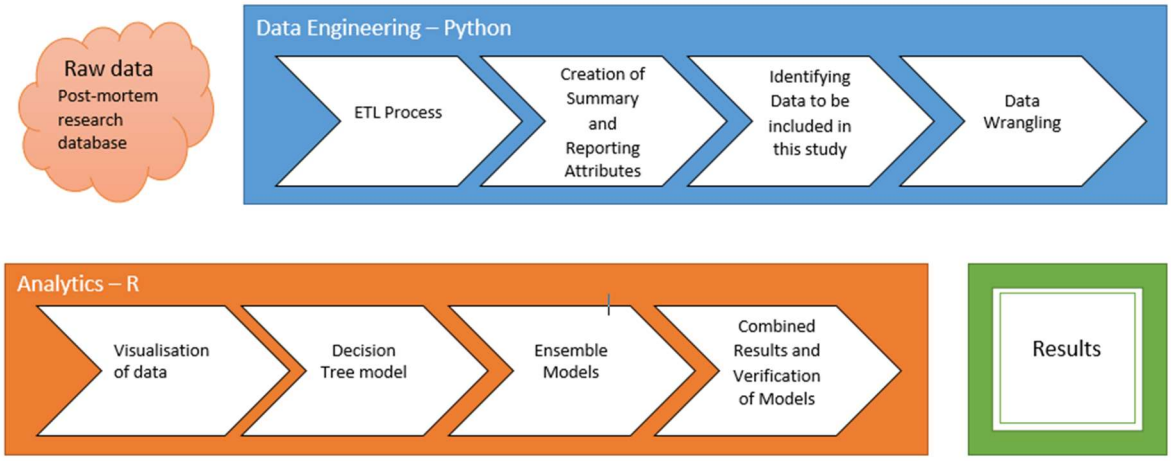


Figure 1 - Project Pipeline

Data Engineering is a major part of this project and the specific challenges faced in undertaking the project has its own chapter. The analytic processes of visualising the data followed by creation of three distinct models tuned to the individual stages of the post-mortem is described primarily using graphics produced during the analytics.

The project is completed by assessing the predictive accuracy of the various models and the relative feature importance at each stage of the post-mortem. The conclusion assess how the project delivered against the projects aims.

The final section evaluates the project process; what went well, what issues were more challenging and the lessons learnt. This section also discusses recommendations for how the project could be developed in the future.

A basic glossary of terms covering both data engineering and healthcare aspects of the project has been included to help the broad range of readers of this report. More detailed descriptions of code layout, ETL process and RDV structures are included in the appendices.

2.4 Aim and Objectives

The aims of this project are:

1. To develop a routine to extract data from the existing Post-mortem Research Database into an entity attribute value schema that will make the data more readily available for data analytics.

2. To apply the Decision Tree Analytical method to the extracted data to develop operational strategies that can be applied to paediatric post-mortems to prioritise which data is required to achieve the target of specifying the cause of death.

3. To investigate ensemble strategies, specifically Random Forests and Gradient Boosting to see how these techniques can improve on the basic Decision Tree method.

3 Methods

3.1 Project Pipeline

This overall project has been divided into a number of sections the output of each section provides the input for the following section to form the project pipeline. The process for each section will be coded in an appropriate environment described below but with the overall aim of creating a fully reproducible set of procedures that lead to a final set of results.

A GIT hub repository has been created for the project and all project code, documents and images are stored and versioned in this repository. A link to the GIT repository is given in Appendix E - Deliverables.

This chapter has been divided into two major sections; Data Engineering and Analytics.

3.2 Data Engineering

The data engineering aspects of this project are undertaken using the Python programing language [*Welcome to Python.org (2019)*], a general purpose programing language that used extensively in the world of data science. The development of python procedures has been carried out using PyCharm [PyCharm: the Python IDE for Professional Developers by JetBrains (2019)] an integrated development environment (IDE).

The data manipulation carried out during this stage uses the structured query language (SQL) and has been instigated using the Python package PyODBC [*mkleehammer (2019)*] which allows the connection to external databases using ODBC connections and the production and return of SQL queries. An example of a PyODBC function is given in Appendix A – Example Project Code.

Where appropriate the data engineering code was broken down into functions that were unit tested prior to implementation. All processes were developed with integral profiling so that processing bottlenecks were readily identified and their effect reduced so that the overall processing of the data was as efficient as possible.

3.2.1 Extract, Transform, Load (ETL) Process

The fundamental section of this stage is the ETL process on the Post-mortem Research database into the Health Analytics Schema (HAS) model using the Entity Attribute Value (EAV) schema.

The HAS is a specific implementation of the EAV schema used at GOSH to store diverse healthcare related data for research.

The basic structure of the process is:

- Create HAS Tables.
- Create Concepts.
- Create Patients and Staff.
- Create Events.
- Create Event Attributes.

A detailed breakdown of the python code developed for the ETL process is given in Appendix B – ETL Process.

The output of this section was the HAS database created and populated from the originating system.

3.2.2 Creation of Summary and Reporting Attributes

Having created the base research data from the original data then a number of summary event attributes were created for reporting and analytic purposes:

- Number of Attributes(ATTRIBUTES)
  - The number of event attributes each event has.
- Cause of Death Summary (COD2_SUMM)
  - A summary of the COD2 attributes into:
    - Not Determined.
    - Determined.
    - Unknown.
    - Not available.
  - More details in the following Data section of the report.
- Macro and Histological Body System Attributes
  - Individual organ internal macro and histological examination results are summarised at the body system level for ease of analytics.
- External and Internal Examinations
  - A simple flag to indicate whether an individual post-mortem event had had an external and or internal examination.

The output of this stage was the addition of a number of event attributes added to the existing set of events and their originating attributes.

3.2.3 Identifying Data to be included in this study

Due to the diverse nature of the data held; for example patients range from foetus with a gestational age of a few weeks to young adults in their teens also some post-mortems are performed on single organs for varying reasons. It was therefore decided to concentrate the project on a well-defined selection of the data.

This process was split into 2 stages:

- Include or exclude data
  - COD2_summ.
    - Only include events where the COD2_summ is either Not Determined or determined.
  - Age category.
    - Only include events for the following age categories:
      - Early Neonatal.
      - Neonatal.
      - Infant.
      - Child - under the age of 2 years.
  - Measurement Outliers.

        - Any numerical values that fall outside what is physically possible.
- Identify for the 4 stages of the post-mortem being considered in this study which features should be included:
    - External.
    - Internal Stage 1 (Organ weights).
    - Internal Stage 2 (Macro examination).
    - Internal Stage 3 (Histological examination).

At this stage the issue of missing data for any chosen event was not addressed.

The output of this stage were four research data views (RDVs), one for each stage of the post-mortem, in the form of CSV files. See Appendix D – RDV structures for more details.

3.2.4 Data Wrangling

The final section of the data engineering stage was to produce the data in the format most appropriate for analytics. Two forms of data wrangling were in the end used:

- One-hot encoding – Categorical features.
    - Rather than each categorical feature having a single column of data with the appropriate category; each category has its own column with either a 1 or 0 depending on whether each event has that feature value.
- Numerical normalisation – Numeric features.
    - Each numeric value will be normalised based on their predicted value for the age of the patient described by each event. This routine means that each numeric value will be in the range 0 – 1 with only outliers having larger values.

It should be noted that Z-Score standardisation of the numeric data was considered but not pursued as it didn't take into account the age of the patient in each event.

The output of this section were four adjusted RDVs, one for each stage of the post-mortem, in the form of CSV files.

The detailed structure of the RDVs in both formats is shown in Appendix D – RDV Structure.

3.3 Analytics

The analytic aspects of this project have been undertaken using the R programming language [*R: The R Project for Statistical Computing (2019)*] a language specifically developed for statistical computing. The development of R scripts was carried out in R Studio [*Open source and enterprise-ready professional software for data science – Rstudio (2019)*] an IDE for the R language.

In this section the key packages that were used and the specific parameters that had to be tuned to obtain an optimised model are described.

A basic tuning procedure was adopted for all three modelling packages:

- Create model using default parameters for each post-mortem stage.
- Define a range for each parameter to be tuned.
- Change each parameter one by one and obtain an optimal value based on predictive accuracy.

- Repeat last step to see whether any changes in the parameters significantly affects each parameter.
- Finalise a set of parameters for each post-mortem stage for each model.

The output for each of the modelling stages was an R function that can be called for that model with a training/test split for each post-mortem stage. The function saves the resulting confusion matrices and relative feature importance in CSV files as well as plots specific to each model as PNG files. An example of one of these model functions is given in Appendix A – Example code.

3.3.1 Visualisation: ggplot2

Ggplot2 [*Create Elegant Data Visualisations Using the Grammar of Graphics • ggplot2 (2019)*] is the principal graphics package used within R and is part of the tidyverse, a collection of packages aiming to bring some semblance of order in the slightly anarchic world of R programing.

This section has three main aims:

- Visualisation of the complete post-mortem data set.
- Visualisation of the sub set of data to be used for this study.
- Develop a basic graphical framework that can be used for all images produced by the various further analytic sections.
    - Colour scheme – viridis [*Garnier, et al; (2018)*] a colour blind friendly colour palette.
    - Theme.classic – a very basic no frills plotting theme.
    - PNG file naming convention for saving plots.

The output of this section were two frames of visualisations saved as PNG files.

3.3.2 Decision Tree: rpart

The rpart package uses recursive partitioning on trees, both classification and regression to achieve an optimum level of complexity for a given set of data [*Atkinson, E., et al. (2019)*], [*Therneau and Atkinson, (1997)*].

The main hyper-parameters that can be tuned are:

- minsplit - the minimum number of observations that must exist in a node in order for a split to be attempted.
- minbucket - the minimum number of observations in any terminal node. Use minsplit / 3.
- cp – complexity parameter, used to define further pruning after the initial tree is produced.

3.3.3 Random Forests: randomForest

Classification and regression based on a forest of trees using random inputs, based on [*Breiman & Cutler, (2018)*].

The main hyper-parameters that can be tuned are:

- Mtry - Number of candidates draw to feed the algorithm. By default, it is the square of the number of columns.
- Maxnodes - Set the maximum amount of terminal nodes in the forest.
- ntree - number of trees in the forest.

3.3.4 Gradient Boosted Decision Tree: XGBoost

Xtreme Gradient Boosting, which is an efficient implementation of the gradient boosting framework [*Chen and Guestrin, (2016)*], [*Chen, (2019)*].

The main hyper-parameters that can be tuned are:

- Eta – controls how much information from a new tree is used in boosting.
- max_depth – controls the maximum depth of a tree.
- gamma - Controls the minimum reduction in the loss function required to grow a new node in a tree.
- min_child_weight - Controls the minimum number of observations (instances) in a terminal node.
- Subsample - This parameter determines if we are estimating a Boosting or a Stochastic Boosting.
- colsample_bytree – Number of features to sample in each new tree.

3.3.5 Combined Results and Verification of Models

Using the functions developed for each model package a model was run for each model package for each post-mortem stage for five different random seeds each deriving a separate training/test data split. The CSV files from each model run were then combined to produce:

- A comparison of model predictive accuracy for changing random seeds.
- A comparison of the change in predictive accuracy of each model at each stage of the post-mortem.
- A comparison of relative feature importance changes for different random seeds for each stage of the post-mortem.
- A final predictive accuracy of cause of death determined or not for each model at each stage of the post-mortem. The predictive of accuracy of both not determined and determined cause of death can also be identified by model by stage of post-mortem.
- A final set of relative feature importance by model by stage of post-mortem.

4 Data Engineering

This section of the report will look in more detail on the data engineering undertaken to prepare the data for analytics.

4.1 ETL Process

The originating database was developed over a number of years and was optimised for data recording. The tables are divided up into subject groups and the overall structure is defined by primary and foreign keys. A lot of the fields contain no data so it is difficult to comprehend how much meaningful data there really is. The following partial schema only shows the major tables and excludes the 134 look-up tables for clarity.

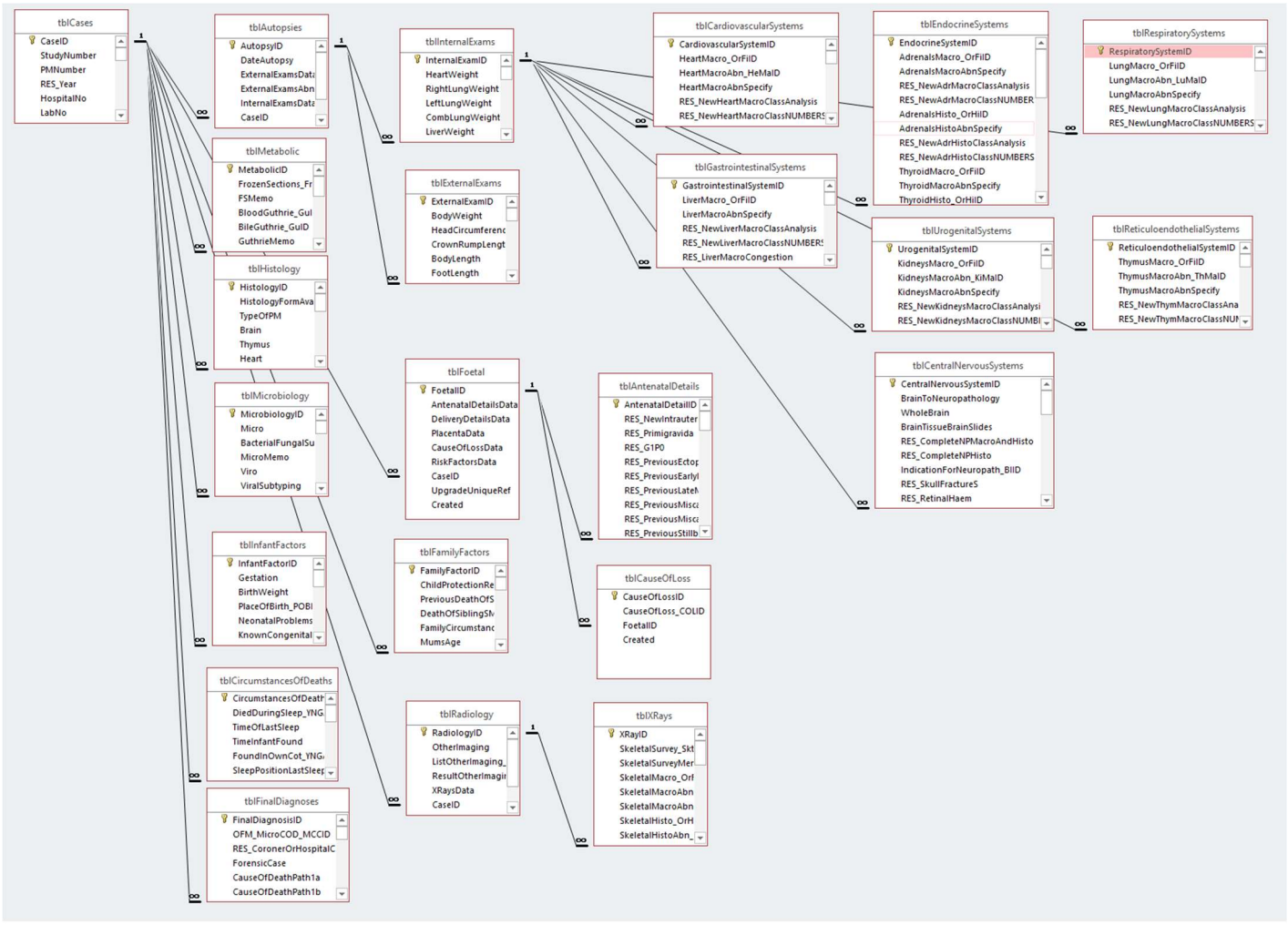


Figure 2 - PM Research Database - Partial Entity Relationship Diagram

The target database will use Healthcare Analytics Schema (HAS) uses the Event Attribute Value (EAV) model. This model has many fewer tables and the overall structure of the data is all encapsulated in the concepts table.

concept_id	category	ha_concepts_parent.code	ha_concepts.code	label	ha_concepts_value.code
22	/EventAttribute/Observation	PostMortem	tblCases	tblCases	Concept
23	/EventAttribute/Observation/PostMortem	tblCases	tblAutopsies	tblAutopsies	Concept
24	/EventAttribute/Observation/PostMortem	tblAutopsies	tblExternalExams	tblExternalExams	Concept
67	/EventAttribute/Observation/PostMortem	tblExternalExams	Nutrition_NutnID	Nutrition_NutnID	Concept
68	/EventAttribute/Observation	PostMortem	Nutrition_NutnID	Nutrition_NutnID	Concept
69	/EventAttribute/Observation/PostMortem/LookUp	Nutrition_NutnID	001	Well-nourished	Concept
778	/EventAttribute/Observation/PostMortem/LookUp	Nutrition_NutnID	002	Slim / Thin	Concept
1255	/EventAttribute/Observation/PostMortem/LookUp	Nutrition_NutnID	003	Wasted	Concept
1017	/EventAttribute/Observation/PostMortem/LookUp	Nutrition_NutnID	006	Plump	Concept
1005	/EventAttribute/Observation/PostMortem/LookUp	Nutrition_NutnID	007	Overweight	Concept
735	/EventAttribute/Observation/PostMortem/LookUp	Nutrition_NutnID	008	Obese	Concept
926	/EventAttribute/Observation/PostMortem/LookUp	Nutrition_NutnID	009	Poorly nourished NOS	Concept
399	/EventAttribute/Observation/PostMortem/LookUp	Nutrition_NutnID	999	N/A	Concept
	Defined in original database but not used.		004	Marasmus	
			005	Kwashiorkor	

Figure 3 - Example of the use of the Concepts Table within the EAV model

The above example demonstrates how both the nutrition field is stored within the overall structure of the data as well as all the possible values the field could take.

The ETL process extracts patient details into patient and patient attribute tables. Every post-mortem is represented as a single event and every field of data is represented as an event attribute with the event attribute type linked back to the concepts table. In the case of look-up values then the value is linked back to concepts table also. Every event attribute has a value; there are no NULL values in the EAV model. In the following entity relationship diagram of the HAS schema all tables are displayed and the concepts table is included twice for clarity.

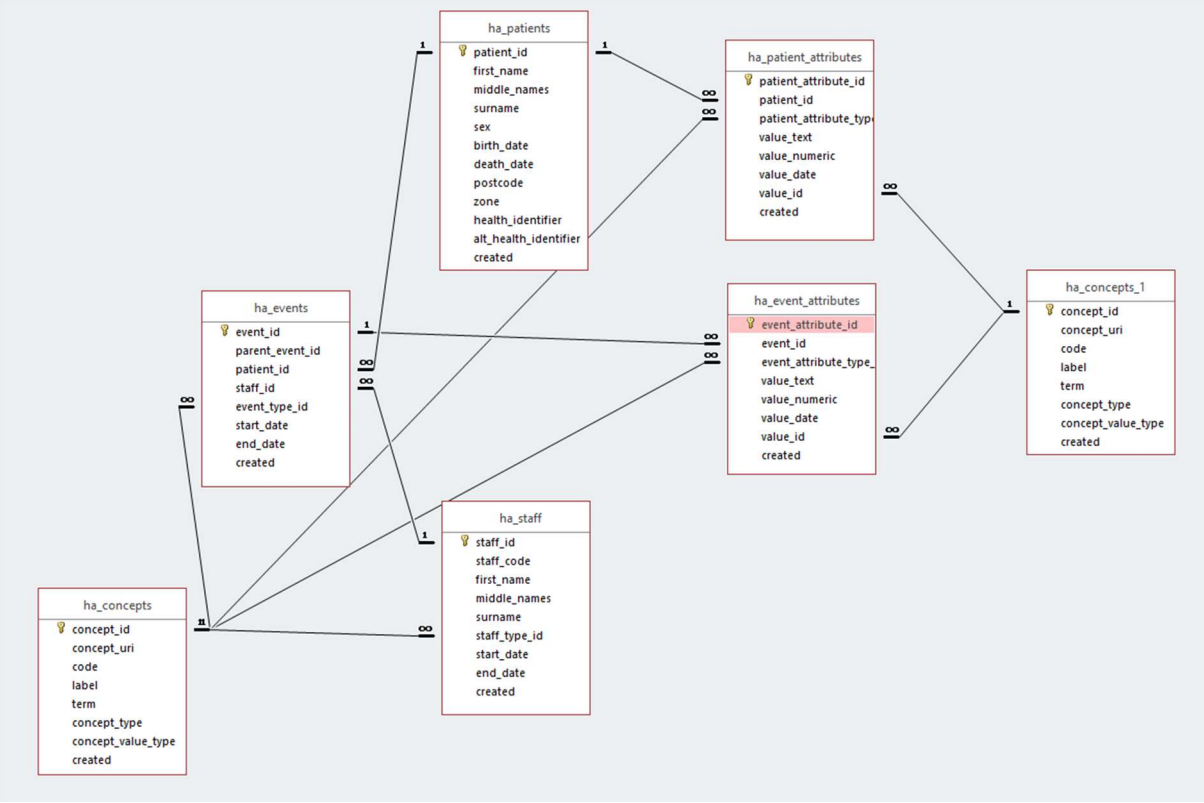


Figure 4 - HAS Schema Entity Relationship Diagram

Development process

- Create HAS_Tables.py.
- Created a separate analytic database to house the HAS schema with linked tables back to the PM Research database. This structure proved to be very inefficient and the bottleneck of the linked tables was identified.

- Changed procedure so that HAS tables still housed in separate database but the PM research database was connected to directly this provided a 10 fold improvement in processing time.
- The details of the EAV process are given in Appendix B.

4.2 Creation of Summary and Reporting Attributes

Having gone through the ETL process to establish the HAS the benefits can start to be realised.

In this section additional attributes were added to the database for use by the analytics process without having to make any structural changes and their relationship to other existing attributes is clearly documented within the concepts table.

Development process

- Modify_events.py
- Process is creating the COD2_SUMM attribute value type concept:
    - Create COD2_SUMM Concepts.
    - Extract all events with event attribute of type COD2_COD2ID.
    - For each event:
        - Apply mapping to COD2_SUMM. Details in Appendix C – Cause of Death Attribute Mapping.
        - Create new event attribute of type COD2_SUMM.
- Process for creating ATTRIBUTES:
    - Create ATTRIBUTES concept value type numeric.
    - Select all event attributes group by event count(attributes).
    - For each event
        - Create new event attribute of type ATTRIBUTES with a value of count(attributes)
- Similar processes were developed for:
    - Macro and histology organ category findings summarised for each body system:
        - Body_system_name + macro_SyFiID and body_system_name + histo_SyHiID
        - Individual organ internal macro and histological examination results are summarised at the body system level.
        - For each system the results for the individual organs were noted and the maximum result, 1-3, for any organ was assigned to the system.
            - 001 - Normal
            - 002 - Abnormal but not COD
            - 003 - Abnormal COD
            - 999 - Other
    - ExternalExam and InternalExam
        - A simple flag to indicate whether an individual post-mortem event had had an external and or internal examination.
            - True/False

4.3 Identifying Data to be included in this study

As with the previous section the EAV model makes it very simple to clearly define which data is to be used for a particular analytic study. Full details of columns in individuals RDV files are given in Appendix D – RDV Structures.

Development process

- Modify_events.py
    - Create INC_IN_STUDY concept of value type concept.
        - Create multiple exclusion types to be able to identify why an event was excluded.
    - Add attribute to every event and set to 'Include'.
        - Check if attribute exists id it does update to 'Include'. This feature allows the process to be run multiple times.
    - Define exclusion attributes.
    - For each event
        - Check exclusion attribute.
            - Update INC_IN_STUDY to appropriate exclusion type if required.
- Create_rdvs.py
    - Initially developed a generic routine that creates a CSV file based on:
        - List of patient attribute filters.
        - List of patient attribute columns.
        - List of event attribute filters.
        - List of event attribute columns.
    - For study RDVs
        - Include filter is event attribute INC_IN_STUDY.
        - Defined columns for each stage of the post-mortem stage.
            - Each new stage added additional columns to previous stage.
        - Produced four CSV files one for each stage.

4.4 Data Wrangling

Although not strictly necessary for the basic decision tree model the advanced ensemble based models needed the data to be modified by:

- Normalisation of numeric variables for Age.
    - Numeric values represent measurements of different aspects of the human body and vary considerably in magnitude [*Furlong, et al., (2016)*].
    - In principal this difference in magnitude could be removed by standardisation of the variables using Z-Score but this method would lose the intrinsic difference between values of different ages of development.
- Apply one-hot encoding.
    - Converts categorical variables to numeric and removes any bias introduced by having different values for each category as this method means that all values can be represented by 1 or 0.

1	2	3	5	7	8	9
event_id	event_start_date	sex	age_in_days	case_id	season	body_weight
1	01/12/2000 12:00	F	143	1	C004	6020
2	01/12/2000 12:00	F	101	2	C004	5900
4	01/12/2000 12:00	F	25	4	C004	5090
5	01/12/2000 12:00	M	225	5	C004	8500
6	01/12/2000 12:00	M	9	6	C004	4345
7	01/12/2000 12:00	M	201	7	C004	5875
9	01/12/2000 12:00	F	5	9	C004	5300
10	15/02/2000 00:00	M	248	10	C004	10500
11	01/12/2000 12:00	M	53	11	C004	5005
12	01/12/2000 12:00	F	77	12	C004	4405
13	01/12/2000 12:00	M	28	13	C004	3920

Figure 5 - Snapshot of the original RDV

1	2	3	4	5	7	9	10	11	12	13	14	15
event_id	event_start_date	sex_f	sex_m	sex_u	age_in_days	case_id	season_c001	season_c002	season_c003	season_c004	season_nan	body_weight
1	01/12/2000 12:00	1	0	0	143	1	0	0	0	1	0	0.0387
2	01/12/2000 12:00	1	0	0	101	2	0	0	0	1	0	0.0316
4	01/12/2000 12:00	1	0	0	25	4	0	0	0	1	0	0.5197
5	01/12/2000 12:00	0	1	0	225	5	0	0	0	1	0	0.0936
6	01/12/2000 12:00	0	1	0	9	6	0	0	0	1	0	0.3542
7	01/12/2000 12:00	0	1	0	201	7	0	0	0	1	0	0.213
9	01/12/2000 12:00	1	0	0	5	9	0	0	0	1	0	0.8071
10	15/02/2000 00:00	0	1	0	248	10	0	0	0	1	0	0.3017
11	01/12/2000 12:00	0	1	0	53	11	0	0	0	1	0	0.1516

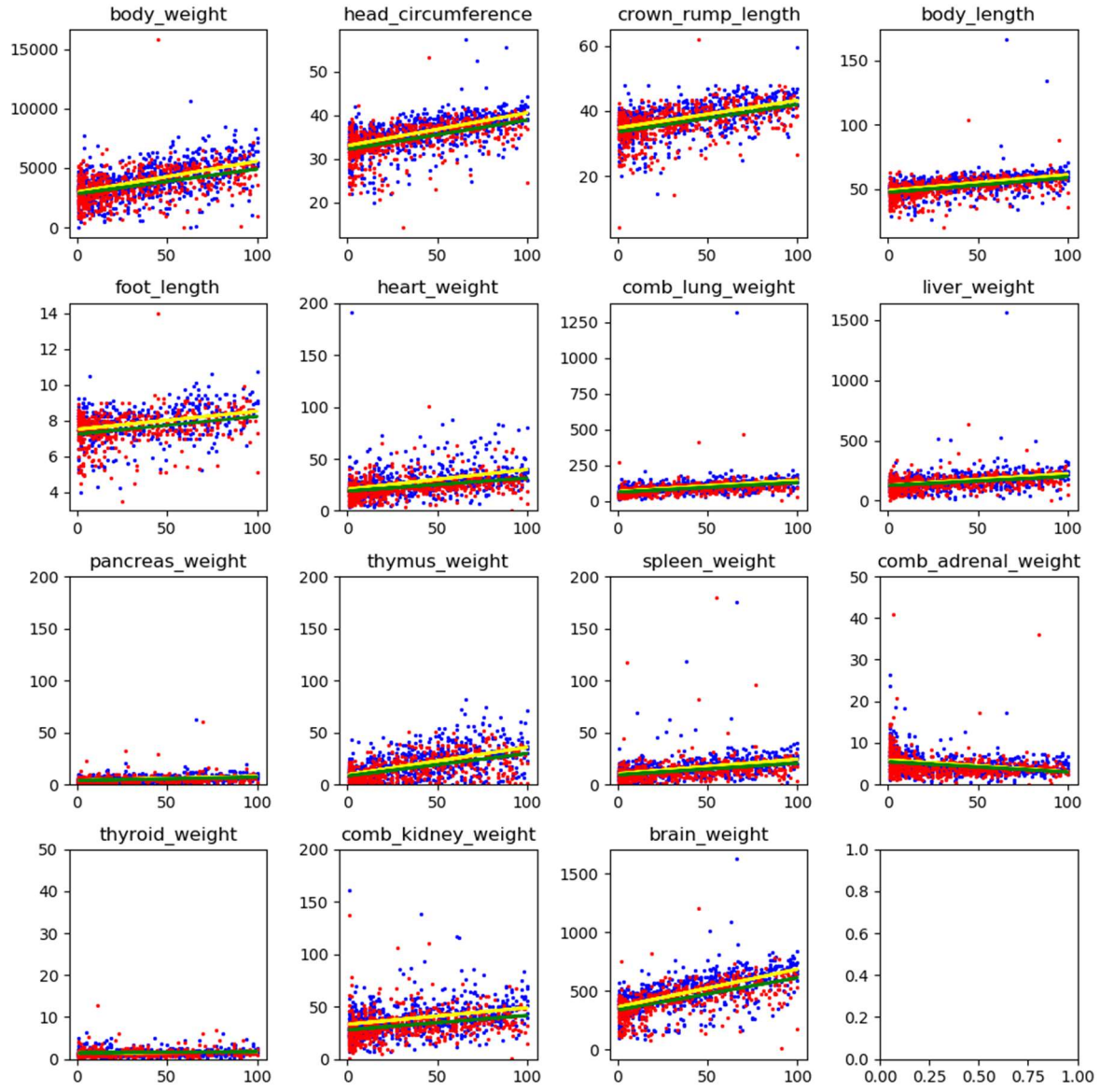
Figure 6 - Snapshot adjusted using normalisation and one-hot encoding

It was decided for comparison of the models that all models should be developed against the same data structure.

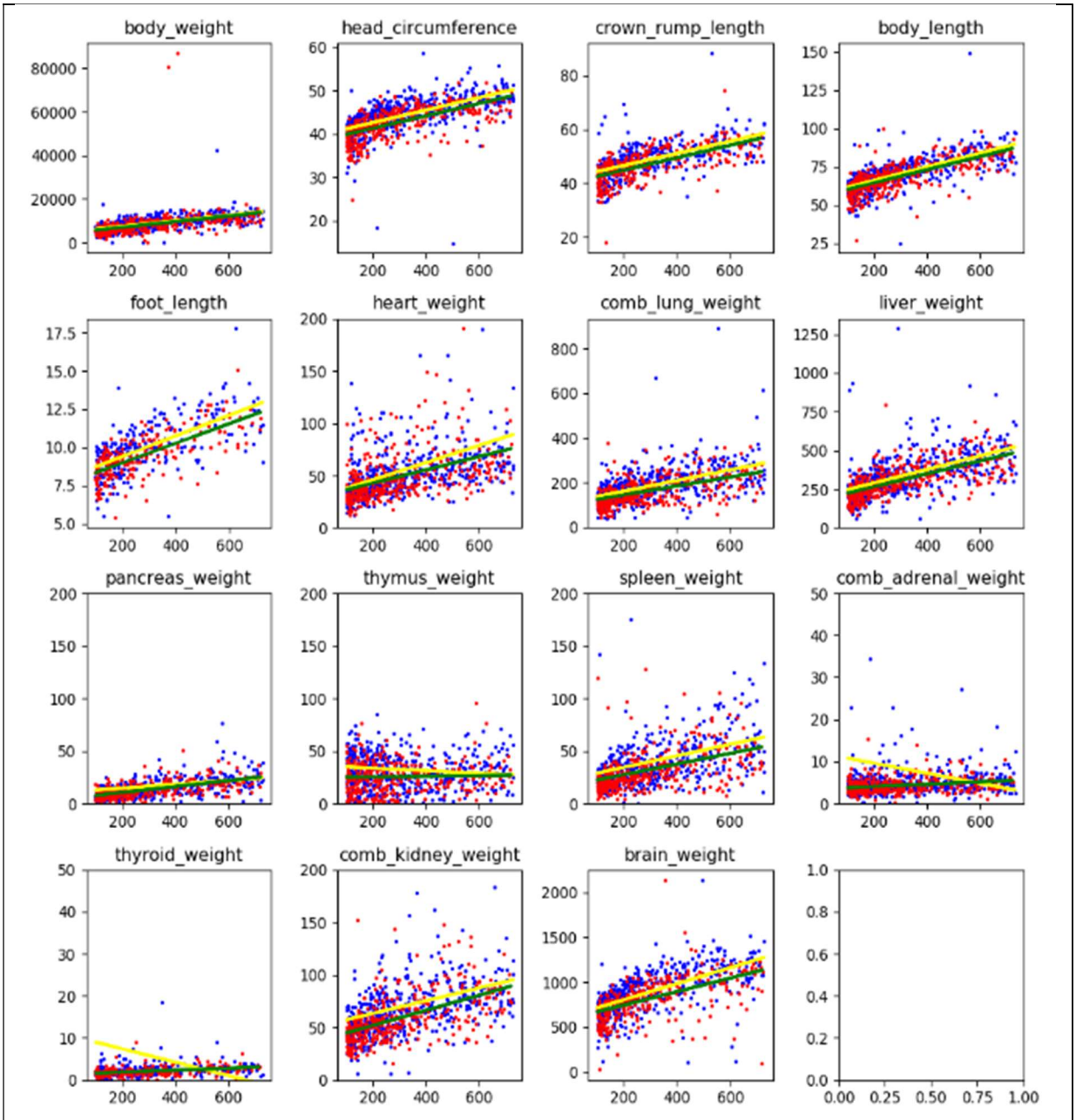
Development process:

- Modify_csv_data.py
- Develop a linear regression model for each measurement feature split by age and sex.
    - Used sklearn.linear_model from python Scikit-Learn package
    - Used matplotlib.pyplot to plot results

Linear Regression plots for ages <= 100 days



Linear Regression plots for ages > 100 days



Blue dots and yellow lines = Male, red dots and green lines = Female

Figure 7 - Linear Regression Plots for Numeric Features

- Read in original CSV file.
- For each row:
    - For each measurement column
        - Add a revised measurement column based on age and sex of patient.
    - For each categorical column.
        - Add a set of columns for all the possible variants of that category used within the original file.
- Write out adjusted CSV file with the appropriately transformed values.

    - For categorical values insert a 1 where the column matches the original category and 0 in all other columns.
    - For measurement values apply the appropriate linear regression parameters based on the sex and age of the patient and store:
        - absolute (actual – predicted)/predicted.
- The final output is an adjusted CSV file for each post-mortem stage.

5 Analysis

The visualisation stage is used to make final checks on the data as presented in the RDV's from the previous stage. Particular attention was given to missing and imbalanced data.

Once the final modifications are determined the focus changes to the first model; the Decision Tree. Due to the transparency of this model the output is reviewed in more detail for each stage of the post-mortem looking at both tabular data and visualisations in the form of the tree produced, confusion matrix for predicted accuracy and relative feature importance.

For the ensemble models of Random Forest and Boosted Gradient Tree the output is limited to the confusion matrix and relative feature importance.

5.1 Visualisation

5.1.1 Complete Data Set

The first step in the analytic process is to get to know the data. Initial visualisations were done on the complete data set but focussed on the features that were used to define the data to be included in the study.

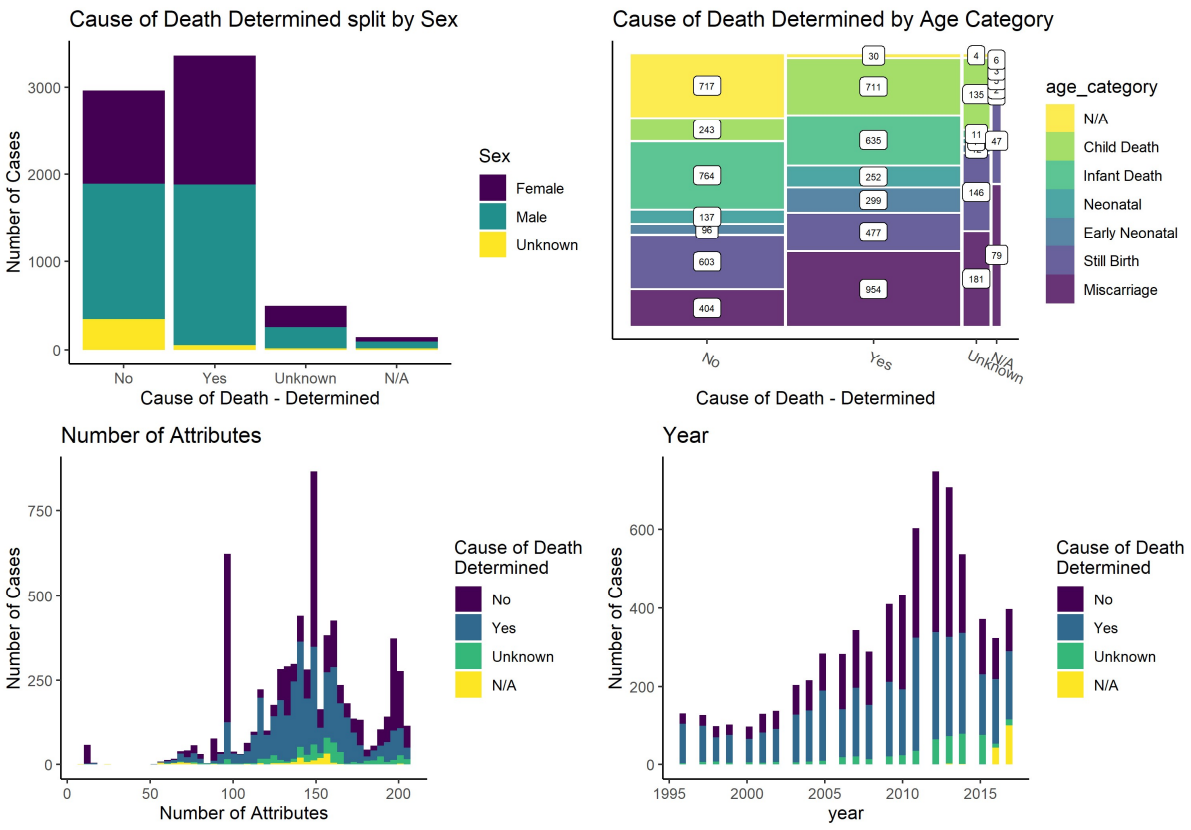


Figure 8 - Visualisations of Complete Data Set

The plots above show that the data is well distributed across cause of death determined and not determined, by age group and by sex. The events have generally a significant number of attributes without any one classification having significantly more than the other and that they are well distributed over time.

5.1.2 Study Data Set

The next set of visualisations were done to focus on the split of data to either be included or excluded from the study and for the reasons why they were excluded.

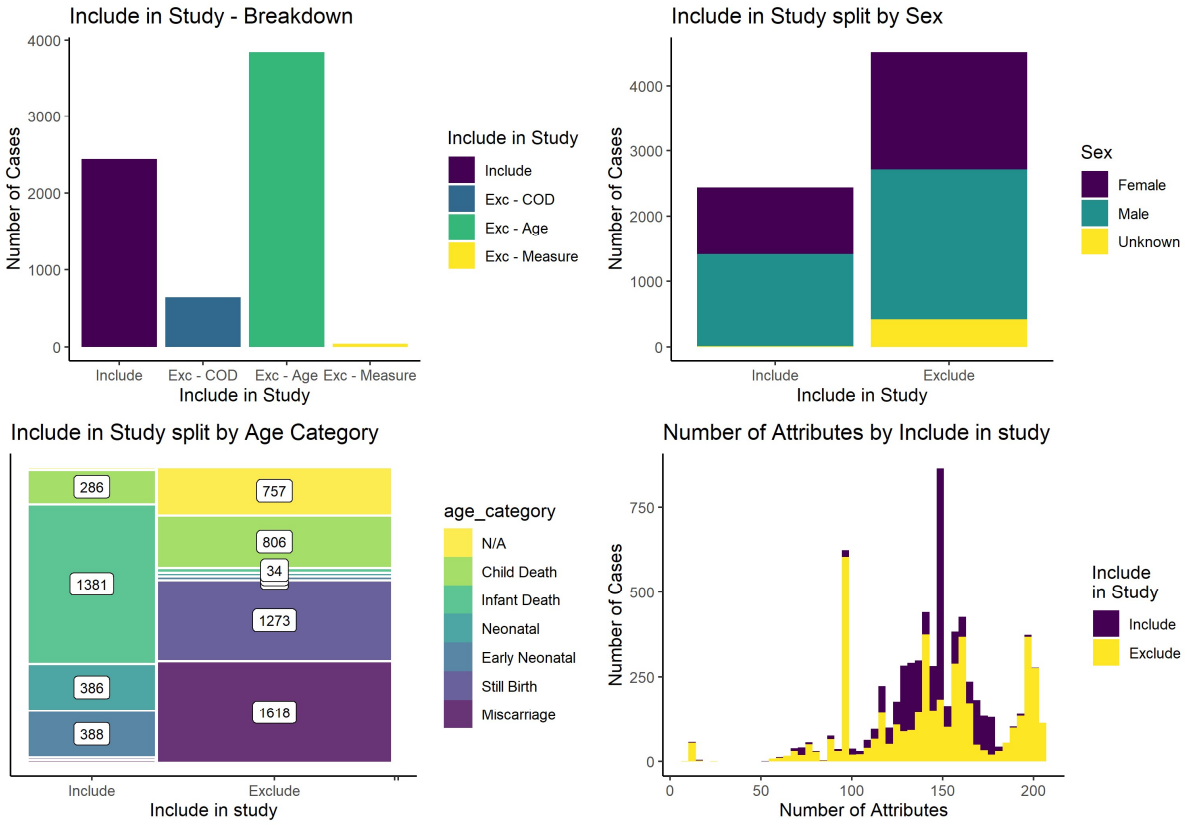


Figure 9 - Visualisation of Study Data Set

The first plot shows that the majority of events were excluded due to the decision to focus on the 1 day to 2 years age group. A significant number of events were excluded due to not having a clear statement about cause of death and then a few events due to incorrect measurements. It is shown that within the events to be included both sexes are well represented and the age groups also good representation. Finally the included events have good numbers of attributes.

5.1.3 Missing data

Having got a study data set defined it should be checked for missing data:

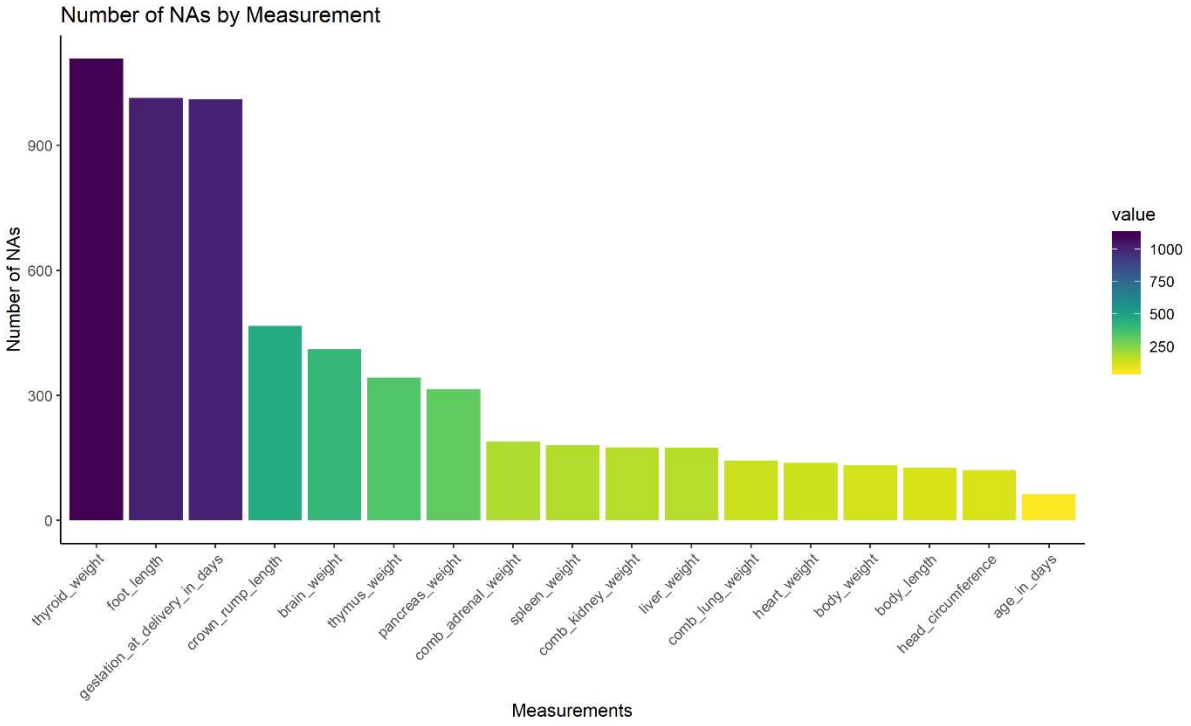


Figure 10 - Number of missing measurement values

Given the significant number of missing values in three features, thyroid_weight, foot_length and gestation_at_delivery_in_days these features will be omitted to maximise the retention of events with other features intact.

It was considered to use imputation of missing values. It was decided that this action could not be applied to gestation_at_delivery_in_days and to only consider post-mortems where the gestational age was known would have significantly biased the data as gestational age is known for hospital cases but not coroner's cases.

5.1.4 Imbalanced data

When using categorical data it is important to look for the presence of imbalanced categories. Below is a plot of all the categorical features used in the study data set.

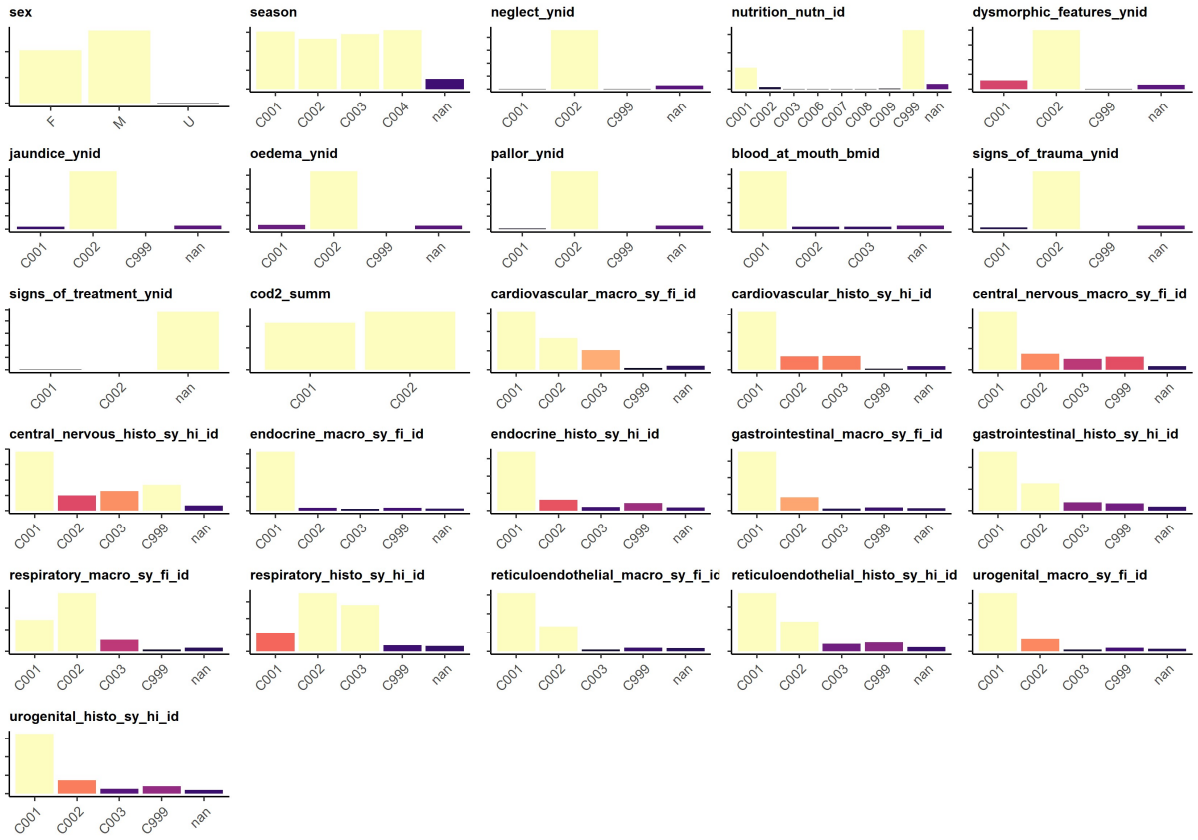


Figure 11 – Variability of Categorical Variables

The light coloured bars indicates that there are more than 500 events with that particular categorical value, the darkest bars indicate very few examples of a particular value exist. By using a categorical value of 'nan' then all categories are represented for all events. For the key categorical feature of 'COD2_SUMM' then it is shown to be reasonable balanced.

5.2 Decision Tree

The output captured for the decision tree model at each post-mortem stage is:

- Decision Tree Details
    - Variable Importance
    - Description of the Node and Split (including # going left or right and even surrogate splits.
    - CP Table
    - Split details
        - Split criteria
        - rows in this node
        - Misclassified
        - Predicted Class
        - % of rows in predicted class for this node.
- Decision Tree plot –a visual representation of the split details above

The split criteria are particularly interesting for example the split of age_in_days is given between 12 and 16 days where as the current clinical split of age category is based on 28 days since birth.

Once the models had be run for all stages then the following outputs were produced:

- Confusion Matrix – predictive accuracy based on training and test data split 80% training 20% test. The predictive accuracy of both C001 – Not determined and C002 – determined is also given.
- Relative Feature importance

5.2.1 External

Decision Tree Details

```
Classification tree:
rpart(formula = cod2_summ ~ ., data = data_train, method = "class",
    control = control)

Variables actually used in tree construction:
[1] age_in_days                 body_weight                 dysmorphic_features_ynid_c001
[4] oedema_ynid_c002

Root node error: 789/1768 = 0.44627

n= 1768

        CP nsplit rel error  xerror     xstd
1 0.110900      0   1.00000 1.00000 0.026492
2 0.060837      2   0.77820 0.80355 0.025558
3 0.040558      3   0.71736 0.75919 0.025223
4 0.016477      4   0.67681 0.71483 0.024839
5 0.010000      5   0.66033 0.71863 0.024874
```

```
n= 1768

node), split, n, loss, yval, (yprob)
      * denotes terminal node

 1) root 1768 789 1 (0.4462670 0.5537330)
   2) age_in_days>=15.5 1362 652 0 (0.5212922 0.4787078)
     4) age_in_days< 275.5 1073 449 0 (0.5815471 0.4184529)
       8) oedema_ynid_c002>=0.5 971 374 0 (0.6148301 0.3851699)
        16) dysmorphic_features_ynid_c001< 0.5 877 311 0 (0.6453820 0.3546180)
          32) body_weight< 0.5913 854 293 0 (0.6569087 0.3430913) *
          33) body_weight>=0.5913 23   5 1 (0.2173913 0.7826087) *
        17) dysmorphic_features_ynid_c001>=0.5 94  31 1 (0.3297872 0.6702128) *
       9) oedema_ynid_c002< 0.5 102  27 1 (0.2647059 0.7352941) *
     5) age_in_days>=275.5 289  86 1 (0.2975779 0.7024221) *
   3) age_in_days< 15.5 406  79 1 (0.1945813 0.8054187) *
```

Decision Tree Plot

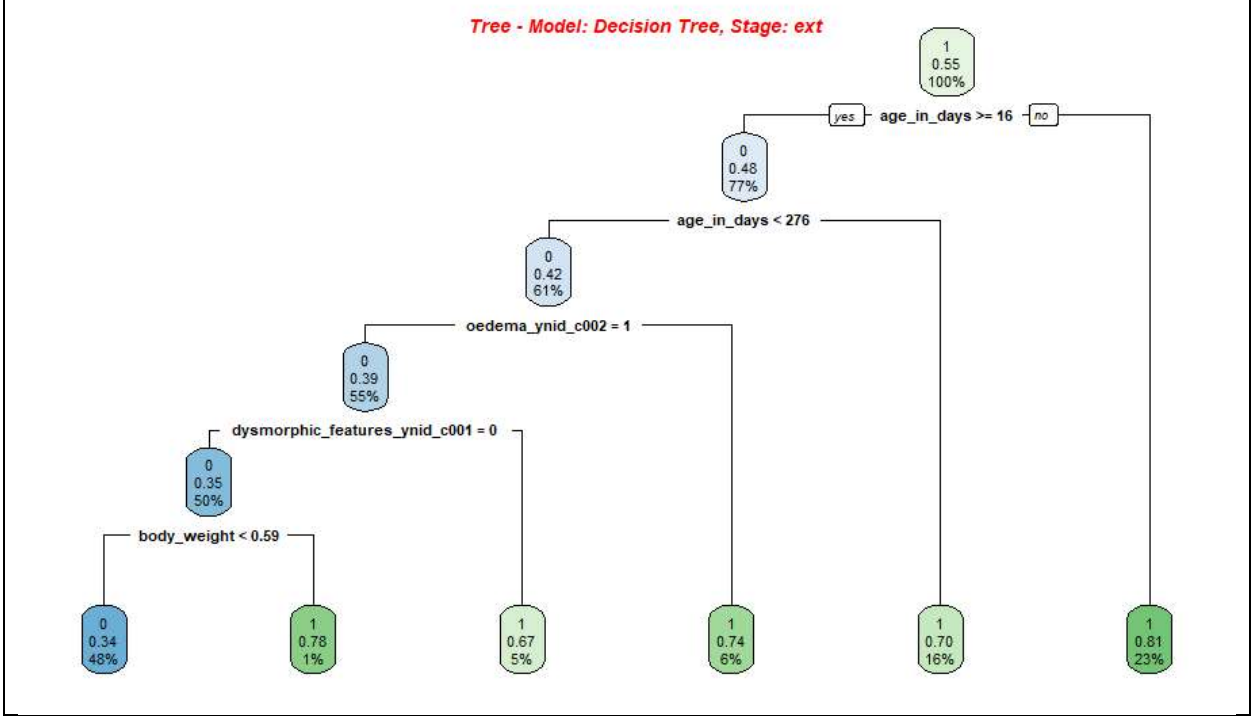


Figure 12 – Decision Tree – External Examination

5.2.2 Internal – Stage 1 – Organs

Decision Tree Details

```
Classification tree:
rpart(formula = cod2_summ ~ ., data = data_train, method = "class",
    control = control)

Variables actually used in tree construction:
[1] age_in_days      body_weight      brain_weight     heart_weight     oedema_ynid_c001

Root node error: 624/1340 = 0.46567

n= 1340

        CP nsplit rel error  xerror     xstd
1 0.246795      0   1.00000 1.00000 0.029263
2 0.089744      1   0.75321 0.76122 0.028062
3 0.060897      2   0.66346 0.71154 0.027613
4 0.014423      3   0.60256 0.63462 0.026767
5 0.011218      5   0.57372 0.64744 0.026921
6 0.010000      6   0.56250 0.63942 0.026825
```

```
n= 1340

node), split, n, loss, yval, (yprob)
      * denotes terminal node

 1) root 1340 624 0 (0.5343284 0.4656716)
   2) age_in_days>=14.5 1062 408 0 (0.6158192 0.3841808)
     4) age_in_days< 310.5 862 280 0 (0.6751740 0.3248260)
       8) heart_weight< 0.61165 808 234 0 (0.7103960 0.2896040)
        16) brain_weight< 0.3006 715 185 0 (0.7412587 0.2587413)
          32) oedema_ynid_c001< 0.5 696 172 0 (0.7528736 0.2471264) *
          33) oedema_ynid_c001>=0.5 19   6 1 (0.3157895 0.6842105) *
        17) brain_weight>=0.3006 93  44 1 (0.4731183 0.5268817)
          34) body_weight< 0.3684 57  22 0 (0.6140351 0.3859649) *
          35) body_weight>=0.3684 36   9 1 (0.2500000 0.7500000) *
       9) heart_weight>=0.61165 54   8 1 (0.1481481 0.8518519) *
     5) age_in_days>=310.5 200  72 1 (0.3600000 0.6400000) *
   3) age_in_days< 14.5 278  62 1 (0.2230216 0.7769784) *
```

Decision Tree Plot

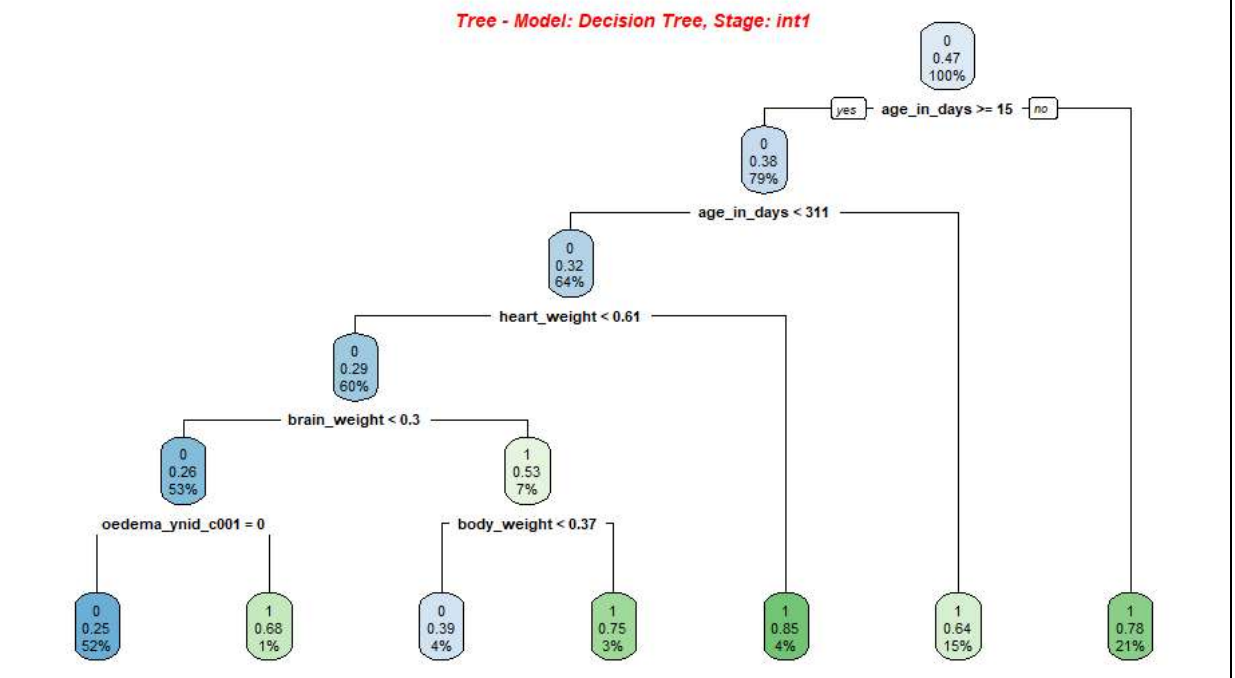


Figure 13 – Decision Tree – Internal Examination – Organs

5.2.3 Internal – Stage 2 – Macro investigation

Decision Tree Details

```
Classification tree:
rpart(formula = cod2_summ ~ ., data = data_train, method = "class",
    control = control)

Variables actually used in tree construction:
[1] age_in_days                          cardiovascular_macro_sy_fi_id_c003
[3] central_nervous_macro_sy_fi_id_c003 comb_lung_weight
[5] respiratory_macro_sy_fi_id_c003

Root node error: 624/1340 = 0.46567
n= 1340
        CP nsplit rel error  xerror     xstd
1 0.243590      0   1.00000 1.00000 0.029263
2 0.107372      1   0.75641 0.75641 0.028022
3 0.078526      2   0.64904 0.66987 0.027178
4 0.056090      3   0.57051 0.59936 0.026314
5 0.038462      4   0.51442 0.54487 0.025527
6 0.014423      5   0.47596 0.51603 0.025065
7 0.010000      7   0.44712 0.49359 0.024682
```

```
n= 1340

node), split, n, loss, yval, (yprob)
      * denotes terminal node

  1) root 1340 624 0 (0.5343284 0.4656716)
    2) age_in_days>=12.5 1090 423 0 (0.6119266 0.3880734)
      4) cardiovascular_macro_sy_fi_id_c003< 0.5 1003 346 0 (0.6550349 0.3449651)
        8) respiratory_macro_sy_fi_id_c003< 0.5 924 282 0 (0.6948052 0.3051948)
         16) central_nervous_macro_sy_fi_id_c003< 0.5 839 222 0 (0.7353993 0.2646007)
           32) comb_lung_weight< 0.50225 781 181 0 (0.7682458 0.2317542)
             64) age_in_days< 310.5 675 128 0 (0.8103704 0.1896296) *
             65) age_in_days>=310.5 106  53 0 (0.5000000 0.5000000)
              130) comb_lung_weight< 0.20725 66  24 0 (0.6363636 0.3636364) *
              131) comb_lung_weight>=0.20725 40  11 1 (0.2750000 0.7250000) *
           33) comb_lung_weight>=0.50225 58  17 1 (0.2931034 0.7068966) *
         17) central_nervous_macro_sy_fi_id_c003>=0.5 85  25 1 (0.2941176 0.7058824) *
        9) respiratory_macro_sy_fi_id_c003>=0.5 79  15 1 (0.1898734 0.8101266) *
      5) cardiovascular_macro_sy_fi_id_c003>=0.5 87  10 1 (0.1149425 0.8850575) *
    3) age_in_days< 12.5 250  49 1 (0.1960000 0.8040000) *
```

Decision Tree Plot

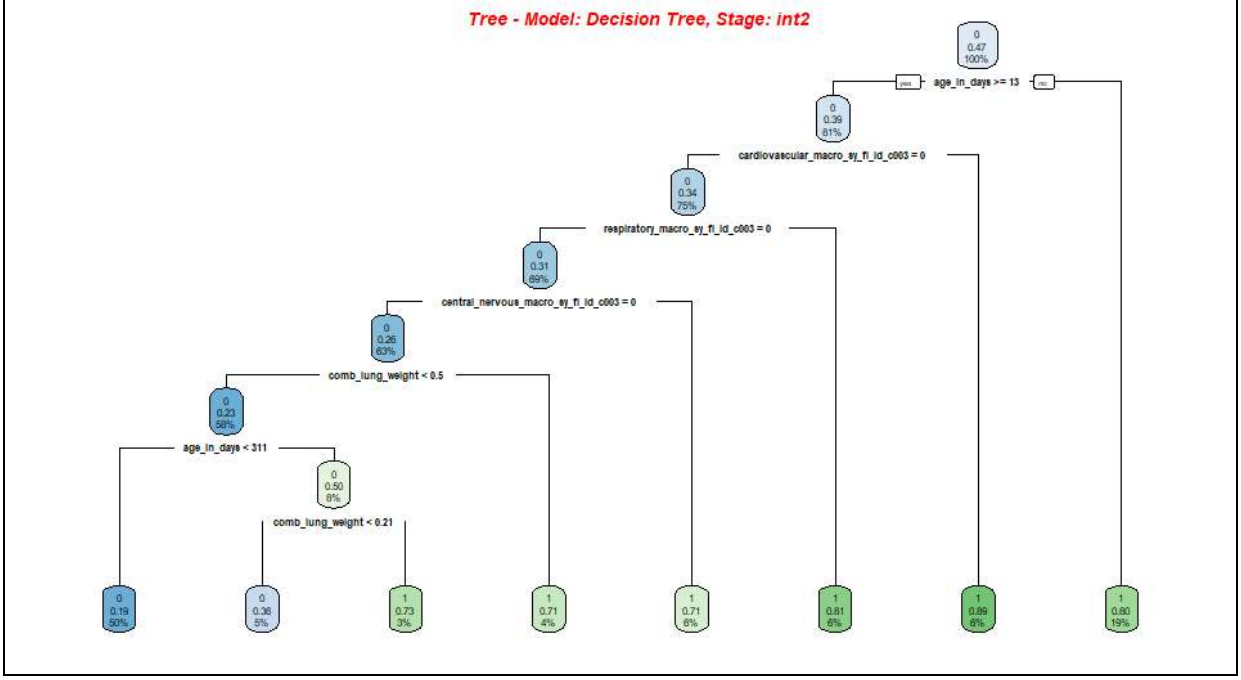


Figure 14 - Decision Tree - Internal Examination - Macro

5.2.4 Internal – Stage 3 – Histological investigation

Decision Tree Details

```
Classification tree:
rpart(formula = cod2_summ ~ ., data = data_train, method = "class",
    control = control)

Variables actually used in tree construction:
[1] age_in_days                          cardiovascular_histo_sy_hi_id_c003
[3] central_nervous_macro_sy_fi_id_c003 respiratory_histo_sy_hi_id_c003

Root node error: 626/1340 = 0.46716

n= 1340

        CP nsplit rel error  xerror     xstd
1 0.372204      0   1.00000 1.00000 0.029175
2 0.089457      1   0.62780 0.62780 0.026622
3 0.055911      2   0.53834 0.60224 0.026294
4 0.028754      3   0.48243 0.52396 0.025142
5 0.010000      4   0.45367 0.48882 0.024547
```

```
n= 1340

node), split, n, loss, yval, (yprob)
      * denotes terminal node

 1) root 1340 626 0 (0.5328358 0.4671642)
   2) respiratory_histo_sy_hi_id_c003< 0.5 889 284 0 (0.6805399 0.3194601)
     4) age_in_days>=12.5 751 187 0 (0.7509987 0.2490013)
       8) cardiovascular_histo_sy_hi_id_c003< 0.5 694 141 0 (0.7968300 0.2031700)
        16) central_nervous_macro_sy_fi_id_c003< 0.5 638 104 0 (0.8369906 0.1630094) *
        17) central_nervous_macro_sy_fi_id_c003>=0.5 56  19 1 (0.3392857 0.6607143) *
       9) cardiovascular_histo_sy_hi_id_c003>=0.5 57  11 1 (0.1929825 0.8070175) *
     5) age_in_days< 12.5 138  41 1 (0.2971014 0.7028986) *
   3) respiratory_histo_sy_hi_id_c003>=0.5 451 109 1 (0.2416851 0.7583149) *
```

Decision Tree Plot

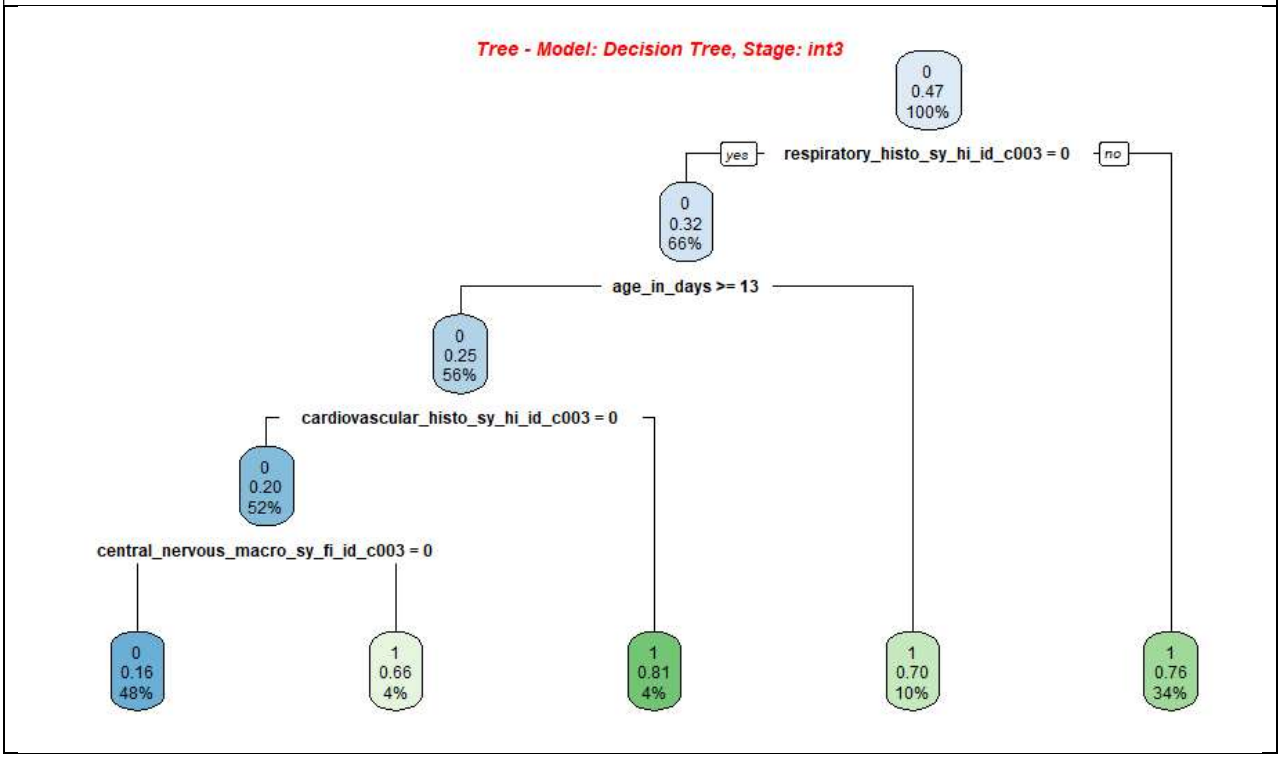


Figure 15 - Decision Tree - Internal Examination – Histology

5.2.5 All Stages

The confusion matrices for each stage give the overall predictive accuracy of the model as well as the predictive accuracy of predicting each individual classification. It can be seen that the models vary considerably in their prediction of the individual classifications.

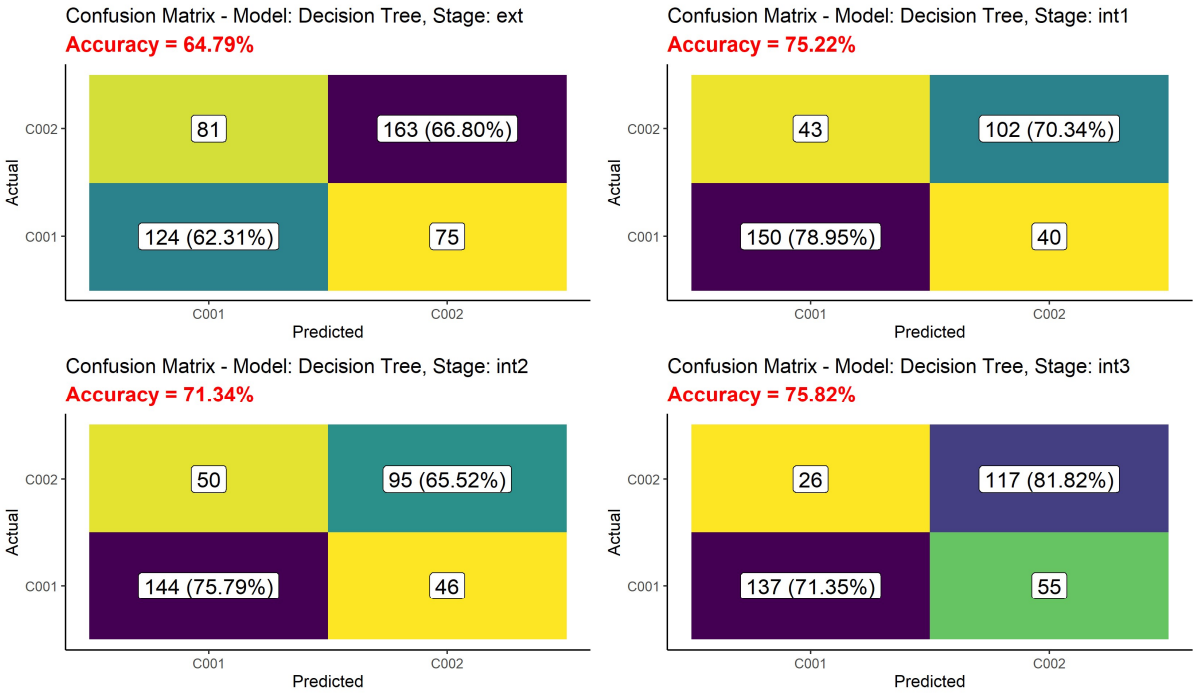


Figure 16 - Decision Tree Confusion Matrices – All stages

For all four stage it can be shown the relative importance of a feature as the different stages of a post-mortem progresses.

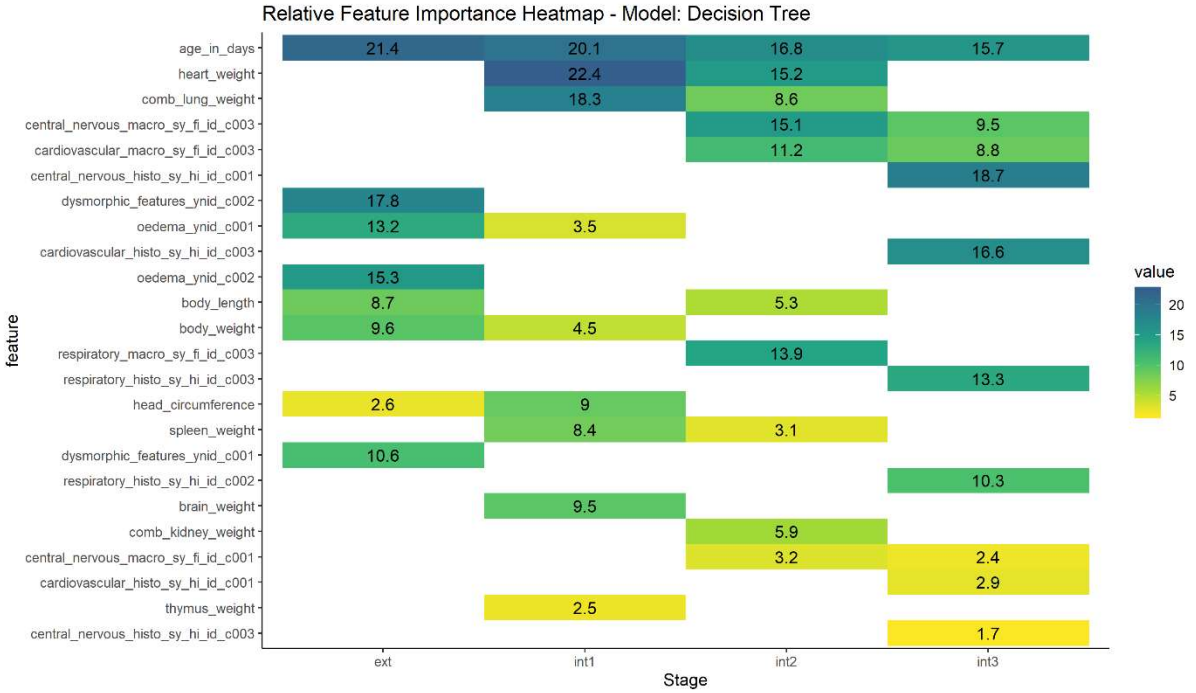


Figure 17 – Decision Tree – Relative Feature Importance – All Stages

From the above plot it can be seen that age_in_days is important across all stages of the post-mortem but other features, for example, heart weight, start of important but by the final stage has been dropped of the list.

5.3 Ensemble Models

For the ensemble models the output is focussed on predictive accuracy shown by the confusion matrices and Relative Feature Importance.

5.3.1 Random Forest

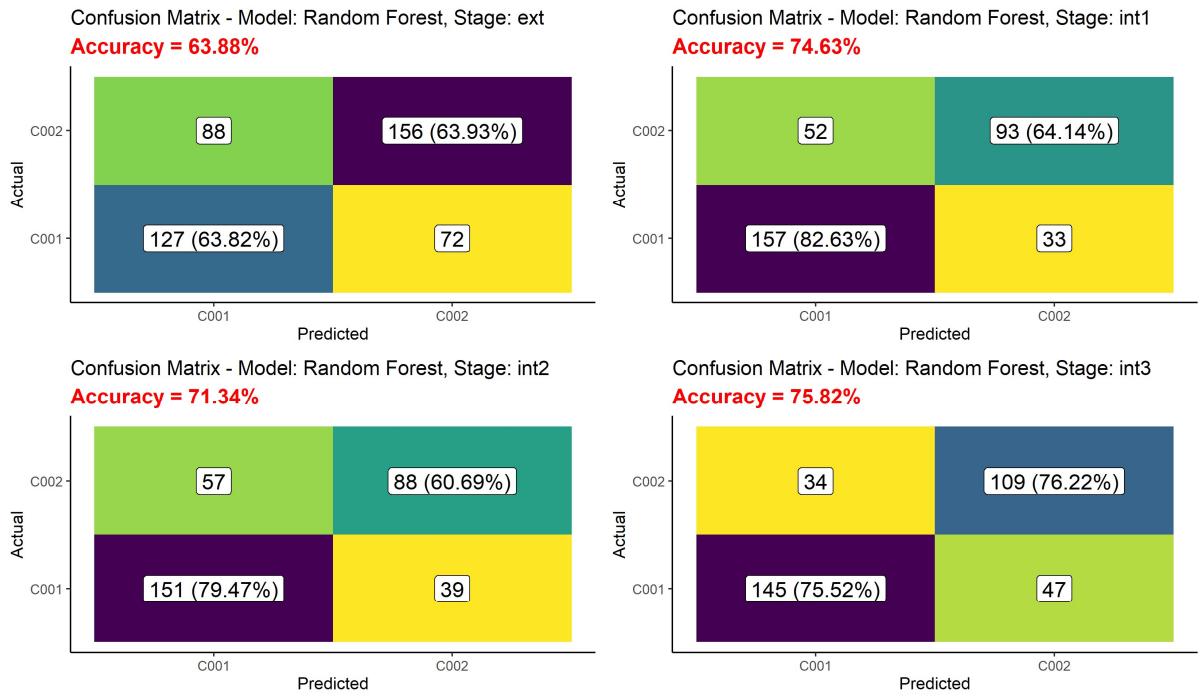


Figure 18 – Random Forest Confusion Matrices – All stages

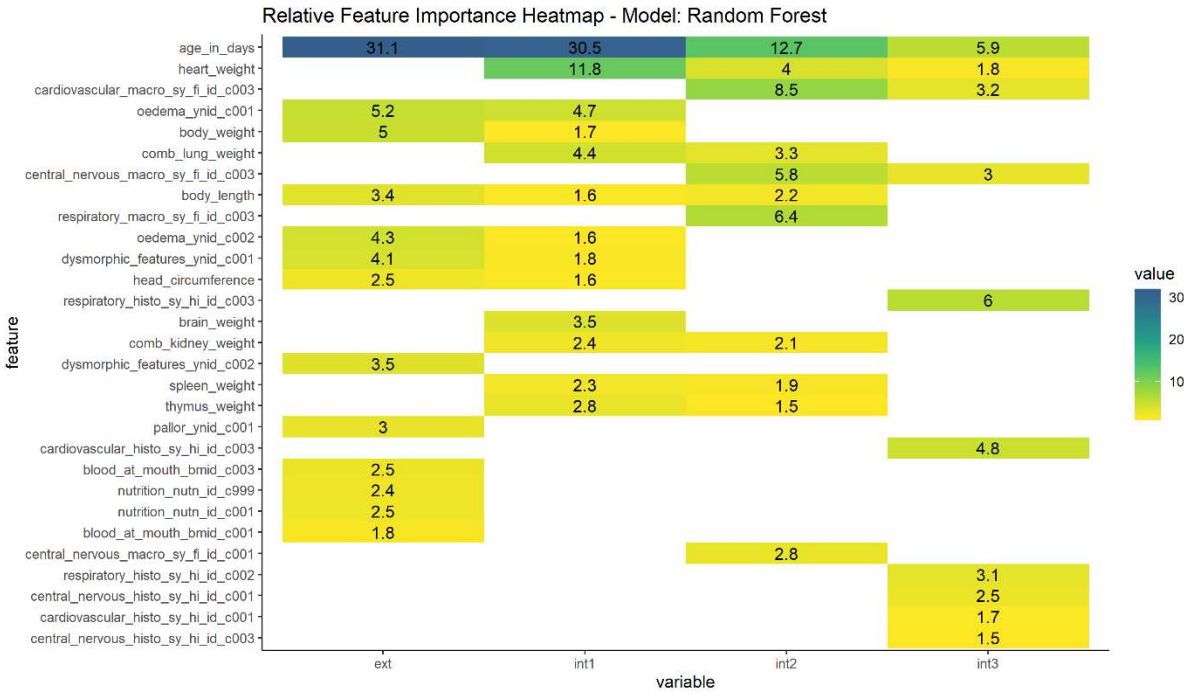


Figure 19 – Random Forest Relative Feature Importance – All Stages

5.3.2 Gradient Boosted Decision Tree

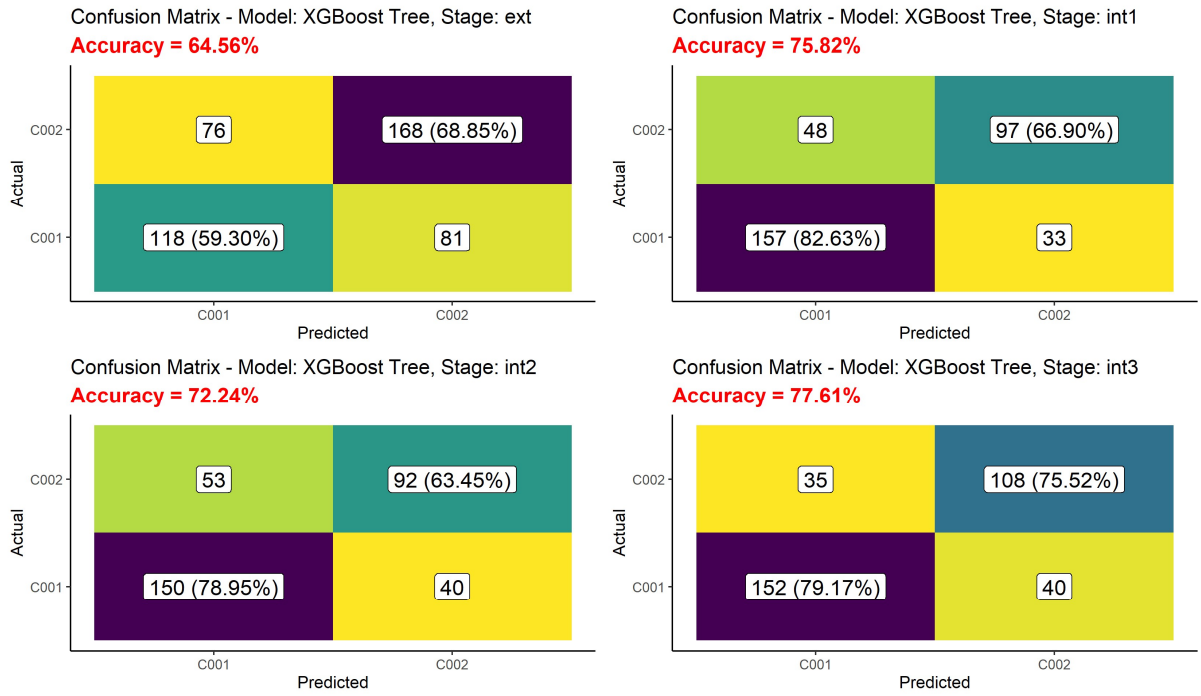


Figure 20 – XGBoost Confusion Matrices – All stages

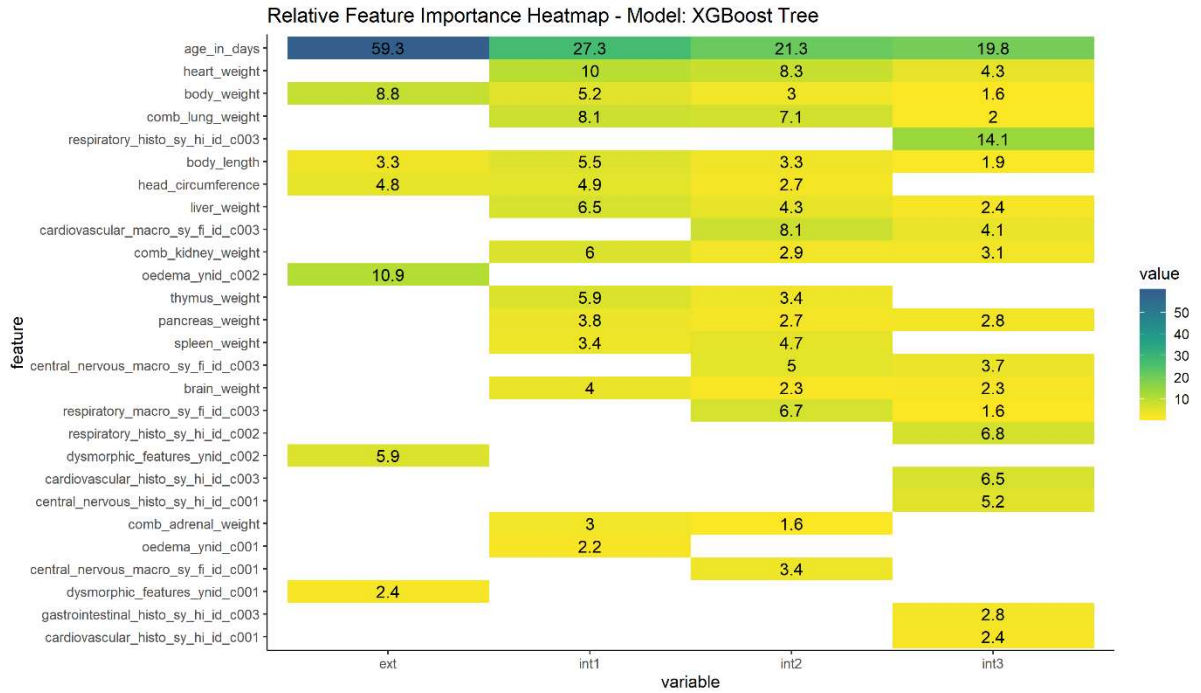


Figure 21 – XGBoost Relative Feature Importance – All Stages

5.4 Compare models with varying random seeds

Each of the models above was run 5 times with different random seeds and the results compared:

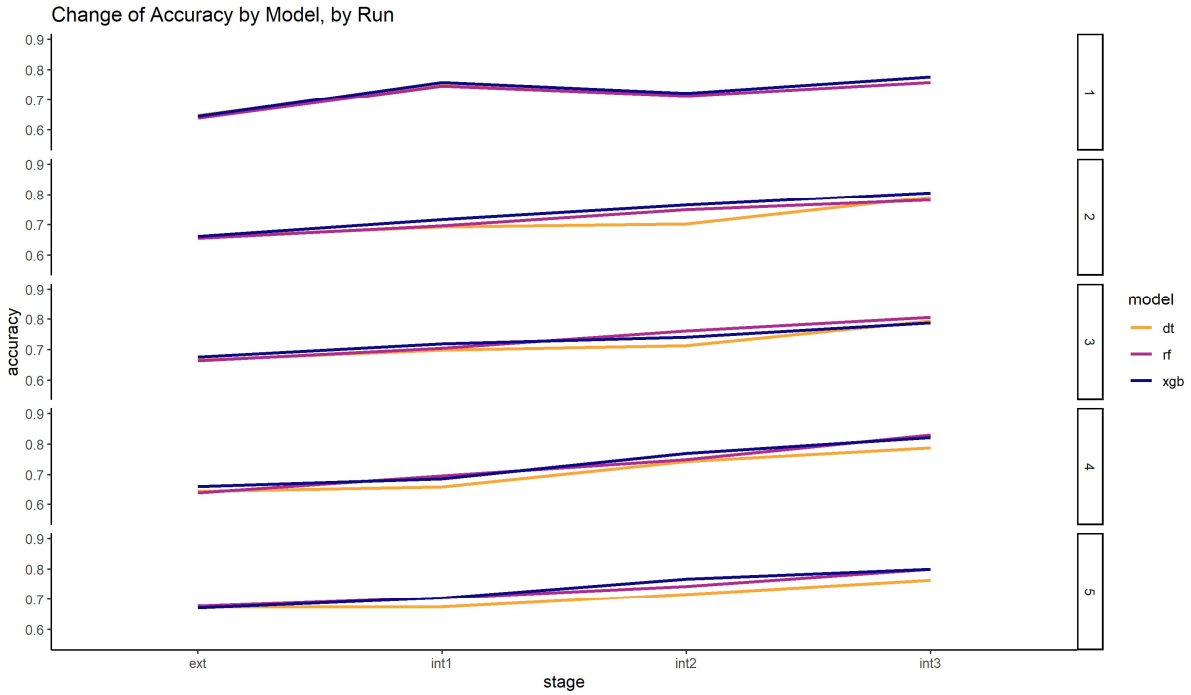


Figure 22 – Predictive Accuracy of each stage by Model by Run

The first plot shows how the different models agree in the increased predictive as the different stages of the post-mortem.

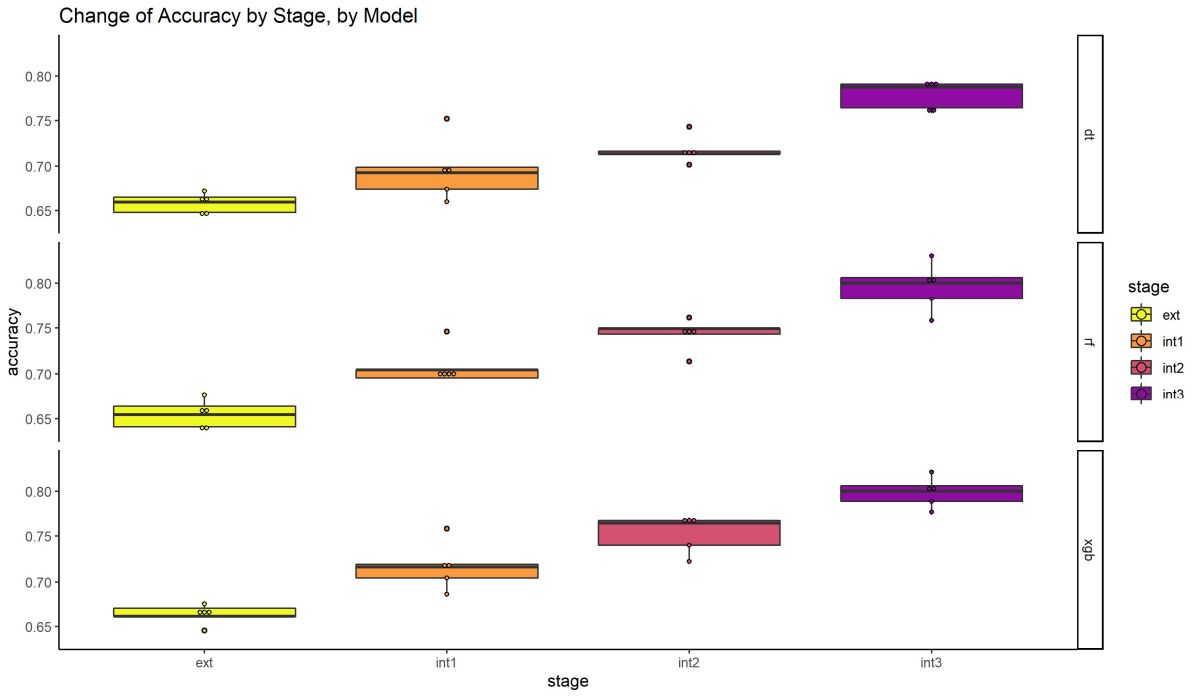


Figure 23 – Variability in Accuracy by Model by Stage

The second plot shows the variability of the predictive accuracy by model by stage for the five different random seeds.

Finally the change in relative feature importance by model for the five different random seeds. The first and last stage have been plotted to demonstrate the relative stability from one random seed to another.

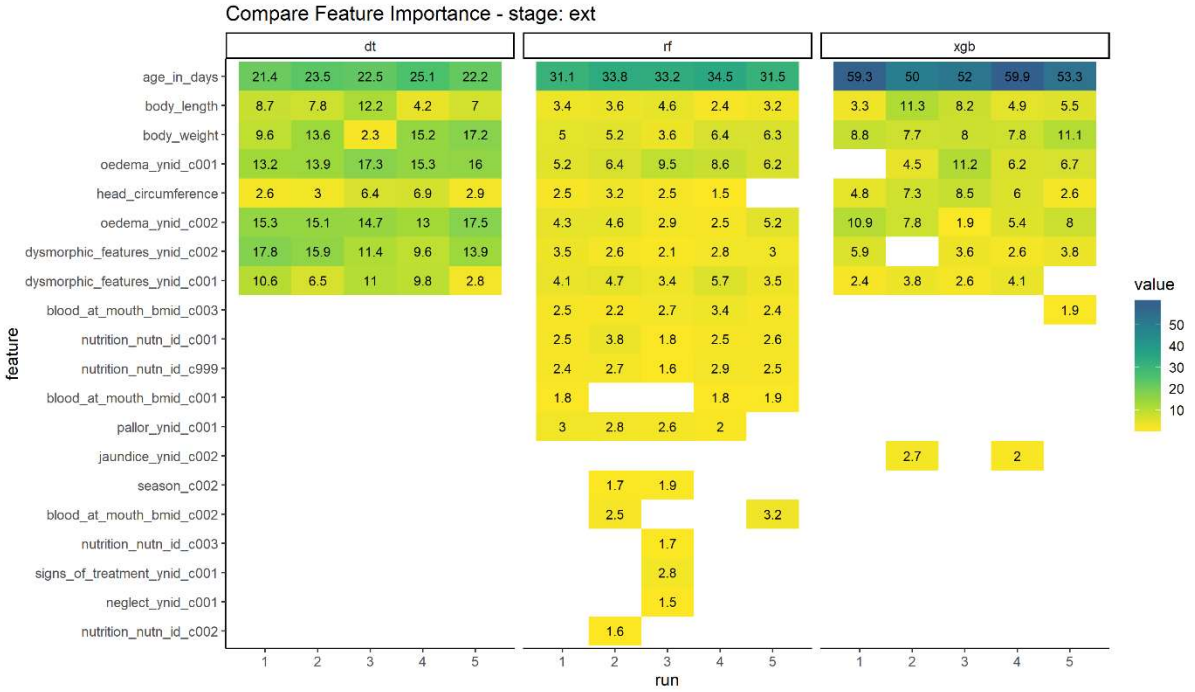


Figure 24 – Compare Feature Importance by Model, Stage: Ext, by Run

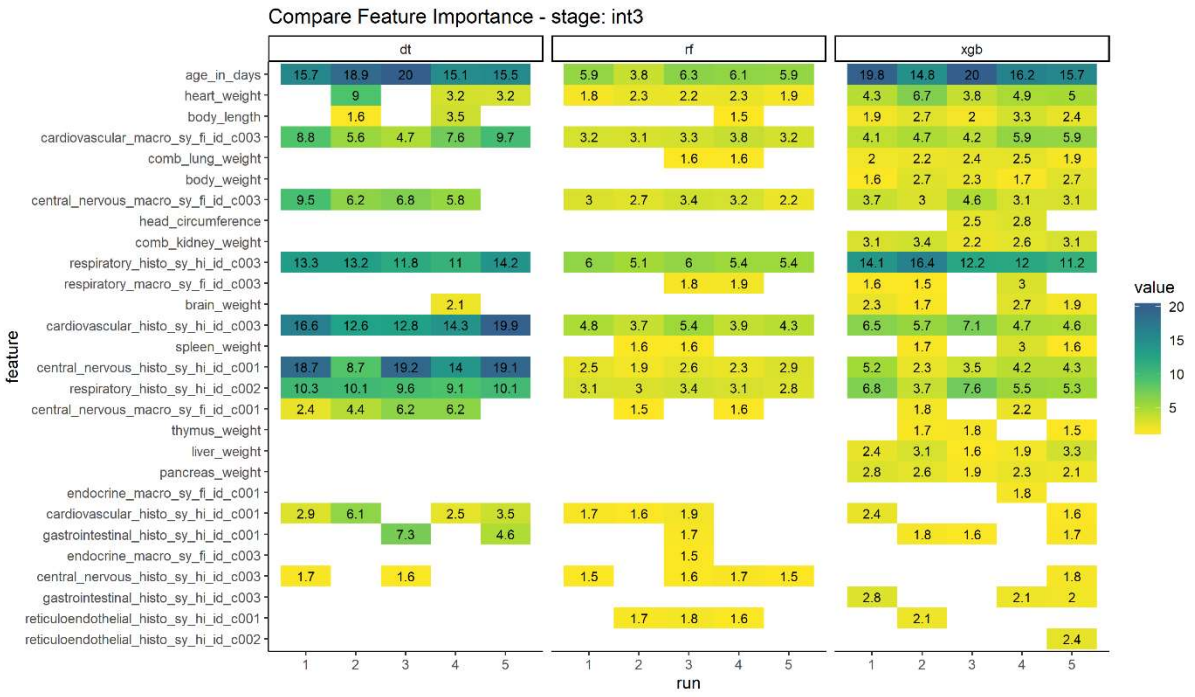


Figure 25 – Compare Feature Importance by Model, Stage: Int3, by Run

6 Results

The results of this project are assessed in terms of the predictive accuracy of the different models and then the relative feature importance and their change as the different stages of the post-mortem.

6.1 Predictive Accuracy

Mean Predictive Accuracy - COD Combined by Model by Stage

	Decision Tree	Random Forest	XGBoost
ext	65.82 %	65.51 %	66.28 %
int1	69.55 %	70.93 %	71.70 %
int2	71.76 %	74.33 %	75.28 %
int3	77.91 %	79.52 %	79.82 %

Mean Predictive Accuracy - COD Not Determined by Model by Stage

	Decision Tree	Random Forest	XGBoost
ext	63.15 %	67.02 %	61.86 %
int1	71.74 %	76.83 %	75.37 %
int2	74.13 %	82.15 %	80.82 %
int3	74.79 %	78.58 %	81.58 %

Mean Predictive Accuracy - COD Determined by Model by Stage

	Decision Tree	Random Forest	XGBoost
ext	68.11 %	64.38 %	70.04 %
int1	67.12 %	63.95 %	67.40 %
int2	68.67 %	64.50 %	68.24 %
int3	81.98 %	80.60 %	77.62 %

From the first table it can be seen that both ensemble methods outperform the basic decision tree model and the XGBoost model outperforms Random Forest but only by a small margin. Also the underlying increase on predictive accuracy as the post-mortem stages progress is reflected across all three models.

When it comes to the predictive accuracy of the individual classifications the picture becomes a little less clear with the Random Forest model outperforming the XGBoost model at certain stages.

Taking these results as a whole the XGBoost model would be classed as the most accurate model. Though the basic analytic interpretation of the decision would still seem to be valid.

6.2 Relative Feature Importance by Stage of Post-mortem

As XGBoost is the most accurate of the models considered then the results for Feature Importance will be used from this model:

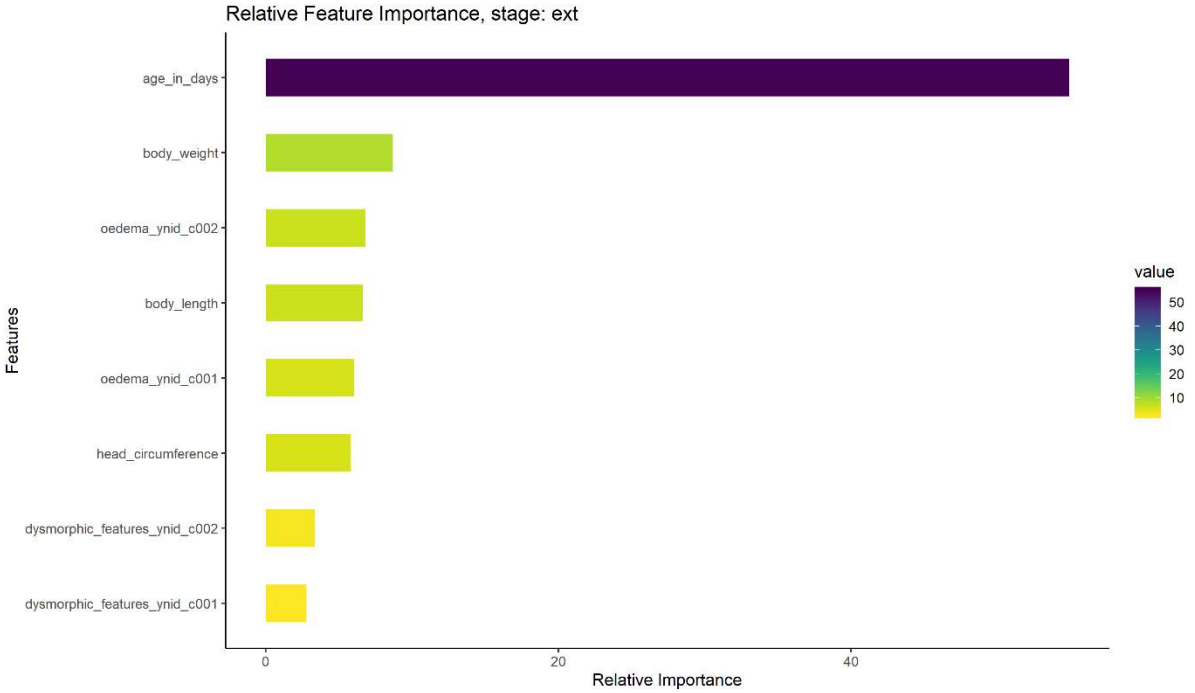


Figure 26 – Relative Feature Importance, Stage: External Examination

At the initial external stage of the post-mortem only the age of the patient has any significant bearing on being able to determine the cause of death. From the decision tree output it can be shown that the main boundary is around 16 days of age, with a secondary boundary of around 276 days.

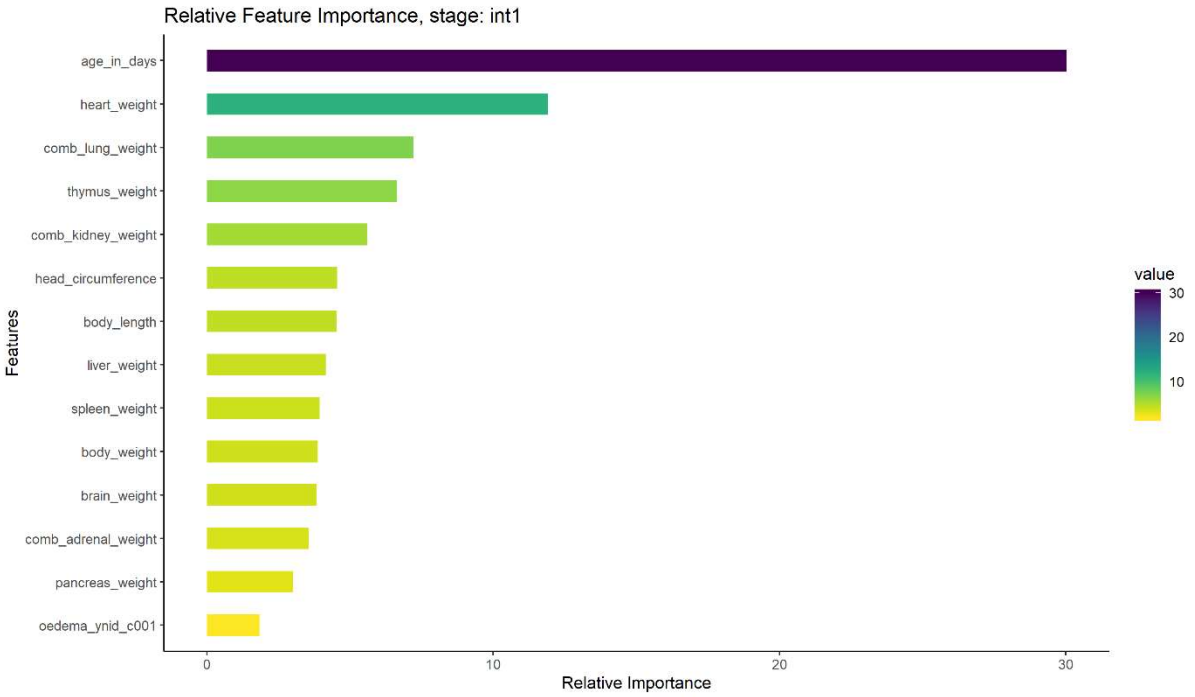


Figure 27 – Relative Feature Importance, Stage: Internal Examination – Organs

At the second stage the age of the patient is till of primary importance but the weight of various organs begins to have some significance. From the results of the decision tree it can be shown that that the feature boundary is when then organ weight varies from the 'normal' by around 30%.

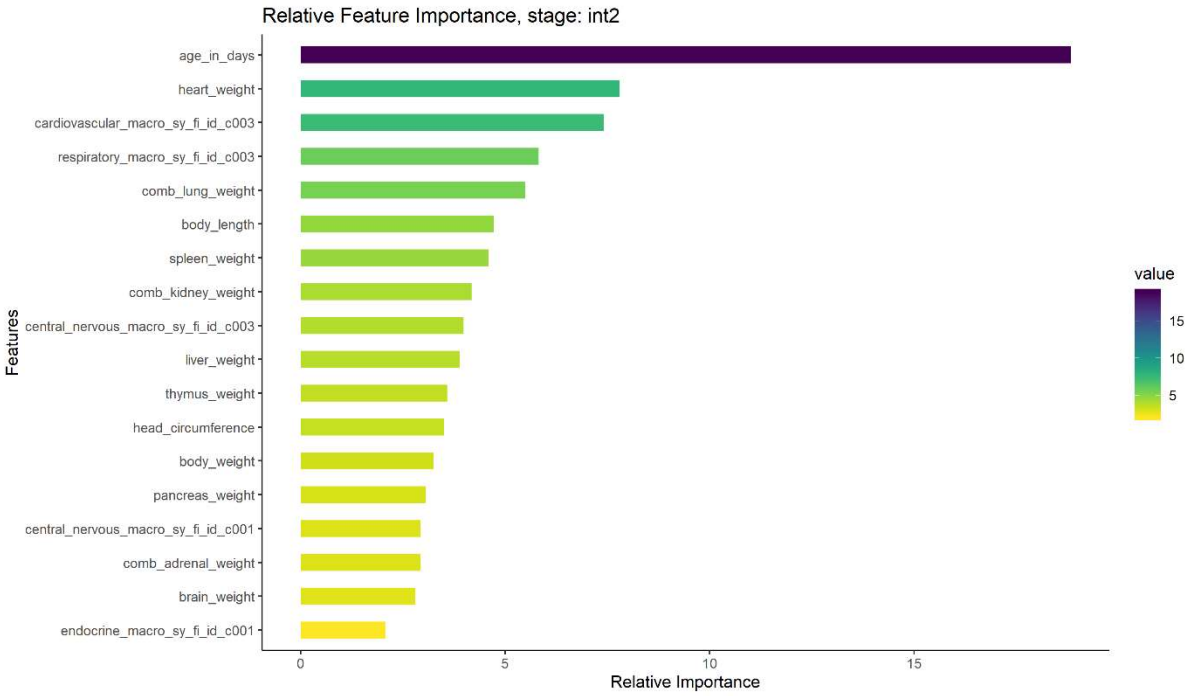


Figure 28 – Relative Feature Importance, Stage: Internal Examination – Macro

From the plot above the results of the macro stage of the internal examination has some effect the important features from the previous stage are still ranked highly.

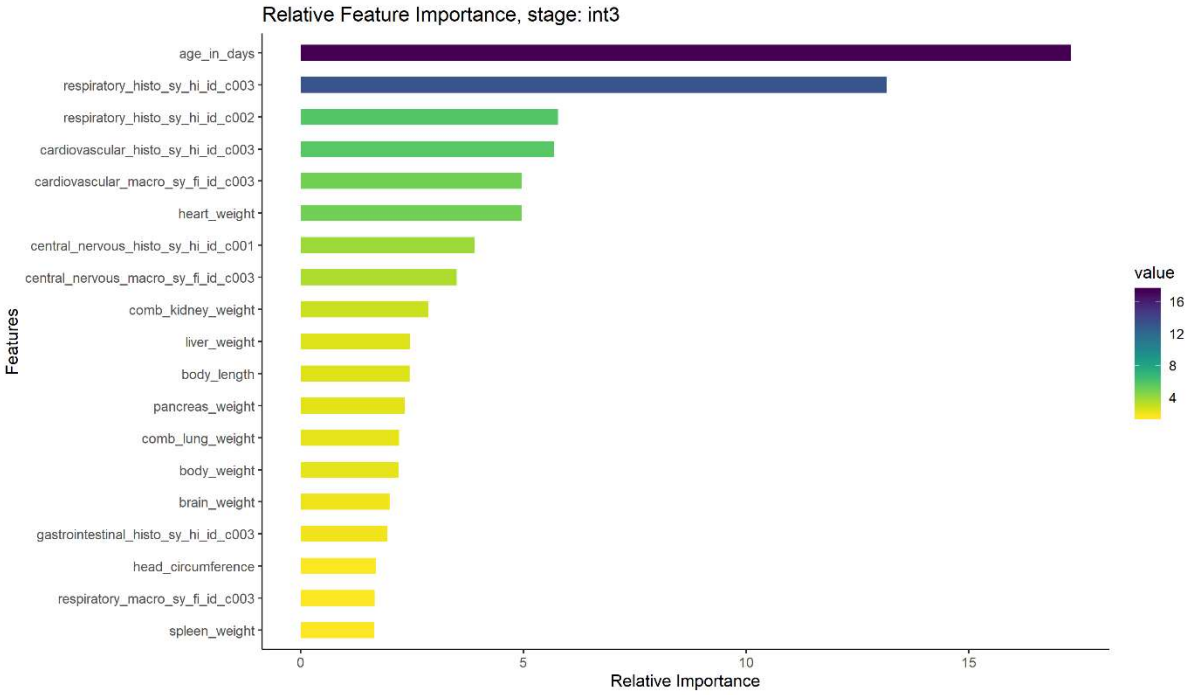


Figure 29 – Relative Feature Importance, Stage: Internal Examination - Histology

In the final stage the histology classifications now are significantly reducing the overall importance of the age of the patient.

7 Project Evaluation

The project will be evaluated against the aims laid out at the start of the project.

7.1 Data Engineering

The aim was to develop a routine to extract data from the existing Post-mortem Research Database into an entity attribute value schema that will make the data more readily available for data analytics.

This project has been able to demonstrate that after some use case specific understanding i.e. underlying structure of the source database and how that data could be mapped onto Patients, Events and Event Attributes the following generic routines could be developed:

- Create and update events and event attributes
- Add additional event attributes
- Create RDVs for different stages for analysis
- Carry out some data wrangling i.e. one-hot encoding.

Using these generic routines it was then very efficient to produce RDV's suitable structure for each model and for each stage of the post-mortem.

7.2 Decision Tree

The aim was to apply the Decision Tree Analytical method to the extracted data to develop operational strategies that can be applied to paediatric post-mortems to prioritise which data is required to achieve the target of specifying the cause of death.

A decision tree was developed for each stage of the post-mortem process and that the structure of each model could be clearly identified and could be used to communicate back to the Pathologists carrying out the post-mortems.

At each stage of the post-mortem it could be seen which features were important for determining cause of death. Although the likely hood of being able to predict a cause of death increased at each stage of the post-mortem one key factor, age in days, that determined whether cause of death could be determined was available without a post-mortem taking place.

7.3 Ensemble Models

The aim was to investigate ensemble strategies, specifically Random Forests and Gradient Boosting to see how these techniques can improve on the basic Decision Tree method.

Both Random Forest and XGBoost ensemble models were developed and they both showed progressive improvements in predictive accuracy over the initial decision tree. Both models could be used to identify the relative feature importance of each stage of the post-mortem but did not have the same transparency of the decision tree models.

7.4 Overall Project Assessment

Finally I would like to consider what aspects of the project went well, which parts went less well and what are the lessons I have learnt from the process of completing this project.

The areas that I feel worked well during the project were:

1. Using an agile approach to project. Each week I set myself objectives for the coming week and at the start of the following week I reviewed progress from last week before setting my next objectives enabling me to respond when certain aspects had taken longer to complete than expected. Obviously with only one team member I couldn't experience all the aspects of an agile approach but I feel it contributed significantly to producing a complete project.
2. Developing a project pipeline. All aspects of the project were coded so that each stage could be repeated any number of times and that each stage fed the following stage. This process became more important towards the end of the project when new aspects came to light and I wished to repeat processes from an early stage of the project.
3. Use of EAV structure. The decision to use the EAV schema gave me a significant amount of up front coding but during the lifetime of the project I feel that I gained that time back in the flexible and reproducible way I could produce the RDVs for the analytics stage.
4. Python and R. I decided that I wanted to develop hands on experience of using both Pythion and R so I used the natural split of the project between data engineering and analytics to split the use of Python and R. I feel that the experience of using PyODBC for SQL manipulation of data and ggplot2 for plotting my analytics will be skills that I will be using extensively in my working life.

Areas that I found particularly challenging were:

1. Developing the ETL process. Although the basic mechanics of the ETL process was relatively easily produced my initial coding would have taken over five days to convert the PM Research database to the HAS schema. It took extensive profiling to highlight bottlenecks to improve the efficiency of the process and get the time so that the process could be completed overnight.
2. Finalising of the adjusted RDV structure. A significant amount of time was taken getting the format of the adjusted RDVs suitable for use by all three of the modelling packages. Of this process the single biggest aspect was getting the measurement features to be age normalised as part of the project pipeline process.
3. Creation and tuning of three model types for four stages of post-mortem. I completely underestimated time to create and tune each model which had to be done for each stage of the post-mortem as the data sets changed considerably from stage to stage. Initially I had expected to consider all age groups and additional post-mortem stages but as part of my agile approach I had to modify my aspirations on what I could achieve.
4. Python versus R. Although I feel that getting experience of using both Python and R was worthwhile I had significantly more experience of Python coming into the project and I found my lack of experience of using the different style of R development quite challenging. The two very different approaches led to an inefficient development process at times.

I would also like to highlight some key lessons learnt during this project's development

1. Modelling takes longer then you imagine. Developing a well-tuned model on real world data is a non-trivial task and although examples based on Iris data are extremely useful to get you going they can lead you to believe that this process is relatively straightforward.
2. Blog posts are not always correct or complete. When approaching a new modelling package extensive research will bring up any number of potentially useful looking examples in an aspiring data scientist's blog posts. Although these can be a useful resource they should be seen merely as a starting point for one's own exploration of the potential of any given package.
3. Visualisation is done for a reason. I left my visualisation stage till quite late in my project and it then highlighted a number of omissions that I had made which meant that I had to go back to a quite early stage in the project process and redo that work. Do visualisations at the right time and do it thoroughly to get to know your data.

8 Conclusion

This project was able to show the benefits of developing a pipeline utilising the appropriate development strategies for the various stages of the project to enable reproducible results.

Data Engineering was a significant part of this project and some of the generic issues encountered during this process were clearly highlighted as well as the benefits of adopting a data schema tailored for research and analytics. The analytic processes of visualising the data followed by creation and distinct models tuned to the individual stages of the post-mortem was then extensively explored.

The project was completed by assessing the predictive accuracy of the various models and the relative feature importance at each stage of the post-mortem and how these results can be used to inform clinical practice.

The overall experience of doing a complete project was by far and away the most rewarding aspect of the overall MSc course.

8.1 Recommendations for Future Work

It is the intention of GOSH to use this project as the basis of future publications. The following areas of further work need to be considered:

1. Improve current models. There are two areas that the models developed during this project could be improved. Firstly by improving the process of measurement normalisation using gestational age at birth where possible. Secondly some use of imputation of missing values for certain measurements by investigating the setting of missing vales to zero in the normalised data set. I would also concentrate on the XGBoost models, the RandomForest package took much longer than the other two packages to run and added very little to the overall results.
2. Use of the full data set. Analyse the complete set of age categories including the foetal groups. Include the additional post-mortem stages of Microbiology and Metabolic investigations. This additional data would also enable the consideration of cause of death by class rather than just the binary determined and not determined.
3. Use other machine learning models. Using the input data developed during this project additional insights could be achieved by considering different analytic algorithms for example support vector machines and neural networks. Un-supervised modelling could also be used to investigate any underlying clustering.

References

Abolfazl Ravanshad, 2018. Ensemble Methods [online] Medium. Available at: https://medium.com/@aravanshad/ensemble-methods-95533944783f *[Accesses 2019-03-24]*

Abolfazl Ravanshad, 2018. Gradient Boosting vs Random Forest. [online] Medium. Available at: https://medium.com/@aravanshad/gradient-boosting-versus-random-forest-cfa3fa8f0d80 *[Accesses 2019-03-24]*

Analytics Vidhya Contributors (2016). A Complete Tutorial on Tree Based Modeling from Scratch. [online] Analytics Vidhya Avaiable at: https://www.analyticsvidhya.com/blog/2016/04/complete-tutorial-tree-based-modeling-scratch-in-python/ *[Accesses 2019-03-24]*

Archer, K.J., 2010. rpartOrdinal: an R package for deriving a classification tree for predicting an ordinal response. *Journal of Statistical Software*, *34*, p.7.

Atkinson, E., Ripley, B. and Therneau, T., 2019, CRAN - Package rpart, viewed 14th September, 2019, <https://cran.r-project.org/web/packages/rpart/index.html>.

Ben-Sasi, K., Chitty, L.S., Franck, L.S., Thayyil, S., Judge-Kronis, L., Taylor, A.M. and Sebire, N.J., 2013. Acceptability of a minimally invasive perinatal/paediatric autopsy: healthcare professionals' views and implications for practice. Prenatal diagnosis, 33(4), pp.307-312.

Borodin, A. and Zavyalova, Y., 2015. On an EAV based approach to designing of medical data model for mobile healthcare service. UBICOMM 2015, p.33.

Breiman, L. & Cutler, A., 2018, CRAN - Package randomForest, viewed 14th September, 2019, <https://cran.r-project.org/web/packages/randomForest/>.

Chen, T., 2019, CRAN - Package xgboost.html, viewed 14th September, 2019, <https://cran.r-project.org/web/packages/xgboost/>

Chen, T. and Guestrin, C., 2016, August. Xgboost: A scalable tree boosting system. In Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining (pp. 785-794). ACM

Create Elegant Data Visualisations Using the Grammar of Graphics • ggplot2, 2019, tidyverse website, viewed 14th September, 2019, < https://ggplot2.tidyverse.org/>.

*Dinu, Valentin; Nadkarni, Prakash (2007), "Guidelines for the effective use of entity-attribute-value modeling for biomedical databases", International Journal of Medical Informatics, **76** (11–12): 769–79.*

Furlong, K.R., Anderson, L.N., Kang, H., Lebovic, G., Parkin, P.C., Maguire, J.L., O'Connor, D.L., Birken, C.S. and TARGet Kids! Collaboration, 2016. BMI-for-age and weight-for-length in children 0 to 2 years. Pediatrics, 138(1), p.e20153809.

Garnier, S, Ross, N & Rudis, B 2018, The viridis color palettes, Centre for Creative Leadership, viewed 14 September, 2019, <https://cran.r-project.org/web/packages/viridis/vignettes/intro-to-viridis.html>

Horn, L.C., Langner, A., Stiehl, P., Wittekind, C. and Faber, R., 2004. Identification of the causes of intrauterine death during 310 consecutive autopsies. European Journal of Obstetrics & Gynecology and Reproductive Biology, 113(2), pp.134-138.

Löper, D., Klettke, M., Bruder, I. and Heuer, A., 2013. Enabling flexible integration of healthcare information using the entity-attribute-value storage model. Health information science and systems, 1(1), p.9.

Lullaby Trust (2018). Facts-and-Figures-for-2015-released-2017. Updated 2018. London: Lullaby Trust.

mkleehammer, 2019, pyodbc: Python ODBC bridge, viewed 14 September, 2019 <https://github.com/mkleehammer/pyodbc>

NHS Contributors (2018). Post Mortem. [online] NHS. Available at: https://www.nhs.uk/conditions/post-mortem [Accessed 2019-03-24].

Open source and enterprise-ready professional software for data science – Rstudio, 2019, RSudio website, viewed 14th September, 2019, <https://www.rstudio.com/>.

Prashant Gupta 2017. Decision Trees in Machine Learning. [online] Medium. Avaiable at: https://towardsdatascience.com/decision-trees-in-machine-learning-641b9c4e8052 *[Accessed 2019-03-24]*

Pryce, J.W., Bamber, A.R., Ashworth, M.T., Kiho, L., Malone, M. and Sebire, N.J., 2014. Reference ranges for organ weights of infants at autopsy: results of> 1,000 consecutive cases from a single centre. BMC clinical pathology, 14(1), p.18.

PyCharm: the Python IDE for Professional Developers by JetBrains 2019, JetBrains s.r.o. website, viewed 14th September, 2019, <https://www.jetbrains.com/pycharm/>.

R: The R Project for Statistical Computing, 2019, The R Foundation website, viewed 14th September, 2019, < https://www.r-project.org/>

RCPath Contributors (2018). Paediatric Pathoogy. [online] Royal College of Pathologists. Available at: https://www.rcpath.org/trainees/examinations/examinations-by-specialty/paediatric-pathology.html *[Accessed 2019-03-24].*

Song, Yan-Yan and Ying Lu.2015. "Decision tree methods: applications for classification and prediction" Shanghai archives of psychiatry vol. 27,2: p130-5.

Therneau, T.M. and Atkinson, E.J., 1997. An introduction to recursive partitioning using the RPART routines.

Weber, M.A., Ashworth, M.T., Risdon, R.A., Hartley, J.C., Malone, M.A.R.I.A.N. and Sebire, N.J., 2008. The role of post-mortem investigations in determining the cause of sudden unexpected death in infancy. Archives of disease in childhood, 93(12), pp.1048-1053.

Weber, M.A., Ashworth, M.T., Risdon, R.A., Malone, M., Burch, M. and Sebire, N.J., 2008. Clinicopathological features of paediatric deaths due to myocarditis: an autopsy series. Archives of disease in childhood, 93(7), pp.594-598.

Weber, M.A., Ashworth, M.T., Risdon, R.A., Brooke, I., Malone, M. and Sebire, N.J., 2009. Sudden unexpected neonatal death in the first week of life: autopsy findings from a specialist centre. The Journal of Maternal-Fetal & Neonatal Medicine, 22(5), pp.398-404.

Weber, M.A., Klein, N.J., Hartley, J.C., Lock, P.E., Malone, M. and Sebire, N.J., 2008. Infection and sudden unexpected death in infancy: a systematic retrospective case review. The Lancet, 371(9627), pp.1848-1853.

Welcome to Python.org 2019, Python Software Foundation website, viewed 14th September, 2019, <http://www.python.org/>.

Wikipedia Contributors (2019). Entity Attribute Model. [online] Wikipedia. Available at: https://en.wikipedia.org/wiki/Entity-attribute-value_model *[Accessed 2019-03-24].*

Yang, Y., Morillo, I.G. and Hospedales, T.M., 2018. Deep neural decision trees. arXiv preprint arXiv:1806.06988.

Bibliography

Caruana, R. and Niculescu-Mizil, A., 2006, June. An empirical comparison of supervised learning algorithms. In Proceedings of the 23rd international conference on Machine learning (pp. 161-168). ACM.

Luellen, J.K., Shadish, W.R. and Clark, M.H., 2005. Propensity scores: An introduction and experimental test. *Evaluation Review*, *29*(6), pp.530-558.

James, G., Witten, D., Hastie, T. and Tibshirani, R., 2013. An introduction to statistical learning (Vol. 112, p. 18). New York: springer.

Quinlan, J.R., 1986. Induction of decision trees. Machine learning, 1(1), pp.81-106.

Glossary of Terms

From the Royal College of Pathologists:

https://www.rcpath.org/discover-pathology/what-is-pathology/glossary-of-terms.html

And Wikipedia:

https://en.wikipedia.org

Term	Description
Data Wrangling	the process of transforming and mapping data from one "raw" data form into another format with the intent of making it more appropriate and valuable for a variety of downstream purposes such as analytics.
Dysmorphic	A dysmorphic feature is a difference of body structure. It can be an isolated finding in an otherwise normal individual, or it can be related to a congenital disorder, genetic syndrome, or birth defect.
Entity Attribute Value (EAV) Schema	Entity–attribute–value model (EAV) is a data model to encode, in a space-efficient manner, entities where the number of attributes (properties, parameters) that can be used to describe them is potentially vast, but the number that will actually apply to a given entity is relatively modest.
ETL	The process of pulling data out of one database and place it into another database. Within this process the data is cleans and transformed to be in a more appropriate schema for analytics.
Histopathology	The branch of pathology that involves looking at tissue under the microscope to diagnose disease. If you have a mole or a breast lump removed, the histopathologist will examine it to work out what it is.
Metabolic	A group of overlapping areas of clinical practice with a common dependence on the detailed understanding of basic biochemistry and medicine. These areas fall within the territory of both physicians and chemical pathologists. They include clinical nutrition, lipid abnormalities, diabetes, metabolic bone disease, porphyria and adult inherited metabolic disorders.
Microbiology	The diagnosis of infection caused by bacteria, fungi, parasites and viruses; identification of the best treatment options for infection; and the monitoring of antibiotic resistance. It also includes testing for how well a patient is responding to treatment of infection.
Oedema	an abnormal accumulation of fluid in the interstitium, located beneath the skin and in the cavities of the body
Structured Query Language (SQL)	A domain-specific language used in programming and designed for managing data held in a relational database management system.

Appendix A – Example Project Code

Python Function with ODBC

```
def return_null_string(variable):
    if pandas.isnull(variable):
        return "NULL"
    else:
        return "'" + variable.replace("'","''") + "'"

def GetConceptID(cnxn, crsr, category, parent_concept_id, code, label = None, value_type_concept_id = 1):
    '''
    Gets Concept id
    If it doesn't exist then adds it.
    :param cnxn: ODBC Connection
    :param crsr: ODBC Cursor
    :param category: String
    :param code: string
    :param label: string - default blank as not required if concept exist
    :param value_type_concept_id: integer; default is concept id
    :return: integer; 0 if not been able to create and doesn't exist.
    '''

    # Create SQL string to find concept
    SQLstring = "SELECT "
    SQLstring += "  concept_id "
    SQLstring += "FROM "
    SQLstring += "  ha_concepts "
    SQLstring += "WHERE "
    SQLstring += "  category = '" + category + "' "
    SQLstring += "  AND code = '" + code + "'"
    SQLstring += ";"

    crsr.execute(SQLstring)

    row = crsr.fetchone()

    #If concept doesn't exist then insert it
    if not row:

        SQLinsert = "INSERT INTO ha_concepts "
        SQLinsert += "  (category, parent_concept_id, code, label, value_type_concept_id) "
        SQLinsert += "VALUES "
```

```
        SQLinsert += "  ('" + category + "', " + str(parent_concept_id) + ", '" + code + "', " + return_null_string(
            label) + ", " + str(value_type_concept_id) + ")"
        SQLinsert += ";"

        crsr.execute(SQLinsert)
        cnxn.commit()

        #Re execute SQL select
        crsr.execute(SQLstring)

        row = crsr.fetchone()

    if row:
        return row.concept_id
    else:
        return 0
```

Python Function to Create COD2_SUMM

```
def create_cod2_Summ_attribute_from_cod2_attribute(cnxn, crsr):
    '''
    :param cnxn:
    :param crsr:
    :return:

    'build an array of existing key values & labels from ha_concepts
    'Create an array of new attribute key values - add to ha_concepts
    'Assign new keys to old keys.
    'process all existing attributes
    '   add new event with mapped value

    ' code label
    ' 1  Unknown - No Abnormal Findings
    ' 2  Unknown - Non-Contributory Findings
    ' 3  Unknown - Possible/Probable Contributory Findings
    ' 4   Infection
    ' 5   CNS
    ' 6   Respiratory System
    ' 7   Cardiovascular System
    ' 8   GIT
    ' 9   Urogenital System
```

```
    '10  Lymphoreticular/Haematological
    '11  Musculoskeletal
    '12  Peripheral Nervous System/Neuromuscular Junction
    '13  Metabolic
    '14  Anaphylaxis
    '15  Immunological/Autoimmune
    '16  Neoplasia Benign
    '17  Neoplasia Malignant
    '18  Chromosomal Abnormality
    '19  Congenital Anomalies/Malformation Syndrome
    '20  Accidental
    '21  NAI/Homicide
    '22  Suicide
    '23  Wigglesworth 1
    '24  Wigglesworth 2
    '25  Wigglesworth 3
    '26  Wigglesworth 4
    '27  Wigglesworth 5
    '28  Normal Fetus
    '29  Traumatic NOS
    '30  Endocrine
    '32  No PM, For PM MRI ONLY
    '33  No PM, see Memo
    '34  Other
    '999 N/A

    'code label
    ' 1  Unknown (1 - 3)
    ' 2  Known   (4 - 30)
    ' 3  Other   (31 - 34)
    '999 N/A

    '''

    # Add Attributes and values
    parent_concept_id = Create_HAS_Tables.GetConceptID(cnxn, crsr, "/EventAttribute/Observation", None, "PostMortem")

    value_type_concept_id = Create_HAS_Tables.GetConceptID(cnxn, crsr, "/", None, "Concept")

    parent_concept_id = Create_HAS_Tables.GetConceptID(cnxn, crsr, "/EventAttribute/Observation/PostMortem",
                                                       parent_concept_id, "tblFinalDiagnoses", "tblFinalDiagnoses",
                                                       value_type_concept_id)

    event_attribute_type_concept_id = Create_HAS_Tables.GetConceptID(cnxn, crsr,
```

```
"/EventAttribute/Observation/PostMortem/tblFinalDiagnoses",
                                                                    parent_concept_id, "COD2_SUMM", "COD2_SUMM",
                                                                    value_type_concept_id)

    parent_concept_id = Create_HAS_Tables.GetConceptID(cnxn, crsr, "/EventAttribute/Observation/PostMortem", None,
                                                       "LookUp")
    parent_concept_id = Create_HAS_Tables.GetConceptID(cnxn, crsr, "/EventAttribute/Observation/PostMortem/LookUp",
                                                       parent_concept_id, "COD2_SUMM", "COD2_SUMM",
                                                       value_type_concept_id)

    value_concept_id = Create_HAS_Tables.GetConceptID(cnxn, crsr,
                                                      "/EventAttribute/Observation/PostMortem/LookUp/COD2_SUMM",
                                                      parent_concept_id, "001", "Unknown", value_type_concept_id)
    value_concept_id = Create_HAS_Tables.GetConceptID(cnxn, crsr,
                                                      "/EventAttribute/Observation/PostMortem/LookUp/COD2_SUMM",
                                                      parent_concept_id, "002", "known", value_type_concept_id)
    value_concept_id = Create_HAS_Tables.GetConceptID(cnxn, crsr,
                                                      "/EventAttribute/Observation/PostMortem/LookUp/COD2_SUMM",
                                                      parent_concept_id, "003", "Other", value_type_concept_id)
    value_concept_id = Create_HAS_Tables.GetConceptID(cnxn, crsr,
                                                      "/EventAttribute/Observation/PostMortem/LookUp/COD2_SUMM",
                                                      parent_concept_id, "994", "N/A", value_type_concept_id)

    # Get all Event Attributes of the original type

    # Get event_attributes for current event
    SQLstring = "SELECT "
    SQLstring += "  event_attribute_id, "  # TODO we seem to have a duplicate CaseID
    SQLstring += "  event_id, "
    SQLstring += "  event_attribute_type_concept_id, "
    SQLstring += "  value_text, "
    SQLstring += "  value_numeric, "
    SQLstring += "  value_datetime, "
    SQLstring += "  value_concept_id, "
    SQLstring += "  value_type_concept_id, "
    SQLstring += "  code, "
    SQLstring += "  label "
    SQLstring += "FROM ha_event_attributes "
    SQLstring += "  LEFT OUTER JOIN ha_concepts "
    SQLstring += "    ON ha_concepts.concept_id = ha_event_attributes.value_concept_id "
    SQLstring += "WHERE "
    SQLstring += "  (ha_concepts.category = '/EventAttribute/Observation/PostMortem/LookUp/COD2_COD2ID') "
    SQLstring += ";"
```

```
    crsr.execute(SQLstring)
    EventAttributeRows = crsr.fetchall()

    row = 0
    print("Processing Events - COD2_SUMM Attribute From COD2 Attribute")

    for EventAttributeRow in EventAttributeRows:

        row += 1
        sys.stdout.write("\r \r {0}".format(str(row)))
        sys.stdout.flush()

        if EventAttributeRow.code in ("001", "002", "003", "999"):
            code = "001"
            text = "Unknown"
        elif EventAttributeRow.code in ("031", "032", "033", "034"):
            code = "003"
            text = "Other"
        else:
            code = "002"
            text = "Known"

        Create_HAS_Tables.AddEventAttribute(cnxn, crsr, EventAttributeRow.event_id,
                                            "Observation/PostMortem/tblFinalDiagnoses", "COD2_SUMM", "COD2_SUMM", "ID",
                                            code, text)
    print("")
    print("Done!")
```

Python Function to Create RDV

```
def populate_event_attributes(cnxn, crsr, stage, EventAttributes):

    EventAttributes.append(
        Create_HAS_Tables.GetConceptID(cnxn, crsr, "/EventAttribute/Observation/PostMortem/tblCases", None,
                                       "CASEID"))
    EventAttributes.append(
        Create_HAS_Tables.GetConceptID(cnxn, crsr, "/EventAttribute/Observation/PostMortem/tblCases", None,
                                       "SSN"))
    EventAttributes.append(
        Create_HAS_Tables.GetConceptID(cnxn, crsr, "/EventAttribute/Observation/PostMortem/tblFinalDiagnoses", None,
                                       "COD2_SUMM"))

    if stage in ("ext_x","ext","int1","int2","int3","int2_s","int3_s"):

        EventAttributes.append(
            Create_HAS_Tables.GetConceptID(cnxn, crsr, "/EventAttribute/Observation/PostMortem/tblExternalExams", None,
                                           "BodyWeight"))
        EventAttributes.append(
            Create_HAS_Tables.GetConceptID(cnxn, crsr, "/EventAttribute/Observation/PostMortem/tblExternalExams", None,
                                           "CrownRumpLength"))
        EventAttributes.append(
            Create_HAS_Tables.GetConceptID(cnxn, crsr, "/EventAttribute/Observation/PostMortem/tblExternalExams", None,
                                           "HeadCircumference"))
        EventAttributes.append(
            Create_HAS_Tables.GetConceptID(cnxn, crsr, "/EventAttribute/Observation/PostMortem/tblExternalExams", None,
                                           "CrownRumpLength"))
        EventAttributes.append(
            Create_HAS_Tables.GetConceptID(cnxn, crsr, "/EventAttribute/Observation/PostMortem/tblExternalExams", None,
                                           "BodyLength"))
        EventAttributes.append(
            Create_HAS_Tables.GetConceptID(cnxn, crsr, "/EventAttribute/Observation/PostMortem/tblExternalExams", None,
                                           "FootLength"))
        EventAttributes.append(
            Create_HAS_Tables.GetConceptID(cnxn, crsr, "/EventAttribute/Observation/PostMortem/tblExternalExams", None,
                                           "Neglect_YNID"))
        EventAttributes.append(
            Create_HAS_Tables.GetConceptID(cnxn, crsr, "/EventAttribute/Observation/PostMortem/tblExternalExams", None,
                                           "Nutrition_NutnID"))
        EventAttributes.append(
            Create_HAS_Tables.GetConceptID(cnxn, crsr, "/EventAttribute/Observation/PostMortem/tblExternalExams", None,
                                           "DysmorphicFeatures_YNID"))
        EventAttributes.append(
```

```
            Create_HAS_Tables.GetConceptID(cnxn, crsr, "/EventAttribute/Observation/PostMortem/tblExternalExams", None,
                                           "Jaundice_YNID"))
        EventAttributes.append(
            Create_HAS_Tables.GetConceptID(cnxn, crsr, "/EventAttribute/Observation/PostMortem/tblExternalExams", None,
                                           "Oedema_YNID"))
        EventAttributes.append(
            Create_HAS_Tables.GetConceptID(cnxn, crsr, "/EventAttribute/Observation/PostMortem/tblExternalExams", None,
                                           "Pallor_YNID"))
        EventAttributes.append(
            Create_HAS_Tables.GetConceptID(cnxn, crsr, "/EventAttribute/Observation/PostMortem/tblExternalExams", None,
                                           "BloodAtMouth_BMID"))
        EventAttributes.append(
            Create_HAS_Tables.GetConceptID(cnxn, crsr, "/EventAttribute/Observation/PostMortem/tblExternalExams", None,
                                           "SignsOfTrauma_YNID"))
        EventAttributes.append(
            Create_HAS_Tables.GetConceptID(cnxn, crsr, "/EventAttribute/Observation/PostMortem/tblExternalExams", None,
                                           "SignsOfTreatment_YNID"))

    if stage in ("int1_x", "int1", "int2", "int3","int2_s","int3_s"):

        EventAttributes.append(
            Create_HAS_Tables.GetConceptID(cnxn, crsr, "/EventAttribute/Observation/PostMortem/tblInternalExams", None,
                                           "HeartWeight"))
        EventAttributes.append(
            Create_HAS_Tables.GetConceptID(cnxn, crsr, "/EventAttribute/Observation/PostMortem/tblInternalExams", None,
                                           "CombLungWeight"))
        EventAttributes.append(
            Create_HAS_Tables.GetConceptID(cnxn, crsr, "/EventAttribute/Observation/PostMortem/tblInternalExams", None,
                                           "LiverWeight"))
        EventAttributes.append(
            Create_HAS_Tables.GetConceptID(cnxn, crsr, "/EventAttribute/Observation/PostMortem/tblInternalExams", None,
                                           "PancreasWeight"))
        EventAttributes.append(
            Create_HAS_Tables.GetConceptID(cnxn, crsr, "/EventAttribute/Observation/PostMortem/tblInternalExams", None,
                                           "ThymusWeight"))
        EventAttributes.append(
            Create_HAS_Tables.GetConceptID(cnxn, crsr, "/EventAttribute/Observation/PostMortem/tblInternalExams", None,
                                           "SpleenWeight"))
        EventAttributes.append(
            Create_HAS_Tables.GetConceptID(cnxn, crsr, "/EventAttribute/Observation/PostMortem/tblInternalExams", None,
                                           "CombAdrenalWeight"))
        EventAttributes.append(
            Create_HAS_Tables.GetConceptID(cnxn, crsr, "/EventAttribute/Observation/PostMortem/tblInternalExams", None,
                                           "ThyroidWeight"))
```

```
        EventAttributes.append(
            Create_HAS_Tables.GetConceptID(cnxn, crsr, "/EventAttribute/Observation/PostMortem/tblInternalExams", None,
                                           "CombKidneyWeight"))
        EventAttributes.append(
            Create_HAS_Tables.GetConceptID(cnxn, crsr, "/EventAttribute/Observation/PostMortem/tblInternalExams", None,
                                           "BrainWeight"))

    if stage in ("int2_x","int2","int3"):

        EventAttributes.append(
            Create_HAS_Tables.GetConceptID(cnxn, crsr, "/EventAttribute/Observation/PostMortem", None,
                                           "CardiovascularMacro_SyFiID"))
        EventAttributes.append(
            Create_HAS_Tables.GetConceptID(cnxn, crsr, "/EventAttribute/Observation/PostMortem", None,
                                           "CentralNervousMacro_SyFiID"))
        EventAttributes.append(
            Create_HAS_Tables.GetConceptID(cnxn, crsr, "/EventAttribute/Observation/PostMortem", None,
                                           "EndocrineMacro_SyFiID"))
        EventAttributes.append(
            Create_HAS_Tables.GetConceptID(cnxn, crsr, "/EventAttribute/Observation/PostMortem", None,
                                           "GastrointestinalMacro_SyFiID"))
        EventAttributes.append(
            Create_HAS_Tables.GetConceptID(cnxn, crsr, "/EventAttribute/Observation/PostMortem", None,
                                           "RespiratoryMacro_SyFiID"))
        EventAttributes.append(
            Create_HAS_Tables.GetConceptID(cnxn, crsr, "/EventAttribute/Observation/PostMortem", None,
                                           "ReticuloendothelialMacro_SyFiID"))
        EventAttributes.append(
            Create_HAS_Tables.GetConceptID(cnxn, crsr, "/EventAttribute/Observation/PostMortem", None,
                                           "UrogenitalMacro_SyFiID"))

    if stage in ("int3_x","int3"):

        EventAttributes.append(
            Create_HAS_Tables.GetConceptID(cnxn, crsr, "/EventAttribute/Observation/PostMortem", None,
                                           "CardiovascularHisto_SyHiID"))
        EventAttributes.append(
            Create_HAS_Tables.GetConceptID(cnxn, crsr, "/EventAttribute/Observation/PostMortem", None,
                                           "CentralNervousHisto_SyHiID"))
        EventAttributes.append(
            Create_HAS_Tables.GetConceptID(cnxn, crsr, "/EventAttribute/Observation/PostMortem", None,
                                           "EndocrineHisto_SyHiID"))
        EventAttributes.append(
            Create_HAS_Tables.GetConceptID(cnxn, crsr, "/EventAttribute/Observation/PostMortem", None,
```

```
                                            "GastrointestinalHisto_SyHiID"))
        EventAttributes.append(
            Create_HAS_Tables.GetConceptID(cnxn, crsr, "/EventAttribute/Observation/PostMortem", None,
                                            "RespiratoryHisto_SyHiID"))
        EventAttributes.append(
            Create_HAS_Tables.GetConceptID(cnxn, crsr, "/EventAttribute/Observation/PostMortem", None,
                                            "ReticuloendothelialHisto_SyHiID"))
        EventAttributes.append(
            Create_HAS_Tables.GetConceptID(cnxn, crsr, "/EventAttribute/Observation/PostMortem", None,
                                            "UrogenitalHisto_SyHiID"))

    if stage in ("int2_s","int3_s"):

        EventAttributes.append(
            Create_HAS_Tables.GetConceptID(cnxn, crsr, "/EventAttribute/Observation/PostMortem", None,
                                            "CaseMacro_CsFiID"))
    if stage == "int3_s":

        EventAttributes.append(
            Create_HAS_Tables.GetConceptID(cnxn, crsr, "/EventAttribute/Observation/PostMortem", None,
                                            "CaseHisto_CsHiID"))
    return EventAttributes

def create_rdv_study(cnxn, crsr, stage, include_null = True):

    # Select Patient Attributes
    EventPatientAttributes = []
    EventPatientAttributes.append(Create_HAS_Tables.GetConceptID(cnxn, crsr, "/PatientAttribute", None, "AC")) # Age Category
    EventPatientAttributes.append(Create_HAS_Tables.GetConceptID(cnxn, crsr, "/PatientAttribute", None, "AG")) # Age in Days
    EventPatientAttributes.append(Create_HAS_Tables.GetConceptID(cnxn, crsr, "/PatientAttribute", None, "GA")) # Gestational
Age at delivery

    # Select Patient Attribute Filters
    EventPatientAttributeFilters = []
    EventPatientAttributeFilterValues = []

    # Select Event Attributes
    EventAttributes = populate_event_attributes(cnxn, crsr, stage,[])

    # Is this necessary if measurements were made we could use them.
    EventAttributeFilters = []
    EventAttributeFilters.append(Create_HAS_Tables.GetConceptID(cnxn, crsr, "/EventAttribute/Observation/PostMortem",
None,"INC_IN_STUDY"))
    EventAttributeFilterValues = []
```

```
    EventAttributeFilterValues.append(Create_HAS_Tables.GetConceptID(cnxn, crsr,
"/EventAttribute/Observation/PostMortem/LookUp/INC_IN_STUDY", None, "001"))

    file_name = "rdv_" + "study_" + stage

    create_rdv(cnxn, crsr, file_name, EventPatientAttributes, EventPatientAttributeFilters, EventPatientAttributeFilterValues,
EventAttributes, EventAttributeFilters, EventAttributeFilterValues)
```

R Model Function

```
# Load libraries
library(dplyr)
library(ggplot2)
library(ggmosaic)
library(grid)
library(gridExtra)
library(gtable)
library(viridis)
library(rpart)
library(rpart.plot)
library(caret)
library(lubridate)
library(reshape2)
# xgboost added:
library(xgboost)
library('DiagrammeR') # NB installed package
library('rsvg') # NB installed package
library('DiagrammeRsvg') # NB installed package

# run.seed - random seed for this run
# rdv.type, - rdv type adjusted or not adjusted (_adj)
# importance.min, - min importance for feature importance plots (1.5)
# source.dir,  - location of RDV files
# results.sub.dir, - location to store results
# file.suffix, - to distinguidh the files for this run
# stage.list, - list of post-mortem stages
# ext.train.index, - training index for external stage
# int1.train.index,
# int2.train.index,
# int3.train.index

RunXGBModel <- function(run.seed,
                        rdv.type,
                        importance.min,
                        source.dir,
                        results.sub.dir,
                        file.suffix,
                        stage.list,
                        ext.train.index,
                        int1.train.index,
                        int2.train.index,
                        int3.train.index
```

```
                        ) {

  set.seed(run.seed)

  model.name = "XGBoost Tree"
  model.abv = "xgb"

  run.str <- substr(file.suffix, nchar(file.suffix) - 1, nchar(file.suffix))

  fimp.matrix <- setup.fimp.matrix(rdv.type, source.dir, run.str)

  results.matrix <- setup.results.matrix(model.abv,length(stage.list))

  for(stage.num in 1:length(stage.list)) {

    stage <- stage.list[stage.num]

    rm.col <- 1

    print(paste0("Run: ", run.str, " Model: ",model.name," Stage: ",stage))

    now <- Sys.time()

    results.matrix[stage.num,rm.col] = run.str
    rm.col = rm.col + 1
    results.matrix[stage.num,rm.col] = format(now, "%Y-%m-%d %H:%M:%S")
    rm.col = rm.col + 1
    results.matrix[stage.num,rm.col] = rdv.type
    rm.col = rm.col + 1
    results.matrix[stage.num,rm.col] = run.seed
    rm.col = rm.col + 1
    results.matrix[stage.num,rm.col] = stage
    rm.col = rm.col + 1

    clean_RDVData <- return_clean_rdvdata(source.dir, stage, rdv.type)

    results.matrix[stage.num,rm.col] = nrow(clean_RDVData)
    rm.col = rm.col + 1

    clean_RDVData$cod2_summ <- as.factor(clean_RDVData$cod2_summ)


    ##########################################################
    # XGBoost
```

```
xgb.data <- clean_RDVData

num_class = length(levels(xgb.data$cod2_summ))
cod2_summ = clean_RDVData$cod2_summ

# Convert from class to numeric
label <- as.integer(xgb.data$cod2_summ) - 1
xgb.data$cod2_summ = NULL

if (stage == "ext") {
  train.index <- ext.train.index
} else if (stage == "int1") {
  train.index <- int1.train.index
} else if (stage == "int2") {
  train.index <- int2.train.index
} else if (stage == "int3") {
  train.index <- int3.train.index
}

train.data = as.matrix(xgb.data[train.index,])
train.label = label[train.index]
test.data = as.matrix(xgb.data[-train.index,])
test.label = label[-train.index]

# Store proportional split of COD2_SUMM for this run
results.matrix[stage.num,rm.col] = prop.table(table(train.label))[1]
rm.col = rm.col + 1
results.matrix[stage.num,rm.col] = prop.table(table(train.label))[2]
rm.col = rm.col + 1
results.matrix[stage.num,rm.col] = prop.table(table(test.label))[1]
rm.col = rm.col + 1
results.matrix[stage.num,rm.col] = prop.table(table(test.label))[2]
rm.col = rm.col + 1

# Transform the two data sets into xgb.Matrix
xgb.train = xgb.DMatrix(data=train.data,label=train.label)
xgb.test = xgb.DMatrix(data=test.data,label=test.label)

# Stored tuned model results

if (stage == "ext") {
  eta.value <- 0.3
  max_depth.value <- 6
```

```
      gamma.value <- 5
      min_child_weight.value <- 4
      subsample.value <- 0.75
      colsample_bytree.value <- 0.50
    } else if (stage == "int1") {
      eta.value <- 0.185
      max_depth.value <- 6
      gamma.value <- 2.65
      min_child_weight.value <- 8
      subsample.value <- 0.725
      colsample_bytree.value <- 0.50
    } else if (stage == "int2") {
      eta.value <- 0.2
      max_depth.value <- 5
      gamma.value <- 1
      min_child_weight.value <- 5
      subsample.value <- 0.775
      colsample_bytree.value <- 0.5
    } else if (stage == "int3") {
      eta.value <- 0.17
      max_depth.value <- 5
      gamma.value <- 3.2
      min_child_weight.value <- 4
      subsample.value <- 0.65
      colsample_bytree.value <- 0.525
    } else {
      # default values
      eta.value <- 0.3
      max_depth.value <- 6
      gamma.value <- 0
      min_child_weight.value <- 1
      subsample.value <- 1
      colsample_bytree.value <- 1
    }

    #Store best values
    results.matrix[stage.num,rm.col] = eta.value
    rm.col = rm.col + 1
    results.matrix[stage.num,rm.col] = max_depth.value
    rm.col = rm.col + 1
    results.matrix[stage.num,rm.col] = gamma.value
    rm.col = rm.col + 1
    results.matrix[stage.num,rm.col] = min_child_weight.value
    rm.col = rm.col + 1
```

```
    results.matrix[stage.num,rm.col] = subsample.value
    rm.col = rm.col + 1
    results.matrix[stage.num,rm.col] = colsample_bytree.value
    rm.col = rm.col + 1

    params=list(
      booster="gbtree",
      eta=eta.value,
      max_depth=max_depth.value,
      gamma=gamma.value,
      min_child_weight=min_child_weight.value,
      subsample=subsample.value,
      colsample_bytree=colsample_bytree.value,
      objective="multi:softprob",
      eval_metric="mlogloss",
      num_class=num_class
    )

    # Train the XGBoost classifer
    xgb.fit=xgb.train(
      params=params,
      data=xgb.train,
      nrounds=1000,
      nthreads=1,
      early_stopping_rounds=10,
      watchlist=list(val1=xgb.train,val2=xgb.test),
      verbose=0
    )

    # Review the final model and results
    # xgb.fit

    # Predict outcomes with the test data
    xgb.pred = predict(xgb.fit,test.data,reshape=T)
    xgb.pred = as.data.frame(xgb.pred)
    colnames(xgb.pred) = levels(cod2_summ)

    # Use the predicted label with the highest probability
    xgb.pred$prediction = apply(xgb.pred,1,function(x) colnames(xgb.pred)[which.max(x)])
    xgb.pred$label = levels(cod2_summ)[test.label + 1]

    # Calculate the final accuracy
    result = sum(xgb.pred$prediction==xgb.pred$label)/nrow(xgb.pred)
```

```
    print(result)

    results.matrix[stage.num,rm.col] = result
    rm.col = rm.col + 1

    # Create confusion matrix
    table_mat <- table(xgb.pred$label, xgb.pred$prediction)

    # Loop over my_matrix
    for(row in 1:nrow(table_mat)) {
      for(col in 1:ncol(table_mat)) {
        results.matrix[stage.num,rm.col] = table_mat[row, col]
        rm.col = rm.col + 1
      }
    }

    importance <- xgb.importance(model = xgb.fit)
    imp <- as.data.frame(importance)

    total_imp = sum(imp$Gain)

    for (imp_row in 1:nrow(imp)){
      res_row = which(fimp.matrix$feature == imp[imp_row,1])
      fimp.matrix[res_row, stage.num + 2] <- (imp[imp_row,2] / total_imp) * 100
      imp[imp_row,2]  <- (imp[imp_row,2] / total_imp) * 100
    }

    # Remove less import features for clarity
    imp <- subset(imp, Gain > importance.min)

    plot.title = paste0("Relative Feature Importance - Model: ",model.name,", Stage: ",stage)

    p <- ggplot(imp, aes(x=reorder(Feature, Gain), y=Gain))
    p <- p + geom_point()
    p <- p + geom_segment(aes(x=Feature,xend=Feature,y=0,yend=Gain))
    p <- p + ggtitle(plot.title)
    p <- p + ylab("Relative Importance")
    p <- p + xlab("Feature")
    p <- p + coord_flip()
    p <- p + theme_classic()

    print(p)

    ggsave(paste0(results.sub.dir, "/", model.abv, "_feature_importance_",stage, file.suffix,".png"))
```

```
    # Single tree plot

    p <- xgb.plot.tree(model = xgb.fit, trees = 0, show_node_id = TRUE)

    print(p)

    gr <- xgb.plot.tree(model=xgb.fit, trees=0, show_node_id = TRUE, render=FALSE)
    export_graph(gr, paste0(results.sub.dir, "/", model.abv, "_tree_",stage, file.suffix,".png"), width=1500, height=1900)

  }

  ##############################
  ## graph combined importance
  ##############################

  data <- fimp.matrix
  # Order results
  data$feature <- with(data, reorder(feature, ext + int1 + int2 + int3))
  # Remove 0 values and create structure to plot
  data.m.ss <- subset(melt(data), value > importance.min)
  # Create plot
  plot.title = paste0("Relative Feature Importance Heatmap - Model: ",model.name)
  p <- ggplot(data.m.ss, aes(x=variable, y=feature))
  p <- p + ggtitle(plot.title)
  p <- p + geom_tile(aes(fill = value))
  p <- p + scale_fill_viridis_c(direction = -1, begin = .3, end = 1)
  p <- p + geom_text(aes(label = round(value, 1)))
  p <- p + theme_classic()

  print(p)

  ggsave(paste0(results.sub.dir, "/", model.abv, "_feature_importance_hm", file.suffix, ".png"))

  ##############################
  ## output results CSV files
  ##############################

  write.csv(results.matrix, file = paste0(results.sub.dir, "/", model.abv, "_results_matrix", file.suffix,
".csv"),row.names=FALSE, na="")
  write.csv(fimp.matrix, file = paste0(results.sub.dir, "/", model.abv, "_feature_importance_matrix", file.suffix,
".csv"),row.names=FALSE, na="")

    file.text <- "_results_matrix"
```

```
  file.suffix <- sprintf("_%02d", run.num)

  p.list <- list()

  for(stage.num in 1:length(stage.list)) {

    stage.abv <- stage.list[stage.num]

    save_confusion_matrix_plot(model.abv, model.name, stage.abv, file.text, results.sub.dir, file.suffix)

    p.list[[stage.num]] <- plot_confusion_matrix_plot(model.abv, model.name, stage.abv, file.text, results.sub.dir,
file.suffix)

  }

  g <- do.call(grid.arrange,p.list)

  ggsave(paste0(results.sub.dir, "/", model.abv, "_confusion_matrix_grid_",stage.abv, file.suffix,".png"),g)

  ##################################

}
```

Appendix B – ETL Process

The ETL process that extracts data from the post-mortem research database, transforms it to the EAV schema and then loads it into the HAS tables is the most extensive piece of python code developed for this project and its success is the foundation for all the analytics that follows.

There follows a detailed break-down of the functions and procedures developed for this process:

Create_HAS_Tables.py

- create_has_tables
    - DROPs and CREATEs HAS Tables in Staging database
    - Creates base concepts
        - Value types
        - Staff types
            - Consultant type
        - Event types & event attribute root types
        - Patients & patient attributes
- runTests
    - Completes unit testing on the following procedures and functions:
        - GetConceptID
            - Checks whether concepts already exists based on category and code if it doesn't then the functions adds the concept.
            - Return concept ID
        - GetStaffID
            - Checks whether staff entity already exists based on staff type code if it doesn't then the functions adds the staff entity.
            - Return staff ID
        - GetPatientID
            - Checks whether patient entity already exists based on their health identity number if it doesn't then the functions adds the patient entity.
            - Return staff ID
        - GetEventID
            - Inserts event and returns ID
        - GetEventAttributeID
            - Returns event id based on event id, concept category and code.
        - AddPatientAttribute
            - Adds patient attribute value for a given patient id and patient attribute code.
            - Option to only add attribute if it doesn't already exist.
        - AddEventAttribute
            - Adds event attribute value for a given event id and event attribute code
        - BuildEventsSQL
            - For a given table in the source database this function builds up the SQL that links the required table back through the levels to the case table.
        - GetAgeCategory

- This function takes a combination of gestational age in days, age in days and referral text and returns an appropriate age category:
  - Miscarriage/Termination of pregnancy (TOP) – less than 24 weeks
  - Still birth – Greater than 24 weeks
  - Early neonatal – Less than 7 days
  - Neonatal – 7 to 28 days
  - Infant – Less than a year
  - Child

- GetPMYear
  - This function returns the year of the post-mortem based on the case number.

- CreateEvents
  - Create root event attribute for each table in source database
  - Get list of all events with basic case, patient and staff information required to create an event
  - For each case
    - Check staff entity is created
    - Check patient entity is created
      - Add patient attributes
    - Create an event
    - Add case event attributes
    - CreateEvent Attributes
      - For each table in source database
        - For each column
          - Create event attribute
            - Depending on type of column do basic checks on data to ensure that it's valid
            - If the column is a categorical variable then ensure that the appropriate lookup concepts are in place.

Appendix C – Cause of Death Attribute Mapping

The analytics for this project is based on predicting whether a cause of death can be determined or not. For the majority of post-mortem cases the cause of death is recorded using a lookup table coded as COD2. COD2 allows the consultant to define a number of different variants for each situation of whether the cause of death is determined or not. The table that follows is how the various variants where mapped to a summary lookup category COD2_SUMM:

Detail			Summary	
category	code	label	code	label
\Event Attribute Type\Post Mortem\Look Up\COD2_COD2ID	001	Unknown - No Abnormal Findings	001	Unknown
\Event Attribute Type\Post Mortem\Look Up\COD2_COD2ID	002	Unknown - Non-Contributory Findings	001	Unknown
\Event Attribute Type\Post Mortem\Look Up\COD2_COD2ID	003	Unknown - Possible/Probable Contributory Findings	001	Unknown
\Event Attribute Type\Post Mortem\Look Up\COD2_COD2ID	004	Infection	002	Known
\Event Attribute Type\Post Mortem\Look Up\COD2_COD2ID	005	CNS	002	Known
\Event Attribute Type\Post Mortem\Look Up\COD2_COD2ID	006	Respiratory System	002	Known
\Event Attribute Type\Post Mortem\Look Up\COD2_COD2ID	007	Cardiovascular System	002	Known
\Event Attribute Type\Post Mortem\Look Up\COD2_COD2ID	008	GIT	002	Known
\Event Attribute Type\Post Mortem\Look Up\COD2_COD2ID	009	Urogenital System	002	Known
\Event Attribute Type\Post Mortem\Look Up\COD2_COD2ID	010	Lymphoreticular/Haematological	002	Known
\Event Attribute Type\Post Mortem\Look Up\COD2_COD2ID	011	Musculoskeletal	002	Known
\Event Attribute Type\Post Mortem\Look Up\COD2_COD2ID	012	Peripheral Nervous System/Neuromuscular Junction	002	Known
\Event Attribute Type\Post Mortem\Look Up\COD2_COD2ID	013	Metabolic	002	Known
\Event Attribute Type\Post Mortem\Look Up\COD2_COD2ID	014	Anaphylaxis	002	Known
\Event Attribute Type\Post Mortem\Look Up\COD2_COD2ID	015	Immunological/Autoimmune	002	Known
\Event Attribute Type\Post Mortem\Look Up\COD2_COD2ID	016	Neoplasia Benign	002	Known
\Event Attribute Type\Post Mortem\Look Up\COD2_COD2ID	017	Neoplasia Malignant	002	Known
\Event Attribute Type\Post Mortem\Look Up\COD2_COD2ID	018	Chromosomal Abnormality	002	Known
\Event Attribute Type\Post Mortem\Look Up\COD2_COD2ID	019	Congenital Anomalies/Malformation Syndrome	002	Known
\Event Attribute Type\Post Mortem\Look Up\COD2_COD2ID	020	Accidental	002	Known
\Event Attribute Type\Post Mortem\Look Up\COD2_COD2ID	021	NAI/Homicide	002	Known

\Event Attribute Type\Post Mortem\Look Up\COD2_COD2ID	022	Suicide	002	Known
\Event Attribute Type\Post Mortem\Look Up\COD2_COD2ID	023	Wigglesworth 1	002	Known
\Event Attribute Type\Post Mortem\Look Up\COD2_COD2ID	024	Wigglesworth 2	002	Known
\Event Attribute Type\Post Mortem\Look Up\COD2_COD2ID	025	Wigglesworth 3	002	Known
\Event Attribute Type\Post Mortem\Look Up\COD2_COD2ID	026	Wigglesworth 4	002	Known
\Event Attribute Type\Post Mortem\Look Up\COD2_COD2ID	027	Wigglesworth 5	002	Known
\Event Attribute Type\Post Mortem\Look Up\COD2_COD2ID	028	Normal Fetus	002	Known
\Event Attribute Type\Post Mortem\Look Up\COD2_COD2ID	029	Traumatic NOS	002	Known
\Event Attribute Type\Post Mortem\Look Up\COD2_COD2ID	030	Endocrine	002	Known
\Event Attribute Type\Post Mortem\Look Up\COD2_COD2ID	032	No PM (For PM MRI ONLY)	003	Other
\Event Attribute Type\Post Mortem\Look Up\COD2_COD2ID	033	No PM (see Memo)	003	Other
\Event Attribute Type\Post Mortem\Look Up\COD2_COD2ID	034	Other	003	Other
\Event Attribute Type\Post Mortem\Look Up\COD2_COD2ID	999	N/A	999	N/A

Appendix D –RDV structures

For each stage of the post-mortem procedure an RDV was produced firstly in a standard format and then an adjusted format with one-hot encoding and normalisation. In this appendix there is the details of each format plus where appropriate the look-up vales of each categorical feature.

Each stage only details the new features added in at that stage the complete RDV's would include all the features from any previous stages.

External

Original Columns		
Name	Type	Description
event_id	integer	Event id
event_start_date	timestamp	Event start date
sex	text	Sex
age_category	text	Age category
age_in_days	integer	Age in days
gestation_at_delivery_in_days	integer	Gestation at delivery in days
case_id	integer	Case id
season	text	Season
body_weight	double precision	Body weight (gm)
head_circumference	double precision	Head circumference
crown_rump_length	double precision	Crown rump length (mm)
body_length	double precision	Body length (mm)
foot_length	double precision	Foot length (mm)
neglect_ynid	text	Neglect
nutrition_nutn_id	text	Nutrition nutn id
dysmorphic_features_ynid	text	Dysmorphic features
jaundice_ynid	text	Jaundice
oedema_ynid	text	Oedema
pallor_ynid	text	Pallor
blood_at_mouth_bmid	text	Blood at mouth bmid
signs_of_trauma_ynid	text	Signs of trauma
signs_of_treatment_ynid	text	Signs of treatment
cod2_summ	text	Cause of death summary

include_in_study	text	Include in study

Lookup Values		
category	code	label
/EventAttribute/Observation/PostMortem/LookUp/COD2_SUMM	001	Unknown
/EventAttribute/Observation/PostMortem/LookUp/COD2_SUMM	002	known
/EventAttribute/Observation/PostMortem/LookUp/COD2_SUMM	003	Other
/EventAttribute/Observation/PostMortem/LookUp/COD2_SUMM	994	N/A
/EventAttribute/Observation/PostMortem/LookUp/DysmorphicFeatures_YNID	001	Yes
/EventAttribute/Observation/PostMortem/LookUp/DysmorphicFeatures_YNID	002	No
/EventAttribute/Observation/PostMortem/LookUp/DysmorphicFeatures_YNID	999	N/A
/EventAttribute/Observation/PostMortem/LookUp/Jaundice_YNID	001	Yes
/EventAttribute/Observation/PostMortem/LookUp/Jaundice_YNID	002	No
/EventAttribute/Observation/PostMortem/LookUp/Jaundice_YNID	999	N/A
/EventAttribute/Observation/PostMortem/LookUp/Neglect_YNID	001	Yes
/EventAttribute/Observation/PostMortem/LookUp/Neglect_YNID	002	No
/EventAttribute/Observation/PostMortem/LookUp/Neglect_YNID	999	N/A
/EventAttribute/Observation/PostMortem/LookUp/Nutrition_NutnID	001	Well-nourished
/EventAttribute/Observation/PostMortem/LookUp/Nutrition_NutnID	002	Slim / Thin
/EventAttribute/Observation/PostMortem/LookUp/Nutrition_NutnID	003	Wasted
/EventAttribute/Observation/PostMortem/LookUp/Nutrition_NutnID	006	Plump
/EventAttribute/Observation/PostMortem/LookUp/Nutrition_NutnID	007	Overweight
/EventAttribute/Observation/PostMortem/LookUp/Nutrition_NutnID	008	Obese
/EventAttribute/Observation/PostMortem/LookUp/Nutrition_NutnID	009	Poorly nourished NOS
/EventAttribute/Observation/PostMortem/LookUp/Nutrition_NutnID	999	N/A
/EventAttribute/Observation/PostMortem/LookUp/Oedema_YNID	001	Yes
/EventAttribute/Observation/PostMortem/LookUp/Oedema_YNID	002	No
/EventAttribute/Observation/PostMortem/LookUp/Oedema_YNID	999	N/A
/EventAttribute/Observation/PostMortem/LookUp/Pallor_YNID	001	Yes
/EventAttribute/Observation/PostMortem/LookUp/Pallor_YNID	002	No
/EventAttribute/Observation/PostMortem/LookUp/Pallor_YNID	999	N/A

Adjusted columns		
Name	Type	Description
event_id	integer	Event id
event_start_date	timestamp	Event start date
sex_f	integer	Sex (f)
sex_m	integer	Sex (m)
sex_u	integer	Sex (u)
age_category	text	Age category
age_in_days	integer	Age in days
gestation_at_delivery_in_days	integer	Gestation at delivery in days
case_id	integer	Case id
season_c001	integer	Season (Spring)
season_c002	integer	Season (Summer)
season_c003	integer	Season (Winter)
season_c004	integer	Season (Autumn)
season_nan	integer	Season (nan)
body_weight	double precision	Body weight (normalised)
head_circumference	double precision	Head circumference
crown_rump_length	double precision	Crown rump length (normalised)
body_length	double precision	Body length (normalised)
foot_length	double precision	Foot length (normalised)
neglect_ynid_c001	integer	Neglect (yes)
neglect_ynid_c002	integer	Neglect (no)
neglect_ynid_c999	integer	Neglect (N/A)
neglect_ynid_nan	integer	Neglect (NaN)
nutrition_nutn_id_c001	integer	Nutrition nutn id c001
nutrition_nutn_id_c002	integer	Nutrition nutn id c002
nutrition_nutn_id_c003	integer	Nutrition nutn id c003
nutrition_nutn_id_c006	integer	Nutrition nutn id c006
nutrition_nutn_id_c007	integer	Nutrition nutn id c007
nutrition_nutn_id_c008	integer	Nutrition nutn id c008
nutrition_nutn_id_c009	integer	Nutrition nutn id c009

nutrition_nutn_id_c999	integer	Nutrition nutn id c999
nutrition_nutn_id_nan	integer	Nutrition nutn id nan
dysmorphic_features_ynid_c001	integer	Dysmorphic features (yes)
dysmorphic_features_ynid_c002	integer	Dysmorphic features (no)
dysmorphic_features_ynid_c999	integer	Dysmorphic features (N/A)
dysmorphic_features_ynid_nan	integer	Dysmorphic features (NaN)
jaundice_ynid_c001	integer	Jaundice (yes)
jaundice_ynid_c002	integer	Jaundice (no)
jaundice_ynid_c999	integer	Jaundice (N/A)
jaundice_ynid_nan	integer	Jaundice (NaN)
oedema_ynid_c001	integer	Oedema (yes)
oedema_ynid_c002	integer	Oedema (no)
oedema_ynid_c999	integer	Oedema (N/A)
oedema_ynid_nan	integer	Oedema (NaN)
pallor_ynid_c001	integer	Pallor (yes)
pallor_ynid_c002	integer	Pallor (no)
pallor_ynid_c999	integer	Pallor (N/A)
pallor_ynid_nan	integer	Pallor (NaN)
blood_at_mouth_bmid_c001	integer	Blood at mouth bmid c001
blood_at_mouth_bmid_c002	integer	Blood at mouth bmid c002
blood_at_mouth_bmid_c003	integer	Blood at mouth bmid c003
blood_at_mouth_bmid_nan	integer	Blood at mouth bmid nan
signs_of_trauma_ynid_c001	integer	Signs of trauma (yes)
signs_of_trauma_ynid_c002	integer	Signs of trauma (no)
signs_of_trauma_ynid_c999	integer	Signs of trauma (N/A)
signs_of_trauma_ynid_nan	integer	Signs of trauma (NaN)
signs_of_treatment_ynid_c001	integer	Signs of treatment (yes)
signs_of_treatment_ynid_c002	integer	Signs of treatment (no)
signs_of_treatment_ynid_nan	integer	Signs of treatment (NaN)
cod2_summ	text	Cause of death summary
include_in_study	text	Include in study

Internal – Stage 1 – Organs

Original columns		
Name	Type	Description
heart_weight	double precision	Heart weight (gm)
comb_lung_weight	double precision	Comb lung weight (gm)
liver_weight	double precision	Liver weight (gm)
pancreas_weight	double precision	Pancreas weight (gm)
thymus_weight	double precision	Thymus weight (gm)
spleen_weight	double precision	Spleen weight (gm)
comb_adrenal_weight	double precision	Comb adrenal weight (gm)
thyroid_weight	double precision	Thyroid weight (gm)
comb_kidney_weight	double precision	Comb kidney weight (gm)
brain_weight	double precision	Brain weight (gm)

Adjusted columns		
Name	Type	Description
heart_weight	double precision	Heart weight (normalised)
comb_lung_weight	double precision	Comb lung weight (normalised)
liver_weight	double precision	Liver weight (normalised)
pancreas_weight	double precision	Pancreas weight (normalised)
thymus_weight	double precision	Thymus weight (normalised)
spleen_weight	double precision	Spleen weight (normalised)
comb_adrenal_weight	double precision	Comb adrenal weight (normalised)
thyroid_weight	double precision	Thyroid weight (normalised)
comb_kidney_weight	double precision	Comb kidney weight (normalised)
brain_weight	double precision	Brain weight (normalised)

Internal – Stage 2 – Macro investigation

Original columns		
Name	Type	Description
cardiovascular_macro_sy_fi_id	text	Cardiovascular macro investigation
central_nervous_macro_sy_fi_id	text	Central nervous macro investigation
endocrine_macro_sy_fi_id	text	Endocrine macro investigation
gastrointestinal_macro_sy_fi_id	text	Gastrointestinal macro investigation
respiratory_macro_sy_fi_id	text	Respiratory macro investigation
reticuloendothelial_macro_sy_fi_id	text	Reticuloendothelial macro investigation
urogenital_macro_sy_fi_id	text	Urogenital macro investigation

Lookup values		
category	code	label
/EventAttribute/Observation/PostMortem/LookUp/CardiovascularMacro_SyFiID	001	Normal
/EventAttribute/Observation/PostMortem/LookUp/CardiovascularMacro_SyFiID	002	Abnormal but NOT contributed to death
/EventAttribute/Observation/PostMortem/LookUp/CardiovascularMacro_SyFiID	003	Abnormal and cause of death
/EventAttribute/Observation/PostMortem/LookUp/CardiovascularMacro_SyFiID	999	N/A
/EventAttribute/Observation/PostMortem/LookUp/CentralNervousMacro_SyFiID	001	Normal
/EventAttribute/Observation/PostMortem/LookUp/CentralNervousMacro_SyFiID	002	Abnormal but NOT contributed to death
/EventAttribute/Observation/PostMortem/LookUp/CentralNervousMacro_SyFiID	003	Abnormal and cause of death
/EventAttribute/Observation/PostMortem/LookUp/CentralNervousMacro_SyFiID	999	N/A
/EventAttribute/Observation/PostMortem/LookUp/EndocrineMacro_SyFiID	001	Normal
/EventAttribute/Observation/PostMortem/LookUp/EndocrineMacro_SyFiID	002	Abnormal but NOT contributed to death
/EventAttribute/Observation/PostMortem/LookUp/EndocrineMacro_SyFiID	003	Abnormal and cause of death
/EventAttribute/Observation/PostMortem/LookUp/EndocrineMacro_SyFiID	999	N/A
/EventAttribute/Observation/PostMortem/LookUp/GastrointestinalMacro_SyFiID	001	Normal
/EventAttribute/Observation/PostMortem/LookUp/GastrointestinalMacro_SyFiID	002	Abnormal but NOT contributed to death
/EventAttribute/Observation/PostMortem/LookUp/GastrointestinalMacro_SyFiID	003	Abnormal and cause of death
/EventAttribute/Observation/PostMortem/LookUp/GastrointestinalMacro_SyFiID	999	N/A
/EventAttribute/Observation/PostMortem/LookUp/RespiratoryMacro_SyFiID	001	Normal
/EventAttribute/Observation/PostMortem/LookUp/RespiratoryMacro_SyFiID	002	Abnormal but NOT contributed to death

/EventAttribute/Observation/PostMortem/LookUp/RespiratoryMacro_SyFiID	003	Abnormal and cause of death
/EventAttribute/Observation/PostMortem/LookUp/RespiratoryMacro_SyFiID	999	N/A
/EventAttribute/Observation/PostMortem/LookUp/ReticuloendothelialMacro_SyFiID	001	Normal
/EventAttribute/Observation/PostMortem/LookUp/ReticuloendothelialMacro_SyFiID	002	Abnormal but NOT contributed to death
/EventAttribute/Observation/PostMortem/LookUp/ReticuloendothelialMacro_SyFiID	003	Abnormal and cause of death
/EventAttribute/Observation/PostMortem/LookUp/ReticuloendothelialMacro_SyFiID	999	N/A
/EventAttribute/Observation/PostMortem/LookUp/UrogenitalMacro_SyFiID	001	Normal
/EventAttribute/Observation/PostMortem/LookUp/UrogenitalMacro_SyFiID	002	Abnormal but NOT contributed to death
/EventAttribute/Observation/PostMortem/LookUp/UrogenitalMacro_SyFiID	003	Abnormal and cause of death
/EventAttribute/Observation/PostMortem/LookUp/UrogenitalMacro_SyFiID	999	N/A

Adjusted columns		
Name	Type	Description
cardiovascular_macro_sy_fi_id_c001	integer	Cardiovascular macro investigation (Normal)
cardiovascular_macro_sy_fi_id_c002	integer	Cardiovascular macro investigation (Abnormal but NOT contributed to death)
cardiovascular_macro_sy_fi_id_c003	integer	Cardiovascular macro investigation (Abnormal)
cardiovascular_macro_sy_fi_id_c999	integer	Cardiovascular macro investigation (N/A)
cardiovascular_macro_sy_fi_id_nan	integer	Cardiovascular macro investigation (NaN)
central_nervous_macro_sy_fi_id_c001	integer	Central nervous macro investigation (Normal)
central_nervous_macro_sy_fi_id_c002	integer	Central nervous macro investigation (Abnormal but NOT contributed to death)
central_nervous_macro_sy_fi_id_c003	integer	Central nervous macro investigation (Abnormal)
central_nervous_macro_sy_fi_id_c999	integer	Central nervous macro investigation (N/A)
central_nervous_macro_sy_fi_id_nan	integer	Central nervous macro investigation (NaN)
endocrine_macro_sy_fi_id_c001	integer	Endocrine macro investigation (Normal)
endocrine_macro_sy_fi_id_c002	integer	Endocrine macro investigation (Abnormal but NOT contributed to death)
endocrine_macro_sy_fi_id_c003	integer	Endocrine macro investigation (Abnormal)
endocrine_macro_sy_fi_id_c999	integer	Endocrine macro investigation (N/A)
endocrine_macro_sy_fi_id_nan	integer	Endocrine macro investigation (NaN)
gastrointestinal_macro_sy_fi_id_c001	integer	Gastrointestinal macro investigation (Normal)
gastrointestinal_macro_sy_fi_id_c002	integer	Gastrointestinal macro investigation (Abnormal but NOT contributed to death)
gastrointestinal_macro_sy_fi_id_c003	integer	Gastrointestinal macro investigation (Abnormal)

gastrointestinal_macro_sy_fi_id_c999	integer	Gastrointestinal macro investigation (N/A)
gastrointestinal_macro_sy_fi_id_nan	integer	Gastrointestinal macro investigation (NaN)
respiratory_macro_sy_fi_id_c001	integer	Respiratory macro investigation (Normal)
respiratory_macro_sy_fi_id_c002	integer	Respiratory macro investigation (Abnormal but NOT contributed to death)
respiratory_macro_sy_fi_id_c003	integer	Respiratory macro investigation (Abnormal)
respiratory_macro_sy_fi_id_c999	integer	Respiratory macro investigation (N/A)
respiratory_macro_sy_fi_id_nan	integer	Respiratory macro investigation (NaN)
reticuloendothelial_macro_sy_fi_id_c001	integer	Reticuloendothelial macro investigation (Normal)
reticuloendothelial_macro_sy_fi_id_c002	integer	Reticuloendothelial macro investigation (Abnormal but NOT contributed to death)
reticuloendothelial_macro_sy_fi_id_c003	integer	Reticuloendothelial macro investigation (Abnormal)
reticuloendothelial_macro_sy_fi_id_c999	integer	Reticuloendothelial macro investigation (N/A)
reticuloendothelial_macro_sy_fi_id_nan	integer	Reticuloendothelial macro investigation (NaN)
urogenital_macro_sy_fi_id_c001	integer	Urogenital macro investigation (Normal)
urogenital_macro_sy_fi_id_c002	integer	Urogenital macro investigation (Abnormal but NOT contributed to death)
urogenital_macro_sy_fi_id_c003	integer	Urogenital macro investigation (Abnormal)
urogenital_macro_sy_fi_id_c999	integer	Urogenital macro investigation (N/A)
urogenital_macro_sy_fi_id_nan	integer	Urogenital macro investigation (NaN)

Internal – Stage 3 – Histological investigation

Original columns		
Name	Type	Description
cardiovascular_histo_sy_hi_id	text	Cardiovascular histological investigation
central_nervous_histo_sy_hi_id	text	Central nervous histological investigation
endocrine_histo_sy_hi_id	text	Endocrine histological investigation
gastrointestinal_histo_sy_hi_id	text	Gastrointestinal histological investigation
respiratory_histo_sy_hi_id	text	Respiratory histological investigation
reticuloendothelial_histo_sy_hi_id	text	Reticuloendothelial histological investigation
urogenital_histo_sy_hi_id	text	Urogenital histological investigation

Lookup values		
category	code	label
/EventAttribute/Observation/PostMortem/LookUp/CardiovascularHisto_SyHiID	001	Normal
/EventAttribute/Observation/PostMortem/LookUp/CardiovascularHisto_SyHiID	002	Abnormal but NOT contributed to death
/EventAttribute/Observation/PostMortem/LookUp/CardiovascularHisto_SyHiID	003	Abnormal and cause of death
/EventAttribute/Observation/PostMortem/LookUp/CardiovascularHisto_SyHiID	999	N/A
/EventAttribute/Observation/PostMortem/LookUp/CentralNervousHisto_SyHiID	001	Normal
/EventAttribute/Observation/PostMortem/LookUp/CentralNervousHisto_SyHiID	002	Abnormal but NOT contributed to death
/EventAttribute/Observation/PostMortem/LookUp/CentralNervousHisto_SyHiID	003	Abnormal and cause of death
/EventAttribute/Observation/PostMortem/LookUp/CentralNervousHisto_SyHiID	999	N/A
/EventAttribute/Observation/PostMortem/LookUp/EndocrineHisto_SyHiID	001	Normal
/EventAttribute/Observation/PostMortem/LookUp/EndocrineHisto_SyHiID	002	Abnormal but NOT contributed to death
/EventAttribute/Observation/PostMortem/LookUp/EndocrineHisto_SyHiID	003	Abnormal and cause of death
/EventAttribute/Observation/PostMortem/LookUp/EndocrineHisto_SyHiID	999	N/A
/EventAttribute/Observation/PostMortem/LookUp/GastrointestinalHisto_SyHiID	001	Normal
/EventAttribute/Observation/PostMortem/LookUp/GastrointestinalHisto_SyHiID	002	Abnormal but NOT contributed to death
/EventAttribute/Observation/PostMortem/LookUp/GastrointestinalHisto_SyHiID	003	Abnormal and cause of death
/EventAttribute/Observation/PostMortem/LookUp/GastrointestinalHisto_SyHiID	999	N/A
/EventAttribute/Observation/PostMortem/LookUp/RespiratoryHisto_SyHiID	001	Normal
/EventAttribute/Observation/PostMortem/LookUp/RespiratoryHisto_SyHiID	002	Abnormal but NOT contributed to death
/EventAttribute/Observation/PostMortem/LookUp/RespiratoryHisto_SyHiID	003	Abnormal and cause of death
/EventAttribute/Observation/PostMortem/LookUp/RespiratoryHisto_SyHiID	999	N/A
/EventAttribute/Observation/PostMortem/LookUp/ReticuloendothelialHisto_SyHiID	001	Normal
/EventAttribute/Observation/PostMortem/LookUp/ReticuloendothelialHisto_SyHiID	002	Abnormal but NOT contributed to death
/EventAttribute/Observation/PostMortem/LookUp/ReticuloendothelialHisto_SyHiID	003	Abnormal and cause of death
/EventAttribute/Observation/PostMortem/LookUp/ReticuloendothelialHisto_SyHiID	999	N/A
/EventAttribute/Observation/PostMortem/LookUp/UrogenitalHisto_SyHiID	001	Normal
/EventAttribute/Observation/PostMortem/LookUp/UrogenitalHisto_SyHiID	002	Abnormal but NOT contributed to death
/EventAttribute/Observation/PostMortem/LookUp/UrogenitalHisto_SyHiID	003	Abnormal and cause of death
/EventAttribute/Observation/PostMortem/LookUp/UrogenitalHisto_SyHiID	999	N/A

Adjusted columns		
Name	Type	Description
cardiovascular_histo_sy_hi_id_c001	integer	Cardiovascular histological investigation (Normal)
cardiovascular_histo_sy_hi_id_c002	integer	Cardiovascular histological investigation (Abnormal but NOT contributed to death)
cardiovascular_histo_sy_hi_id_c003	integer	Cardiovascular histological investigation (Abnormal)
cardiovascular_histo_sy_hi_id_c999	integer	Cardiovascular histological investigation (N/A)
cardiovascular_histo_sy_hi_id_nan	integer	Cardiovascular histological investigation (NaN)
central_nervous_histo_sy_hi_id_c001	integer	Central nervous histological investigation (Normal)
central_nervous_histo_sy_hi_id_c002	integer	Central nervous histological investigation (Abnormal but NOT contributed to death)
central_nervous_histo_sy_hi_id_c003	integer	Central nervous histological investigation (Abnormal)
central_nervous_histo_sy_hi_id_c999	integer	Central nervous histological investigation (N/A)
central_nervous_histo_sy_hi_id_nan	integer	Central nervous histological investigation (NaN)
endocrine_histo_sy_hi_id_c001	integer	Endocrine histological investigation (Normal)
endocrine_histo_sy_hi_id_c002	integer	Endocrine histological investigation (Abnormal but NOT contributed to death)
endocrine_histo_sy_hi_id_c003	integer	Endocrine histological investigation (Abnormal)
endocrine_histo_sy_hi_id_c999	integer	Endocrine histological investigation (N/A)
endocrine_histo_sy_hi_id_nan	integer	Endocrine histological investigation (NaN)
gastrointestinal_histo_sy_hi_id_c001	integer	Gastrointestinal histological investigation (Normal)
gastrointestinal_histo_sy_hi_id_c002	integer	Gastrointestinal histological investigation (Abnormal but NOT contributed to death)
gastrointestinal_histo_sy_hi_id_c003	integer	Gastrointestinal histological investigation (Abnormal)
gastrointestinal_histo_sy_hi_id_c999	integer	Gastrointestinal histological investigation (N/A)
gastrointestinal_histo_sy_hi_id_nan	integer	Gastrointestinal histological investigation (NaN)
respiratory_histo_sy_hi_id_c001	integer	Respiratory histological investigation (Normal)
respiratory_histo_sy_hi_id_c002	integer	Respiratory histological investigation (Abnormal but NOT contributed to death)
respiratory_histo_sy_hi_id_c003	integer	Respiratory histological investigation (Abnormal)
respiratory_histo_sy_hi_id_c999	integer	Respiratory histological investigation (N/A)
respiratory_histo_sy_hi_id_nan	integer	Respiratory histological investigation (NaN)
reticuloendothelial_histo_sy_hi_id_c001	integer	Reticuloendothelial histological investigation (Normal)
reticuloendothelial_histo_sy_hi_id_c002	integer	Reticuloendothelial histological investigation (Abnormal but NOT contributed to death)
reticuloendothelial_histo_sy_hi_id_c003	integer	Reticuloendothelial histological investigation (Abnormal)
reticuloendothelial_histo_sy_hi_id_c999	integer	Reticuloendothelial histological investigation (N/A)
reticuloendothelial_histo_sy_hi_id_nan	integer	Reticuloendothelial histological investigation (NaN)

urogenital_histo_sy_hi_id_c001	integer	Urogenital histological investigation (Normal)
urogenital_histo_sy_hi_id_c002	integer	Urogenital histological investigation (Abnormal but NOT contributed to death)
urogenital_histo_sy_hi_id_c003	integer	Urogenital histological investigation (Abnormal)
urogenital_histo_sy_hi_id_c999	integer	Urogenital histological investigation (N/A)
urogenital_histo_sy_hi_id_nan	integer	Urogenital histological investigation (NaN)

Appendix E – Deliverables

Project GIT HUB Repository

All files associated with this project have been versioned controlled using a GIT HUB repository which is available at the following URL:

https://github.com/jbooth04BBK/MScProject

The structure of files in the GIT repository is in the same structure as stored on the attached CD the details of which are given blow. With the exception that the folder '' is not reflected in the GIT repository.

Files on attached CD

MScProject			**Python Scripts produced during the project**
MScProject	create_has_tables	.py	Python script to create and populate HAS tables
MScProject	create_rdvs	.py	Python script to create RDV CSV and XML files
MScProject	modify_csv_data	.py	Python script to modify original RDV files creates revised CSV and XML file
MScProject	modify_events	.py	Python script to add new event attributes to existing events
MScProject	readme	.md	Markup text file
MScProject	test_code	.py	Python script
MScProject	test_lin_reg	.py	Python script
MScProject	test_os_walk	.py	Python script
MScProject	test_pyodbc	.py	Python script
MScProject\Data			**RDV CSV and XML files used to produced output for the project report**
MScProject\Data	rdv_demo_selection_02	.csv	Comma separated text data file
MScProject\Data	rdv_demo_selection_02	.xml	Table definition file
MScProject\Data	rdv_study_ext	.csv	Comma separated text data file
MScProject\Data	rdv_study_ext	.xml	Table definition file
MScProject\Data	rdv_study_ext_adj	.csv	Comma separated text data file
MScProject\Data	rdv_study_ext_adj	.xml	Table definition file
MScProject\Data	rdv_study_ext_adj_columns	.csv	Comma separated text data file

MScProject\Data	rdv_study_ext_adj_lrparams	.csv	Comma separated text data file
MScProject\Data	rdv_study_int1	.csv	Comma separated text data file
MScProject\Data	rdv_study_int1	.xml	Table definition file
MScProject\Data	rdv_study_int1_adj	.csv	Comma separated text data file
MScProject\Data	rdv_study_int1_adj	.xml	Table definition file
MScProject\Data	rdv_study_int1_adj_columns	.csv	Comma separated text data file
MScProject\Data	rdv_study_int1_adj_lrparams	.csv	Comma separated text data file
MScProject\Data	rdv_study_int2	.csv	Comma separated text data file
MScProject\Data	rdv_study_int2	.xml	Table definition file
MScProject\Data	rdv_study_int2_adj	.csv	Comma separated text data file
MScProject\Data	rdv_study_int2_adj	.xml	Table definition file
MScProject\Data	rdv_study_int2_adj_columns	.csv	Comma separated text data file
MScProject\Data	rdv_study_int2_adj_lrparams	.csv	Comma separated text data file
MScProject\Data	rdv_study_int3	.csv	Comma separated text data file
MScProject\Data	rdv_study_int3	.xml	Table definition file
MScProject\Data	rdv_study_int3_adj	.csv	Comma separated text data file
MScProject\Data	rdv_study_int3_adj	.xml	Table definition file
MScProject\Data	rdv_study_int3_adj_columns	.csv	Comma separated text data file
MScProject\Data	rdv_study_int3_adj_lrparams	.csv	Comma separated text data file
MScProject\Docs			**Project report files**
MScProject\Docs	boothj_ds_report_v03	.docx	Word Document file
MScProject\Docs	project_pipeline	.docx	Word Document file
MScProject\Docs	project_presentation_20190823	.pptx	Power point file
MScProject\Docs	project_stuff	.docx	Word Document file
MScProject\Docs	project_tracker	.docx	Word Document file
MScProject\Docs	prop_boothj_ds	.docx	Word Document file
MScProject\Docs\Images			**Images files used in project report**
MScProject\Docs\Images	adjustedl_data_set	.png	Image file
MScProject\Docs\Images	all_data_vis_grid	.png	Image file
MScProject\Docs\Images	has_schema	.png	Image file
MScProject\Docs\Images	inc_data_vis_grid	.png	Image file
MScProject\Docs\Images	lr_infants	.png	Image file

MScProject\Docs\Images	lr_nn	.png	Image file
MScProject\Docs\Images	na_measurements	.png	Image file
MScProject\Docs\Images	nutrition_concepts	.png	Image file
MScProject\Docs\Images	original_data_set	.png	Image file
MScProject\Docs\Images	pm_research_partial_schema	.png	Image file
MScProject\Docs\Images	project_pipeline	.png	Image file
MScProject\Docs\Images	study_categoric_values	.png	Image file
MScProject\Docs\Images\run_01			**Support files output by R scripts produced during a complete run and used in the project report**
MScProject\Docs\Images\run_01	accuracy_table	.png	Image file
MScProject\Docs\Images\run_01	accuracy_table_crop	.png	Image file
MScProject\Docs\Images\run_01	comb_accuracy_model_run	.png	Image file
MScProject\Docs\Images\run_01	comb_accuracy_run_model	.png	Image file
MScProject\Docs\Images\run_01	compare_feature_importance_stage_ext	.png	Image file
MScProject\Docs\Images\run_01	compare_feature_importance_stage_int3	.png	Image file
MScProject\Docs\Images\run_01	dt_confusion_matrix_ext_01	.png	Image file
MScProject\Docs\Images\run_01	dt_confusion_matrix_grid_int3_01	.png	Image file
MScProject\Docs\Images\run_01	dt_confusion_matrix_int1_01	.png	Image file
MScProject\Docs\Images\run_01	dt_confusion_matrix_int2_01	.png	Image file
MScProject\Docs\Images\run_01	dt_confusion_matrix_int3_01	.png	Image file
MScProject\Docs\Images\run_01	dt_feature_importance_ext_01	.png	Image file
MScProject\Docs\Images\run_01	dt_feature_importance_hm_01	.png	Image file
MScProject\Docs\Images\run_01	dt_feature_importance_int1_01	.png	Image file
MScProject\Docs\Images\run_01	dt_feature_importance_int2_01	.png	Image file
MScProject\Docs\Images\run_01	dt_feature_importance_int3_01	.png	Image file
MScProject\Docs\Images\run_01	dt_tree_cp_ext_01	.txt	Text file
MScProject\Docs\Images\run_01	dt_tree_cp_int1_01	.txt	Text file
MScProject\Docs\Images\run_01	dt_tree_cp_int2_01	.txt	Text file
MScProject\Docs\Images\run_01	dt_tree_cp_int3_01	.txt	Text file
MScProject\Docs\Images\run_01	dt_tree_ext_01	.png	Image file
MScProject\Docs\Images\run_01	dt_tree_fit_ext_01	.txt	Text file
MScProject\Docs\Images\run_01	dt_tree_fit_int1_01	.txt	Text file

MScProject\Docs\Images\run_01	dt_tree_fit_int2_01	.txt	Text file
MScProject\Docs\Images\run_01	dt_tree_fit_int3_01	.txt	Text file
MScProject\Docs\Images\run_01	dt_tree_int1_01	.png	Image file
MScProject\Docs\Images\run_01	dt_tree_int2_01	.png	Image file
MScProject\Docs\Images\run_01	dt_tree_int3_01	.png	Image file
MScProject\Docs\Images\run_01	rf_confusion_matrix_grid_int3_01	.png	Image file
MScProject\Docs\Images\run_01	rf_feature_importance_ext_01	.png	Image file
MScProject\Docs\Images\run_01	rf_feature_importance_hm_01	.png	Image file
MScProject\Docs\Images\run_01	rf_feature_importance_int1_01	.png	Image file
MScProject\Docs\Images\run_01	rf_feature_importance_int2_01	.png	Image file
MScProject\Docs\Images\run_01	rf_feature_importance_int3_01	.png	Image file
MScProject\Docs\Images\run_01	xgb_confusion_matrix_ext_01	.png	Image file
MScProject\Docs\Images\run_01	xgb_confusion_matrix_grid_int3_01	.png	Image file
MScProject\Docs\Images\run_01	xgb_confusion_matrix_int1_01	.png	Image file
MScProject\Docs\Images\run_01	xgb_confusion_matrix_int2_01	.png	Image file
MScProject\Docs\Images\run_01	xgb_confusion_matrix_int3_01	.png	Image file
MScProject\Docs\Images\run_01	xgb_feature_importance_ext_01	.png	Image file
MScProject\Docs\Images\run_01	xgb_feature_importance_hm_01	.png	Image file
MScProject\Docs\Images\run_01	xgb_feature_importance_int1_01	.png	Image file
MScProject\Docs\Images\run_01	xgb_feature_importance_int2_01	.png	Image file
MScProject\Docs\Images\run_01	xgb_feature_importance_int3_01	.png	Image file
MScProject\Docs\Images\run_01	xgb_feature_importance_mean_ext	.png	Image file
MScProject\Docs\Images\run_01	xgb_feature_importance_mean_int1	.png	Image file
MScProject\Docs\Images\run_01	xgb_feature_importance_mean_int2	.png	Image file
MScProject\Docs\Images\run_01	xgb_feature_importance_mean_int3	.png	Image file
MScProject\RCode			**R Scripts produced during the project**
MScProject\RCode	combine_run_models_results	.r	R script
MScProject\RCode	comb_study_heatmaps	.r	R script
MScProject\RCode	confusion_matrix	.r	R script
MScProject\RCode	dtree_study	.r	R script - Decision tree model function
MScProject\RCode	dtree_study_all	.r	R script
MScProject\RCode	dtree_study_ext	.r	R script

MScProject\RCode	dtree_study_ext_adj	.r	R script
MScProject\RCode	dtree_study_int1	.r	R script
MScProject\RCode	dtree_study_play	.r	R script
MScProject\RCode	featrure_importance_result	.r	R script
MScProject\RCode	gboost_example	.r	R script
MScProject\RCode	gboost_study	.r	R script - XGBoost model function
MScProject\RCode	gboost_study_all	.r	R script
MScProject\RCode	gboost_study_ext	.r	R script
MScProject\RCode	gboost_study_ext_adj	.r	R script
MScProject\RCode	gboost_study_int1	.r	R script
MScProject\RCode	gboost_study_int1_adj	.r	R script
MScProject\RCode	gboost_study_tune	.r	R script
MScProject\RCode	gboost_study_tune_part2	.r	R script
MScProject\RCode	rforest_study	.r	R script - Random Forest model function
MScProject\RCode	rforest_study_all	.r	R script
MScProject\RCode	rforest_study_ext	.r	R script
MScProject\RCode	rforest_study_ext_adj	.r	R script
MScProject\RCode	rforest_study_int1	.r	R script
MScProject\RCode	run_models	.r	R script - Runs models for multiple random.seeds and combines results
MScProject\RCode	study_functions	.r	R script - R functions used across all models
MScProject\RCode	test_basics	.r	R script
MScProject\RCode	test_data_frames	.r	R script
MScProject\RCode	test_heatmaps	.r	R script
MScProject\RCode	test_linear_regression	.r	R script
MScProject\RCode	test_list_files	.r	R script
MScProject\RCode	test_lm	.r	R script
MScProject\RCode	vis_plots	.r	R script