**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.

The report may be freely copied and distributed provided the source is explicitly acknowledged.

Acknowledgements

Contents

[Table of Figures 5](#_Toc18749340)

[1 Abstract 6](#_Toc18749341)

[2 Introduction 7](#_Toc18749342)

[2.1 Background 7](#_Toc18749343)

[2.2 Literary Review 7](#_Toc18749344)

[2.2.1 Paediatric Post-mortems 7](#_Toc18749345)

[2.2.2 Data Extraction 7](#_Toc18749346)

[2.2.3 Data Wrangling 8](#_Toc18749347)

[2.2.4 Analytics 8](#_Toc18749348)

[2.3 Report outline 8](#_Toc18749349)

[2.4 Aim and Objectives 8](#_Toc18749350)

[3 Methods 10](#_Toc18749351)

[3.1 Project Pipeline 10](#_Toc18749352)

[3.2 Data Engineering 10](#_Toc18749353)

[3.2.1 Extract, Transform, Load (ETL) Process 10](#_Toc18749354)

[3.2.2 Creation of Summary and Reporting Attributes 11](#_Toc18749355)

[3.2.3 Identifying Data to be included in this study 11](#_Toc18749356)

[3.2.4 Data Wrangling 12](#_Toc18749357)

[3.3 Analytics 12](#_Toc18749358)

[3.3.1 Visualisation: ggplot2 13](#_Toc18749359)

[3.3.2 Decision Tree: rpart 13](#_Toc18749360)

[3.3.3 Random Forests: randomForest 13](#_Toc18749361)

[3.3.4 Gradient Boosted Decision Tree: xgboost 13](#_Toc18749362)

[3.3.5 Combined Results 14](#_Toc18749363)

[4 Data Engineering 15](#_Toc18749364)

[4.1 ETL Process 15](#_Toc18749365)

[4.2 Creation of Summary and Reporting Attributes 17](#_Toc18749366)

[4.3 Identifying Data to be included in this study 17](#_Toc18749367)

[4.4 Data Wrangling 18](#_Toc18749368)

[5 Analysis 21](#_Toc18749369)

[5.1 Visualisation 21](#_Toc18749370)

[5.1.1 Complete Data Set 21](#_Toc18749371)

[5.1.2 Study Data Set 22](#_Toc18749372)

[5.1.3 Missing data 23](#_Toc18749373)

[5.1.4 Imbalanced data 24](#_Toc18749374)

[5.2 Decision Tree 25](#_Toc18749375)

[5.3 Ensemble Models 28](#_Toc18749376)

[5.3.1 Random Forest 28](#_Toc18749377)

[5.3.2 Gradient Boosted Decision Tree 29](#_Toc18749378)

[5.4 Compare models with varying random seeds 30](#_Toc18749379)

[6 Results 32](#_Toc18749380)

[6.1 Predictive Accuracy 32](#_Toc18749381)

[6.2 Relative Feature Importance by Stage of Post-mortem 33](#_Toc18749382)

[7 Conclusion 35](#_Toc18749383)

[7.1 Project Summary 35](#_Toc18749384)

[7.1.1 Data Engineering 35](#_Toc18749385)

[7.1.2 Decision Tree 35](#_Toc18749386)

[7.1.3 Ensemble Models 35](#_Toc18749387)

[7.2 Project Evaluation 35](#_Toc18749388)

[7.3 Recommendations for Future Work 35](#_Toc18749389)

[References 36](#_Toc18749390)

[Glossary of Terms 37](#_Toc18749391)

[Appendix A – Example Project Code 38](#_Toc18749392)

[Appendix B – ETL Process 38](#_Toc18749393)

[Appendix C – Cause of Death Attribute Mapping 38](#_Toc18749394)

[Appendix D –RDV structures 38](#_Toc18749395)

[External 38](#_Toc18749396)

[Internal – Stage 1 – Organs 38](#_Toc18749397)

[Internal – Stage 2 – Macro investigation 38](#_Toc18749398)

[Internal – Stage 3 – Histological investigation 38](#_Toc18749399)

[Appendix E – Detailed output from an analytic process 39](#_Toc18749400)

[Appendix F – Deliverables 39](#_Toc18749401)

# Table of Figures

[Figure 1- PM Research Database - Partial Entity Relationship Diagram 15](#_Toc18686116)

[Figure 2- Example of the use of the Concepts Table within the EAV model 16](#_Toc18686117)

[Figure 3 - HAS Schema Entity Relationship Diagram 16](#_Toc18686118)

[Figure 4 - Snapshot of the original RDV 19](#_Toc18686119)

[Figure 5 - Snapshot adjusted using normalisation and one-hot encoding 19](#_Toc18686120)

[Figure 6 - Linear Regression Plots for Numeric Features 20](#_Toc18686121)

[Figure 7 - Visualisations of Complete Data Set 21](#_Toc18686122)

[Figure 8 - Visualisation of Study Data Set 22](#_Toc18686123)

[Figure 9 - Number of missing measurement values 23](#_Toc18686124)

[Figure 10 - Variability of Categorical Variables 24](#_Toc18686125)

[Figure 11 - Decision Tree - External Examination 26](#_Toc18686126)

[Figure 12 - Decision Tree - Internal Examination – Organs 26](#_Toc18686127)

[Figure 13 - Decision Tree - Relative Feature Importance - All Stages 27](#_Toc18686128)

[Figure 14 - Random Forest Confusion Matrices - All stages 28](#_Toc18686129)

[Figure 15 - Random Forest Relative Feature Importance - All Stages 28](#_Toc18686130)

[Figure 16 - XGBoost Confusion Matrices - All stages 29](#_Toc18686131)

[Figure 17 - XGBoost Relative Feature Importance - All Stages 29](#_Toc18686132)

[Figure 18 - Predictive Accuracy of each stage by Model by Run 30](#_Toc18686133)

[Figure 19 - Variability in Accuracy by Model by Stage 30](#_Toc18686134)

[Figure 20 - Compare Feature Importance by Model, Stage: Ext, by Run 31](#_Toc18686135)

[Figure 21 - Compare Feature Importance by Model, Stage: Int3, by Run 31](#_Toc18686136)

[Figure 22 - Relative Feature Importance, Stage: Ext 33](#_Toc18686137)

[Figure 23 - Relative Feature Importance, Stage: Int1 33](#_Toc18686138)

[Figure 24 - Relative Feature Importance, Stage: Int2 34](#_Toc18686139)

[Figure 25 - Relative Feature Importance, Stage: Int3 34](#_Toc18686140)

# 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.

# Introduction

## 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.

The purpose of this project is to use data science analytic techniques 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.

## Literary Review

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

### 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.

### 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)]*

### 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]*

### 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.*]

## Report outline

Highlight some key aspects of the report, include some interesting graphics clearly annotated.

NB Glossary of terms that cover the basic healthcare concepts covered in this document.

## Aim and Objectives

Take from project proposal; amended based on experience undertaking project?

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.

# Methods

## 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 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 D - Deliverables.

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

## Data Engineering

The data engineering aspects of this project will be undertaken using the Python programing language, a general purpose programing language that used extensively in the world of data science. The development of python procedures will be carried out using PyCharm an integrated development environment (IDE).

The data manipulation carried out during this stage will be using structured query language (SQL) and will be instigated using the Python package PyODBC which allows the connection to external databases using ODBC connections and the production and return of SQL queries. An example of a PyODBC procedure is given in Appendix A – Example Project Code.

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

### 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 basic structure will be:

* 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 will be the HAS database created and populated from the originating system.

### Creation of Summary and Reporting Attributes

Having created the base research data from the original data then a number of summary event attributes will be 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 will be the addition of a number of event attributes added to the existing set of events and their originating attributes.

### Identifying Data to be included in this study

This process will be 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 will not be addressed.

The output of this stage will be four research data views (RDVs), one for each stage of the post-mortem, in the form of CSV files.

### Data Wrangling

The final section of the data engineering stage will be to produce the data in the format most appropriate for analytics. Two forms of data wrangling will be 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 notes 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 will be 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 C – RDV Structure.

## Analytics

The analytic aspects of this project will be undertaken using the R programming language a language specifically developed for statistical computing. The development of R scripts will be carried out in R Studio an IDE for the R language.

In this section I will describe the key packages that I will be using and the specific parameters that have to be tuned to obtain an optimised model.

A basic tuning procedure will be 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 is an R function that can be called for that model with a training/test split for each post-mortem stage. The function will save the resulting confusion matrices and relative feature importance in CSV files as well as plots specific to each model as PNG files.

### Visualisation: ggplot2

Ggplot2 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 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 are two frames of visualisations saved as PNG files.

### 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.

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

### Random Forests: randomForest

Classification and regression based on a forest of trees using random inputs, based on Breiman (2001)

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

### Gradient Boosted Decision Tree: xgboost

xtreme Gradient Boosting, which is an efficient implementation of the gradient boosting framework from Chen & Guestrin (2016)

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.

### Combined Results

Using the functions developed for each model package a model can be run for each model package for each post-mortem stage for a number of random seeds deriving the training/test data split. The CSV files from each model run can then be 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.

# Data Engineering

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

## 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 1- 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.



Figure 2- 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 3 - 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.

## 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 we add additional attributes 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:
  + system\_name + macro\_SyFiID and 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

## 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.

## 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.
  + 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.



Figure 4 - Snapshot of the original RDV



Figure 5 - 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 6 - 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.

# Analysis

Explain what analysis was undertaken and why.

## Visualisation

### 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.

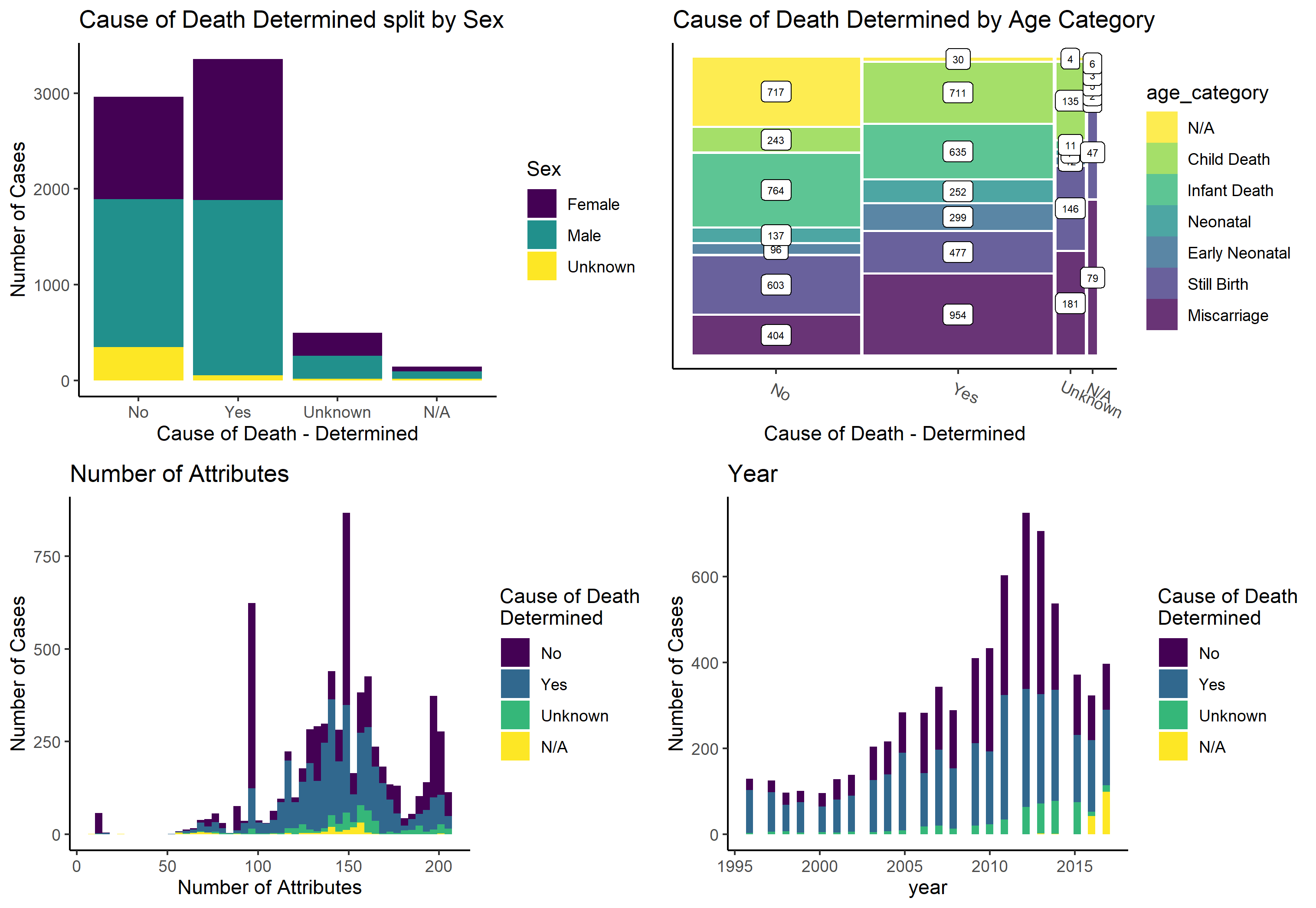


Figure 7 - Visualisations of Complete Data Set

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.

### 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.

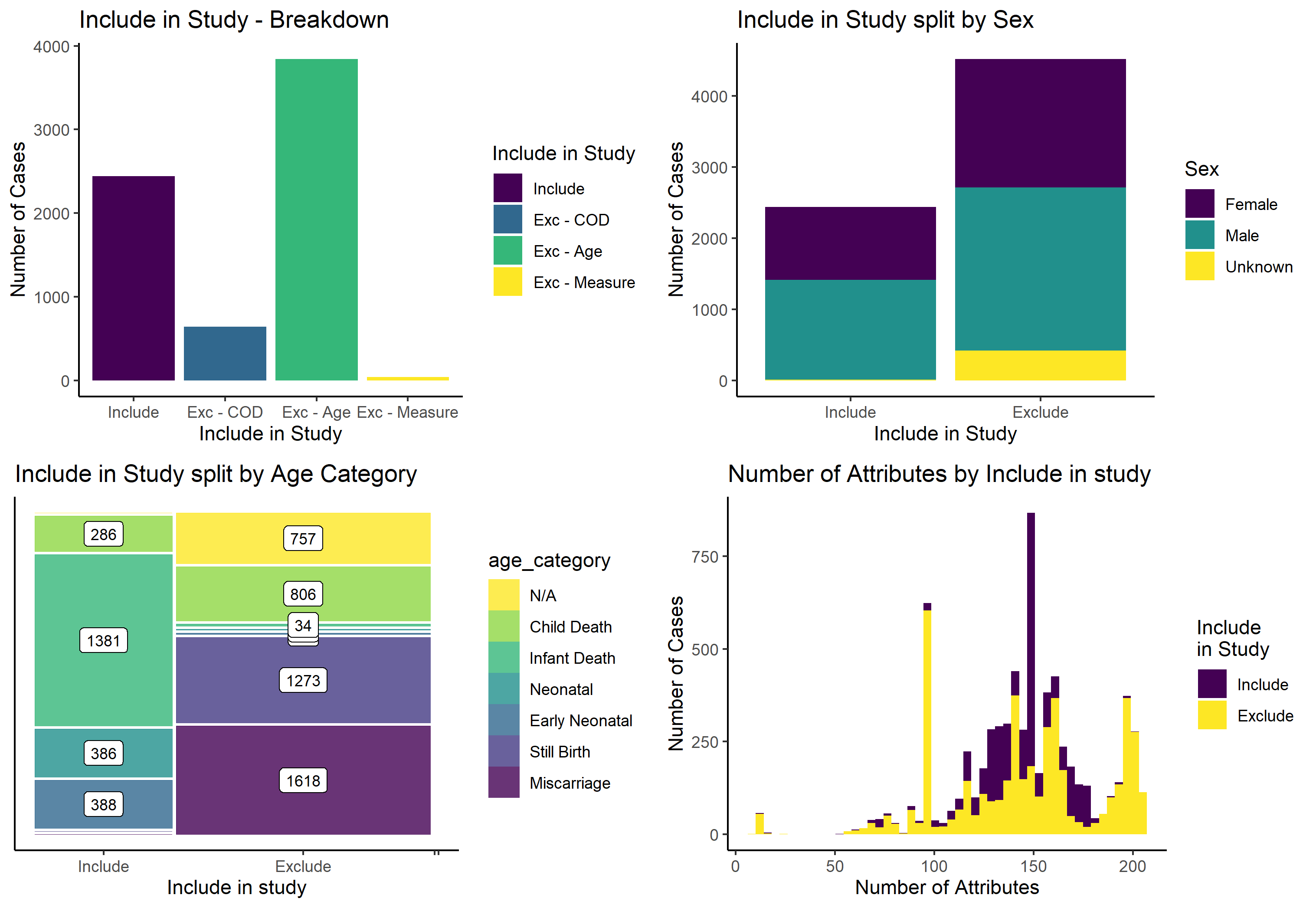


Figure 8 - Visualisation of Study Data Set

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.

### Missing data

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

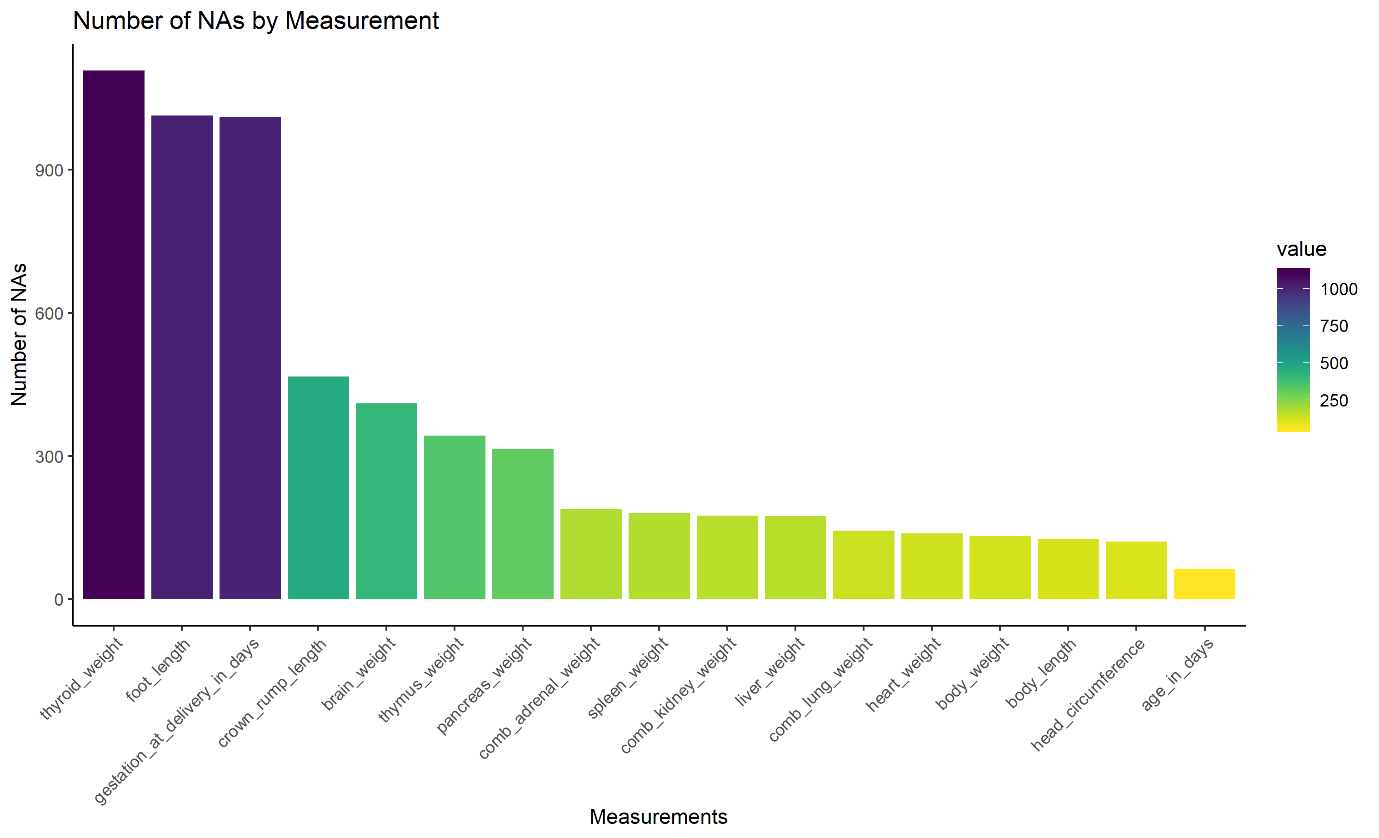


Figure 9 - Number of missing measurement values

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.

### 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.

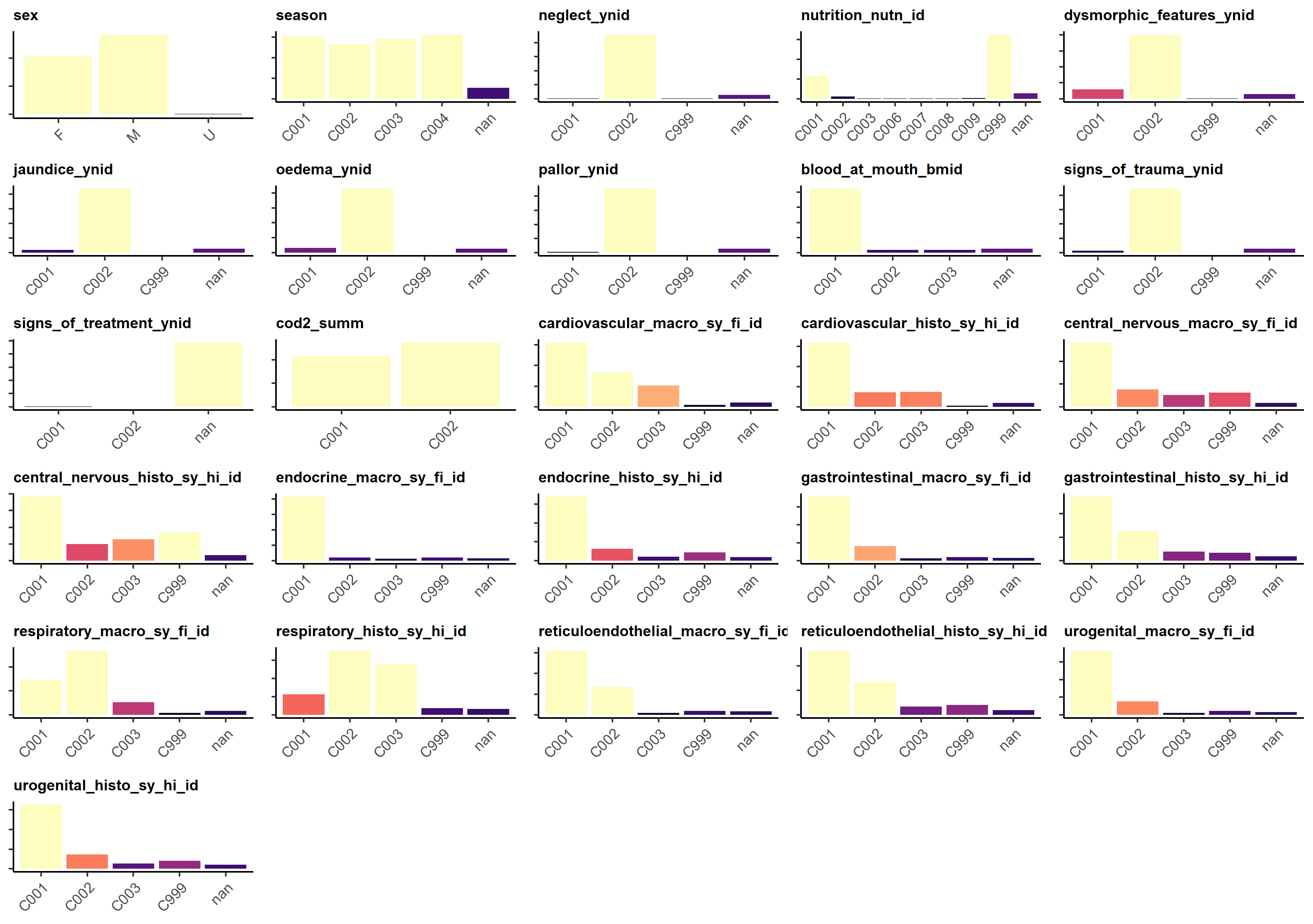


Figure 10 – Variability of Categorical Variables

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.

## Decision Tree

|  |
| --- |
| 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\_c002  [4] oedema\_ynid\_c001 pallor\_ynid\_c002  Root node error: 792/1768 = 0.44796  n= 1768  CP nsplit rel error xerror xstd  1 0.118056 0 1.00000 1.00000 0.026401  2 0.047980 2 0.76389 0.78409 0.025343  3 0.027778 3 0.71591 0.73864 0.024981  4 0.013889 4 0.68813 0.73485 0.024948  5 0.010101 5 0.67424 0.71843 0.024803  6 0.010000 6 0.66414 0.71843 0.024803 |
| n= 1768  node), split, n, loss, yval, (yprob)  \* denotes terminal node  1) root 1768 792 1 (0.4479638 0.5520362)  2) age\_in\_days>=15.5 1360 644 0 (0.5264706 0.4735294)  4) age\_in\_days< 224.5 989 401 0 (0.5945399 0.4054601)  8) oedema\_ynid\_c001< 0.5 937 356 0 (0.6200640 0.3799360)  16) dysmorphic\_features\_ynid\_c002>=0.5 823 288 0 (0.6500608 0.3499392)  32) body\_weight< 0.5913 796 269 0 (0.6620603 0.3379397)  64) pallor\_ynid\_c002>=0.5 786 260 0 (0.6692112 0.3307888) \*  65) pallor\_ynid\_c002< 0.5 10 1 1 (0.1000000 0.9000000) \*  33) body\_weight>=0.5913 27 8 1 (0.2962963 0.7037037) \*  17) dysmorphic\_features\_ynid\_c002< 0.5 114 46 1 (0.4035088 0.5964912) \*  9) oedema\_ynid\_c001>=0.5 52 7 1 (0.1346154 0.8653846) \*  5) age\_in\_days>=224.5 371 128 1 (0.3450135 0.6549865) \*  3) age\_in\_days< 15.5 408 76 1 (0.1862745 0.8137255) \* |
| Decision Tree Plot |
|  |
| Relative Feature Importance |
|  |
| Confusion Matrix |
|  |

Figure 11 – Decision Tree – External Examination

Figure 12 – Decision Tree – Internal Examination – Organs



Figure 13 – Decision Tree – Relative Feature Importance – All Stages

## Ensemble Models

NB Reduced output just consider predictive accuracy and Relative Feature Importance.

### Random Forest

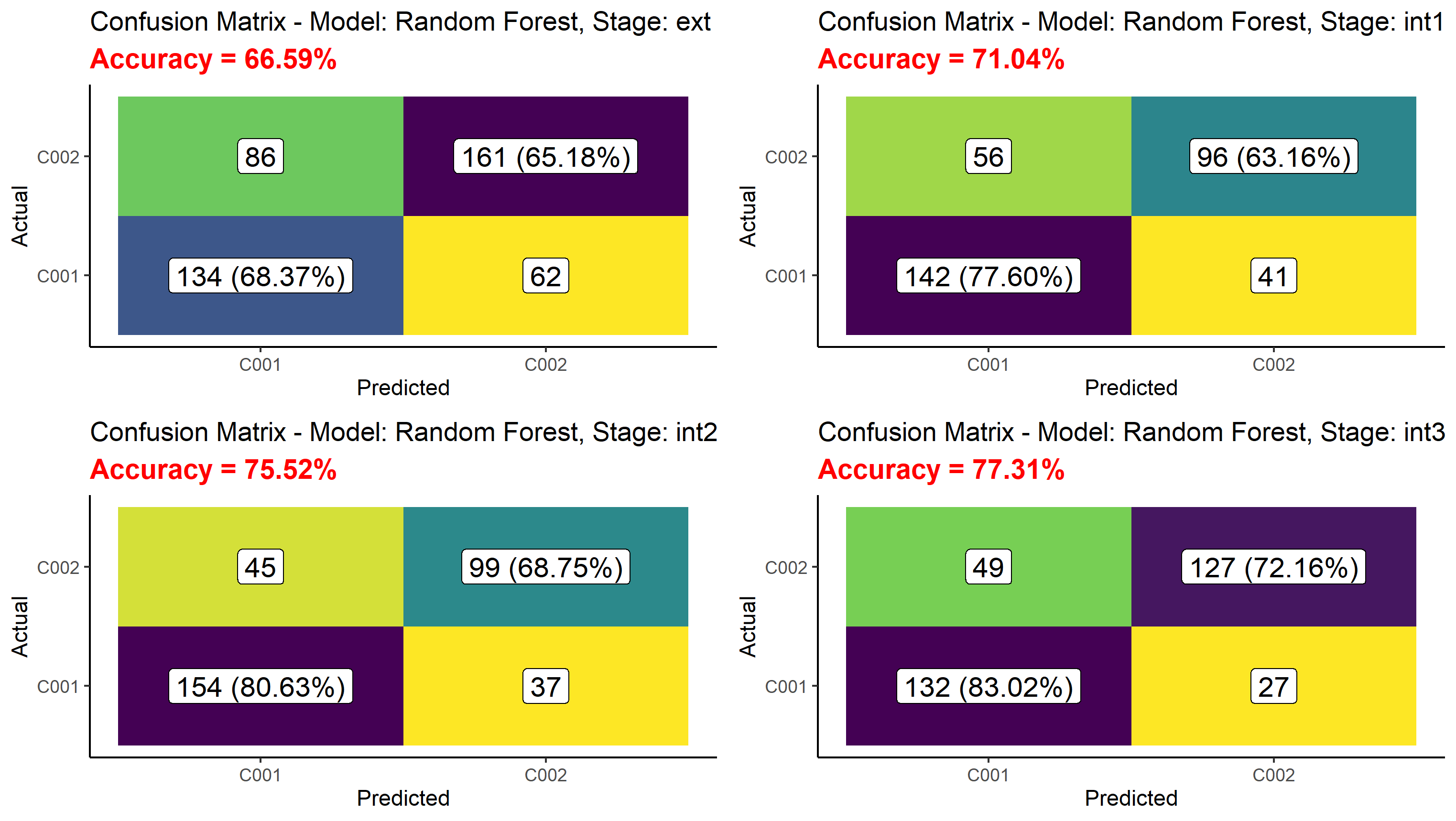


Figure 14 – Random Forest Confusion Matrices – All stages

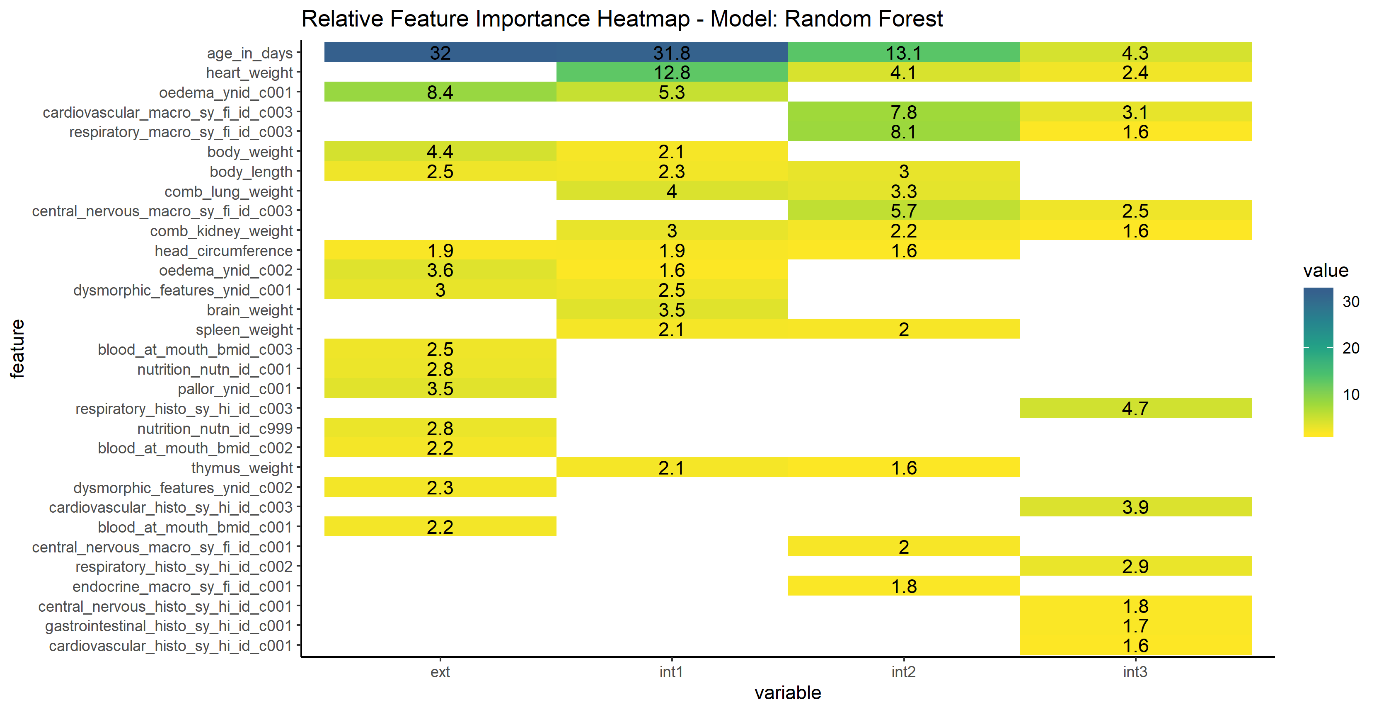


Figure 15 – Random Forest Relative Feature Importance – All Stages

### Gradient Boosted Decision Tree

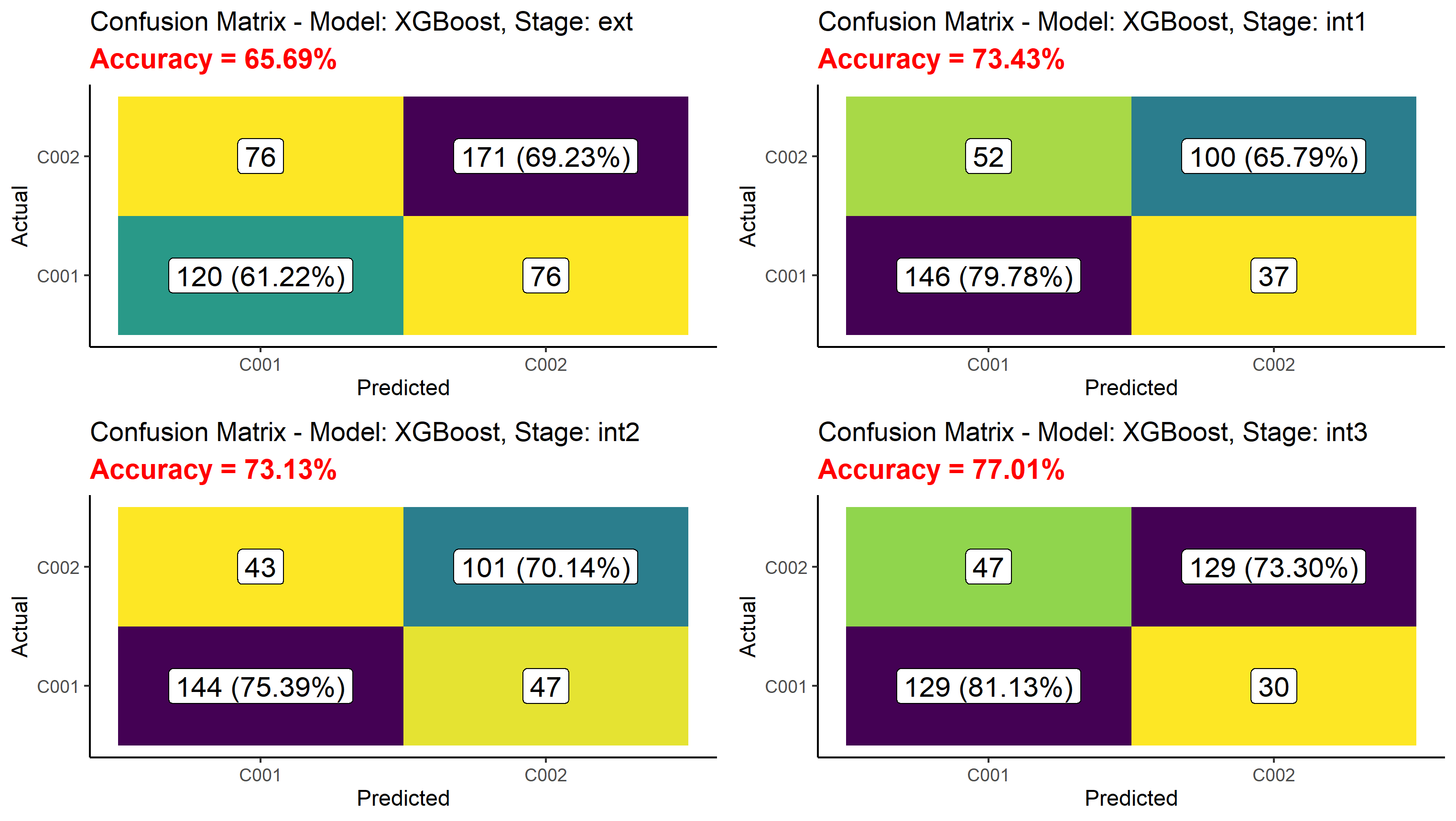


Figure 16 – XGBoost Confusion Matrices – All stages



Figure 17 – XGBoost Relative Feature Importance – All Stages

## Compare models with varying random seeds

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

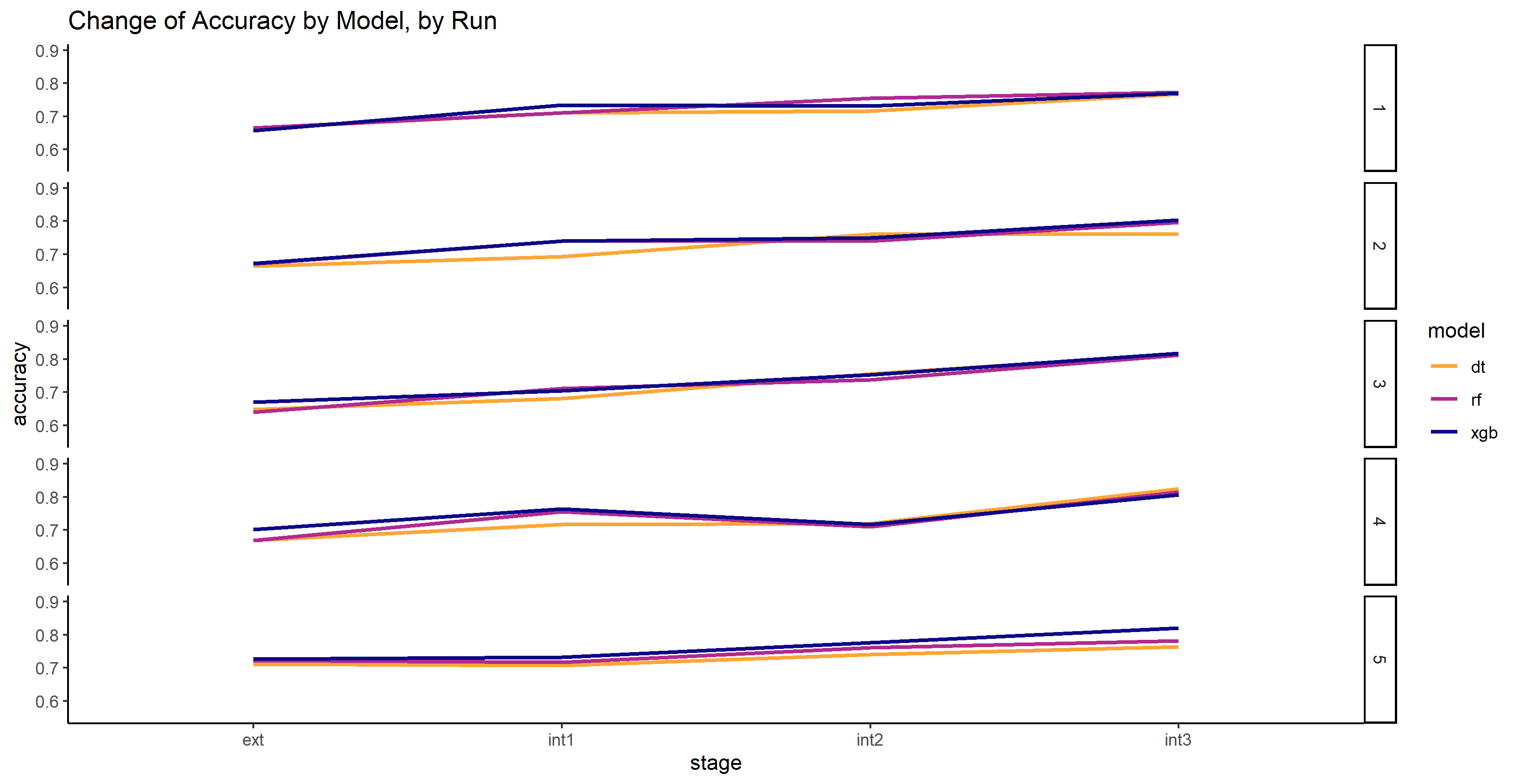


Figure 18 – Predictive Accuracy of each stage by Model by Run

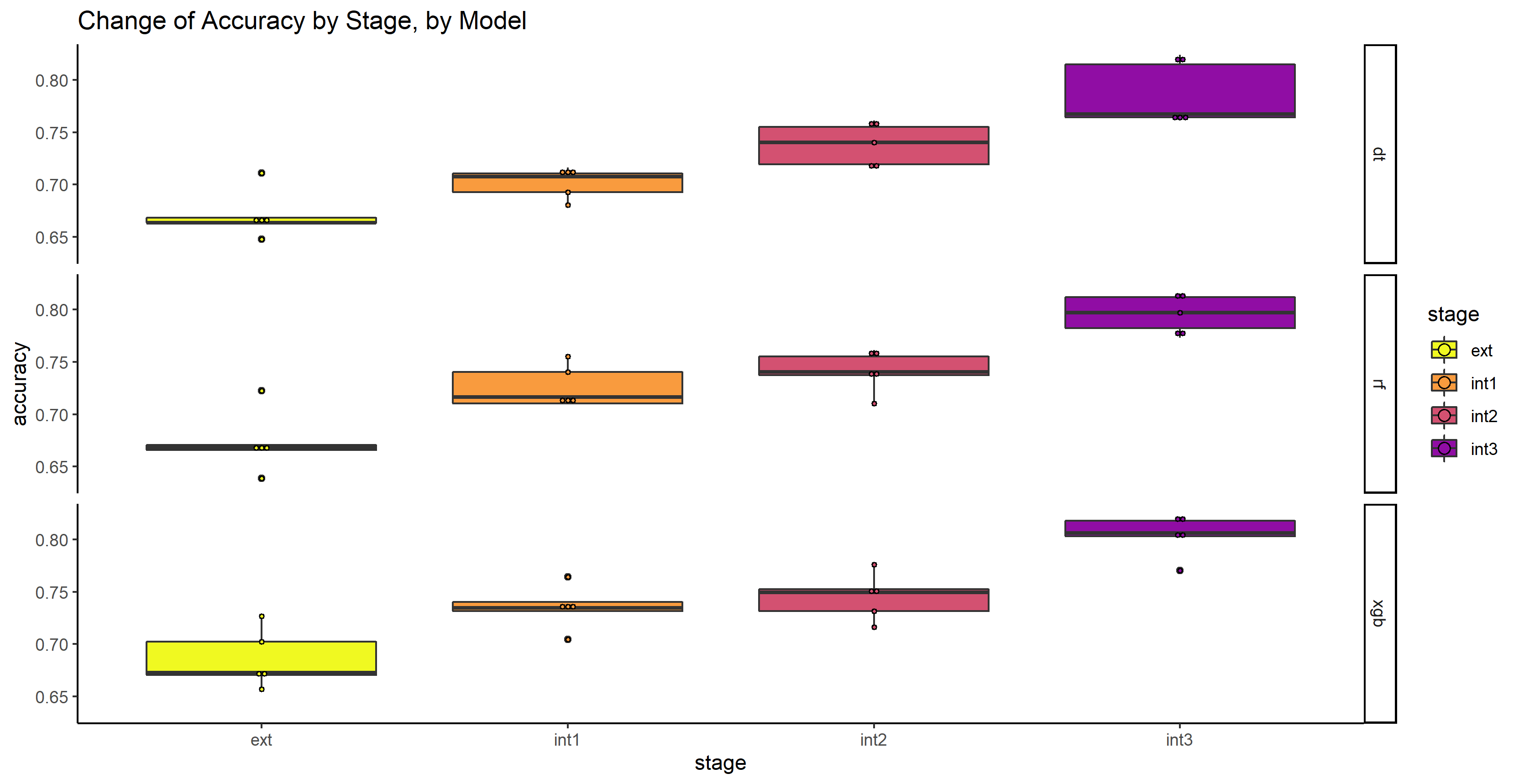


Figure 19 – Variability in Accuracy by Model by Stage

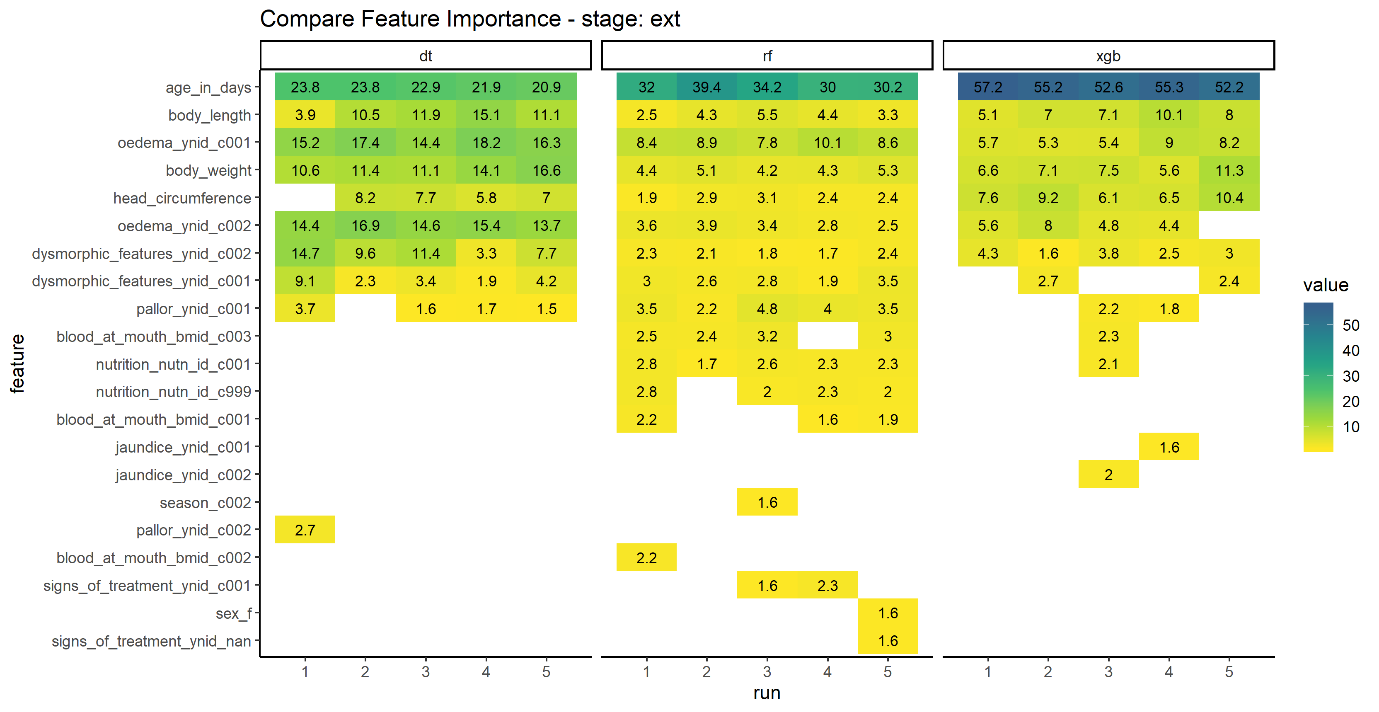


Figure 20 – Compare Feature Importance by Model, Stage: Ext, by Run

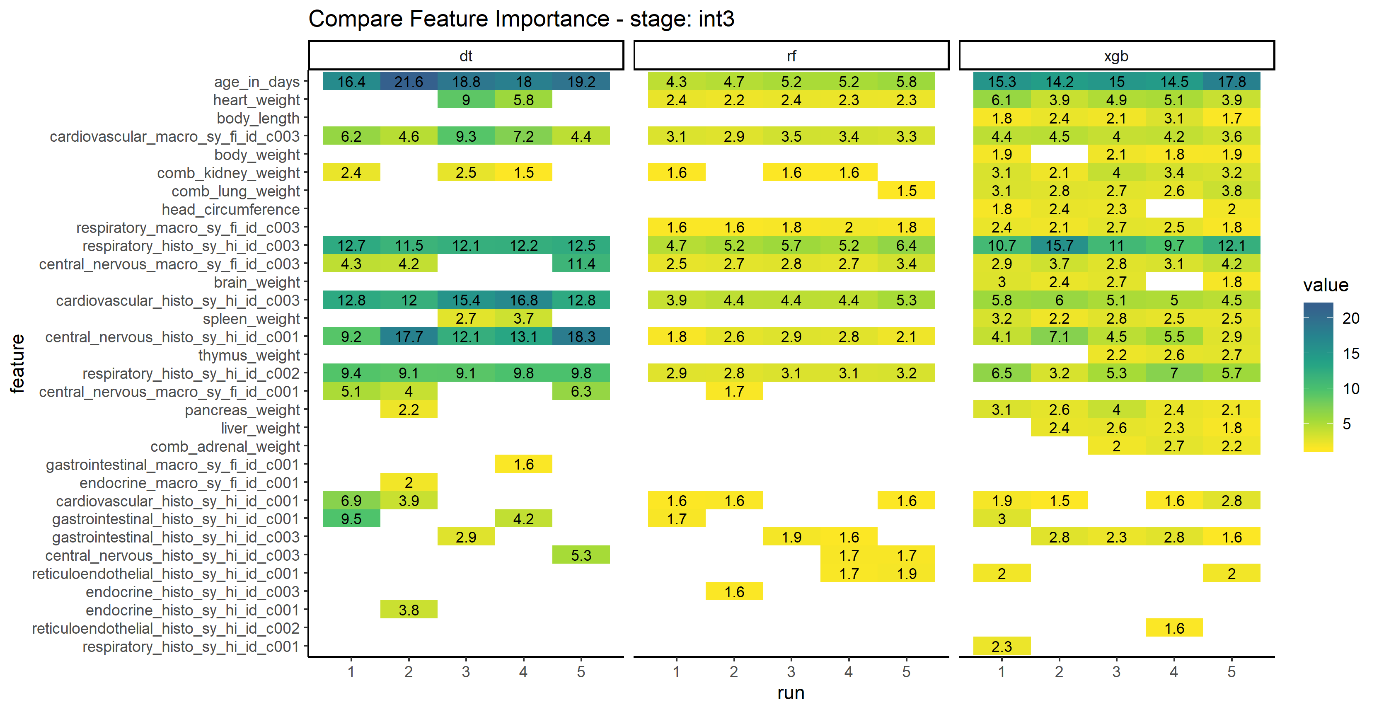


Figure 21 – Compare Feature Importance by Model, Stage: Int3, by Run

# Results

## Predictive Accuracy

Now the text should flow alongside the image?

## Relative Feature Importance by Stage of Post-mortem

As XGBoost is the most accurate then use this model’s results for Feature Importance

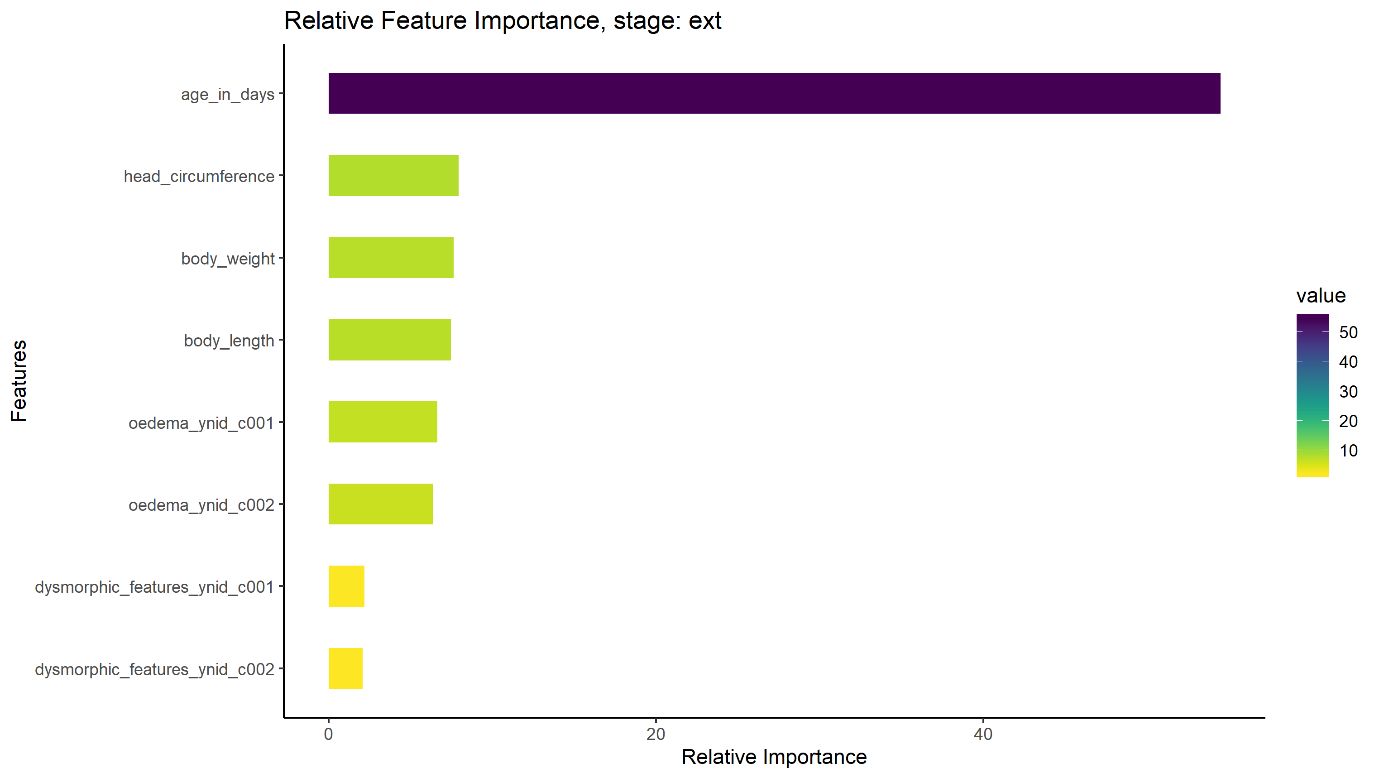


Figure 22 – Relative Feature Importance, Stage: Ext

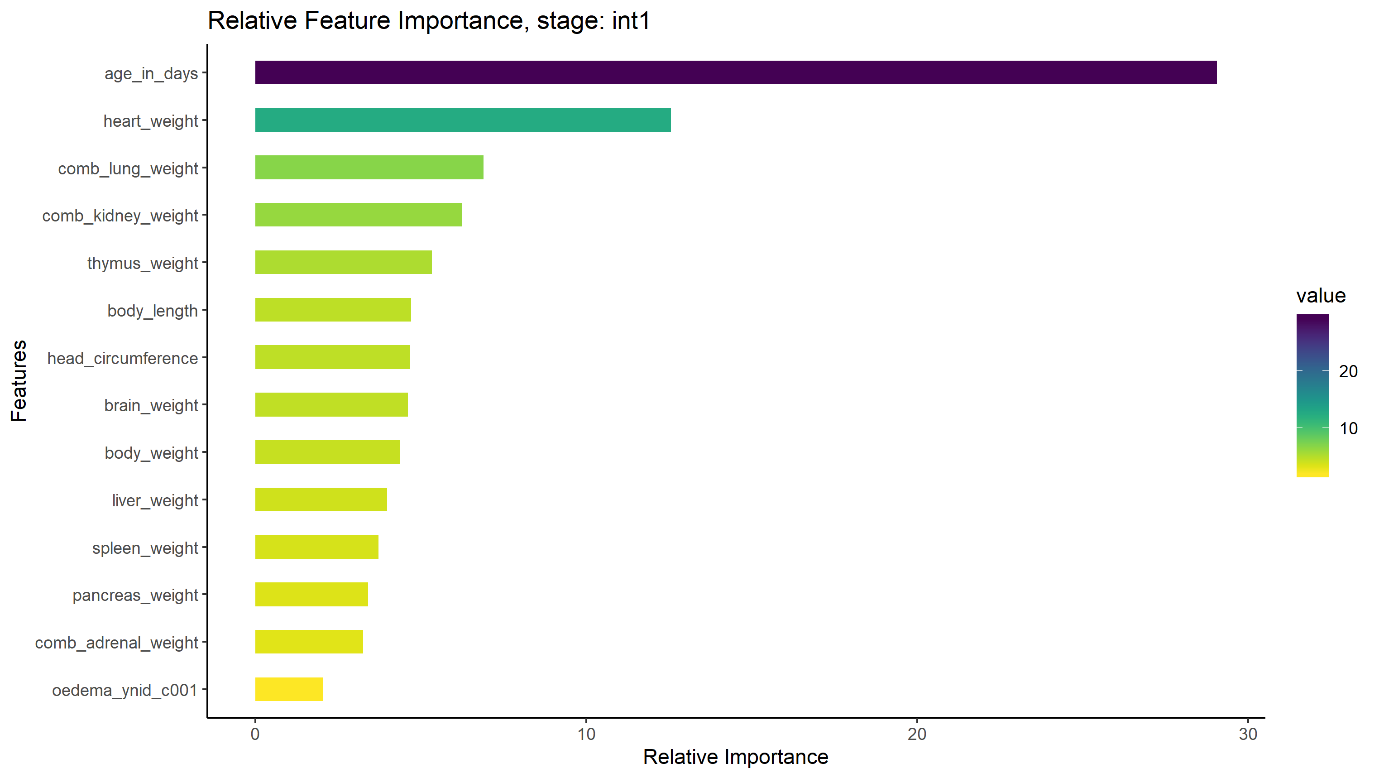


Figure 23 – Relative Feature Importance, Stage: Int1

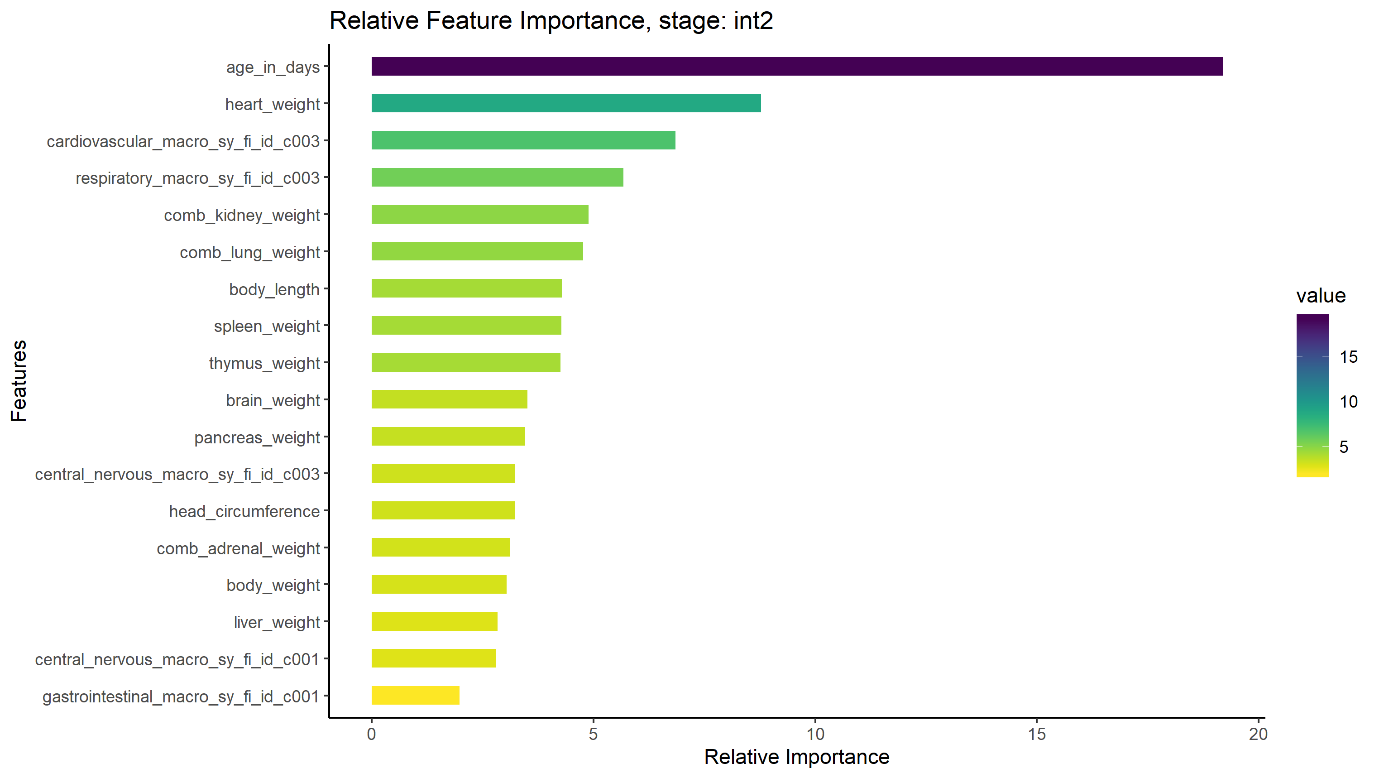


Figure 24 – Relative Feature Importance, Stage: Int2

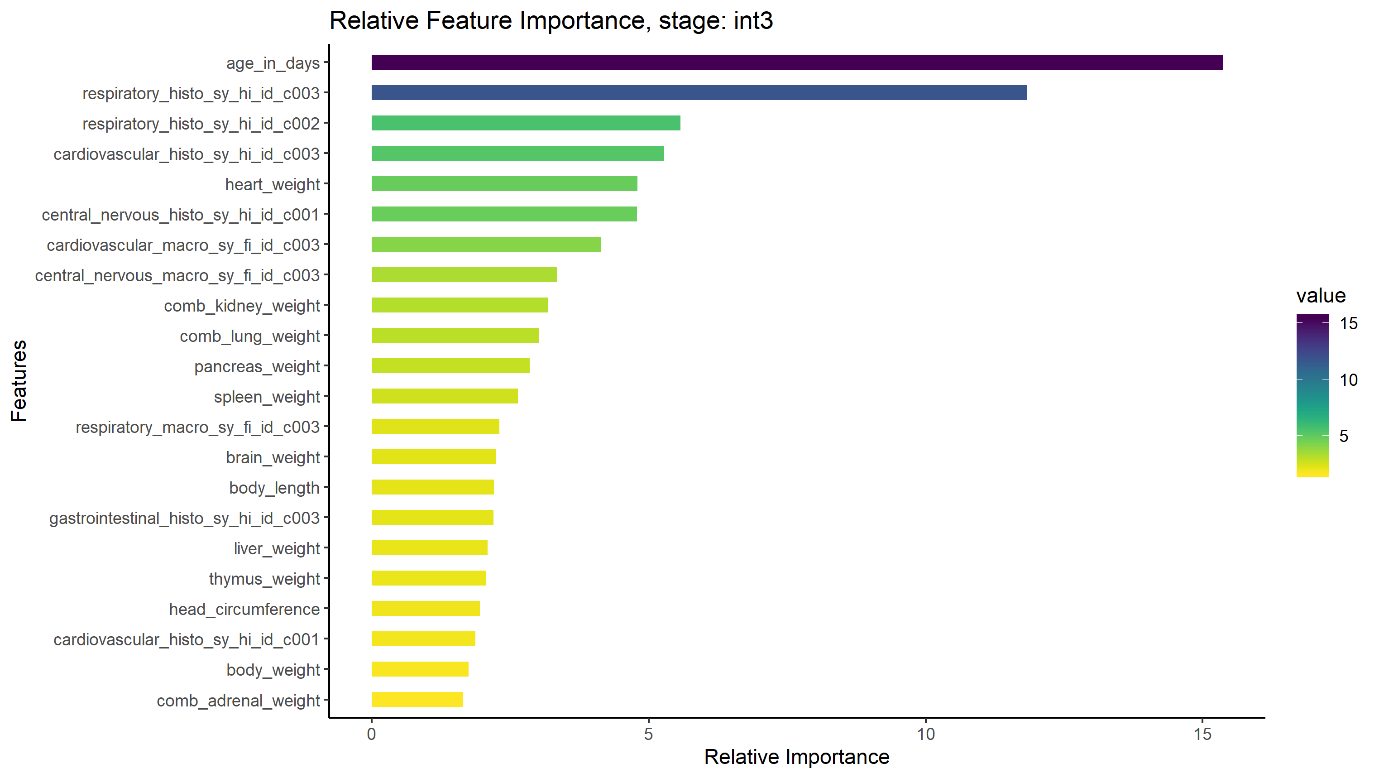


Figure 25 – Relative Feature Importance, Stage: Int3

# Conclusion

## Project Summary

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

### 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.

* 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.
* Generic routines to create and update events and event attributes
* Generic routines to add additional event attributes
* Generic routines to create RDVs for different stages for analysis
* Generic routines to carry out some data wrangling i.e. one-hot encoding.

### 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.

### 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.

## Project Evaluation

Identify areas for improvement; discuss what you could have done better (particularly important if you failed some of your targets or your results as not as expected)

* Python versus R
* Agile approach
  + Reviewed progress from last week
  + Set objectives for the coming week
* Underestimated time to tune each model
  + Needed a separate model for each stage.

## Recommendations for Future Work

* Improved measurement normalisation
* Imputation of missing values for certain measurements.
* Incorporate latest post-mortem results.

# References

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.

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.

(Example of using RPART package)

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

List of reference to be included:

Python

Pycharm

pyODBC

R

R studio

ggplot2

The viridis color palettes:

<https://cran.r-project.org/web/packages/viridis/vignettes/intro-to-viridis.html>

Rpart

<https://cran.r-project.org/web/packages/rpart/index.html>

Randomforest

<https://cran.r-project.org/web/packages/randomForest/>

<https://cran.r-project.org/web/packages/randomForest/randomForest.pdf>

XGBoost

<https://cran.r-project.org/web/packages/xgboost/>

<https://cran.r-project.org/web/packages/xgboost/xgboost.pdf>

Tuning xgboost in R: Part I

<https://insightr.wordpress.com/2018/05/17/tuning-xgboost-in-r-part-i/>

# Glossary of Terms

From the Royal College of Pathologists

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

|  |  |
| --- | --- |
| 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. |
| 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. |
| SQL |  |
|  |  |

# Appendix A – Example Project Code

Python Functions

Python ODBC

R Model

# Appendix B – ETL Process

# Appendix C – Cause of Death Attribute Mapping

# Appendix D –RDV structures

## External

## Internal – Stage 1 – Organs

## Internal – Stage 2 – Macro investigation

## Internal – Stage 3 – Histological investigation

# Appendix E – Detailed output from an analytic process

RDM file output

# Appendix F – Deliverables

Random Thoughts

* Python vs R
  + Why used when
* R code; evolved through three stages
  + Individual model
  + Model run for all PM stages
  + Model as a function called with multiple random keys.