REPORT

Exploring the Effect of the Characteristics of Convicts in Norfolk Island Penal Colony on Their Sentence Length

Haoting Chen

ANU | U7227871

Exploring the Effect of the Characteristics of Convicts in Norfolk Island Penal Colony on Their Sentence Length

Abstract

Norfolk Island was historically a penal colony for the most dangerous criminals from the British. This study aims to explore a demographic record of the island during its second operation as a penitentiary and find out how the characteristics of the convicts on this penal station affect their sentence length. This study is conducted in three steps: initial exploration through association mining to detect the important features associated with the sentence length, outlier detection through k-prototype (k-mean + k-mode) clustering to detect any surprising individuals, and finally, sentence length prediction through neural networks with the supply of elaborate features and the removal of outliers. The outcome of this study shows that criminal record, religion, literacy, and whether the convict would commit additional offences in prison are more or less associated with their sentence length. In addition, two neural network models have been built. One can predict whether a convict is sentenced to a fixed-term or lifetime sentence with an accuracy of 0.757 and an F1 of 0.691. The other can predict the actual length of the sentence for all fixed-term convicts with a mean average error of only 2.75 years. These show that the characteristics of convicts do have a strong relationship with the sentence length. The outcome of this study may assist historians in recovering the missing sentence length data in a convict record on the same island as other penal stations with highly similar demographic features. The relations between convict characteristics and their sentence length may also assist legal researchers to estimate the law at that time and study whether a trial was fair and just. The artifact of this research is available at <https://github.com/glowing-sea/norfolk-island>.

1. Introduction

During 1825 to 1855, Norfolk became a penal colony for the most desperate criminals from the British (Britannica, 2025). It was governed by Van Diemen's Land (Tasmania) and used as a place for secondary punishment for convicts in Van Diemen's Land and New South Wales (Channers on Norfolk, n.d.). A study conducted by Dr Tim Causer et al. has presented a valuable demographic dataset, named “VDL Norfolk Island Penal Colony” about the characteristics of the 6472 convicts in Norfolk Island during this penal colony period (Finnane, et al., n.d.)

. This research builds upon Dr Tim Causer’s work to further extract the interesting patterns from the dataset, particularly the effect of convicts’ characteristics on their sentence lengths. The goals are not only to find associations but also to train a model to predict the sentence length based only on the other characteristics of the convicts.

The resultant prediction model may help historical researchers to recover the missing information about the sentences of other convicts on the island, or other islands with a highly similar demographic distribution. The predicted sentence length is useful to track down the outcome of each convict, e.g., whether it was transferred to another penal station or potentially became an Australian citizen after being released from prison, which may further contribute to the study of other penal stations or regions where prisoners worked and lived after being released.

The outcome of this study can also help legal research to generate a picture of legislation and the judiciary. This becomes the final model built, which takes the convict’s characteristics, such as offence, criminal history, occupation, and outputs the sentence, essentially acting like a “judging machine”, and whether or not the sentence is fair and just. This model may only be used to study the convict at the same period, and the misuse of this model, e.g., judging a modern criminal, can lead to serious consequences because the law and society have changed significantly. This is a major ethical concern of this study.

To achieve the final goal of building a sentence length prediction model. This study has also set the three sub-goals, where each helps our study progress towards the final goals and corresponds to different data mining techniques.

- Goal 1: determining whether there are any relations (association) between convict characteristics and their sentence length, and what the key attributes are that affect their sentence length. (Association Mining via Apriori)
- Goal 2: exploring whether there are any unexpected individuals and perhaps relations, but excluding them when attempting Goal 3 (Outlier Removal via clustering)
- Goal 3: determining whether there is a strong relationship between a selected set of convict characteristics and their sentence length by evaluating how well a model can fit into the data for prediction. Through the evaluation of the model, it should also show how well the data mining outcome (the sentence length prediction) model can be used in practice for historical and legal researchers. (Regression and Classification via Neural Networks)

The remainder of this article is structured as follows: Section 2 covers the steps for preprocessing the initial dataset from Dr Tim Causer. Section 3 presents a general summary of the dataset. Section 4 outlines the detailed methods for exploring the effect of a convict's characteristics on sentence length, from association mining, outlier detection via clustering, and prediction via regression and classification. Section 5 presents the results. Section 6 concludes the article, and Section 7 provides directions for future work.

2. Data Preprocessing

The Norfolk Island Penal Colony dataset about the 6472 convicts used in this study consists of two files, "VDL_Norfolk_Island_Penal_Colony.txt" and "VDL_Norfolk_Island_Penal_Colony.csv". One is about the description of the dataset, and the other is the actual data. They are originally presented through the Dr Tim Causer et al. and are currently managed by The Australian Data Archive's (ADA) Australian Historical Criminal Justice Data Dataverse (Finnane, et al., n.d.). However, our study uses a pre-processed version, called "VDL_Norfolk_Island_Penal_Colony_pp.csv", presented by Dr Kerry Taylor and her teaching teams, which contains all the original columns and some additional columns with names postfix, "pp", which has pre-processed some unstructured and non-unified nominal values into a more meaningful form and convert some numerical values, e.g., height, currency, into modern units (Taylor, 2025). Even so, the data processing is still limited and requires further data cleaning. Therefore, this section covers the additional pre-processing steps done in Dr Kerry Taylor et al.'s version of the dataset. The Python code for performing data pre-processing is available at "./NLPC_pp.ipynb" in the artifact repository. The pre-processed dataset is stored as "./VDL_Norfolk_Island_Penal_Colony_Cleaned.csv".

The Norfolk Island Penal Colony dataset contains 46 attributes, excluding the added "pp" attributes. After our preprocessing, only 21 attributes (columns) are kept, and all values are non-null. This can be divided into 31 steps. The process log is shown in Appendix 1.

1. **Checking** "trial_id" to ensure it is unique and not null. All 6472 rows have passed, and thus we use it as the convict ID.
2. **Dropping** "def_firstname", "def_othername", "def_surname". The convict's names may be used to study how a family background affects a person to become a criminal, but we need more information to identify if two people with the same family name belong to the same household, as there are many common surnames.
3. **Dropping** "register_name". All convicts are registered to VLD because Norfolk Island was managed by Van Diemen's Land. It does not show where the original penal station was for each convict.
4. **Dropping** "def_sex". All non-null values in this column are MALE. According to the historical data, there were no women convicts in this penal colony (Wright, 2025).

5. **Dropping** “note”. The convict transfer notes are very interesting. However, it is unstructured, and we may need some natural language processing tools to convert it into more machine-readable values. This can be done easily nowadays by prompting a generative LLM, such as ChatGPT. However, none of the generative LLMs were available by the time this research was conducted.
6. **Processing** “offence_pp” into only 36 categories, including an added UNKNOWN category. Further, it groups “offence_pp” into a new column called “offence_pp_general,” which consists of only 10 categories. The original “offence_pp” contains too many (205) distinct attributes, where many of them do not make sense, such as “X” and “SENT”, while many of them are duplicated, such as “HOUSE”, “HOUSEBREAKING”, and “BURGLARY”. Since it may be too time-consuming to check through all distinct attributes, we only clean the most common distinct attribute, while setting others to UNKNOWN, assuming that we do not know it and it may be noise. This is also part of the outlier removal process for building a more robust prediction model based on Mean Squared Errors. Within these 30 distinct attributes, we further clean and merge them if necessary. However, we found that some attributes, such as “STEALING” and “HOUSE_STEALING” have hierarchical relations. Therefore, we further merge them into a new column where each distinct category value is at the same hierarchical level, while keeping the original columns for detailed offences. Offence type is highly related to sentence length in a common sense. **Dropping** the original “offence”.
7. **Processing** “sentence_pp” to drop any 1614 rows with null values where it is also null in “sentence_years_pp”. **Dropping** the original “sentence”, “sentence_years”, and “sentence_years_pp”. We must drop all rows without sentence length because this is the target label for our models.
8. **Processing** “trial_date” by splitting it into “trial_year” and “trial_month” and dropping 111 rows with null values. This may be useful as the law can change over time, affecting the sentence year.
9. **Processing** “trial_place” by cleaning and grouping into 17 distinct places and placing UNKNOWN for missing or infrequent values. The reason is the same as 6. However, we did not create new columns for 6. This may be useful as different place may have different laws, affecting the sentence year.
10. **Dropping** “verdict” as it is empty.
11. **Dropping** “line_number”, “fas_id”, and “manual_source” as they are the attribution data about the current data keepers, not about the convicts.
12. **Dropping** “ship”: the ship name that sent the convict to Australia may not be relevant to the sentence length.
13. **Dropping** “police_number”: the prison police may not be relevant to the original sentence length, and this column seems to be unique.
14. **Processing** “pris_ht_pp” (convict’s height) by replacing all missing values with the median height of 165.1. There are too many missing heights. Simply detecting all rows with null values is too expensive, and many other features may still be useful, such as detecting the sentence length. Median is used because it may represent the data’s mean and is robust to outliers. This may have some indirect relation to sentence length, so we tend to keep it, similarly for 15,16, 17, 18, 19, 20, 21, 28. **Dropping** “pris_ht”.
15. **Processing** “def_age” (convict’s age) and **dropping** “def_age_pp” in a similar way to 14.
16. **Processing** “def_literacy_pp” (convict’s literacy) by converting it into four levels, filling all non-values with the median. **Dropping** “def_literacy”.
17. **Processing** “def_religion_pp” (convict’s religion) by cleaning their names and placing UNKNOWN for all ambiguous or null values. **Dropping** “def_religion”.
18. **Processing** “martial_status_pp” (convict’s marital status) and **dropping** “martial_status” in a similar way to 17.
19. **Processing** “children_nr” (convict’s children count) in a similar way to 14.
20. **Processing** “cash_sav” (convict’s savings) and **dropping** “cash_sav” in a similar way to 14.
21. **Processing** “occupation” and **dropping** “occupation” in a similar way to 6.
22. **Dropping** “birth_country”, there are 1417 unique values, and it must contain duplication. This column can be interesting, but time and tool limitations have prevented us from using this column.
23. **Processing** “coloffence_info” (colony offence info) by converting it into a Boolean. If it is empty or contains just a dot, set it to False. This affects the convict's behaviour and may reflect their sentence length, similarly for 27.
24. **Dropping** “prev_transp” (whether the convict has been transported previously). The only unique values are [nan 'Y' 'AUSTRALIA'], which is a bit ambiguous.

25. **Dropping** “prev_transp” (criminal history). This column can highly affect sentence length. However, it is in text and require more advanced NLP pre-processing techniques.
26. **Dropping** “depni_date”, “depni_ship”, “arrivni_date”, as there are too many null values.
27. **Processing** “offence_ni” (additional offence in Norfolk Island) into Boolean, similarly to 23.
28. **Processing** “dead_in_custody_pp” (whether the convict died in custody) into Boolean, similarly to 23. **Dropping** “dead_in_custody_pp”
29. **Dropping** “probat_dur”. This column is too unstructured as it requires more advanced NLP.
30. **Based on** “date_arrived_aus” (date when arriving in Australia), “probat_date” (date of starting probation), “tl_date” (date of getting the tick of leave), **create** two new columns by computation: “length_of_stay_until_probat” and “length_of_stay_until_tl”. These two attributes are more informative and may reflect the behaviours of the convict in prison, and hence the sentence length for their original offence. **Dropping** the original columns. Inserting the median values for all null values.
31. **By matching** “date_1stkconv” (date of the first conviction) and “trial_month” and “trial_year” (the trial date of the conviction that led the person to Norfolk Island), determining whether a convict has additional previous offences before the current offence. Creating a new Boolean column called “previous_conviction”. This is highly relevant to sentence length. Dropping “trial_court”, “date_1stkconv”, “offence_1stkconv”, “trialplace_1stkconv”,”sentence_1stkconv”, “sentduryears_1stkconv” , and “court_1stkconv".

3. Data Summary

As for the original unprocessed dataset, as demonstrated in the previous section, it is complex and unstructured because it includes not only numerical and nominal values, but also unstructured text, which makes it hard for traditional algorithmic mining tools to find interesting patterns. In addition, many numerical values, such as heights and savings, contain non-unified units, and many nominal values contain duplicated instances with different name variations, typos and ambiguous names. Null values exist everywhere in the dataset. Therefore, the dataset quality is low. It is even hard to perform basic statistical analysis without encountering errors.

After the preprocessing, the dataset now contains 21 attributes and 4747 observations (convicts). There is no null value in the dataset; null values are either replaced by UNKNOWN or the mean values. The raw Python data summary and Rattle summary are shown in Appendices 2 and 3, respectively. The description of the new columns and their partial summary is as follows. N=Numerical, C=Category, N/B=Binary but treated as Numerical, N/O=Ordinal but treated as Numerical. (XX/XX%) after “—” indicate the number and percentage of UNKNOWN values or dummy median values that are inserted into rows with null or too infrequent values. All binary attributes have (0/0.00%) because FALSE include the meaning of UNKNOWN. This can reflect how much these attributes tell meaningful information about the data.

1. (N) trial_id: Convict’s trial ID, but also used for convict ID. – (0/0.00%)
2. (C) offence_pp_general: a set of 10 distinct offence types, including UNKNOWN. – (1940/40.86%)
3. (C) offence_pp: a set of 36 more detailed offence types, including UNKNOWN. – (1940/40.87%)
4. (N) pp_sentence_years: sentence years from 1 to 99 (Lifetime). – (0/0.00%)
5. (N) trial_month: month of the trial. – (0/0.00%)
6. (N) trial_year: year of the trial, ranging from 1825 to 1855. – (0/0.00%)
7. (C) trial_place: place of the trial. – (3784/79.71%)
8. (N) pris_ht_pp: height in cm. – (1794/36.84%)
9. (N) def_age_pp: age from 9 to 70. – (203/4.28%)
10. (N/O) def_literacy: literacy level from 0 to 4. – (746/15.72%)
11. (C) def_religion_pp: a set of 16 religion types, including UNKNOWN. – (836/17.61%)
12. (C) marital_status_pp: either S(single), M(married), W(Widowed), or UNKNOWN. – (690/14.54%)
13. (N) children_nr: number of children. – (4004/84.35%)
14. (N) cash_sav_pp: amount of savings. – (4674/98.46%)

15. (C) occupation_pp: a set of 22 occupation distinct types, including UNKNOWN. – (1941/40.89%)
16. (N/B) coloffence_info: whether the person committed an additional offence in a penal colony. – (0/0.00%)
17. (N/B) offence_ni: whether the person committed an additional offence in Norfolk Island. – (0/0.00%)
18. (N/B) death_in_custody_pp: whether the person died in custody, with 753 true values. – (0/0.00%)
19. (N) length_of_stay_until_probat: the length of stay until the probation starts. – (4028/84.85%)
20. (N) length_of_stay_until_tl: the length of stay until getting the ticket of leave. – (4283/90.23%)
21. (N/B) previous_convictions: whether the person has previous convictions in addition to the current conviction. – (935/19.70%)

Based on the partial summary above, the 6472 convict records (which has been reduced to 4747 after processing) was like censored from the whole population of the convicts in Norfolk Island during the approximate whole penal station period (1825-1855) because the range of the trial years (trial_year) of convicts is 1822 – 1859 and additional information shows that there were at most 2000 convicts in the island (Channers on Norfolk, n.d.). 753 convicts died in Norfolk Island, suggesting that the environment is harsh, and the convicts may be tortured. The minimum age is 9, indicating that there may be a lack of legal protection for children. The minimum sentence year is 1, indicating that even minor criminals were sent to the island, showing that punishment may be unfair.

As for the data quality, the quality of the attributes "length_of_stay_until_tl", "length_of_stay_until_probat", "cash_sav_pp", "children_nr", and "trial_place" is poor, as it contains too many dummy values (originally null). This may generate a lot of uninvesting rules, e.g., "UNKNOWN -> UNKNOWN" during later association mining, distort the distance between two data points during clustering, and waste the regression model capacity (neurons) through fitting to dummy patterns. Therefore, these attributes need to be carefully handled or disabled.

Although the basic statistics have provided some useful observations for understanding this penal colony, they do not show any direct relation between sentence length and the rest of the convict attributes.

4. Methodology

In order for exploring the effect of convict characteristics on their sentence length and building a sentence length prediction model, this study has divided into three main stages: (1) association minting via the apriori algorithm for finding any important features or feature sets associated with sentence length (2) outlier detection and removal via k-prototype (k-mean + k-mode) for finding unexpected individual as well as pre-processing data for step 3. (3) sentence length prediction via neural network regressors and classifiers to determine whether the effect between convict characteristics and sentence length is strong enough to fit a model with high performance.

4.1 Association Mining via Aprico

The idea behind association mining is to find out whether the occurrence of an attribute or a set of attributes is associated with the occurrence of another attribute or a set of attributes in a set of transactions. The set of transactions here is the whole dataset, and each transaction corresponds to the record of each convict, consisting of an itemset of attribute values. Therefore, association is generated based on the possible combination of an element in a powerset of attributes between the RHS and the LHS, many association rules are meaningless but valid, such as {} -> {}, and we are only interested in the association rules with high support (0-1) and confidence (0-1). because many association rules are meaningless but valid, such as {} -> {}. Support indicates the rule's coverage in the dataset, and confidence means how valid the rule is. In addition, we only interested in rules with high lift, where lift =1 indicate the LHS and the RHS are independent and hence the association rule is not interesting, lift > 1 indicates that the occurrence of LHS likely lead to the occurrence of RHS, and lift < 1 indicates that the occurrence of LHS likely lead to the absence of the RHS.

To perform association mining, we have chosen the Apriori algorithm. The Apriori algorithm is one of the widely accepted tools for association mining and has been implemented in Rattle, a data mining toolkit (Taylor, 2025). Compared to the naïve rule that exhaustedly finds all association rule that satisfies the minimal support and confidence requirement. Apriori prunes the search space intelligently. For example, if a LHS->RHS rule is not frequent (support < min_support), any rules with LHS and the RHS that are the superset of this rules must not be frequent.

Association Mining through the Apriori algorithm is quite computationally feasible and, with modern computers, can generally finish in a second, provided that only rules satisfying a certain limit are generated. It can serve as the initial and quick glance at the relation between convict characteristics and sentence length, or potentially more interesting relations that can lead to future study. In our case, the primary purpose of the association is to detect important features to assist the feature selection during neural network training in step 3. However, association mining only works for nominal variables. Therefore, further data pre-processing is required.

4.2 Association Mining in Python (Pre-Processing) + Rattle

The first step for association mining is to convert all numerical variables, such as “pp_sentence_years” into categorical through grouping them into bins, convert ordinal variables, such as “pp_literacy” into categorical through ignoring ordering, and convert all binary variables into two meaning values, such “Has Colony Offence” and “No Colony Offence”. The python script “./NLPC_to_all_nominal.ipynb” in the repository contains the code for converting the original pre-pre-processed dataset “VDL_Norfolk_Island_Penal_Colony_Cleaned.csv” into “VDL_Norfolk_Island_Penal_Colony_Nominal.csv”.

After converting all 21 variables except ID (trial_id) into categories. The next step is to load the dataset into Rattle for association mining. Before running the Aprico algorithm, disable the default portion setting to use all rows and set all variables except ID (trial_id) to input.

Two rounds of association running of Aprico have been decided to maximise the chance of finding interesting rules. In the first round, set the support to 0.3, confidence to a high value of 0.8, and maximum number of rules to 3000 and use all the 20 available features. Then, search for any rule with “pp_sentence_years” in the RHS. This aims to find any strong association rules. However, some features, e.g., “place_of_trial” with many dummy values, may result in most of these 3000 tops essentially being the rules containing UNKNOWN, making them less meaningful, making it hard for humans to find more meaningful rules from them. Therefore, in the second round, the support and confidence are both set to low values of 0.1 and all five low-quality attributes “length_of_stay_until_tl”, “length_of_stay_until_probat”, “cash_sav_pp”, “children_nr”, and “trial_place” are disabled. This may help us discover more potentially interesting rules, even with low support, but with a plausible confidence.

Although association can help discover important features for building neural networks in the later stages, a feature that does not appear in top association rules does not necessarily mean it is not important for neural networks because neural networks are more capable of understanding and relating features to sentence length.

4.3 Outlier Detection and Removal via K-Prototype Clustering

The second stage aims to detect unexpected individuals and help pre-process the data for stage 3. This is because a Mean Squared Error (MSE), which is used for the error function when training a neural network regression model for predicting sentence length in stage 3, can suffer from outliers (Taylor, 2025).

The idea behind using clustering for outlier detection is that clustering is about finding similar objects in groups, and hence, objects that are far away from any group are likely outliers. There are different clustering methods, such as DBSCAN (Density-Based Spatial Clustering of Applications with Noise), K-means, K-modes, K-medoids, and K-prototypes (K-means + K-modes). DBSCAN can effectively detect outliers as it can intrinsically not classify an object to any class, whereas the K family algorithm can only estimate outliers by looking at object that is relatively distant from any cluster. However, DBSCAN is very sensitive to choices of hyperparameters so that it may be time-consuming. K-means is relatively easier to implement than K-mode. However, the K-means itself may be affected by outliers due to the use of the L2 norm. Given that the dataset consists of a combination of numerical and nominal variables, the K-prototype method is used because it can handle both cases.

The idea behind k-prototype is to repeat two steps iteratively: (1) for each point, finding a closest centroid for the point and assign the group of the centroid to the point, and (2) for each group of data points recomputing a new centroid through the mode (for nominal variable) of the group member or their average value (for numerical variable). The distance to a centroid is computed by combining the Hamming distance for the category part of the data point vectors and the Euclidean distance for the numerical part of the data points. Hamming distance is about counting the number of matching nominal attributes between two data point vectors and is the square root of their L2 norm.

4.4 Outlier Detection and Removal in Python

The Python script "./outlier_detection.ipynb" contains all the necessary code for building the K-prototype clustering model as well as finding and removing outliers. It first removes the ID and target columns, "trial_id" and "pp_sentence_years". The reason for excluding this column during clustering is that if we has removed any outlier rows from the dataset that is somewhat based on the "pp_sentence_years" attributes and later use the dataset for training a sentence length prediction model, we essentially expose some information about the test set and validation set to the model during, which can cause the model to overfit and our validation and test result become invalid. Since we assume that other characteristics are somewhat related to sentence length, an outlier detected based on the attributed excused "pp_sentence_years" should still be an outlier when all attributes are used. After removing "trial_id" and "pp_sentence_year", all numerical variables are normalised, which makes the computation of distance error weigh equally for all numerical variables.

The K-prototype is trained a number of times with different K values ranging from 5 to 20. The K value that is located at the elbow point in the Cost (Inertia) vs Number of Cluster (K) graph is selected as the final clustering model. Inertia is the loss function of K-means, indicating the sum of squared distances from each point to its assigned cluster centre.

After getting the best K value, the K-prototype algorithm is run again with the optimal value. Then, the distance (Euclidean/Hamming) between a data point and its centroid is computed for each data point. The top 5% distant data points are marked as outliers. Before removing these data points, we also check the features of these top 10 data points to see if there is any expected finding.

4.5 Sentence Length Detection via Neural Network Classifier and Regressor

After marking important features and removing outliers, the final step is to train a regressor based on some or all features to predict the sentence year value. However, the sentence years consist of a huge jump between fixed-term (around 1 to 20) sentences or lifetime sentences (recorded as 99 in this dataset). Mathematically, the sentence length function contains a discontinuity and a linear function, regardless of whether a single linear regression model or a neural model is used, can find it changing to fit this sentence length function. Therefore, two neural network models are trained, one determining whether the sentence is fixed-term or lifetime, the other estimating the actual sentence length for a fixed-term convict only.

There are many ways to perform classification and regression. For classification, we have logistic regression, feed-forward neural network with sigmoid/SoftMax output layer, support vector machine, etc. For regression, we have least-squares linear regression, feed-forward neural network, etc. Logistic and least-squares linear regression can be performed extremely fast on modern computers. However, they may be too simple to feed data with complex features.

Neural Network is one of the best choices because it is not only simple and runs fast on modern computers but also allows us to flexibly select a network architecture, which may lead to more improvement opportunities.

4.6 Sentence Length Detection in Python

The code for training a fixed-term vs lifetime classifier is located at "./sentence_length_prediction_lifetime_or_not.ipynb" in the repository. The code for training a sentence length predictor for all fixed-term convicts is located at "./sentence_length_prediction_no_lifetime.ipynb". The naïve implementation, which directly feeds a regressor to all sentences, regardless of whether they are fixed-term or lifetime, is located at "sentence_length_prediction_naive.ipynb" for comparison.

As for building the neuron network fixed-term vs lifetime classifier, since neural networks only accept numerical values, the first step is to convert all category values to numerical values. One of the methods is to convert each numerical variable into one-hot encoding, i.e., creating k more dimensions to represent a numerical variable, where k is the number of unique values in those numerical values. A data point with a specific original nominal value only has its corresponding dimension marked as 1, while other dimensions of that nominal variable are zeros. The drawback of this method is that it may make these dimensions weigh more than another dimension that only uses one dimension. However, this should be better than simply converting nominal variables into numerical values based on the random order of each distinct nominal value. This will inject random ordering information into the dataset and prevent the model from focusing on more interesting features. Next, all numerical attributes are normalised, telling the model that all features (excluding those one-hot columns transformed from nominal data) are equally important to the model. The last step before training the model is to split the dataset into training, validation, and testing sets. Ideally, feature selection should be done after the dataset splitting. This is because feature selection is an action of validating a model. If we select a feature before splitting the dataset, we may leak our validation and testing sets to the model during training. However, after converting all categorical variables to one-hot encoding, it can be hard to perform feature selection. Therefore, our solution is to set a seed for the dataset splitting function. In this case, it can be placed before a feature selection, before it is deterministic.

As for the model validation process, we have chosen the traditional training, validation, and testing set instead of a training and testing set only, but with 10-fold cross-validation. This is because we have large enough samples. However, the drawback is that the validation score for each model will be a fixed value instead of a distribution for the 10-fold cross-validation case. Therefore, we cannot make a statistically significant judgment about which models are better. The selection of an optimal model will mostly be based on inspection. Even so, the validation set method is roughly 10 times faster and enables us to test even more combinations of hyperparameters.

The adjusted hyperparameters are the network architecture, a set of enabled features, and the maximum epoch for early stopping. Instead of performing a grid search to search all possible combinations exhaustedly, trial-and-error is used, i.e., first select a reasonable default hyperparameter setup and dynamically choose a value to adjust to see if the loss is reduced. In this method, we ourselves act as an optimiser to work through the function space of hyperparameters. Finally, the model with the best hyperparameter setup is tested on the test set, and the score for the model is obtained.

The process for training a neural network sentence length regressor is similar, except that there is no need to add a SoftMax activation function to the output layer, and MSE is used as the error function instead of Cross Entropy Loss.

5. Results

5.1 Association Mining Results

The raw results for the two rounds of association mining are located at "./0.3-0.8-3000.pdf" and "./0.1-0.1-1000-LessUnknown.pdf" in the repository.

In the first round of association, minting there is no top 3000 rules with "pp_sentence_length" on the RHS. Among these 3000 rules, most of the rules are as expected, such as {coloffence_info = No Colony Offence} is associated with {previous_convictions: No Previous Conviction} with support = 0.48, confidence = 0.73 and lift=1.31, which indicates that this is a positive correlation. However, these top 3000 rules with all features enabled are dominated by binary variables and nominal variables with many UNKNOWNS. This is potentially because each of the specific values of a variable has a high occurrence. This makes sense for a binary variable that has a limited domain of only two values, and hence each value should have very high occurrences.

In the second round of association mining, where the attributes with many dummy values, including "length_of_stay_until_tl", "length_of_stay_until_probat", "cash_sav_pp", "children_nr", and "trial_place", and lower the support and confidence requirements, we have obtained some rules with "pp_sentence_length" on the RHS. For example, we have found that {previous_convictions=Has Previous Convictions} is positively correlated with {pp_sentence_years=(15.0, 99.0]} with support = 0.25, confidence = 0.56, and lift = 1.29, which aligns with common sense as a convict with previous convictions is likely sentenced to a long prison term. Other association rules includes:

1. {def_religion_pp=Protestant} => {pp_sentence_years=(15.0, 99.0]} S=0.23, C=0.41, L=0.94 (A protestant convict is unlikely to have a long sentence length.)
2. {def_literacy=Read or Both Little} => {pp_sentence_years=(15.0, 99.0]} S=0.18, C=0.50, L=1.15 (Criminals who commit serious crimes may need to be smart enough.)
3. {previous_convictions=No Previous Convictions} => {pp_sentence_years=(0.999, 10.0]} S=0.23, C=0.42, L=1.19 (Criminals without previous convictions are like having short sentence length, which is as expected)
4. {offence_ni=Has Offence in NI} => {pp_sentence_years=(0.999, 10.0]} S=0.17, C=0.39, L=1.11 (Criminals who commit additional offences in Norfolk Island were not sentenced to a long sentence. This is surprising, but it may also show that a secondary offence may not make their sentence length longer.)
5. {marital_status_pp=S, coloffence_info=No Colony Offence, previous_convictions=No Previous Convictions} => {pp_sentence_years=(0.999, 10.0]} S=0.17, C=0.45, L=1.27 (Single convicts with no colony offence and no previous convictions are likely sentenced to a short sentence length)

From association mining, we can find that the sentence length is likely affected by criminal record, religion, literacy, and whether the convict would commit additional offences. However, many of these rules are weak, e.g., with low support and confidence. However, help us mark out the important features that are potentially useful for training our final model. In addition, association mining can more explicitly tell us the relation between convict characteristics and sentence length, whereas neural networks typically act as a black box, and it can be hard to extract any relational rules for studying the specific features affecting the sentence length and how they affect.

5.2 Outlier Removal via Clustering Results

The results for outlier removal via clustering can be accessed through the same script of the cluster model, i.e, "./outlier_detection.ipynb". Figure 1 shows the cost of the k-prototype with different cluster (K) setups. The elbow of the graph is not very clear, indicating that there are potentially too many features in the algorithm to fit the data. However, by carefully observing, we find that the elbow point is approximately K=12. Therefore, we run K-prototype with K=12 and mark out the outlier as the top 5% distant object from its centroid. Among the top 10 distant data points, there are some unexpected findings. For example, the person with tiral_id = 698533 whose offence is "RECEIVING_STOLEN_GOODS" was sentenced to lifetime, which seems to be excessive punishment. This person has zero literacy capability, which means there is a possibility of being framed because of their ignorance.

After the outlier removal process, 238 outliers are removed, leaving 4509 convict records.

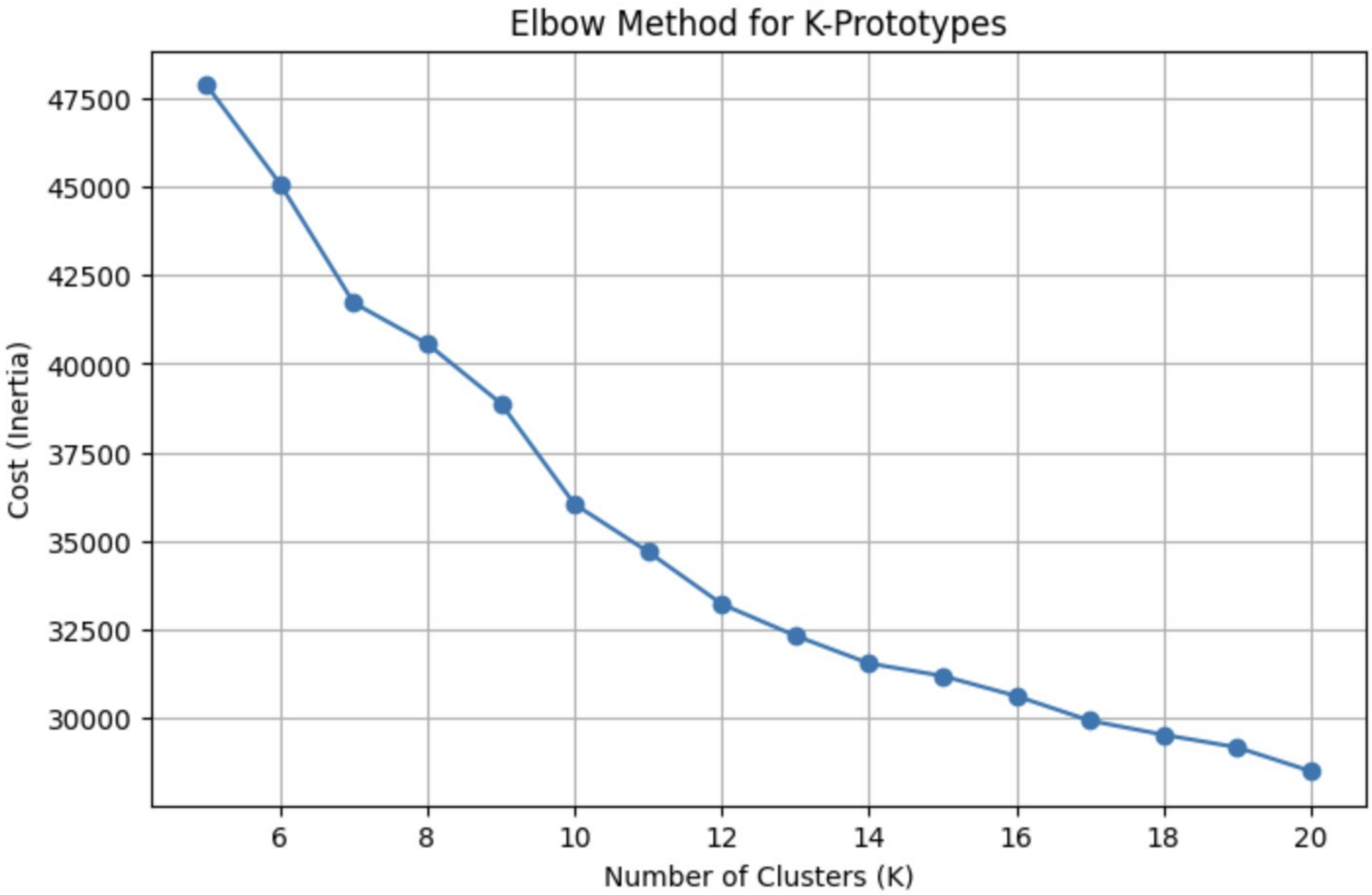

Figure 1: Cost vs. Number of Clusters

5.3 Naïve Sentence Length Prediction Results

Initially, all convicts, regardless of whether they are lifetime (99) or not, are used to train the model. The feature that we have adjusted is:

1. The set of enabled features. We have selected three set of features. (Some) The first set only exclude the 5 attributes with high dummy values, (Minimal) the second set only select the import features found in association mining, (All) the third set consists of all features. The details of these set can be found in Appendix 4.
2. The network architectures, starting from the default (75,50). Some variations include (1000,), (1000,50), and (75, 75, 50, 50)
3. By default, the maximum number of epochs starts from 1000. Some variations include 100, 200, and 500.

The complete validation log can be found in Appendix 5. Through several different trials, it was found that the model likely performs the best when all features are enabled, even though some of the features are less meaningful than others. This shows the strong capability of neural networks in learning features. The default model setup (75, 50) seems to be better than all other setups because this is based on the rule of thumb that the first hidden layer dimension is about 0.75 * the input dimension. In comparison, the second layer is about 0.5 * the input dimension. After one-hot encoding, when all features are enabled, the dimension of the actual input data is 119. Making the hidden layer smaller than the input layer can force the model to summarise the data by finding common features and avoid overfitting. The number of maximum epochs is also set to 100 to avoid overfitting. However, even though we have tried our best to adjust our hyperparameters, the model gets stuck at about 30 mean absolute error (MAE) of sentence length, which is very high. The final model with 100 maximum epochs, (75,50) architecture, and all features available has an MSE of 1279 and an MAE of 28 on the test set, which is unacceptable.

5.4 Fix-Term vs. Lifetime Classification

To overcome the discontinuity in the function space of sentence length, we first train a classifier to classify fixed-term and lifetime sentences only. A similar validation process is performed, and the final result shows that the neural network with architecture (75, 50), max_iter=200, and all features available performs the best with an accuracy of 0.757, an F-score of 0.676, and an AUC of 0.805, which is relatively fine. The complete validation log can be found in Appendix 6.

5.5 Sentence Length Prediction for Fix-Term Convicts Only

Similarly, we have found that the neural network with the architecture (75, 50), and max_iter=200 performs the best and surprisingly, the minimal feature set is chosen. The complete validation log can be found in Appendix 5. The final result on the test set is a MSE of 10.64 and MAE of 2.63 years, which is far less than 28 years for the naïve sentence length prediction case.

6. Conclusion

This study was conducted on the Norfolk Island Penal Colony dataset to explore the effect of convicts' characteristics on their sentence length. By performing association mining on the dataset, we have found that criminal record, religion, literacy, and additional offence in the colony are likely associated with their sentence length. By performing outlier removal, we have found some outliers, e.g., the trial_id = 698533 case. By performing classification and numerical prediction, we have successfully trained neural models to predict whether a convict was sentenced to a fixed-term sentence or not, and if so, what is their sentence length. Both the classification and numerical models have a fairly acceptable performance.

Regarding the three initial goals we have set, we have determined that any association between convict characteristics and these highlighted attributes has helped us select features when building the sentence length prediction model for fixed-term convicts. As for Goal 2, we have successfully explored any unexpected individuals and removed outliers for potentially better model training for sentence length prediction. As for the goal, we have built the neural network models that fit the training data and have acceptable results in the test data. This indicates a strong relationship between some and all convict features and their sentence. Although the sentence length prediction model for fixed-term convicts has a relatively low MAE of 2.63 years, the model for predicting whether a convict is fixed-term or lifetime does not have a very high accuracy (0.757) and F-score (0.676). This may lead to a few possibilities: there may be a better model available to fit the data well, the convict characteristics do not completely reflect their sentence length, or the dataset is too noisy. The result of misclassifying whether a convict is lifetime or fixed-term is serious. Therefore, the result may need further improvement before being used in practice, i.e., by legal and historical researchers. However, if we already know that a convict is fixed-term, the sentence length prediction model for fixed-term convicts can be deployed first, and its performance is relatively high.

As already highlighted in the introduction section, the result is useful for historical researchers to recover any missing historical demographic data on the same penal colony or another penal colony with highly similar demographic feature distributions. The model built may act as a "judging machine" at that specific time, determining a criminal sentence length based on their characteristic, including their offence and criminal history. This can also help legal research to recover any missing law at that time as well as determine whether the trials at that time were fair and just.

Future work continues to explore the effect of the characteristics of convict on their sentence length, but with many improvements. During data pre-processing, many potentially interesting features, such as “note,” have been ignored due to limited NLP capability. Future studies may leverage the advances of LLM to extract more useful and structured information or clean any attribute cells, typos, duplicated names, etc. During the data mining part, future work can try more data mining techniques, e.g., different classification and regression models, to hopefully extract more useful information. When doing hyperparameter tuning, a more systematic method may be used instead of trial and error. Future work may also explore a different aspect of the dataset, such as estimating the death rate. This is because many signals of patterns have been discovered in the basic dataset summary part.

Bibliography

Britannica, 2025. *Norfolk Island.* [Online]
Available at: https://www.britannica.com/place/Norfolk-Island
[Accessed 4 May 2025].

Channers on Norfolk, n.d.. *Norfolk Island A Brief History.* [Online]
Available at: https://www.channersonnorfolk.com/about-norfolk
[Accessed 4 May 2025].

Finnane, M., Causer, T. & Durnian, L., n.d. [Online].

Taylor, K., 2025. *Assignment 2.* [Online]
Available at: https://wattlecourses.anu.edu.au/mod/folder/view.php?id=3353588
[Accessed 4 May 2025].

Taylor, K., 2025. *Association Mining.* [Online]
Available at: https://wattlecourses.anu.edu.au/mod/book/view.php?id=3353513
[Accessed 5 May 2025].

Taylor, K., 2025. *Classification & Prediction: Linear Regression & Neural Nets.* [Online]
Available at: https://wattlecourses.anu.edu.au/mod/book/view.php?id=3353585
[Accessed 5 May 2025].

Wright, R., 2025. *Norfolk Island during the Second Settlement (1825-1855).* [Online]
Available at: https://www.postcolonialweb.org/australia/wright3.html
[Accessed 4 May 2025].

Appendix

Appendix 1: Data Pre-Processing Log

```
1 - Checking: trial_id
    trial_id is unique and not null
2 - Dropped Columns: def_firstname, def_othername, def_surname
3 - Dropped Columns: register_name
4 - Dropped Columns: def_sex
5 - Dropped Columns: notes
6 - Checking: offence
    Number of unique offences in offence_pp: 205
    Number of unique offences in offence_pp: 36
    Number of unique offences in offence_pp_general: 10
    Unique offences in offence_pp_general:
        STEALING: 2490
        UNKNOWN: 1940
        ASSAULT: 667
        ROBBERY: 570
        SEXUAL: 203
        FORGERY: 172
        PROCESSING: 169
        DESTRUCTION: 125
        ESCAPE: 105
        FINANCE: 31
    Unique offences in offence_pp:
        UNKNOWN: 1940
        BURGLARY: 1252
        STEALING: 779
        HIGHWAY_ROBBERY: 411
        ASSAULT: 206
        CATTLE_STEALING: 190
        ROBBERY: 159
        RAPE: 153
        SHEEP_STEALING: 135
        FORGERY: 124
        HORSE_STEALING: 115
        ARSON: 105
        RECEIVING_STOLEN_GOODS: 100
        MANSLAUGHTER: 77
        MURDER: 59
        UTTERING: 54
        CUTTING: 52
        STABBING: 50
        PICKING: 48
        SHOOTING: 45
        POSSESSING: 42
        ABSCONDING: 36
        BESTIALITY: 35
        DESERTION: 35
        BUSHRANGING: 34
        KILLING: 34
        PIRACY: 28
        OBTAINING: 27
        WOUNDING: 23
        BREAKING: 20
        PERJURY: 20
        STRIKING: 19
        SHOPLIFTING: 19
        COINING: 16
        SODOMY: 15
        EMBEZZLEMENT: 15
7 - Checking: sentence, sentence_pp, sentence_years, and sentence_years_pp
    Dropped 1614 rows with null pp_sentence_years values
    All pp_sentence_years values are now numeric and not null
8 - Checking: trial_date
    Dropped 111 rows with null trial_date values
    Min trial year: 1822
    Max trial year: 1859
9 - Checking: trial_place
    Number of null trial places has been replaced to 'UNKNOWN': 2202
    Unique trial places:
        UNKNOWN: 3784
        CENTRAL CRIMINAL COURT: 220
        YORKSHIRE: 127
        LIVERPOOL: 105
        IRELAND: 100
        LONDON: 79
        AUSTRALIA: 69
        STAFFORD: 46
        WORCESTER ASSIZES: 31
        WARWICK ASSIZES: 28
        DURHAM ASSIZES: 24
        LINCOLN ASSIZES: 24
        CHELMSFORD ASSIZES: 23
        DERBY ASSIZES: 23
        SCOTLAND: 23
        MAIDSTONE ASSIZES: 21
        CHESTER ASSIZES: 20
10 - Dropped Columns: verdict
11 - Dropped Columns: line_number, fas_id, manual_source
12 - Dropped Columns: ship
13 - Dropped Columns: police_number
14 - Checking: pris_ht        pris_ht_pp
```

```
    1749 rows with null pris_ht_pp values, replaced with the median height 165.1
15 - Checking: def_age, def_age_pp
    203 rows with null def_age_pp values, replaced with the median age 23.0
16 - Checking: def_literacy, def_literacy_pp
    746 rows with null def_literacy values, replaced with the median literacy 2.0
17 - Checking: def_religion, def_religion_pp
18 - Checking: marital_status, marital_status_pp
    690 rows with null marital_status_pp values, replaced with the mode value UNKNOWN
19 - Checking: children_nr
    4004 rows with null children_nr values, replaced with the median value 2.0
20 - Checking: cash_sav, cash_sav_pp
    4674 rows with null cash_sav_pp values, replaced with the median value 48.0
21 - Checking: occupation, occupation_pp
    Number of UNKNOWN values in occupation_pp: 1941
22 - Dropped Columns: birth_country
    Number of unique birth countries: 1417
23 - Checking: coloffence_info
24 - Dropped Columns: prev_transp
25 - Dropped Columns: priorconvictions_text
26 - Dropped Columns: depni_date, depni_ship, arrivni_date
27 - Checking: offence_ni
28 - Checking: death_notes
    death_in_custody_pp contains only True or False
29 - Dropped Columns: probat_dur
30 - Checking: date_arrived_aus, probat_date, tl_date
    4028 rows with null length_of_stay_until_probat values
    4283 rows with null length_of_stay_until_tl values
31 - Checking: trial_court, date_1stkconv, offence_1stkconv, trialplace_1stkconv, sentence_1stkconv, sentduryears_1stkconv, court_1stkconv
    935 convictions's previous convictions are UNKNOWN
    2102 convictions has previous convictions
    1710 convictions has no previous convictions
```

Appendix 2: Python Data Summary

```
Column 1: trial_id, Imputed Value Count: 0, Percentage: 0.00%
    Range: 695735 - 714857
    Mean: 704675.7345692016
    Median: 699089.0
    Mode: 695735
    Standard deviation: 7691.5461735420695
Column 2: offence_pp_general, Imputed Value Count: 1940, Percentage: 40.87%
    Number of unique values: 10
    Unique values:
        STEALING: 2429
        ASSAULT: 639
        ROBBERY: 564
        UNKNOWN: 367
        SEXUAL: 198
        FORGERY: 165
        PROCESSING: 160
        DESTRUCTION: 122
        ESCAPE: 73
        FINANCE: 30
    Imputed Value Count: 367
Column 3: offence_pp, Imputed Value Count: 1940, Percentage: 40.87%
    Number of unique values: 36
    Unique values:
        BURGLARY: 1241
        STEALING: 741
        HIGHWAY_ROBBERY: 406
        UNKNOWN: 367
        ASSAULT: 191
        CATTLE_STEALING: 182
        ROBBERY: 158
        RAPE: 148
        SHEEP_STEALING: 132
        FORGERY: 122
        HORSE_STEALING: 114
        ARSON: 103
        RECEIVING_STOLEN_GOODS: 94
        MANSLAUGHTER: 76
        MURDER: 55
        CUTTING: 52
        UTTERING: 49
        PICKING: 48
        STABBING: 47
        SHOOTING: 45
        POSSESSING: 40
        BESTIALITY: 35
        DESERTION: 35
        KILLING: 34
        BUSHRANGING: 33
        OBTAINING: 26
        WOUNDING: 23
        PIRACY: 23
        PERJURY: 20
        BREAKING: 19
        STRIKING: 19
        SHOPLIFTING: 19
        COINING: 16
        SODOMY: 15
        EMBEZZLEMENT: 14
        ABSCONDING: 5
    Imputed Value Count: 367
Column 4: pp_sentence_years, Imputed Value Count: 0, Percentage: 0.00%
```

```
  Range: 1.0 - 99.0
  Mean: 47.08573836107015
  Median: 15.0
  Mode: 99.0
  Standard deviation: 43.377905302757696
Column 5: trial_month, Imputed Value Count: 0, Percentage: 0.00%
  Range: 1 - 12
  Mean: 6.253001895934275
  Median: 7.0
  Mode: 3
  Standard deviation: 3.3588930938079105
Column 6: trial_year, Imputed Value Count: 0, Percentage: 0.00%
  Range: 1822 - 1859
  Mean: 1841.1468295765746
  Median: 1843.0
  Mode: 1839
  Standard deviation: 5.295138062581291
Column 7: trial_place, Imputed Value Count: 3784, Percentage: 79.71%
  Number of unique values: 17
  Unique values:
    UNKNOWN: 3784
    CENTRAL CRIMINAL COURT: 220
    YORKSHIRE: 127
    LIVERPOOL: 105
    IRELAND: 100
    LONDON: 79
    AUSTRALIA: 69
    STAFFORD: 46
    WORCESTER ASSIZES: 31
    WARWICK ASSIZES: 28
    DURHAM ASSIZES: 24
    LINCOLN ASSIZES: 24
    CHELMSFORD ASSIZES: 23
    DERBY ASSIZES: 23
    SCOTLAND: 23
    MAIDSTONE ASSIZES: 21
    CHESTER ASSIZES: 20
  Imputed Value Count: 3784
Column 8: pris_ht_pp, Imputed Value Count: 1749, Percentage: 36.84%
  Range: 121.92 - 200.66
  Mean: 165.01331788497998
  Median: 165.1
  Mode: 165.1
  Standard deviation: 6.322965946898381
Column 9: def_age_pp, Imputed Value Count: 203, Percentage: 4.28%
  Range: 9.0 - 70.0
  Mean: 25.49610280176954
  Median: 23.0
  Mode: 23.0

  Standard deviation: 8.414447345248856

Column 10: def_literacy, Imputed Value Count: 746, Percentage: 15.72%

  Range: 0.0 - 4.0

  Mean: 1.998525384453339

  Median: 2.0

  Mode: 3.0

  Standard deviation: 1.1011802347174124

Column 11: def_religion_pp, Imputed Value Count: 836, Percentage: 17.61%

  Number of unique values: 16

  Unique values:

    Protestant: 2699

    Roman Catholic: 1076

    UNKNOWN: 836

    Presbyterian: 83

    Jewish: 12

    Church of England: 9

    Methodist: 6

    Baptist: 6

    Pagan: 5

    Wesleyan Methodist: 4

    Independent: 3

    Lutheran: 3

    Quaker: 2

    Unitarian: 1

    Dissenter: 1

    Muslim: 1

  Imputed Value Count: 836

Column 12: marital_status_pp, Imputed Value Count: 690, Percentage: 14.54%

  Number of unique values: 4

  Unique values:

    S: 3092

    M: 835

    UNKNOWN: 690

    W: 130

  Imputed Value Count: 690

Column 13: children_nr, Imputed Value Count: 4004, Percentage: 84.35%

  Range: 0.0 - 12.0

  Mean: 2.1301874868337896

  Median: 2.0

  Mode: 2.0

  Standard deviation: 0.824399324563902

Column 14: cash_sav_pp, Imputed Value Count: 4674, Percentage: 98.46%

  Range: 0.0 - 214.0

  Mean: 48.282704866231306

  Median: 48.0

  Mode: 48.0

  Standard deviation: 6.821247420029412
```

```
Column 15: occupation_pp, Imputed Value Count: 1941, Percentage: 40.89%
  Number of unique values: 22
  Unique values:
    UNKNOWN: 1941
    LABOURER: 965
    FARM LABOURER: 634
    ERRAND: 149
    SHOEMAKER: 130
    GROOM: 110
    DOMESTIC: 101
    TAILOR: 94
    WEAVER: 70
    CARPENTER: 66
    BUTCHER: 63
    SOLDIER: 62
    SEAMAN: 47
    BAKER: 46
    CLERK: 45
    SHEPHERD: 38
    STONEMASON: 34
    STABLE: 32
    BRICKLAYER: 30
    BOATMAN: 30
    SAILOR: 30
    COLLIER: 30
  Imputed Value Count: 1941
Column 16: coloffence_info, Imputed Value Count: 0, Percentage: 0.00%
  Range: False - True
  Mean: 0.24773541183905626
  Median: 0.0
  Mode: False
  Standard deviation: 0.43174279942349836
Column 17: offence_ni, Imputed Value Count: 0, Percentage: 0.00%
  Range: False - True
  Mean: 0.4451232357278281
  Median: 0.0
  Mode: False
  Standard deviation: 0.4970317717760989
Column 18: death_in_custody_pp, Imputed Value Count: 0, Percentage: 0.00%
  Range: False - True
  Mean: 0.15862650094796713
  Median: 0.0
  Mode: False
  Standard deviation: 0.365365920056314
Column 19: length_of_stay_until_probat, Imputed Value Count: 4028, Percentage: 84.85%
  Range: 1.7178082191780821 - 30.92876712328767
  Mean: 3.8472182863870774
  Median: 3.649315068493151
```

```
   Mode: 3.649315068493151
   Standard deviation: 1.5171074367282256
Column 20: length_of_stay_until_tl, Imputed Value Count: 4283, Percentage: 90.23%
   Range: 2.9095890410958902 - 27.432876712328767
   Mean: 7.915694988327162
   Median: 7.795890410958904
   Mode: 7.795890410958904
   Standard deviation: 1.5114942902475113
Column 21: previous_convictions, Imputed Value Count: 935, Percentage: 19.70%
   Range: False - True
   Mean: 0.4428059827259322
   Median: 0.0
   Mode: False
   Standard deviation: 0.4967704007697412
```

Appendix 3: Rattle Data Summary

```
trial_id      offence_pp_general  offence_pp       pp_sentence_years  trial_month      trial_year
 Min.   :695735   STEALING:2429     Length:4747       Min.   : 1.00     Min.   : 1.000   Min.   :1822
 1st Qu.:696922   ASSAULT : 639     Class :character   1st Qu.:10.00     1st Qu.: 3.000   1st Qu.:1839
 Median :699089   ROBBERY : 564     Mode  :character   Median :15.00     Median : 7.000   Median :1843
 Mean   :704676   UNKNOWN : 367                        Mean   :47.09     Mean   : 6.253   Mean   :1841
 3rd Qu.:712305   SEXUAL  : 198                        3rd Qu.:99.00     3rd Qu.: 9.000   3rd Qu.:1845
 Max.   :714857   FORGERY : 165                        Max.   :99.00     Max.   :12.000   Max.   :1859
                  (Other) : 385
```

```
trial_place     pris_ht_pp      def_age_pp    def_literacy           def_religion_pp
 UNKNOWN               :3784   Min.   :121.9   Min.   : 9.0   Min.   :0.000   Protestant       :2699
 CENTRAL CRIMINAL COURT: 220   1st Qu.:162.6   1st Qu.:20.0   1st Qu.:2.000   Roman Catholic   :1076
 YORKSHIRE             : 127   Median :165.1   Median :23.0   Median :2.000   UNKNOWN          : 836
 LIVERPOOL             : 105   Mean   :165.0   Mean   :25.5   Mean   :1.999   Presbyterian     :  83
 IRELAND               : 100   3rd Qu.:167.6   3rd Qu.:28.0   3rd Qu.:3.000   Jewish           :  12
 LONDON                :  79   Max.   :200.7   Max.   :70.0   Max.   :4.000   Church of England:   9
 (Other)               : 332                                                  (Other)          :  32
```

```
marital_status_pp  children_nr     cash_sav_pp     occupation_pp      coloffence_info offence_ni
 M      : 835      Min.   : 0.00   Min.   :  0.00   Length:4747        Mode :logical   Mode :logical
 S      :3092      1st Qu.: 2.00   1st Qu.: 48.00   Class :character   FALSE:3571      FALSE:2634
 UNKNOWN: 690      Median : 2.00   Median : 48.00   Mode  :character   TRUE :1176      TRUE :2113
 W      : 130      Mean   : 2.13   Mean   : 48.28
                   3rd Qu.: 2.00   3rd Qu.: 48.00
                   Max.   :12.00   Max.   :214.00
```

```
 death_in_custody_pp length_of_stay_until_probat length_of_stay_until_tl previous_convictions
 Mode :logical       Min.   : 1.718              Min.   : 2.910          Mode :logical
 FALSE:3994          1st Qu.: 3.649              1st Qu.: 7.796          FALSE:2645
 TRUE :753           Median : 3.649              Median : 7.796          TRUE :2102
```

```
Mean   : 3.847        Mean   : 7.916
3rd Qu.: 3.649        3rd Qu.: 7.796
Max.   :30.929        Max.   :27.433
```

Appendix 4: Three Feature Sets (Some, Minimal, All)

```
# A mapping to boolean indicating that whether a feature is enabled, do not use enabled_features
ENABLED_FEATURE_MAPPING = {
    'offence_pp_general': True,
    'offence_pp': True,
    'pp_sentence_years': True,
    'trial_month': True,
    'trial_year': True,
    'trial_place': True,
    'pris_ht_pp': True,
    'def_age_pp': True,
    'def_literacy': True,
    'def_religion_pp': True,
    'marital_status_pp': True,
    'children_nr': False,
    'cash_sav_pp': False,
    'occupation_pp': True,
    'coloffence_info': True,
    'offence_ni': True,
    'death_in_custody_pp': True,
    'length_of_stay_until_probat': False,
    'length_of_stay_until_tl': False,
    'previous_convictions': True
}
# 4 disabled features


ENABLED_FEATURE_MAPPING = {
    'offence_pp_general': True,
    'offence_pp': False,
    'pp_sentence_years': True,
    'trial_month': False,
    'trial_year': False,
    'trial_place': False,
    'pris_ht_pp': False,
    'def_age_pp': True,
    'def_literacy': True,
    'def_religion_pp': True,
    'marital_status_pp': True,
    'children_nr': False,
    'cash_sav_pp': False,
    'occupation_pp': False,
    'coloffence_info': True,
    'offence_ni': True,
    'death_in_custody_pp': True,
    'length_of_stay_until_probat': False,
    'length_of_stay_until_tl': False,
    'previous_convictions': True
}
# # offence_pp_general, def_age_pp, def_literacy, def_religion_pp, marital_status_pp, coloffence_info, offence_ni, death_in_custody_pp, previous_convictions


ENABLED_FEATURE_MAPPING = {
    'offence_pp_general': True,
    'offence_pp': True,
    'pp_sentence_years': True,
    'trial_month': True,
    'trial_year': True,
    'trial_place': True,
    'pris_ht_pp': True,
    'def_age_pp': True,
    'def_literacy': True,
    'def_religion_pp': True,
    'marital_status_pp': True,
    'children_nr': True,

    'cash_sav_pp': True,

    'occupation_pp': True,

    'coloffence_info': True,

    'offence_ni': True,

    'death_in_custody_pp': True,

    'length_of_stay_until_probat': True,

    'length_of_stay_until_tl': True,

    'previous_convictions': True

}
```

Appendix 5: Validation Log (Naïve Sentence Length Prediction)

```
============ (Some Feature)

First Column: Max Itertion

1000 Epoches Architecture: (1000,), MSE: 2324.4454276986835, MAE: 37.1584469349519251

1000 Architecture: (500,), MSE: 2181.95853019564, MAE: 35.99034732203381

500 Architecture: (1000,), MSE: 1932.3934069157153, MAE: 33.865427466247425

￼ 200 Architecture: (1000,), MSE: 1615.05489950153, MAE: 31.470802826295447

100 Architecture: (1000,), MSE: 1478.7479848709177, MAE: 30.699558931551657

50 Architecture: (1000,), MSE: 1438.0062482402577, MAE: 31.127513013572

100 Architecture: (100,), MSE: 1440.0273704409508, MAE: 31.291837112408327

100 Architecture: (100,), MSE: 1560.431409866534, MAE: 31.447067932695038

100 Architecture: (2000,), MSE: 1718.8692000533042, MAE: 31.98332350524717

50 Architecture: (1000, 500), MSE: 1686.6705418832212, MAE: 31.408358563081
```

```
1000 Architecture: (50, 20), MSE: 2479.2243299871207, MAE: 38.47705596126912

============ (Minimal Features)

200 Architecture: (1000,), MSE: 1540.7836708109999, MAE: 32.49223126294314


1000 Architecture: (), MSE: 1677.6973524338503, MAE: 37.11726095411442

1000 Architecture: (15, 15, 7), MSE: 1619.6085397249078, MAE: 33.129402323097224


============ (All Features)

ALL

1000 Architecture: (75, 50), MSE: 2551.7823971712646, MAE: 38.14995821550021

500 Architecture: (75, 50), MSE: 2095.904746922236, MAE: 34.80440770488276


200 Architecture: (75, 50), MSE: 1585.117727638199, MAE: 30.039100045922726

    100 Architecture: (75, 50), MSE: 1404.0056179416201, MAE: 29.446296671772668 , max_iter: 100, actual_iter: 100

                Final evaluation on test set, Architecture: (75, 50), MSE: 1279.7627051022773, MAE: 28.04515672698975


70 Architecture: (75, 50), MSE: 1383.8403876166178, MAE: 29.823632084518707

50 Architecture: (75, 75, 75, 50, 50), MSE: 1780.407529108055, MAE: 29.77973664939731

100 Architecture: (75, 75, 75, 50, 50), MSE: 2156.243190320355, MAE: 30.79972777529074
```

Appendix 6: Validation Log (Fix-Term vs. Lifetime Classifier)

```
===== (Some Features)

Arc=(75, 50):  ACC=0.710,  F1=0.630,  ROC-AUC=0.764, Threshold=0.500, max_iter=200, actual_iter=48
```

```
====== (Minimal Features)

Arc=(75, 50):  ACC=0.700,  F1=0.577,  ROC-AUC=0.731, Threshold=0.500, max_iter=200, actual_iter=17

======


  Arc=(75, 50):  ACC=0.746,  F1=0.676,  ROC-AUC=0.805, Threshold=0.500, max_iter=200, actual_iter=37

        Final evaluation: ACC=0.757, F1=0.691, ROC-AUC=0.747, Threshold=0.500, max_iter=200, actual_iter=37


Arc=(1000,):  ACC=0.741,  F1=0.655,  ROC-AUC=0.807, Threshold=0.500, max_iter=200, actual_iter=30

Arc=(75, 75, 50, 50):  ACC=0.734,  F1=0.669,  ROC-AUC=0.797, Threshold=0.500, max_iter=200, actual_iter=25

Arc=50:  ACC=0.738,  F1=0.658,  ROC-AUC=0.800, Threshold=0.500, max_iter=200, actual_iter=35
```

Appendix 7: Validation Log (Fix-Term vs. Lifetime Classifier)

```
====== (Some Features)

Architecture: (75, 50), MSE: 19.27299976698429, MAE: 3.4514246777272026, max_iter: 1000, actual_iter: 449

======= (Minimal Features)

Architecture: (75, 50), MSE: 15.027725170372658, MAE: 3.063528730296996, max_iter: 1000, actual_iter: 504

Architecture: (75, 50), MSE: 12.972420014547186, MAE: 2.8437690324737495, max_iter: 200, actual_iter: 200

  Architecture: (75, 50), MSE: 10.772805025181253, MAE: 2.6038333448758095, max_iter: 100, actual_iter: 100
        Final evaluation on test set, Architecture: (75, 50), MSE: 11.826674711145602, MAE: 2.7466945541026093

Architecture: (75, 50), MSE: 10.646290874885588, MAE: 2.6393319020040127, max_iter: 50, actual_iter: 50

======= (All Features)

Architecture: (75, 50), MSE: 16.83820040466013, MAE: 3.197953742791245, max_iter: 1000, actual_iter: 447
```