FACULDADE DE ENGENHARIA DA UNIVERSIDADE DO PORTO

Development of a predictive model for multimorbidity in adolescents

Rodrigo Melo Abrantes

Masters in Informatics Engineering

Supervisor: Prof. Rui Camacho

Co-Supervisor: Dr. Susana Santos

Co-Supervisor: Dr. Carolina Anele

July 25, 2024

Development of a predictive model for multimorbidity in adolescents

Rodrigo Melo Abrantes

Masters in Informatics Engineering

July 25, 2024

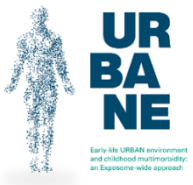
Abstract

Multimorbidity is a condition that refers to the presence of two or more long-term diseases in an individual. A very large amount of variables can contribute to the appearance of these long-term conditions, and it is not trivial to pinpoint precisely which factors play a more prominent role in the probability of these disorders appearing. The environment and a person's exposure to its different factors substantially influence the appearance of long-term diseases. Multiple urban environmental exposures, such as air pollution, greenness, noise, traffic, and ambient temperature, can affect a person's health. The contact with these various factors during pregnancy and childhood has been related to the appearance of a diverse set of health outcomes. To investigate this further, ISPUP researchers gathered environmental data concerning hundreds of children from birth until they were 13 years old. The purpose was to assess the associations of multiple urban environmental exposures from birth onward with multimorbidity in adolescence, with a particular interest in identifying the urban exposures that impact the appearance of health conditions the most and the critical windows of vulnerability to such exposure. By applying data mining techniques to this dataset, more specifically, different classification algorithms, we intend to understand which environmental exposures correlate the most with the appearance of multimorbidity. This research aims to provide insights that can lead to the future implementation of preventive measures, hopefully reducing the probability of multimorbidity cases appearing in a person's life as early as possible.

Acknowledgments

The work presented in this thesis was embedded in the Generation XXI (G21) birth cohort study. We gratefully acknowledge the families enrolled in Generation XXI for their kindness, the participating hospitals and their staff for their help and support, and all previous and current members of the research and field team for their enthusiasm and perseverance. The G21 was funded by Programa Operacional de Saúde – Saúde XXI, Quadro Comunitário de Apoio III and Administração Regional de Saúde Norte (Regional Department of Ministry of Health). The G21 is coordinated and conducted by several research groups from the Epidemiology Research Unit (EPIUnit) of the Institute of Public Health of the University of Porto (ISPUP), which was supported by FCT - Fundação para a Ciência e Tecnologia, I.P. through the projects with references UIDB/04750/2020 and LA/P/0064/2020 and DOI identifiers
https://doi.org/10.54499/UIDB/04750/2020 and https://doi.org/10.54499/LA/P/0064/2020. All phases of the study complied with the Ethical Principles for Medical Research Involving Human Subjects expressed in the Declaration of Helsinki. The baseline and follow-up evaluations were approved by the University of Porto Medical School/ S. João Hospital Centre Ethics Committee, except the 13-year follow-up that was approved by the ISPUP Ethics Committee. At baseline and follow-up evaluations, all procedures were explained to participants, and an informed consent was signed by one of the parents or legal guardians (at 13 years, it was also signed by the participants). The baseline evaluation was additionally approved by the Data Protection National Commission and the study follows the present EU General Data Protection Regulation under close supervision of the Data Protection Office of ISPUP.

This work is part of the URBANE project (European Union´s Horizon Europe Research and Innovation Programme under the Marie Sklodowska-Curie Postdoctoral Fellowship Grant Agreement No. 101109136). Views and opinions expressed are however those of the authors only and do not necessarily reflect those of the European Union or the European Research Executive Agency (REA). Neither the European Union nor the granting authority can be held responsible for them.



Contents

List of Figures

List of Tables

Abbreviations and symbols

URBANE	Early-life URBAN environmental and childhood multimorbidity
ISPUP	Instituto de Saúde PÚblica do Porto
CRISP-DM	CRoss-Industry Standard Process for Data Mining
CART	Classification And Regression Trees
SVM	Support Vector Machines
SVC	Support Vector Classifier
XGBoost	Extreme Gradient Boosting
AdaBoost	Adaptative Boosting
SMOTE	Synthetic Minority Over-sampling Technique
multiSN	Binary multimorbidity classification problem
multi3cat	Three-class multimorbidity classification problem
KNN	K-Nearest Neighbors
IQR	Inter-quartile Range
RFECV	Recursive Feature Elimination with Cross-Validation
MICE	Multiple Imputation by Chained Equations

Chapter 1

Introduction

The goal of this investigation is, as suggested by the thesis title, to develop a machine-learning model that can predict the appearance of multimorbidity in adolescents [2]. First, it is important to define multimorbidity. This condition refers to the presence of two or more long-term diseases in an individual. In medical terms, a long-term disease is a condition that affects an individual for three or more months. Diseases considered for multimorbidity can be physical, such as diabetes or cancer; mental, such as schizophrenia or depression; disabilities, like autism or blindness; or even persistent pain, like chronic back pain. A study conducted in 2017 [22], conducted over 11 years on a population of roughly 1100 individuals aged 78 years or older, found that:

- Multimorbidity was the most prevalent condition among the population, found in 70% of the individuals;
- 69% of the deaths that occurred during the study could be attributable to multimorbidity;
- Individuals with this condition lived on average 7.5 years less than their counterparts;
- 81% of the years following the contraction of the condition were lived with a disability.

This study highlights the impact of chronic conditions, particularly in the later stages of life. Reduced life expectancy and diminished quality of life constitute a primary concern for health professionals. This challenge can be addressed in two ways: treatment or prevention. The latter consists of promoting lifestyle modifications that can decrease the likelihood of developing chronic diseases. For the healthcare sector, the ability to prevent disease manifestation is far more cost-effective than subsequent treatment [19]. Preventive strategies not only reduce the economic burden associated with chronic disease management but also enhance individual quality of life, as even successful treatments can leave lasting adverse side effects.

In line with this preventive approach, medical experts at ISPUP have idealized a study that shifts the traditional focus of multimorbidity research, which typically involves older populations, to investigate factors that contribute to the development of this condition earlier in a patient's life. This perspective aims to pinpoint determinants of multimorbidity at a younger age, providing

insights that can trigger early intervention strategies and ultimately reduce the emergence of these chronic diseases.

The study carried out in this thesis uses real-world data from the *URBANE* project, led by Dr. Susana Santos and Dr. Carolina Anele from ISPUP. This research is specifically focused on the correlation between urban environments and their impact on multimorbidity during adolescence. The dataset used in our study, *Geração 21*, covers approximately 8600 children, monitored from pregnancy until 13 years of age. The primary objective of this data collection was to enable researchers to examine the impact of environmental exposures throughout the whole childhood so that conclusions could be drawn not only on which factors contribute the most but also at which moment in time their impact was more significant.

This research initiative has two main objectives. The first one is identifying the key factors that significantly influence the development of multimorbidity. The second objective is to identify the key time frames where contact with those exposures is more critical. To achieve this, we will train machine learning algorithms to facilitate a deep understanding of the underlying decision-making processes.

The ISPUP experts expect that the insights gained from this study will be valuable to medical researchers, as they will provide a means to address some of the causes of multimorbidity. To achieve this, we have to integrate the expertise of data science and medical domain knowledge to develop a comprehensive solution that offers insights to use for early interventions and subsequent preventions.

1.1 Motivation

The motivation behind this work is to reduce the prevalence of multimorbidity in the overall population. By conducting an innovative study on a younger population, we expect to find new information on how to prevent the appearance of multimorbidity. This will be achieved by discovering which factors have a higher impact on the contraction of the disease. We also aim to identify which populations and sub-groups are more susceptible to these factors and, therefore, are more sensitive to these exposures. Finally, we want to promote and expand the collaboration between health professionals and data scientists so that we can derive solutions and arrive at conclusions together that can have a strong impact on the quality of life of the overall population.

1.2 Objectives

We will apply data mining algorithms to the *Geração 21* dataset to pursue our motivation and achieve our research goals. We will also look to pinpoint precisely which urban factors contribute the most to the appearance of multimorbidity and in which time window exposure is critical. To improve the quality of the data mining models to be generated, we will continuously improve the models by discussing the intermediate results with the specialists and using their feedback and insights to improve our model's performance and degree of explainability.

1.3 Document Structure

This document will begin by defining what data is, and the methodology for extracting knowledge from it. Then, we will detail the data mining tasks, algorithms, and evaluation metrics required for model development, referencing [17] and [16].

The structure of this document will follow the CRISP-DM framework. In the Business Understanding section, we will delineate the investigation's objectives and goals, which will guide the selection of data mining tasks. The Data Understanding section will involve exploring and visualizing the raw dataset, detailing its characteristics. Data cleaning and transformation tasks will be described in the Data Preparation chapter. The Modelling chapters will detail the algorithms, inputs, and performance of the models. Finally, the Evaluation chapter will include a discussion of the drawn conclusion by the health experts.

Chapter 2

State of the art

Nowadays, data is considered one of the most important assets available. Corporations worldwide invest in resources and processes to collect and store data. The interest behind this behavior is to apply data mining processes that will allow them to extract valuable knowledge to give them a business edge.

The importance of data has been evolving [10]. In the late 80s, we can find the first references to data mining in the research community. The definition was still fresh, covering a broad collection of data processing techniques. The goal was clear: Find the hidden patterns in information, also called extracting knowledge from the data.

The first data mining applications appeared in sectors like Retail [8], with the Market Basket Analysis, where Retailers analyze the associations between frequently bought products. Then, with this information, they could restructure and optimize the layout of the supermarket and their promotional offerings to generate more revenue. The Financial sector was also one of the first to apply data mining to assess the creditworthiness of their client's loan approvals or to detect fraud in financial statements [21].

From there, data mining has reached almost every industry and has various applications. New techniques are constantly appearing, and with the evolution in data gathering and the increasing computing power available to process it, knowledge extraction is more powerful and business-centric than ever before.

A dataset constitutes a collection of attributes categorized by type and scale. Data types can be either binary, discrete, or continuous. Binary data comprises two values, frequently used to represent "yes-no" cases. Discrete data consists of a finite, non-divisible number of possible values, usually few. An example is the months in a year or the days in a week. Contrary to this concept, continuous data represents any real value in a range. Time, weight, and temperature are attributes that can be defined continuously.

The scale of an object's data refers to the relative significance of its numbers. Scale is divided into qualitative and quantitative.

The qualitative scale comprises nominal and ordinal values. Nominal values are numeric representations of non-numerical values, such as "yes-no" as (0,1), where the numbers have no mean-

ingful numerical significance. The ordinal scale is also a numeric representation of non-numeric values. Here, the numbers have significance when related to one another, such as a rating from 1-5 on a review site.

The quantitative scale comprises interval and ratio. Interval gives meaning to the gap between numbers and allows researchers to interpret them accordingly. For example, the difference between a student obtaining 16 or 18 on a test scored from 0 to 20 is much more significant than if the score ranged from 0 to 100. A ratio scale gives numbers an absolute meaning by establishing a true zero. For example, the Kelvin temperature scale has a true zero point (0 K), representing the absolute zero at which molecular motion ceases.

2.1 Data Mining Methodology

Data mining methodologies refer to the steps followed to extract knowledge from a data set. It incorporates a series of tasks and techniques designed to transform raw data into meaningful insights.

Multiple methodologies exist, but most follow the same general structure. We will focus only on *CRISP-DM*, as it is used the most in the industry, and researchers consider it one of the best practices to conduct data mining work [26]. The typical CRISP-DM process can be visualized in Figure 2.1.

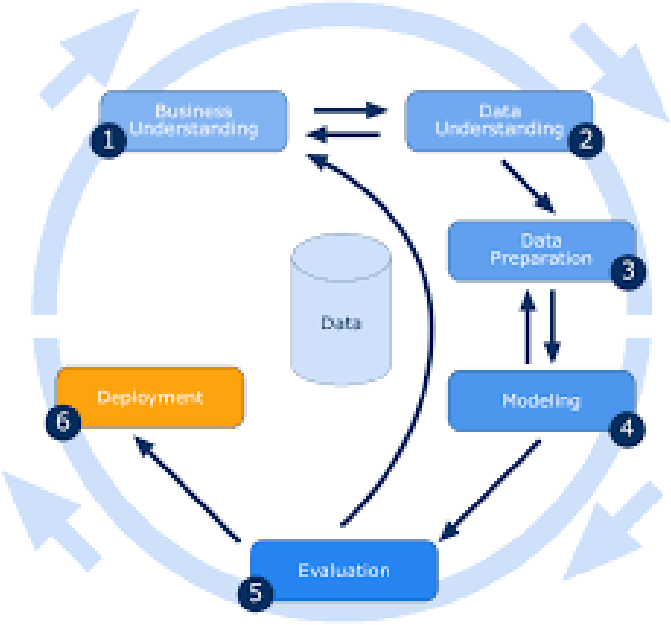


Figure 2.1: CRISP-DM: phases and workflow

CRISP-DM comprises six main phases of development, each equally important to achieve the final result.

The first phase is *Business Understanding*, where the researcher focuses on understanding the project objectives and requirements from a business perspective. It involves talking to the customer

to fully comprehend what are the goals from a business standpoint, what resources are available to make it possible to accomplish, the risks involved in the process as well as the contingency measures to deal with them, and the goal of the investigation from a data mining perspective.

After finishing *Business Understanding*, the work moves on to *Data Understanding*, where the researcher becomes familiar with the initial data collection. This phase aims to examine the overall structure of the dataset, explore the data in that same dataset, and identify data quality problems. The researcher starts by looking at the data and documenting its properties, like the number of records in the dataset, the format, and the entities of the fields. After doing so, researchers visualize the data by generating plots or graphs that give a better perspective on the distribution of values for a given field and explore possible relationships between fields in the dataset. The researcher should also investigate the data quality by examining it and finding out how clean/dirty it is.

Moving to *Data Preparation*, the researcher looks to prepare the final dataset for the *Modeling* stage. The preparation starts with selecting the relevant subsets of data for the study. Not all of the data in a dataset is relevant to the object of the study, so maintaining it would introduce noise, which could obscure the knowledge extraction while also demanding extra computing power. Having selected the relevant data, the researcher must perform data-cleaning tasks to ensure the dataset's quality. This includes dealing with missing or duplicate values, inconsistent data formats, outliers, and denormalized values. Following the cleaning process, it can be helpful for the researcher to enrich the dataset. This enriching step includes creating new fields based on existing ones, integrating dispersed data from other internal systems, or including third-party data from external sources, such as adding social media, demographic, and geographical data.

Upon finishing the dataset preparations, it is time to move on to the *Modeling* phase. Here, the researcher will apply various modeling techniques, calibrating their parameters to optimal values. The first step is to decide which algorithms to use on the data. Next, a dataset is split between training and testing data, fitting such data to the chosen algorithm and generating a model. The researcher will compare this model against other models developed with different algorithms/parameters, and it is the researcher's job to assess which one behaved the best based on domain knowledge, the pre-defined success criteria, and the test design.

When the researcher is confident that it produced a good model, he moves on to the *Evaluation* phase. The model will be evaluated on a business value spectrum by assessing whether it fulfills the desired business success criteria. They also review the data mining to ensure they correctly execute all steps and no one overlooks any details. Then, the researcher decides on whether to deploy the obtained model or re-iterate the process. If everything complies, the model can move on to the *Deployment* phase.

Here, the goal is to deliver the work performed to the customer. It implies the creation of a document with the plan for the model deployment, a document for the monitoring and maintenance of the model to deploy, and a document detailing the project findings containing a final presentation of data mining results. The researchers should also perform a retrospective after finishing the process to reflect on what went well and what could have been better and draft future actions to improve the overall data mining process.

In our case, since we are only interested in deriving conclusions from the decision-making done by the data mining models, we will end our *CRISP-DM* cycle at the *Evaluation* phase, where the results will be commented by the ISPUP experts, and future actions mapped.

2.2 Classification

Classification is a predictive task in which a classifier gives an object a label representing a particular class or category based on the value of the predictive attributes chosen to categorize the object.

A binary classification task aims to categorize objects into one of two distinct classes: the positive class and the negative class. Typically, the positive class is the primary focus of research interest. For instance, in a classification task designed to distinguish between healthy and sick individuals based on a set of attributes, the "sick" class is designated as the positive class, as it represents the primary interest of the investigation.

The process of constructing a classifier involves two main steps. The first step is the learning phase, where the classifier learns from training data consisting of pre-labeled objects and their associated attributes. During this phase, the classifier samples the training data randomly from the dataset under study, with the proportion of data allocated for training determined by the researchers. Since the classifier knows the class's label for each object, this method is considered supervised learning.

This initial learning phase aims to establish a mapping between a set of attributes and a label, which is of interest to researchers to identify the rules that guide the model's classification decisions.d

The second stage involves evaluating the classifier by estimating its predictive accuracy. This evaluation uses a test set comprising the data not used in the training data. The accuracy of the model is determined by the percentage of correctly classified test objects, and if it is deemed satisfactory, researchers can confidently employ the classifier to predict the class of unlabeled objects.

2.2.1 Algorithms

2.2.1.1 Decision Trees

Decision trees are commonly used classification algorithms that create a tree-like structure where nodes represent conditional tests on attributes, branches denote the outcomes of these tests, and leaves contain the labels for classification [18]. Predicting an object's label involves traversing the tree from the root to a leaf node.

Decision trees are advantageous because they do not require domain knowledge and can handle multi-dimensional data. Attribute selection measures are crucial for enhancing decision-making within the tree, focusing on attributes that best partition the data into distinct classes.

These attribute selections aim to choose attributes that result in the purest partitions, where each partition ideally contains objects of the same class. Measures such as Information Gain and the Gini Index are used to evaluate attributes. Information Gain calculates the reduction in entropy (a measure of randomness) after splitting the data, favoring attributes that produce the most significant decrease in entropy. The Gini Index assesses the impurity of partitions, selecting attributes that achieve the most significant reduction in impurity, which is given by the heterogeneity within a partition.

Pruning techniques are applied to prevent overfitting, which occurs when the model is too complex. Prepruning halts the tree's expansion based on criteria like Information Gain or the Gini Index, converting nodes to leaves if further splitting doesn't meet a threshold. Postpruning, more commonly used, involves replacing subtrees with leaves if the cost complexity, determined by a ratio of leaves to error rate, justifies it.

Pruning is essential for simplifying the tree, reducing noise, and improving generalization, ensuring the decision tree remains accurate and efficient.

CART is a specific type of decision tree that constructs only binary trees, where each internal node splits the dataset into two subsets based on a single attribute and threshold [5]. This algorithm follows a greedy approach, selecting the best split at each step without considering the overall tree structure's global optimization.

2.2.1.2 Neural Networks - Backpropagation

Backpropagation is a neural network learning algorithm that adjusts the weights of connections in a multilayer feed-forward neural network to classify input data correctly. The network comprises an input layer, one or more hidden layers, and an output layer, with weights adjusted to minimize classification errors. Key advantages of neural networks include their ability to handle unknown input relationships and continuous-valued inputs and outputs. However, they require significant training time and complex network topology determination and are often difficult to interpret.

The *Backpropagation* process involves feeding input data through the network, calculating errors at the output, and propagating these errors back through the network to adjust weights and biases. Normalization of input values and using activation functions are essential to improve performance. The network topology, including the number of hidden layers and units, significantly affects model performance and is usually optimized through trial and error. Cross-validation and techniques like hill-climbing can help refine the network architecture. The learning rate controls the adjustment of weights to prevent the network from getting stuck in local optima. Case updating modifies weights after each training object, while epoch updating adjusts weights after processing the entire training set, with case updating generally providing better results.

2.2.1.3 Support Vector Machines

Support Vector Machines, or *SVMs*, constitute a method that allows for the linear and non-linear data classification [11]. It applies a non-linear mapping to the training data to transform it into a

higher dimension. In this new dimension, it searches for the linear hyperplane that best separates the objects from one class from the other. With appropriate nonlinear mapping, a hyperplane can always separate two classes. The *SVM* finds this hyperplane using support vectors(the training set objects) and margins. Margin is the sum of the distances between the hyperplane and each "side", with a "side" being a parallel plane representing either class's closest training object. Support vector machines have the advantage of being highly accurate, but they have the drawback of being rather slow to train.

A given dataset can be either linearly or non-linearly separable. For the linearly separable datasets, a straight line/plane that separates all the objects of the negative class from the positive class exists, also called the hyperplane. There is an infinite number of these separation layers, so the optimal one is found by searching for the maximum marginal hyperplane, meaning the one that maximizes the hyperplane's margin. The objects that define the hyperplane's margins are called the support vectors and are equally distant from the hyperplane. These objects are the hardest to classify, so they subsequently provide the most information regarding the classification criteria.

For non-linearly separable datasets, the opposite occurs, where no straight plane exists so that it separates the classes. When this is the case, the kernel trick can achieve separation. Here, we map the input features into a higher-dimensional space where linear separation may be achievable. Some of the most common kernels include the polynomial kernel, the Gaussian radiation function kernel, and the sigmoid kernel.

2.2.2 Ensemble Methods

After measuring how accurate the classifier is, it is of interest to the researchers to increase its predictive performance until it reaches a desirable level. Ensemble methods are an efficient way to achieve this, and techniques include *Bagging*, *Boosting*, and *Random Forests*. The idea behind an ensemble method is to combine the predictive ability of various classifiers generated, each with its partition of an initial dataset. All classifiers predict the class for a given object. Then, the final result is obtained by, for example, majority voting, where the selected class corresponds to the one predicted by the highest number of classifiers. The advantages of using this technique reside in the fact that to predict the class of an object erroneously, more than half of the classifiers mispredict it(for a binary class case). Results are best if the correlation between the classifiers is lower.

2.2.2.1 Boosting

Boosting ensembles iteratively decide the predicted class through a majority of votes, weighted by the accuracy of the model that casts it [6]. The first classifier is trained on the initial dataset and attributes weights to its objects depending on whether they were correctly classified. An object correctly classified will have a lower weight, while a correctly classified one will have a higher weight. Then, the following classifier will take into consideration these weights when training. In the end, each model will cast a vote that is also weighted with the accuracy that it obtained. The downside of this approach is that it can lead to overfitting.

2.2.2.2 Random Forests

Random Forests constitute a collection of separate decision trees, with the final class prediction being made through majority voting [4]. Each tree is generated by selecting a random attribute to determine the split at each node. *Random Forests* can be constructed using Bagging, where, for each tree, a set of training objects is selected from the initial dataset using Bootstrap. Then, for each node in the tree, a split is made based on one or more of the object's attributes, chosen at random. The trees will be expanded until only leaf nodes are in the bottom, without pruning. *Random Forests* offer a way to reduce the possibility of overfitting while also providing a fast and accurate way to classify objects on large datasets. Its accuracy depends on the individual predictive strength of its tree constituents and is higher when the correlation between trees is lower.

2.2.2.3 XGBoost

XGBoost is a machine learning algorithm that utilizes gradient boosting to improve model performance[9]. At its core, *XGBoost* sequentially builds a series of decision trees, where each subsequent tree corrects the errors made by the previous ones. Initially, it starts with a single decision tree, which predicts the target variable. It then calculates the residuals or errors between the predicted values and the actual target values so the next tree can be trained considering the learning mistakes from the prior tree, effectively refining the overall prediction. This process continues iteratively, with each new tree improving the model's accuracy by focusing on the residuals left by its predecessors. *XGBoost* also employs several key optimizations to enhance performance. An example is regularization techniques to control overfitting, such as pruning trees. It supports parallel processing, making it efficient for large datasets while also incorporating an objective function that balances model complexity and prediction accuracy.

2.2.2.4 AdaBoost

AdaBoost is an ensemble method that combines multiple weak learners to create a robust predictive model, in which each subsequent learner focuses more on misclassified instances from its predecessor [23]. *AdaBoost* starts by assigning equal weights to each training sample in the dataset. In each iteration, it trains a weak learner on this weighted dataset and calculates its error rate, attributing higher weights to misclassified instances and lower weights to correctly classified instances. After training all the weak learners, the *AdaBoost* model proceeds to a weighted majority voting scheme, where each weak learner's contribution is weighted based on its accuracy. This approach highlights the strengths of multiple weak learners, compensating for their individual limitations and enhancing the model's ability to generalize to new, unseen data. *AdaBoost* is particularly valuable in scenarios where interpretability is important and when datasets that are not excessively large or noisy, as it can be sensitive to outliers and noisy data points.

2.2.3 Sampling Train/Test data

An issue that can affect the classifier's performance is the presence of an imbalanced class distribution on the dataset. This happens in datasets where the negative class is much more prominent than the positive class. The issue with this situation is that classification usually aims to reduce the number of erroneous predictions without considering whether it was a False Positive or a False Negative. But when the positive class is rare, predicting a false negative wrongly is much more impactful than predicting a false positive. For example, considering the health study performed for this paper, it is worse to indicate that a child will not develop multimorbidity when it will happen than to predict that it will happen but end up not occurring. To address this balancing challenge, multiple sampling techniques allow a better approximation of characteristics relevant to the research question [25].

Oversampling/Undersampling measures produce a more balanced dataset. Imagining a dataset with ten positive class objects and one hundred negative class objects, we can replicate the positive class objects until we have one hundred objects of each class(*oversampling*) or randomly remove ninety objects from the negative class(*undersampling*). Both methods aim to produce a set of objects where each class gets equal representation. This can be achieved with *SMOTE*, a technique that generates synthetic samples from an imbalanced dataset until classes are balanced [7].

Bootstrap is a technique that samples the training set uniformly, allowing for replacement, meaning each object in the dataset is equally likely to get chosen to be on the training set every time an object is selected. The classifier can choose the same object more than once. All objects not added to the training set are used for the testing set. This method works best on small data sets but tends to be overly optimistic.

The *Holdout* method separates the dataset into training and testing data, usually allocating two-thirds of the objects to the training task and the remaining to the testing task. The training data is used to generate the model, and the test data is used to evaluate its performance. This estimate is usually pessimistic as only a portion of the initial dataset is used to train the model. To improve this, we can perform *Random subsampling*, which mirrors the *Holdout* method behavior but repeats it k times, with the final model accuracy given by the average accuracy obtained in each iteration.

Cross-validation follows a similar concept, splitting the initial data into a k number of partitions and then using each partition to test the classifier and all the others for the training step [3]. The average of all the runs gives the final accuracy. This way, we ensure that each dataset partition is used the same number of times for training and once for testing. *Cross-validation* can also include stratification to reduce bias and variance, in which each partition will have approximately the same proportion of label distribution as the initial dataset.

2.2.4 Model Evaluation

Evaluating the classifier is a crucial step in the classification data mining process, as it allows us to assess its ability to predict object classes correctly. For an evaluation process to be as truthful as

possible, it is essential to test the classifier on data it has not used for training to avoid cases where overfitting can induce overoptimistic estimates of the model's performance.

The evaluation metrics rely on the count of True Positives, False Positives, True Negatives, and False Negatives. True Positives correspond to objects predicted to belong to the positive class, which were correctly labeled. False Positives are the objects predicted to belong to the positive class but were wrongly labeled. The same logic applies to the True Negatives and False negatives.

Using these values, we can compute various evaluation measures [15]. *Accuracy*, or recognition rate, demonstrates the percentage of objects the classifier predicted correctly:

$$accuracy = \frac{TruePositives + TrueNegatives}{Positives + Negatives}$$

Contrary to this, we have the *error rate*, or misclassification rate, which refers to incorrectly classified objects. It can be calculated by subtracting the classifier's accuracy from 1 or by the following formula:

$$errorrate = \frac{FalsePositives + FalseNegatives}{Positives + Negatives}$$

These two previous measures work well in datasets with balanced label distribution. Still, in cases where the positive class represents, for example, only 5% of the total dataset labels, further measures are required to assure the accuracy of the classifier.

We can compute the *sensitivity* and *specificity* measures to accommodate these cases. *Sensitivity* measures the True Positive recognition rate exclusively, and *specificity* measures the True Negative recognition rate exclusively.

$$sensitivity = \frac{TruePositives}{TruePositives + FalseNegatives}$$

$$specificity = \frac{TrueNegatives}{TrueNegatives + FalsePositives}$$

Another set of widely used evaluation measures are *precision* and *recall*. *Precision* measures the ratio of positive class objects that were identified as such. It is calculated following the formula:

$$precision = \frac{TruePositives}{TruePositives + FalsePositives}$$

Recall, on the other hand, aims to measure how many of the positive objects present in the dataset were identified and follows the same formula as the one used for the sensitivity. *Precision* and *recall* can be combined by calculating the F-score. The F-score represents a harmonic mean between *precision* and *recall* by describing them as equally weighted. It is calculated through the formula:

$$F - score = \frac{2 * precision * recall}{precision + recall}$$

It is also possible to use *F-score* weighted more to one of the two measures by following the formula:

$$F - score(\theta) = \frac{(1+\theta)^2 * 2 * precision * recall}{\theta^2 * precision * recall}$$

This way, by computing the F-score(2), we would weight recall twice as much as precision.

Receiver operating characteristic curves or *ROC* curves are a way to plot the trade-off between the True positive rate (given by sensitivity) and the False positive rate (provided by 1 - specificity), with the area under the curve showing the accuracy of the classifier. The usefulness of this metric relies upon the ability to visualize, for two-class datasets, the trade-off of accurately identifying positive cases versus the incorrect identification of negative cases as positive. Increasing the True positive rate always comes at the expense of increasing the False positive rate. The *area under the ROC curve*, commonly known as *AUC*, is a metric used to evaluate the predictive performance of the classifier. The *AUC* value ranges between 0 and 1. An *AUC* value of 0.5 means the classifier performs as well as a random prediction of an object's class. Below 0.5, the model would be predicting worse than a random predictor. Above 0.5, which is the desirable outcome, the model displays a discriminatory power that performs better than a random choice, and the higher the value, the better the predictor.

Chapter 3

Business Understanding

The investigative phase commenced with a meeting involving ISPUP experts, where the study's primary objectives were presented. These objectives aimed to address two key questions:

- Which environmental exposures significantly influence the appearance of multimorbidity?
- What time-frames of exposure to these environmental factors correlate most strongly with the appearance of multimorbidity?

Multimorbidity was the focal point of the investigation, defined as the presence of two or more of the seven specified adverse health outcomes. The study approached this problem from two perspectives:

- **multiSN** Label: This involved binary classification of multimorbidity, where individuals were categorized as positive if they exhibited 2 or more adverse health outcomes and negative otherwise.
- **multi3cat** Label: This entailed a three-class classification of multimorbidity:
  - **Class 0**: Individuals with 0 adverse health outcomes.
  - **Class 1**: Individuals with exactly 1 adverse health outcome.
  - **Class 2**: Individuals with 2 or more adverse health outcomes.

The objective was to develop classifiers for these two labels using data mining models trained on the environmental exposure data. These exposures encompassed various categories, including *Social Environment*, *Meteorology*, *Air Pollution*, *Built Environment*, *Natural Space*, and *Traffic/Noise*.

3.1 Data Understanding

The ISPUP experts provided two datasets for investigation: one containing environmental measurements and another containing information on adverse health outcomes for individuals measured at the end of the study.

Both datasets were linked using the unique identifiers "*familiaID*" and "*membroID*," ensuring a one-to-one correspondence across all rows between the CSV files. The environmental dataset included records for 8,647 individuals, containing 55 distinct attributes. Most attributes were measured 13 times throughout the study period, while a few were recorded 12 times, resulting in 711 measurements per individual. Annex A.1 provides a comprehensive list of these attributes, their definitions, and the number of measurements recorded.

The outcomes dataset comprised data from 3,788 individuals across ten distinct attributes. Due to missing values in some entries, individuals lacking complete health outcome data were excluded from the study, leaving 2,849 individuals for analysis. The dataset includes seven attributes of health outcomes recorded after the study and three variables used to quantify multimorbidity in individuals. Annex A.2 provides a comprehensive list of these outcomes, their definitions, and their coding.

Data Understanding commenced with exploratory data analysis, including plotting to visualize the distribution of health outcomes in the dataset. The bar chart in Figure 3.1 illustrates the distribution of individuals based on the number of adverse health conditions they contracted.

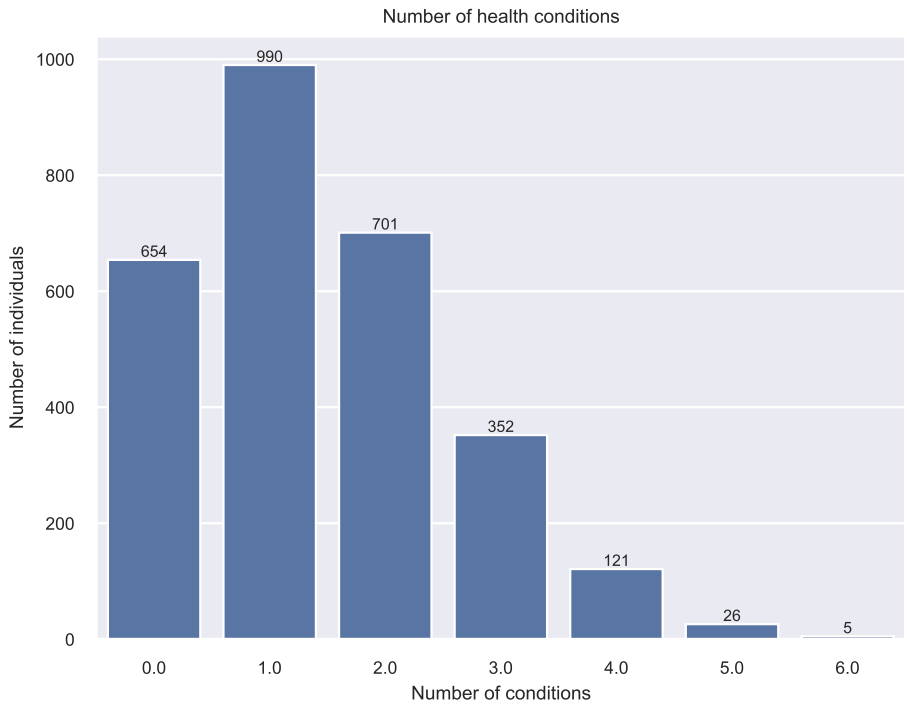


Figure 3.1: Number of individuals with specific adverse health outcomes.

The x-axis represents the number of conditions (ranging from 0 to 6), while the y-axis shows the frequency of individuals with each respective number of adverse health outcomes. Notably, no individual contracted all seven health conditions. The highest frequency occurs among individuals with one health condition (990 individuals), followed by those with two conditions (701 individuals) and those with none (654 individuals). The average number of adverse health outcomes per individual is 1.45.

The multimorbidity analysis is approached from two dimensions. The first dimension categorizes individuals into two classes: the negative class includes individuals with zero or one adverse health outcome, and the positive class comprises individuals with two or more adverse health outcomes.

Examining the binary distribution within this dimension, present in Figure 3.2, reveals that the dataset is relatively balanced, with the positive class representing approximately 42% of the individuals. This balanced distribution is advantageous for subsequent data analysis tasks. It ensures that the machine learning model encounters each class equally during training, reducing the risk of bias. Additionally, a balanced dataset enhances the model's capacity to generalize and perform effectively on unseen data.

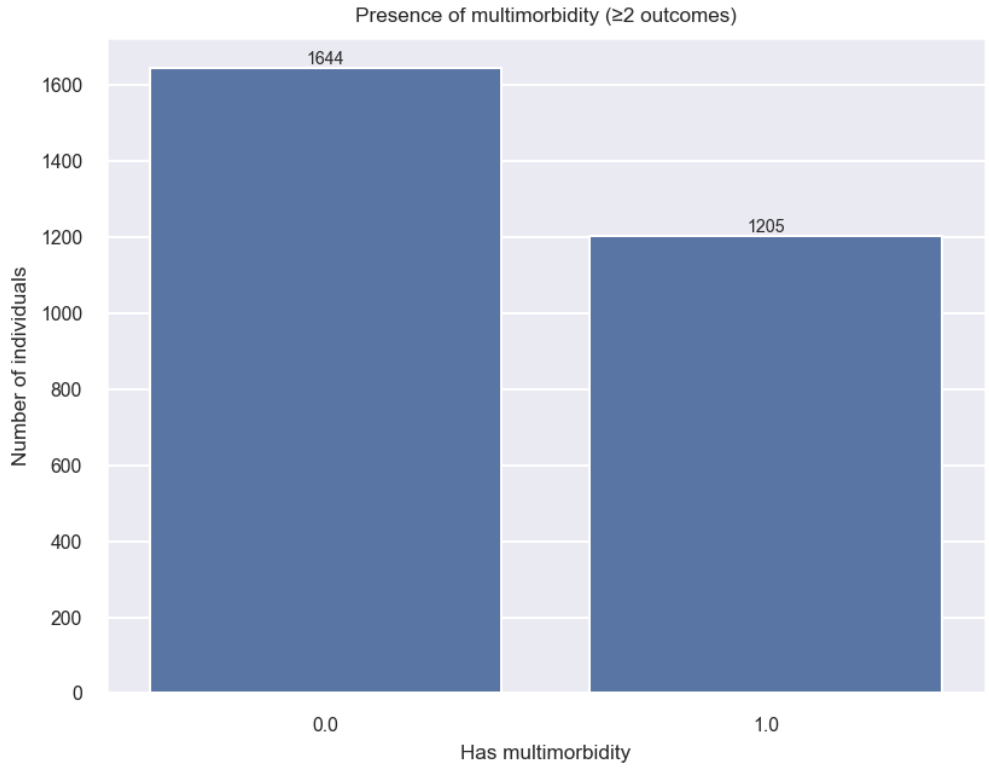


Figure 3.2: Number of individuals in each of the binary multimorbidity classes.

In the second dimension of analysis, individuals are categorized into three distinct classes based on their adverse health outcomes. The classes are defined as follows:

- **Class 0**: Individuals with no adverse health outcomes.
- **Class 1**: Individuals with exactly one adverse health outcome.
- **Class 2**: Individuals with two or more adverse health outcomes.

This classification scheme allows a more detailed understanding of multimorbidity across different severity levels. It enables the investigation of factors influencing the progression from fewer to multiple adverse health outcomes, providing insights into the varying impacts of environmental exposures on health outcomes. This approach supports the study's goal of identifying significant contributors to multimorbidity within the dataset. The number of individuals in each of these classes can be seen in Figure 3.3.

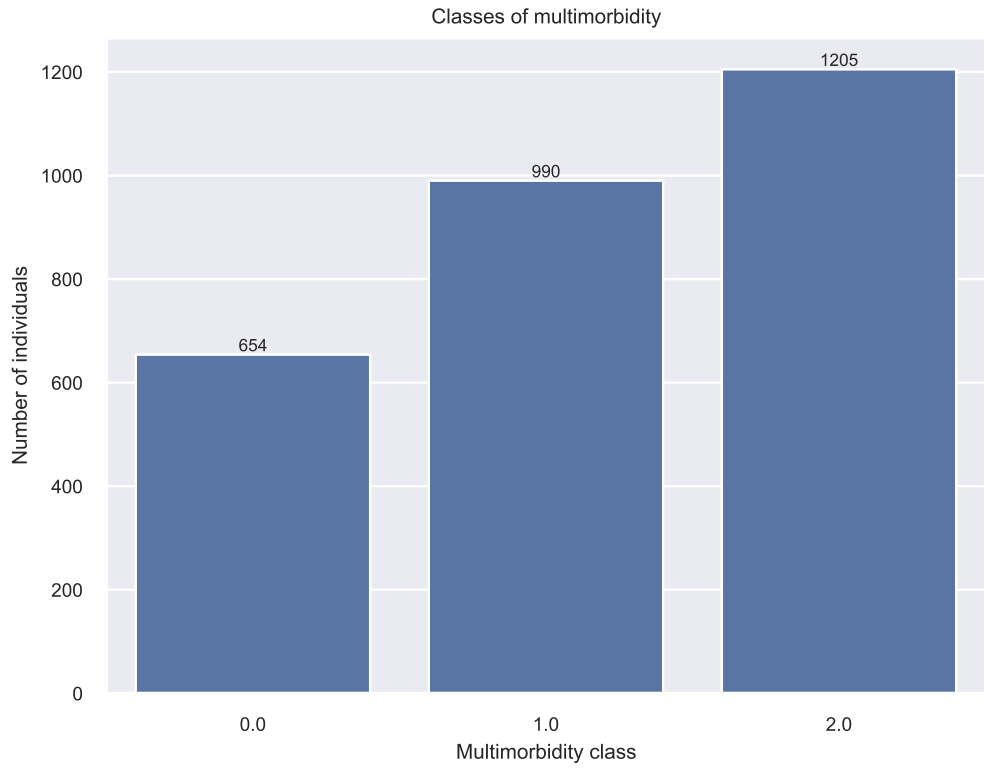


Figure 3.3: Number of individuals in each of the three multimorbidity classes.

After visualizing the distribution of multimorbidity cases, we proceeded to plot the relative frequency of each health outcome among individuals with multimorbidity in Figure 3.4.

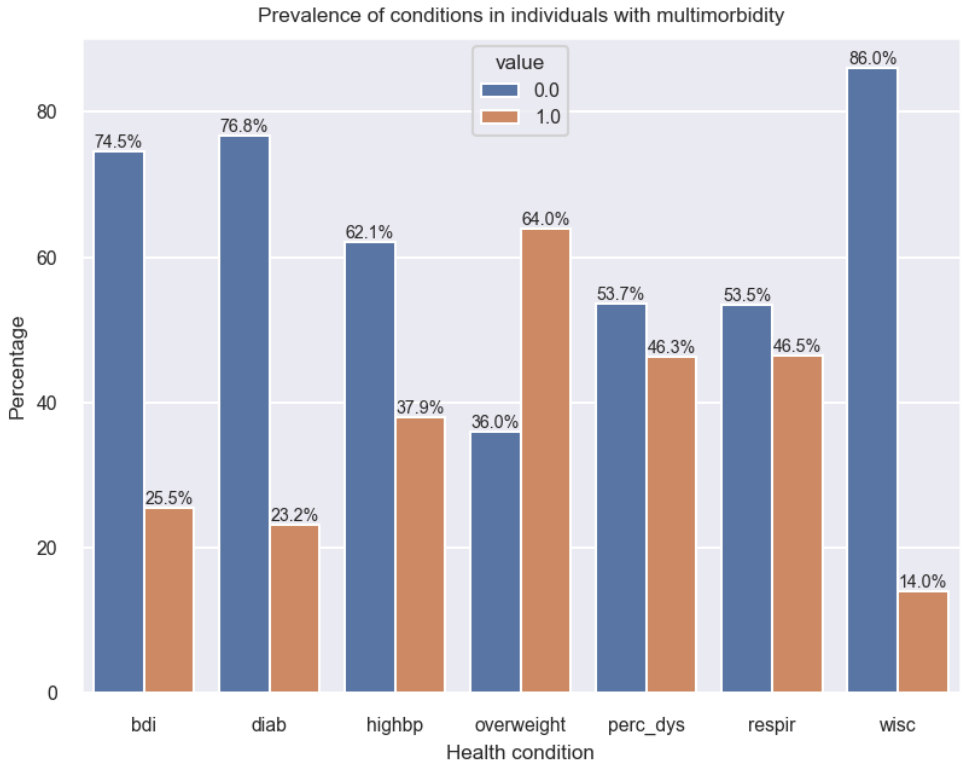


Figure 3.4: Specific outcome prevalence in individuals with multimorbidity.

After obtaining a general overview of the multimorbidity dataset, our analysis focused on exploring the environmental dataset and understanding its contents.

The environmental dataset predominantly comprises continuous variables, with only four variables categorized as categorical. Upon initial inspection, it became apparent that many variables contained a significant number of zeros as values. To quantify this, we conducted an analysis using a *Python* script to calculate the relative frequency of zeros within each column. The results highlighted variables where more than 50% of the entries were zeros. Subsequently, we discussed these findings with ISPUP experts to determine appropriate actions, resulting in the decision to exclude these variables from the final dataset. All attributes listed in Table 3.1 exhibit continuous characteristics.

Table 3.1: Frequency of 0s in environmental variables(only for variables with more than 50%).

Variable name	Percentage of 0s
airpt	89.9%
port	87.9%
blue_yn300	84.7%
water	84.6%
vldres	77.1%
stops_100	60.8%
lines_100	51.5%

Following our initial dataset analysis, several key observations were made regarding missing values, data distribution, and attribute correlations. Here's a detailed breakdown of our findings and future actions to take:

- **Missing Values Analysis**: We identified missing values in certain columns, with the highest percentage in any column being 10.3%. This discovery prompted discussions with ISPUP experts, leading to the decision to proceed only with individuals with complete data across all attributes in the dataset. Additionally, alternative datasets were generated using different techniques to impute missing values, focusing on maintaining a dataset with no missing values for primary analysis. More details on these techniques are provided in the Data Preparation section.

- **Data Distribution Analysis**: We computed each variable's skewness to understand their distribution better. Variables with skewness values between -0.5 and 0.5 were considered symmetric, while those outside this range were deemed asymmetric. For symmetric continuous attributes, we calculated the mean and standard deviation. The results are recorded in Annex A.4. We computed the median, 25th, and 75th percentiles for asymmetric continuous attributes. The results are recorded in Annex A.5. Categorical attributes were analyzed by computing the relative frequency of each possible value. The results are recorded in Annex A.3.

- **Correlation Analysis**: We examined correlations between attributes using correlation plots. Initially, we plotted correlations among all attributes for year 12. Additionally, we explored correlations over different years to determine the stability of exposures across time. Correlation coefficients range from -1 to 1, with values closer to 1 indicating stronger positive correlations and values closer to -1 indicating stronger negative correlations.

Below, Table 3.2 shows the top ten most correlated attribute pairs, and Table 3.3 shows the top 10 least correlated attribute pairs:

Table 3.2: Most correlated environmental variables

Variable 1	**Variable 2**	**Correlation**
uvddc_12	uvdvc_12	1.0
uvdec_12	uvdvc_12	0.999
uvddc_12	uvdec_12	0.998
qq_12	tx_12	0.991
pp_12	qq_12	0.99
pp_12	tx_12	0.987
hu_12	rr_12	0.986
qq_12	uvddc_12	0.976
pp_12	uvddc_12	0.975
qq_12	uvdvc_12	0.971

We observed significant correlations among attributes, particularly within the Meteorology category, where 18 out of the top 20 most correlated attribute pairs were found. This suggests substantial redundancy or colinearity among these variables, indicating potential opportunities to reduce the dataset's dimensionality.

High correlation among attributes implies that some variables may convey redundant information to data mining algorithms without contributing unique insights. Removing redundant variables can simplify computational processes, reduce model complexity, and enhance computational efficiency. This approach is crucial for optimizing the dataset for subsequent data mining tasks, ensuring that models can focus on relevant and informative features.

Therefore, identifying and removing these highly correlated attributes can lead to a more efficient and effective data analysis pipeline, facilitating clearer insights into the relationships between environmental exposures and multimorbidity outcomes.

Table 3.3: Least correlated environmental variables

Variable 1	**Variable 2**	**Correlation**
rr_12	qq_12	-0.991
hu_12	uvddc_12	-0.987
hu_12	uvdvc_12	-0.985
hu_12	pp_12	-0.982
hu_12	uvdec_12	-0.981
rr_12	tx_12	-0.98
hu_12	qq_12	-0.98
rr_12	uvddc_12	-0.978
hu_12	tx_12	-0.975
rr_12	uvdvc_12	-0.975

We observed numerous meteorology-related attributes exhibiting strong correlations, particularly between humidity (*hu*) and precipitation (*rr*). This correlation is anticipated, as humidity levels often correspond to precipitation amounts. Conversely, there is a lack of correlation between these two attributes and the other meteorological attributes, such as those measuring sun-related conditions (e.g., UV light exposure or temperature).

This correlation information is pivotal for reducing the final dataset's dimensionality, thus streamlining the data mining process. However, it is essential to validate that conclusions drawn from the year 12 data are consistent across all other years. To achieve this, we calculated the correlation of each variable's values across the years. If most variables exhibit an average correlation close to 1 throughout the years, we can confidently extend these conclusions to the entire study period.

Therefore, we computed the average correlation for each variable and identified those with an average correlation below 0.5 across the years. These results, presented in Table 3.4, indicate which variables might be inconsistent over time and require careful consideration in the data analysis.

Table 3.4: Average same variable correlations below 0.5

Variable	**Correlation**
tx	0.24
tg	0.23
tn	0.23
uvdec	0.16
uvdvc	0.14
qq	0.14
uvddc	0.12
rr	0.11
hu	0.07
pp	0.05

It was observed that a small number of variables exhibited low correlation over the years. The table above shows that 18% of all variables fall into this category, all belonging to the *Meteorology* category. Meteorological data is expected to vary over time, hence a lower correlation between values across years. However, our earlier conclusions remain valid as we have established that the high correlation between meteorological variables is due to inherent causal relationships rather than undiscovered data patterns. For instance, humidity is inherently influenced by high precipitation levels, a relationship that remains constant regardless of annual variations in precipitation, as humidity will correspondingly adjust.

A meeting with ISPUP experts was convened to discuss these findings, leading to several decisions regarding variable inclusions. Certain variables were removed due to a lack of justifiable impact on the emergence of adverse health conditions. An example is the variable *pp*, which measured average sea level pressure. Two approaches were considered for the highly correlated meteorological variables. One approach was to retain all variables and allow the feature selection

process to determine their inclusion. The other approach was to select a representative variable for any group of variables with correlations exceeding 0.95. We chose *uvddc* as the representative variable in this case.

Regarding missing values, it was decided to proceed primarily with individuals with no missing values in the multimorbidity dataset, preferably the same for the environmental dataset. This approach will be our main focus in concluding the study. Nonetheless, different techniques for handling missing values will be applied and discussed in the next section to explore their potential impact on the model's performance.

Chapter 4

Data preparation

This chapter details the methodologies employed to preprocess the dataset, ensuring it is ready for integration into data mining algorithms. Key tasks in these processes include managing missing values, detecting outliers, implementing dataset-specific modifications following consultations with ISPUP experts, and merging the environmental and multimorbidity datasets. Each of these steps is detailed below, accompanied by descriptions of the chosen strategies:

- **Handling Missing Values**: Techniques such as removal of entries with missing values, imputation using statistical measures (e.g., mean, median, and mode), and more advanced methods like *KNN* imputation were applied. These strategies aimed to minimize data loss while preserving the dataset's information [12].
- **Outlier Detection**: Statistical methods were utilized to identify outliers that could impact results. Outliers were corrected for further stages to ensure they did not influence model training.
- **Dataset-Specific Modifications**: Adjustments following insights from ISPUP experts were implemented to align the dataset with the domain-specific requirements and research objectives.
- **Integration of Datasets**: Environmental and multimorbidity datasets were merged into a single dataset.

These strategies were chosen to optimize data quality, enhance the reliability of subsequent analyses, and ensure the dataset's suitability for modeling in line with ISPUP's research goals.

4.1 Multimorbidity Dataset

The multimorbidity dataset required almost no preparation effort, as it only contained data related to the study's seven adverse health outcomes and three labels derived from them. To ensure the integrity of the dataset, a coherence check was made between the three labels and the remaining adverse health outcomes to confirm they were consistent across measurements.

Regarding missing values, the original dataset contained 3787 entries, of which 938 had missing values for at least one of the seven outcomes. In consultation with ISPUP experts, it was decided to remove these entries from further analysis, proceeding only with individuals with complete data for all health outcomes. Table 4.1 details the percentage of missing values for each of the seven adverse health outcome variables. Some percentages may overlap due to entries missing more than one outcome.

Table 4.1: Percentage of missing values for each outcome in the multimorbidity dataset

Variable	Percentage of missing values
wisc	13.9%
diab	7.2%
perc_dys	5.6%
overweight	2.3%
respir	2%
bdi	1.6%
highbp	0.5%

Removing the entries with missing values leaves the dataset with 2849 records.

4.1.1 Constraints

After the previous steps, at the request of ISPUP experts, all entries sharing the same *familiaID* were excluded from the dataset. These entries represented twins, who are medically more likely to contract adverse health outcomes. This exclusion aimed to reduce variance in the correlation between environmental exposures and health outcomes.

As a result of this removal, the dataset was reduced to 2772 records, finalizing the structure of the multimorbidity dataset for subsequent modeling tasks. This approach ensured that the dataset maintained its integrity and was aligned with the specific research objectives requested by ISPUP experts.

4.2 Environmental Dataset

Following discussions with ISPUP experts after data visualization, several measures were agreed upon to prepare the datasets for subsequent stages:

- **Removal of Columns with High Amount of Zeros and Unwanted Exposures**: Columns populated by a high amount of zeros and environmental exposures that are hard to scientifically correlate with outcomes were removed. Specifically, columns such as *airpt*, *hdres*, *ldres*, *vldres*, *indtr*, *trans*, *port*, *other*, *urbgr*, *agrgr*, *natgr*, *water*, and *pp*.
- **Exclusion of Year 13 Columns**: Columns ending with *_13* were removed from the dataset as the health outcomes were measured that year.
- **Removal of Duplicate familiaID Entries**: Entries sharing the same *familiaID* were removed from the dataset to eliminate data redundancies, particularly focusing on twins who may introduce unwanted bias.
- **Selection of SES Features**: To simplify the dataset, only variables related to area-level socioeconomic status (SES) in quintiles were kept. This decision to choose the quintile features over tertiles was because of their finer granularity in representing the information.

These adjustments were implemented to refine the dataset, ensuring it aligns with ISPUP experts' research objectives and facilitates analysis of environmental exposures' potential impacts on adverse health outcomes.

4.2.1 Outlier detection

The outlier detection phase followed the initial preparation step. To do so, a script was run which counted, for each attribute in the dataset, how many attributes were considered outliers. This was done following the IQR range method explained below:

4.2.1.1 Detection of Outliers Using the IQR Method

First, the dataset is sorted in ascending order. The first quartile ($Q1$) and the third quartile ($Q3$) are then determined. These quartiles represent the 25th and 75th percentiles of the data, respectively [13].

$$Q1 = \text{25th percentile of the data}$$
$$Q3 = \text{75th percentile of the data}$$

The IQR is calculated as the difference between the third quartile and the first quartile:

$$\text{IQR} = Q3 - Q1$$

Outliers are defined as observations that are below the lower bound or above the upper bound. These bounds are determined using the following formulas:

$$\text{Lower Bound} = Q1 - 1.5 \times \text{IQR}$$
$$\text{Upper Bound} = Q3 + 1.5 \times \text{IQR}$$

Any data points that fall below the lower bound or above the upper bound are considered outliers.

$$\text{Outliers} = \{x \mid x < \text{Lower Bound or } x > \text{Upper Bound}\}$$

Out of the 540 columns examined, 112 exhibited more than 5% of values identified as outliers, with the top column containing 1756 outliers. The number of outliers in the dataset was considerable, making complete removal of these entries impossible as it would greatly reduce the amount of data available for modeling. Therefore, it was decided to update these outliers by adjusting their values to the nearest minimum or maximum value considered normal based on the lower and upper bounds established for each column.

This approach ensured that the dataset kept sufficient information while mitigating the impact of outliers.

4.2.2 Handling missing values

From the beginning, ISPUP investigators recommended discarding all entries with missing values in the dataset and proceeding only with patients with complete information. This approach is common in medical practice, where imputing missing values in either exposures or outcomes is uncommon. By removing entries with missing values, 1258 individuals were eliminated from the dataset.

Despite this approach, we generated additional datasets using two different strategies to handle missing values, aiming to achieve higher performance than the dataset without missing values.

- **Mean/Median/Mode Imputation**: This strategy involved filling missing continuous values with the mean for attributes following a symmetric distribution and with the median for attributes following an asymmetric distribution. Categorical attributes were filled with the mode.

- **KNN Imputation**: This method imputes missing values based on similarity with neighboring data points [1]. The *KNN* algorithm calculates the Euclidean distance between an entry with missing values and all other complete records and identifies the K-nearest neighbors

that are closest to the entry. It then imputes the missing value using the mean of these neighbors' values. We used 54 neighbors to evaluate each missing value, corresponding to the square root of the number of patients.

These strategies were employed to experiment with alternative approaches to handling missing data and to evaluate their impact on model performance.

The final three datasets resulting from the handling missing values were

- Dataset 1: No missing values. (preferred approach of the ISPUP experts).
- Dataset 2: Mean/median/mode imputation of values.
- Dataset 3: *KNN* imputation of values from the 54 nearest neighbors.

4.2.3 Data normalization

When analyzing the dataset attributes, significant differences in orders of magnitude were observed among the data. Some attributes were percentages between 0 and 1, while others represented distances or densities. These differences in scale impact how modeling algorithms interpret and learn from the dataset.

Dataset normalization standardizes attributes to ensure they contribute equally to analysis, particularly in gradient-based methods like *Neural Networks* and distance-based methods like *Support Vector Machines* [24]. Normalization typically scales features to a common range, commonly between 0 and 1, or adjusts them to have a mean of 0 and a standard deviation of 1. This practice prevents attributes with larger ranges from dominating the learning process of the model.

Z-score normalization centers the data around zero by subtracting the mean and scales it based on the standard deviation. This approach helps mitigate the influence of outliers, as it rescales the data relative to its distribution.

Additionally, the class imbalance was addressed by balancing the dataset's classes used as labels. Imbalanced datasets, where one class is more prevalent than others, can bias the model towards the majority class and result in poorer performance for minority classes, which are often the class of interest.

To mitigate this issue, *SMOTE* was applied. *SMOTE* generates synthetic samples for minority classes based on existing instances, oversampling them to achieve a more balanced class distribution. This approach ensures that modeling algorithms learn from a more balanced dataset, improving overall performance.

These normalization and balancing techniques were implemented to improve the reliability and predicting power of models in capturing meaningful patterns and relationships within the dataset.

4.2.4 Data transformation

The data transformation process started by merging the environmental and multimorbidity datasets using a combined key consisting of *familiaID* and *membroID*.

Additionally, the label class was chosen for the data mining tasks. The *multiSN* dataset represents a binary class indicating whether an individual has multimorbidity or not. The *multi3cat* dataset indicated a three-class label indicating individuals with multimorbidity, one adverse health outcome, or no adverse health outcomes. Columns related to specific adverse health outcomes were removed from the analysis.

These transformations were applied to each of the three datasets generated, each with its different strategies for handling missing values. In the Modeling section, a comparison of performance between these strategies will be done, continuing only with the strategy that gets the best results.

Chapter 5

Modeling Preparation

In this stage of the *CRISP-DM* workflow, we will explore multiple modeling approaches, and their performance results will be compared to determine the most effective algorithm for a given dataset. Emphasis will be placed on interpretable models, as they facilitate the extraction of feature contributions to the classifier's decisions. The objective is to identify the environmental exposures that most influence the decision-making process and to determine the specific years in which these exposures were most impactful.

5.1 Missing value strategies analysis

As detailed in the previous chapter, three strategies were used to handle missing values: complete removal, imputation with mode/mean/median, and imputation with the *KNN* algorithm. Two distinct datasets were generated, one for each label, to identify the most effective strategy for handling missing values. A *RandomForestClassifier* from *scikit-learn* was trained on all these datasets using the same hyperparameters. The performance of each strategy was evaluated and compared based on 10-fold cross-validation accuracy.

The *RandomForestClassifier* was applied with the following hyperparameters:

- **Number of estimators**: 500
- All other hyperparameters were left as **default**.

The results obtained can be seen in Table 5.1:

Table 5.1: Model accuracy with different missing value strategies.

Label	Remove Missing Values	KNN imputation	Mean/mode/median
multi3cat	53.69%	46.99%	46.56%
multiSN	57.34%	50.38%	50.03%
Average	**55.52%**	48.69%	48.3%

The results indicate that removing entries with missing values produces better performance than imputing them with either the mode/mean/median or the *KNN* algorithm. This finding is consistent with the desired approach of ISPUP experts for handling missing data. Moving forward, only the datasets without missing values will be utilized in analyses, while those generated with imputation strategies have been discarded.

5.2 Dataset shape

5.2.1 Consider less years

Upon analyzing the dataset, it was observed that the data for a lot of features had a high amount of repetition throughout the years. Due to this correlation, we tried to reduce redundancy by removing certain years, which was considered an interesting approach by the ISPUP experts. The strategy was to keep only the years 0, 4, 7, and 10, as these coincided with the intervals when health measurements were collected from the individuals. A dataset containing only these years, and no missing values, was generated. The analysis was re-conducted using the *RandomForestClassifier*.

The results can be seen in Table 5.2:

Table 5.2: Model accuracy for different year selection strategies.

Label	Benchmark	Specific Years
multi3cat	53.69%	46.85%
multiSN	57.34%	45.75%
Average	**55.52%**	46.3%

It is possible to conclude that selecting only specific years does not improve the overall model accuracy. So, moving forward, all years are considered for the model training.

5.2.2 Consider less features

Many features in the dataset exhibited high correlations, approaching values close to 1, indicating redundancy in the information they provide to data mining algorithms. So we attempted to simplify

the dataset by removing highly correlated features. As mentioned in the data visualization chapter, features such as *uvddc* and *hu* were retained, while correlated features like *uvdec*, *uvdvc*, *qq*, *pp*, *tx*, *rr*, and *tg* (with correlations above 0.95) were removed.

Additionally, another set of features measured distances, such as *stops_100*, *stops_300*, and *stops_500*. Given that measurements at 300m were common in ISPUP studies, features associated with other distances (100m and 500m) were excluded from the dataset. This led to the removal of columns *stops_100*, *stops_500*, *lines_100*, *lines_500*, *bdens_100*, *ndvi100_mean*, *ndvi500_mean*, and *connind100*.

A dataset without these features was generated. The same classifier used in the previous analysis was then applied to this dataset, producing the results in Table 5.3:

Table 5.3: Model accuracy for different manual feature selection strategies.

Label	Benchmark	Specific features
multi3cat	53.69%	45.77%
multiSN	57.34%	50.44%
Average	**55.52%**	48.11%

The dataset without this proposed feature selection had better performance, so we proceeded with the complete dataset. The next step involved automatic feature selection.

5.3 Automatic Feature Selection

Since manual feature selection did not enhance the model's predictive performance for the selected labels, a set of automatic feature selection methods will be explored and compared. The objective is to identify a technique capable of reducing the dataset's dimension while maintaining or enhancing the model's accuracy.

Here are the chosen feature selection techniques, along with their descriptions:

- **Extremely Randomized Trees**
  - *ExtraTrees* feature selection identifies important features by averaging the feature importance scores from multiple extremely randomized decision trees built using random data splits. It builds the ensemble using the whole original dataset and, at each node, performs a random split from a random subset of features.
  - Implemented with *sklearn*'s *ExtraTreesClassifier* method.
- **Logistic Regressor based on L1**
  - *Logistic Regressor* based on L1 regularization, also known as *Lasso Logistic Regression*, includes a penalty term to shrink some coefficients to zero [20].

  - Implemented with *sklearn*'s *Logistic Regressor* method, with penalty set to *L1* and solver to *saga*.

- **Random Forest**

  - *Random Forests* identify important features by calculating the average decrease in impurity/increase in prediction accuracy when each feature is used in the ensemble of decision trees, allowing it to rank features based on their contribution to the model's performance. It builds the ensemble using bootstrap sampling (with replacement) and, at each node, performs the best split from a random subset of features.
  - Implemented with *sklearn*'s *RandomForestClassifier* method to select the number of features necessary to retain 95% of the dataset's information.

- **Recursive Feature Elimination with Cross-Validation**

  - *RFECV* is a feature selection method that recursively removes the least important features and uses cross-validation to determine the optimal number of features that yield the best model performance based on specified criteria (model accuracy in our case).
  - Implemented with *sklearn*'s *RFECV* method

Then the *RandomForestClassifier* was trained in the datasets generated by each of the different feature selection techniques, producing the results found in Table 5.4:

Table 5.4: Automatic feature selection accuracy for different feature selection techniques.

Label	Extra Trees	Logistic Regression	Random Forest	RFECV
multi3cat	55.13%	55%	54.83%	55.26%
multiSN	60.04%	58.49%	59.67%	59%
Average	**57.59%**	56.75%	57.25%	57.13%

After selecting *ExtraTrees* as the preferred feature selection method due to its higher performance when compared to other techniques, the next step involves optimizing its hyperparameters to maximize its efficacy in retaining dataset information with a minimal set of selected features.

To achieve this, *sklearn*'s *GridSearchCV* will be utilized. This method conducts a search over a specified parameter grid to identify the optimal hyperparameters for the model. The best configuration is selected based on cross-validation accuracy, using a 3-fold cross-validation strategy.

The grid consists of combinations of potential values for the following attributes (as per the *sklearn* documentation):

- **n_estimators**: The number of trees in the forest.
- **min_samples_split**: The minimum number of samples required to split an internal node.

- **min_sample_leaf**: The minimum number of samples required at a leaf node. A split point at any depth will only be considered if it leaves at least min_samples_leaf training samples in each left and right branch.
- **criterion**: The function to measure the quality of a split. Supported criteria are *gini* for the Gini impurity, *log_loss* and *entropy* both for the Shannon information gain.
- **max_features**: The number of features to consider when looking for the best split. If "*log2*", then max_features=log2(n_features). If "*sqrt*", then max_features=sqrt(n_features).

Table 5.5 shows the optimal configurations for the *ExtraTrees* algorithm, applied to the two datasets.

Table 5.5: Optimal configurations for the *ExtraTrees* feature selection algorithm.

Parameter	**multi3cat**	**multiSN**
n_estimators	700	700
min_samples_split	2	3
min_samples_leaf	1	1
criterion	entropy	entropy
max_features	log2	sqrt

By optimizing the hyperparameters of the feature selection method, accuracy improved. In the multi3cat dataset, the accuracy increased from 55.13% to 55.56%, with the selection of 348 features from the initial set of 514. Similarly, in the multiSN dataset, the accuracy increased from 60.04% to 60.26%, while the number of features was reduced from 514 to 330.

5.4 Train/Test splits

After reducing the dataset, selecting an appropriate ratio between training and testing data is important. The 90/10 split was chosen for our case. In smaller datasets, a higher train-test split ratio, such as 90/10, allocates more data points to the training set. This enables the model to learn from a larger amount of patterns and variations in the data.

With a larger training set, the model can potentially develop a more accurate and generalized understanding of the relationships among features, enhancing its predictive capacity. However, this split strategy has limitations: since the test set is small it may lead to less reliable performance due to higher variance and less representation of the overall data distribution.

Still, a significant portion of the data remains dedicated to testing, and cross-validation will be used to mitigate these concerns. This approach ensures that all data points contribute to evaluating the model's performance, providing a complete assessment of its predictive accuracy and generalization ability.

Chapter 6

Modeling - 1st iteration

After all the preparatory steps, the investigation advances to the modeling stage. At this stage, multiple algorithms will be applied to the generated datasets to extract information that addresses the queries of ISPUP experts.

Similarly to the feature selection process, the initial step involves tuning the hyperparameters of the data mining algorithms. This is accomplished using *scikit-learn*'s *GridSearchCV* with 3-fold cross-validation. The models will be evaluated based on *accuracy* to identify the optimal parameter set from the specified grid.

After determining the optimal configuration, various algorithms will be trained, and performance metrics will be extracted to evaluate and compare their efficacy [15]. The metrics utilized for this comparative analysis include:

- **Accuracy**: The proportion of correctly classified instances among the total instances.
- **Accuracy standard deviation**: The measure of the variability or dispersion of accuracy values across multiple trials or datasets.
- **Sensitivity**: The proportion of true positive instances correctly identified out of all actual positive instances.
- **Specificity**: The proportion of true negative instances correctly identified out of all actual negative instances.
- **Precision**: The proportion of true positive instances among all instances that were predicted as positive.

These metrics are adequate for evaluating the performance of a data mining model in our domain since they provide insights into the model's ability to correctly identify both healthy (specificity) and multimorbidity (sensitivity) individuals, the reliability of positive predictions (precision), the overall correctness of predictions (accuracy), and the consistency of the model's performance across different folds (accuracy standard deviation).

ROC curves and Area Under the Curve (AUCs) [14] will also be used to assess model performance. ROC curves measure sensitivity versus specificity trade-offs across different thresholds, while AUC evaluates the model's discriminating power. This metric is interesting especially for multi-class scenarios, as it provides insights into the model's capability to differentiate between classes.

6.1 Random Forest classifier

6.1.1 Hyperparameter tuning

Again, using *sklearn*'s *GridSearchCV*, we will tune the hyperparameters for the classification task. The grid consists of combinations of potential values for the following attributes (as per the *sklearn* documentation):

- **n_estimators**: The number of trees in the forest.
- **min_samples_split**: The minimum number of samples required to split an internal node.
- **min_sample_leaf**: The minimum number of samples required at a leaf node. A split point at any depth will only be considered if it leaves at least min_samples_leaf training samples in each left and right branch.
- **criterion**: The function to measure the quality of a split. Supported criteria are *gini* for the Gini impurity, *log_loss* and *entropy* both for the Shannon information gain.
- **max_features**: The number of features to consider when looking for the best split. If "*log2*", then max_features=log2(n_features). If "*sqrt*", then max_features=sqrt(n_features).

The optimal configurations can be found in Table 6.1.

Table 6.1: Optimal configurations for the *RandomForestClassifier*.

Parameter	multi3cat	multiSN
n_estimators	500	900
min_samples_split	2	2
min_samples_leaf	1	1
max_features	log2	log2
criterion	gini	gini

6.1.1.1 Modeling results

Table 6.2: Model performance for the *RandomForestClassifier*.

Parameter	Accuracy	Accuracy standard deviation	Sensitivity	Specificity	Precision
multi3cat	58.58%	11.44%	59.36%	79.82%	62.23%
multiSN	62.19%	9.03%	60.48%	64.64%	62.55%

6.1.1.2 Performance analysis

Results can be seen in Table 6.2. Both models exhibit moderate performance in accuracy, sensitivity, and precision, indicating that they are not highly capable of identifying true positive cases, which is critical in our health-related context. The high specificity suggests the model is better at identifying non-multimorbidity individuals. However, the high accuracy standard deviations, especially for multi3cat, highlight the need for further model refinement to achieve more stable and reliable performance across different data folds.

Next, we will explore neural networks, as they are good at identifying intricate and nonlinear relationships within the data, which is common in health-related datasets. They can handle large and high-dimensional data effectively, making them a good fit for complex health-related problems. They often achieve higher performance metrics, such as accuracy, sensitivity, and precision, which are critical for correctly predicting the appearance of diseases.

6.2 Neural Network

To implement the neural network, we will use *sklearn*'s *MLPClassifier*.

6.2.1 Hyperparameter tuning

- **hidden_layer_sizes**: The ith element represents the number of neurons in the ith hidden layer
- **activation**: Activation function for the hidden layer.
- **alpha**: Strength of the L2 regularization term. The L2 regularization term is divided by the sample size when added to the loss.
- **learning_rate**: Learning rate schedule for weight updates.
- **max_iter**: Maximum number of iterations.

Once again, we will use *sklearn*'s *GridSearchCV*, with 3-fold cross-validation. The optimal configurations are displayed in Table 6.3.

Table 6.3: Optimal configurations for the *MLPClassifier*.

Parameter	multi3cat	multiSN
hidden_layer_sizes	(300, 150)	(300, 300)
activation	tanh	relu
alpha	0.01	0.01
learning_rate	constant	constant
max_iter	500	500

6.2.2 Modeling results

Table 6.4: Model performance for the *MLPClassifier*.

Parameter	Accuracy	Accuracy standard deviation	Sensitivity	Specificity	Precision
multi3cat	57.4%	12.18%	57.38%	78.69%	57.5%
multiSN	60.55%	7.77%	65.8%	55.27%	59.02%

6.2.2.1 Performance analysis

The performance of the neural networks, shown in Table 6.4, shows a small decline compared to the random forest model. However, the difference in overall performance is negligible. Given these results, it was decided to move forward with the more intelligible algorithm to ensure better interpretability of the model's decisions. The opaque nature of neural networks limits the ability to see which specific features are driving predictions and limits our ability to get actionable insights that can respond to the queries of the ISPUP experts.

Our next experiment will be with SVCs due to their ability to handle high-dimensional datasets, a characteristic particularly advantageous in our case, which has diverse and complex variables. SVCs can construct optimal hyperplanes to separate classes in multidimensional feature spaces, facilitating the classification tasks.

6.3 Support vector classification

We will use *sklearn*'s *SVC* to implement the support vector classifier.

6.3.1 Hyperparameter tuning

- **C**: Regularization parameter. The strength of the regularization is inversely proportional to C.
- **kernel**: Specifies the kernel type to be used in the algorithm.
- **gamma**: Kernel coefficient for 'rbf', 'poly' and 'sigmoid'.
- **degree**: Degree of the polynomial kernel function ('poly').
- **coef0**: Independent term in kernel function. It is only significant in 'poly' and 'sigmoid'.
- **max_iter**: Hard limit on iterations within solver, or -1 for no limit.

Once again, we will use *sklearn*'s *GridSearchCV*, with 3-fold cross-validation. The optimal hyperparameter configuration can be seen in Table 6.5.

Table 6.5: Optimal configurations for *SVC*.

Parameter	multi3cat	multiSN
C	20	10
kernel	rbf	poly
degree	N/A	4
gamma	auto	auto
coef0	0	0
max_iter	-1	-1

6.3.2 Modeling results

Table 6.6: Model performance for *SVC*.

Parameter	Accuracy	Accuracy standard deviation	Sensitivity	Specificity	Precision
multi3cat	58.8%	9.81%	58.79%	79.4%	58.95%
multiSN	61.92%	9.46%	69.19%	54.61%	59.76%

6.3.3 Performance analysis

The Support Vector Classifier model shows improvements over neural networks and similar results to the random forest in accuracy and sensitivity, as seen in Table 6.6. However, extracting feature importances directly from a Support Vector Classifier (SVC) without a linear kernel, can be challenging. SVCs with non-linear kernels (e.g., ploy or rbf) do not have coefficients that directly indicate feature importance like linear models, such as Random Forests, do.

6.4 Final remarks

Moving forward, random forests have been identified as the best model choice based on their ability to handle complex relationships, provide clear insights into feature importance, and maintain consistent performance across various metrics. Despite testing different algorithms, none yielded significantly better results when compared to others, with the performance metrics generally remaining in the 60% range. This highlights the critical need to enhance feature engineering efforts and potentially address problems in raw data quality.

Following discussions with the project supervisor and the ISPUP experts, an agreement has been reached to explore new feature engineering hypotheses and introduce alternative model algorithms. These steps aim to improve predictive accuracy in the following phases of the project.

Chapter 7

Modelling - 2nd iteration

Following the satisfactory results observed in the initial modeling iteration and the feedback gathered after the first iteration, we have a set of strategies to enhance the model's performance. These strategies include improvements in data preparation methodologies and new experiments with modeling techniques.

- **Data preparation**
  - Establish a threshold to eliminate features not meeting certain information gain criteria.
  - Add six new features gathered by the ISPUP experts.
- **Modelling**
  - Experiment with the CART algorithm.
  - Experiment with Adaboost with Decision Trees
  - Experiment with the XGBoost algorithm.

We will start with modifications in data preparation methodologies and evaluate their efficacy using the Random Forest algorithm used in the previous iteration. If these adjustments improve model performance, they will be implemented definitively.

7.1 Eliminate features based on an information gain threshold

For this analysis, we will execute a *RandomForestClassifier* on the complete dataset to compute the information gain associated with each feature. We will aggregate the information gained across all measured years for each feature. Afterward, we will assess whether there is a point where the reduction in information gain becomes substantial, and we will remove features beyond that point.

7.1.1 Multi3cat dataset

For the multi3cat dataset, the plot presented in Annex A.13 reveals several points where there is a decline in the information gain of subsequent features. Based on this observation, we have selected four distinct subsets of features, each defined by a cutoff point.

The subsets were generated starting from the top feature(the one with the most information gain) until and including:

- *connind100*
- *ndvi500_mean*
- *blue_size*
- All features

The performance obtained is presented in Table 7.1:

Table 7.1: Comparison of metrics for different feature subsets for the multi3cat dataset.

Metric	Accuracy	Accuracy standard deviation	Sensitivity	Specificity	Precision	Average
connind100	56.54%	11.61%	61.02%	80.27%	60.92%	64.69%
ndvi500_mean	56.9%	11.86%	59.22%	79.22%	58.36%	63.43%
blue_size	56.21%	11.63%	58.46%	79.08%	58.05%	62.95%
All features	58.58%	11.44%	59.36%	79.82%	62.23%	**65**%

7.1.2 MultiSN dataset

For the multi3cat dataset, the plot presented in Annex A.14 reveals points where there is a decline in the information contribution of subsequent features. Based on this observation, we have identified four distinct subsets of features, each defined by a unique cutoff point.

The subsets were generated starting from the top feature(the one with the most information gain) until and including:

- *popdens*
- *bdens300*
- *lines_300*
- All features

The performance obtained is presented Table 7.2:

Table 7.2: Comparison of metrics for different feature subsets for the multiSN dataset.

Metric	Accuracy	Accuracy standard deviation	Sensitivity	Specificity	Precision	Average
popdens	60.04%	9.55%	60.15%	63.04%	61.22%	61.88%
bdens300	60.85%	9.2%	60.9%	65.22%	63.09%	**62.52**%
lines_300	60.11%	9.63%	57.14%	66.67%	62.01%	61.48%
All features	62.19%	9.03%	60.48%	64.64%	62.55%	62.47%

7.1.3 Performance analysis

The comparison of performance metrics across different subsets of features and the complete dataset shows small differences in model effectiveness. The dataset using all features achieves the highest accuracy, indicating that including the complete range of features improves the model's ability to classify instances compared to subsets with fewer variables.

Interestingly, while the dataset with all features performs well in accuracy, it also exhibits a slightly higher standard deviation. This suggests that including more features introduces complexity that may affect model consistency across different iterations or folds of cross-validation.

Analyzing sensitivity and specificity metrics, the dataset with all features demonstrates balanced values, indicating robust performance in correctly identifying positive and negative instances.

Precision shows consistent results across all datasets, with the dataset using all features achieving the highest precision, which suggests that including the complete set of features contributes to more accurate predictions of positive outcomes.

In summary, while subsets of features show moderate performance metrics, the dataset incorporating all features demonstrates superior overall performance across various evaluation metrics.

7.2 Introduce new variables to the dataset

Following the discussion after the first iteration, it became evident that the available data was insufficient to achieve better results for the selected label problems. Consequently, ISPUP experts proposed to introduce six new features to enhance the model's predictive accuracy and improve its ability to classify instances correctly.

The variables were the following:

- **idademae**: Mother's age at the moment of birth.
- **imc**: Mother's IMC measured in the first three months of pregnancy.

- **SEXONASC**: Gender of the newborn.
- **ESCOLARIDADEMAE_CAT**: Mother's education level.
- **fumagestRED**: Whether the mother smoked or not during the pregnancy. 0 for the cases it did not smoke, 1 for the cases where she smoked until three months of pregnancy, and 2 for the cases where it smoked during the whole pregnancy.
- **RENDIMENTO_RED_CAT**: Family's liquid income level. 0 for the cases where it fell between 0 and 1000€, 1 for the cases where it fell between 1001€ and 2000€, and 2 for the cases where it was higher than 2000€.

The dataset provided contains some missing values. We imputed these values using the *MICE* algorithm after suggestions from the ISPUP experts. Subsequently, these new variables were added to the datasets generated in the previous sections, and the analysis was conducted again to evaluate the performance with and without the new features.

7.2.1 Multi3cat Dataset

We will test the dataset with the newly added features against the benchmark obtained in the previous section.

The performance obtained is presented in Table 7.3:

Table 7.3: Comparison metrics for the multi3cat dataset with and without the new features.

Metric	Accuracy	Accuracy standard deviation	Sensitivity	Specificity	Precision	Average
New features	57.29%	12.13%	61.44%	81.37%	62.83%	**65.73**%
Benchmark	58.58%	11.44%	59.36%	79.82%	62.23%	65%

7.2.2 MultiSN Dataset

Here, we will test the new dataset generated against the benchmark, as shown in the results obtained in the previous section.

The performance obtained is presented in Table 7.4:

Table 7.4: Comparison metrics for the multiSN dataset with and without the new features.

Metric	Accuracy	Accuracy standard deviation	Sensitivity	Specificity	Precision	Average
New features	60.52%	10.2%	60.15%	67.39%	63.85%	62.98%
Benchmark	60.59%	10.03%	59.4%	70.29%	65.02%	**63.83%**

7.2.3 Performance analysis

The dataset with new features achieved a slightly lower accuracy than the benchmark's. This suggests that the new features may provide additional information but have not significantly improved the model's overall accuracy. The same can be seen in the standard deviation, which was also slightly higher when adding the new features.

However, sensitivity and specificity slightly improved with the new features. This improvement indicates a better ability to identify positive and negative instances correctly.

Precision remained nearly the same, suggesting that the new features did not significantly affect the precision of positive predictions.

Overall, the enhancements are negligible. We decided to discard these new features after presenting these results to the ISPUP experts. They preferred to treat these features as covariates, and not as predictors, and given the minimal increase in performance, it was concluded that excluding these features would be best to avoid the impact on the remaining results.

We believe the issue is not related to the quantity or quality of features but to the insufficient number of entries in the dataset, which fails to represent all the possible profiles this investigation includes. This issue will be elaborated upon further in the next chapter of this work.

7.3 CART

After experimenting with data preparation approaches with little success, our final attempt to improve performance focused on the modeling stage. Based on a suggestion from the project's supervisor, we decided to evaluate the performance of the *CART*, *AdaBoost with Decision Trees*, and *XGBoost* algorithms.

7.3.1 Hyperparameter tuning

The optimal hyperparameters for the *CART* algorithm can be found in Table 7.5.

Table 7.5: Optimal configurations for *CART*.

Parameter	multi3cat	multiSN
splitter	random	random
min_samples_split	2	2
min_samples_leaf	1	1
criterion	gini	entropy
max_features	sqrt	log2

7.3.2 Modeling results

Table 7.6: Model performance for *CART*.

Parameter	Accuracy	Accuracy standard deviation	Sensitivity	Specificity	Precision
multi3cat	46.36%	6.14%	46.36%	73.18%	45.86%
multiSN	56.13%	5.5%	59.1%	53.14%	55.61%

7.3.3 Performance analysis

Looking at Table 7.6, while the performance metrics for both datasets show some differences, none of them demonstrate strong predictive power. The multiSN dataset shows relatively better overall performance than the multi3cat dataset, particularly in sensitivity and precision.

However, accuracy and consistency are insufficient for model performance. In this particular case, the model performs only slightly better than a random predictor, which is not the desirable outcome.

We will utilize the *AdaBoost* algorithm with decision trees as the base learners to explore performance improvements with this technique. By using the ensemble method *AdaBoost*, we aim to improve the base learner's (Decision Trees) overall predictive capabilities.

7.4 Adaboost with Decision Trees

To implement the *Adaboost* algorithm, we will use *sklearn*'s *AdaBoostClassifier*.

7.4.1 Hyperparameter tuning

- **learning_rate**: Weight applied to each classifier at each boosting iteration. A higher learning rate increases the contribution of each classifier.
- **n_estimators**: The maximum number of estimators at which boosting is terminated.
- **algorithm**: Algorithm to use for *AdaBoostClassifier*. Can be either *SAMME* or *SAMME.R*

The optimal hyperparameter configuration can be found in Table 7.7.

Table 7.7: Optimal hyperparameter configurations for the *AdaBoostClassifier*.

Parameter	multi3cat	multiSN
learning_rate	0.1	1
n_estimators	100	300
algorithm	SAMME	SAMME

7.4.2 Modeling results

Table 7.8: Model performance for the *AdaBoostClassifier*.

Parameter	Accuracy	Accuracy standard deviation	Sensitivity	Specificity	Precision
multi3cat	50.21%	11.55%	50.2%	75.1%	50.65%
multiSN	56.27%	8.98%	59.45%	53.06%	49.5%

7.4.3 Performance analysis

Looking at the results in Table 7.8, we can conclude that the application of AdaBoost contributed to an increase in performance for both datasets, suggesting that the ensemble method improved the model's classification accuracy in this context. Still, they could not surpass the previous predicting capabilities obtained by the best-performing models.

These results point to data quality and quantity as potential bottlenecks in achieving higher model performance. Insufficient or inadequate data most likely limits the model's ability to generalize effectively and make accurate predictions across different datasets.

7.5 XGBoost

In a final effort to improve performance, we opted to implement *XGBoost* as a recommendation from the investigation's supervisor..

7.5.1 Hyperparameter tuning

The optimal hyperparameters for the *XGBoost* algorithm are presented in Table 7.9.

Table 7.9: Optimal configurations for the *XGBoost*.

Parameter	multi3cat	multiSN
subsample	0.9	0.6
reg_lambda	0.5	0.5
reg_alpha	0	0
n_estimators	300	300
learning_rate	0.3	0.1
gamma	0	0
colsample_bytree	0.9	0.9

7.5.2 Modeling results

Table 7.10: Model performance for the *XGBoost*.

Parameter	Accuracy	Accuracy standard deviation	Sensitivity	Specificity	Precision
multi3cat	54.54%	10.49%	54.52%	77.26%	55.66%
multiSN	59.67%	9.64%	61.07%	58.22%	58.4%

7.6 Results analysis

Looking at Table 7.10, the results obtained with *XGBoost* for the multi3cat and multiSN datasets show moderate improvements in performance metrics compared to previous models. While these results are satisfactory, they do not outperform the top data mining algorithms tested in earlier analyses.

Chapter 8

Modeling - 3rd iteration

The challenge of classifying multimorbidity is dependent on data quantity and quality. Multimorbidity involves the presence of multiple chronic conditions in individuals, making it a complex health issue that requires diverse datasets with many different outcome configurations to train a model for accurate classification successfully.

Training a classifier for this problem is challenging due to the many possible disease combinations and the varying environmental exposure profiles associated with each combination. For instance, individuals can have different combinations of 2, 3, 4, 5, or even 6 diseases, each forming a unique health profile.

Each of these disease combinations can be influenced by distinct sets of environmental exposures. Despite having different environmental exposure profiles, all these diverse combinations of diseases may fall under the same label of multimorbidity. This heterogeneity within the same class makes it difficult for a classifier to learn consistent patterns and make accurate predictions.

The issue described demonstrates the imbalance within the dataset, specifically on the configurations of outcomes among the 2400 individuals that constitute the final dataset. Out of the total 128 possible configurations of outcomes, only 98 are present in this dataset. This indicates that certain configurations of health conditions are missing from the available data.

Focusing on configurations involving exactly 2 health conditions, we have 21 possible configurations. Among these 21 configurations, only 9 have a sample size larger than the average number of individuals per configuration (24). This imbalance suggests that specific combinations of health conditions, even within the subset of exactly 2, are not adequately represented in the dataset. The same is verified in configurations with exactly three health conditions, where for 35 possible configurations, only 21 are represented in the dataset. Of these 21 configurations, only 3 have a sample size larger than the average number of individuals per configuration (24).

Figure 8.1 shows the distribution of outcome configurations, sorted from highest to lowest. The average number of entries per configuration is visible in the horizontal red line.

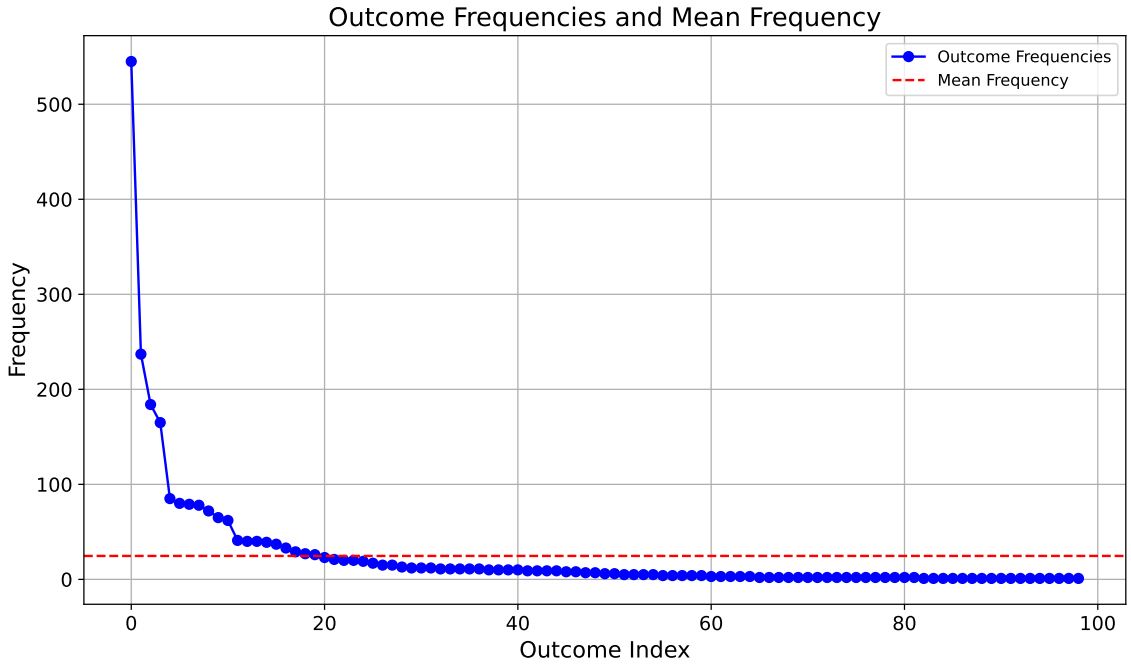


Figure 8.1: Frequency of the outcome configurations.

Approximately 20% of the configurations exceed the mean, while the remaining 80% fall below it. This distribution demonstrates the difference in the frequency of diseases, whether they occur in isolation or in combination with other conditions.

However, it is desirable that the model is able to capture patterns for as many outcome combinations as possible, which is challenging given the limited representation for most configurations.

SMOTE, in the way that it is implemented, cannot effectively address the challenge of synthesizing meaningful samples that capture the variability in these different subgroups, as they are all abstracted within the same class label.

Having these subgroups abstracted into the same single class label makes it difficult for the classifier to differentiate and learn from the patterns within each subgroup. The lack of a more balanced representation of these subgroups hurts the classifier's ability to learn the complete set of environmental exposure profiles.

To address these data constraints and the challenges encountered, we decided to explore two strategies for this third modeling iteration:

- **Strategy 1 - Oversampling Minority Health Outcome Configurations**: The first strategy involves identifying each unique health outcome configuration within the dataset. Using the Synthetic Minority Over-sampling Technique (SMOTE), we will oversample the under-represented configurations until they reach the average number of entries per configuration. Only configurations with less than 24 entries in the initial dataset will be considered for

this process. This approach aims to mitigate the imbalance by artificially increasing the instances of minority configurations.

- **Strategy 2 - Analysis of Specific Health Outcomes**: The second strategy involves applying data mining algorithms to individual health outcomes separately. By focusing on one specific outcome at a time, we can avoid the complexities associated with the imbalance of multiple outcome configurations. This approach addresses the limitations encountered in the multimorbidity study, while letting us identify key associations between environmental factors and a specific health outcome.

8.1 Oversampling of Minority Health Outcome Configurations

After performing the oversampling of minority outcome configurations, all 98 distinct configurations present in the dataset will have at least 24 entries representing them. We will now finely tune the hyperparameters for the *RandomForestClassifier*, before assessing the model's performance. The optimal configuration can be found in Table 8.1.

8.1.1 Hyperparameter tuning

Parameter	multi3cat	multiSN
n_estimators	500	200
min_samples_split	2	3
min_samples_leaf	1	1
max_features	log2	sqrt
criterion	gini	entropy

Table 8.1: Optimal configurations for the *RandomForestClassifier*.

8.1.1.1 Modeling results

Parameter	Accuracy	Accuracy standard deviation	Sensitivity	Specificity	Precision
multi3cat	88.34%	11.8%	88.34%	94.17%	88.92%
multiSN	84.31%	16.02%	74.43%	94.15%	90.25%

Table 8.2: Model performance for the *RandomForestClassifier*.

Examining the results presented in Table 8.2, we observe an improvement in model performance following the oversampling of some of the rarest outcome configurations. This strategy resulted in

an approximate 20% increase in the obtained metric compared to previous analyses. By oversampling the rarer configurations, the model was able to learn additional patterns in the data, leading to the observed performance improvement.

This finding highlights the impact of a lack of data and domain representation on model performance. While the oversampling approach had better results, it altered the natural distribution of diseases and their combinations. Certain outcomes and configurations are rarer than others, and modifying these ratios can potentially cause the model to overfit. Nonetheless, the extent of oversampling was controlled, with underrepresented classes being oversampled to only 5% of the size of the majority class, mitigating the risk of overfitting.

8.2 Analysis of Specific Health Outcomes

As stated above, to handle the issue of imbalance in outcome configurations, we will now analyze each individual outcome.

The plot in Figure 8.2 illustrates the frequency of each outcome in the dataset.

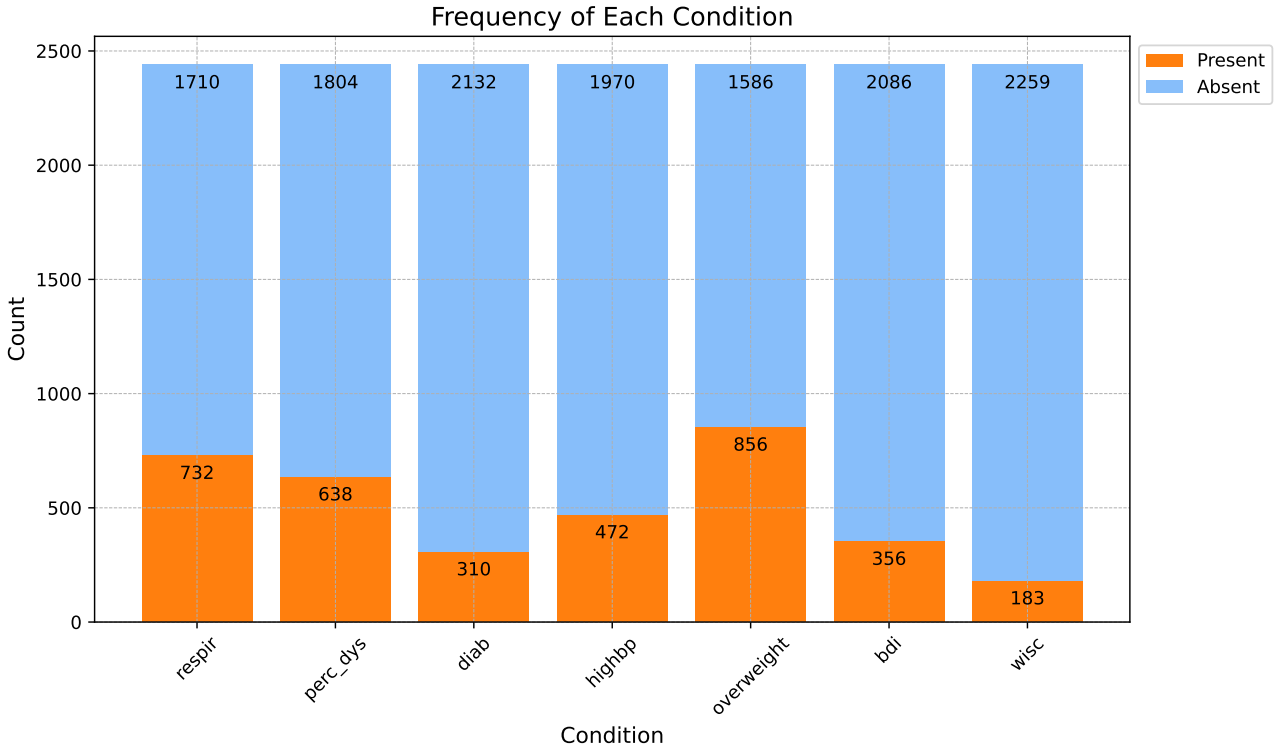


Figure 8.2: Specific outcome count for each of the binary labels.

The prevalence of outcomes varies across the dataset, ranging from 7.5% for the rarest outcome (*wisc*) to 35% for the most common one (*overweight*). To address this imbalance, SMOTE will be employed once again to oversample the positive outcome classes until balance is achieved.

8.2.1 Feature selection hyperparameter tuning

In Table 8.3 are the optimal hyperparameter configurations for the *ExtraTreesClassifier* feature selection model.

Table 8.3: *ExtraTrees* feature selection performance metrics for all specific outcome datasets

Dataset	n_estimators	min_sample_split	min_sample_leaf	max_features	criterion
bdi	400	2	1	log2	entropy
diab	800	2	1	log2	gini
highbp	800	2	1	log2	gini
overweight	800	2	1	sqrt	entropy
perc_dys	400	2	1	log2	entropy
respir	400	2	1	log2	gini
wisc	600	2	1	log2	entropy

8.2.2 Random Forest hyperparameter tuning

In Table 8.4 are the optimal hyperparameter configurations for the *RandomForestClassifier*.

Table 8.4: Optimal hyperparameter configuration for the *RandomForestClassfier*.

Dataset	n_estimators	min_sample_split	min_sample_leaf	max_features	criterion
bdi	1000	2	1	log2	entropy
diab	800	2	1	log2	entropy
highbp	1000	2	1	log2	gini
overweight	200	5	1	sqrt	entropy
perc_dys	600	2	1	sqrt	entropy
respir	400	2	1	sqrt	entropy
wisc	600	2	1	log2	entropy

8.2.3 Performance analysis

Table 8.5 are performance metrics gathered for the distinct outcomes analyzed.

Table 8.5: Model performance metrics for all specific outcome datasets.

Dataset	Accuracy	Accuracy standard deviation	Sensitivity	Specificity	Precision
bdi	93.48%	7.82%	91.83%	95.11%	94.87%
diab	94.33%	5.98%	93.84%	94.82%	94.42%
highbp	89.82%	8.21%	88.8%	90.81%	90.65%
overweight	69.71%	13.66%	69.17%	70.17%	67.83%
perc_dys	84.49%	11.27%	81.13%	87.84%	85.39%
respir	79.38%	11.35%	79.22%	79.56%	78.59%
wisc	97.36%	4.82%	96.59%	98.13%	98.06%

With the results obtained, we can conclude that focusing on individual health outcomes rather than tackling the multimorbidity problem had superior model performance.

Specifically, outcomes such as *bdi*, *diab*, and *wisc* reported high model accuracy and precision. These outcomes also exhibited the lowest frequency of positive outcomes, suggesting a potential correlation between model performance and the imbalance in the outcome classes. It is important to consider that the higher performance observed in these cases may be influenced by a slight overfitting to the minority class.

Furthermore, our analyses revealed a higher specificity compared to sensitivity across all outcomes. This finding highlights the model's tendency to more precisely identify instances of the negative class (true negatives).

Chapter 9

Evaluation

This chapter will use the best model and dataset configuration obtained to evaluate performance according to the selected metrics. We will then gather the importance of each feature in the model and answer the questions that started this investigation.

The results in this chapter were generated using the *RandomForestClassifier*, with the appropriate hyperparameters, for the datasets obtained by applying the *ExtraTreesClassifier* to select the most important features.

To effectively visualize the results, three distinct plots were generated:

- **Plot 1: Aggregated Feature Importance by Attribute Across All Years**: This plot presents an attribute's aggregated feature importance scores throughout the study. The feature importance of each attribute is normalized by the number of measurements taken for that attribute, ensuring that attributes with varying measurement frequencies are equally compared. The attributes are color-coded based on their respective categories, allowing the visualization of the contribution of each category to the overall feature importance.

- **Plot 2: Time Frame Importance**: This plot presents the time-frame importance score by aggregating the importance of the features that fall within that time frame. The importance of each time frame is normalized by the number of measurements taken for that time frame, ensuring that time frames with a different number of features are equally compared.

- **Plot 3: Feature Importance of Attributes by Time-Frame**: This plot illustrates the feature importance scores of individual attributes within each of the three time-frames (0-3 years, 4-7 years, and 8-12 years). Each attribute's importance score is normalized by the number of measurements for that attribute within the respective time frame. The attributes are color-coded by categories, allowing the visualization of the contribution of each category to the overall feature importance.

9.1 Multi3cat and multiSN analysis

9.1.1 Multi3cat

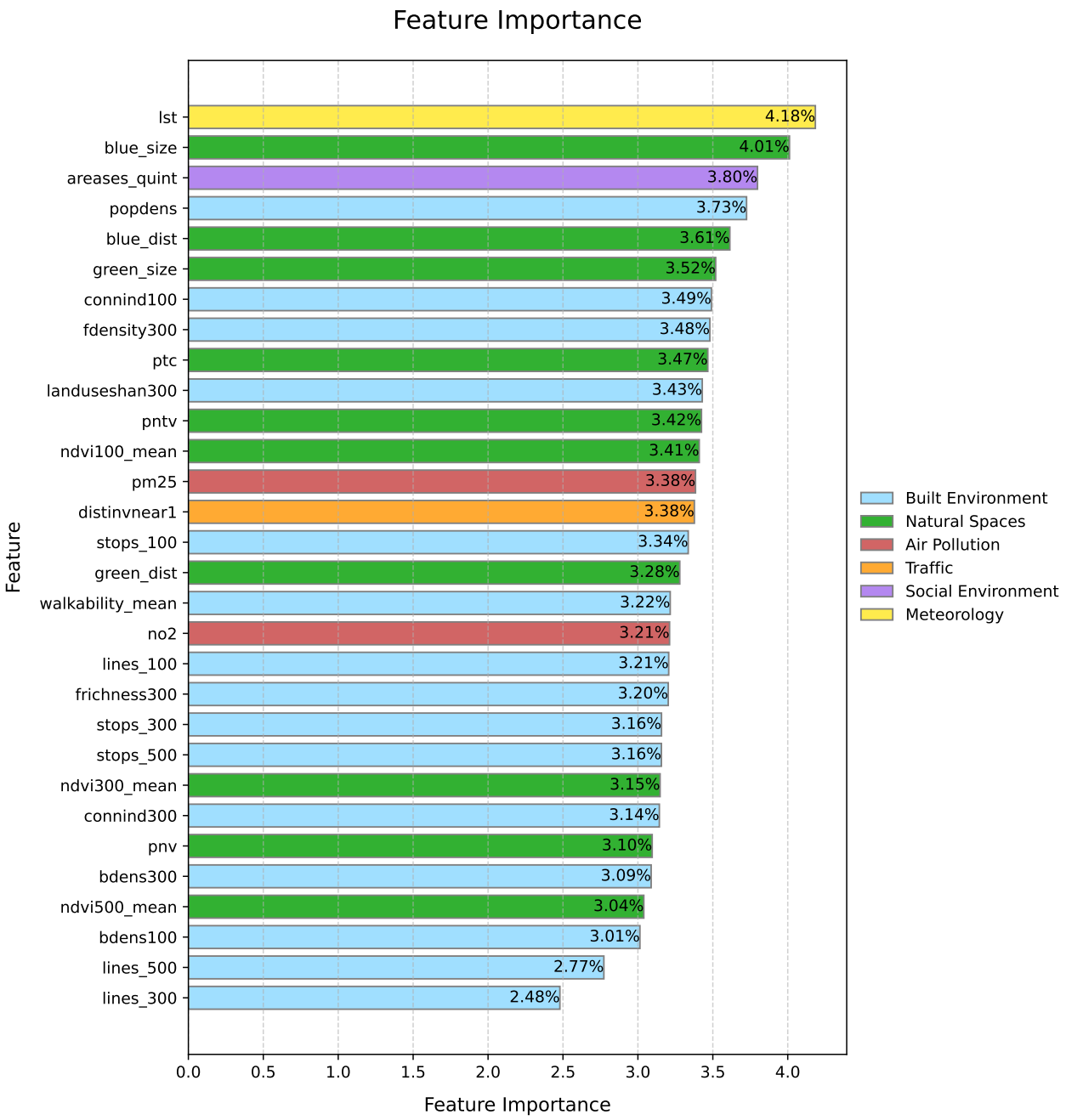


Figure 9.1: Feature importances for the multi3cat dataset, colored by feature type.

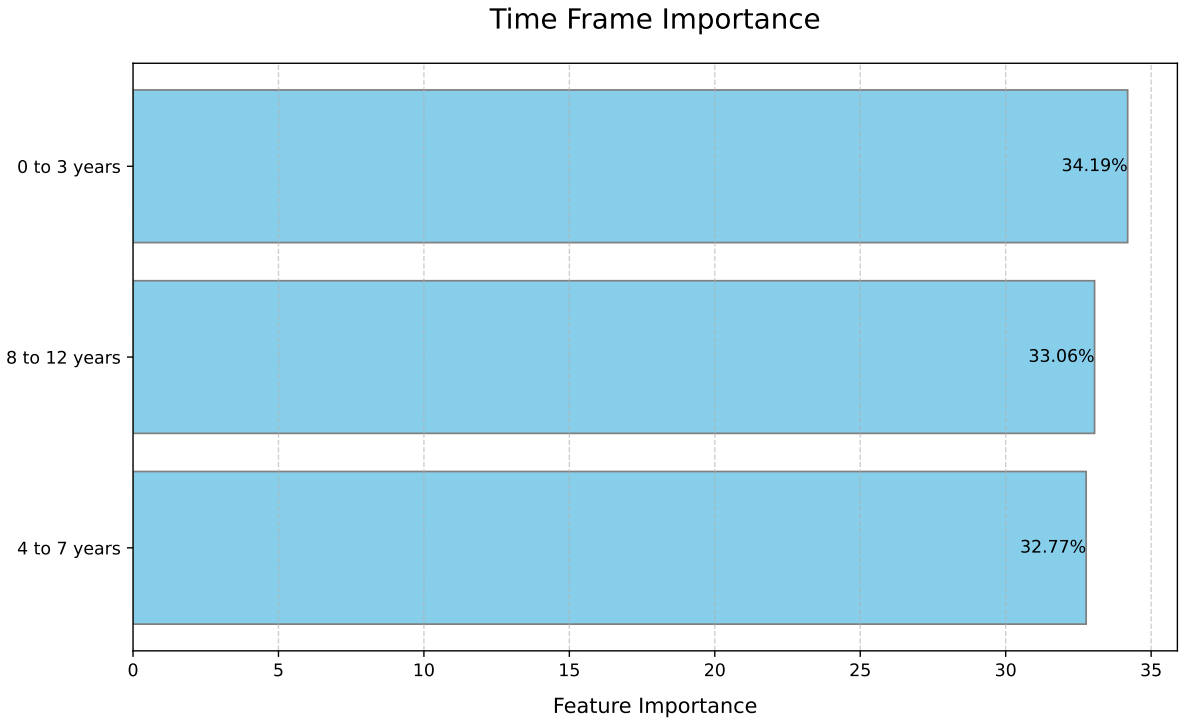


Figure 9.2: Time-frame importances for the multi3cat dataset.

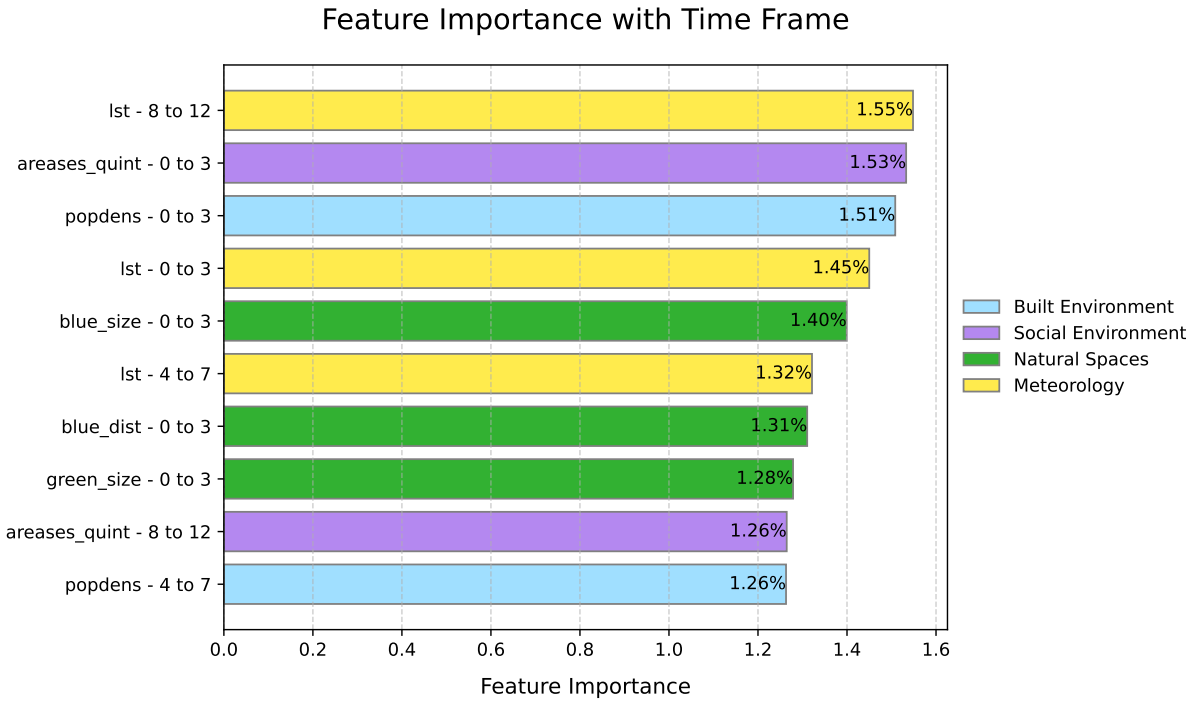


Figure 9.3: Time-frame specific feature importances for the multi3cat dataset.

9.1.2 MultiSN

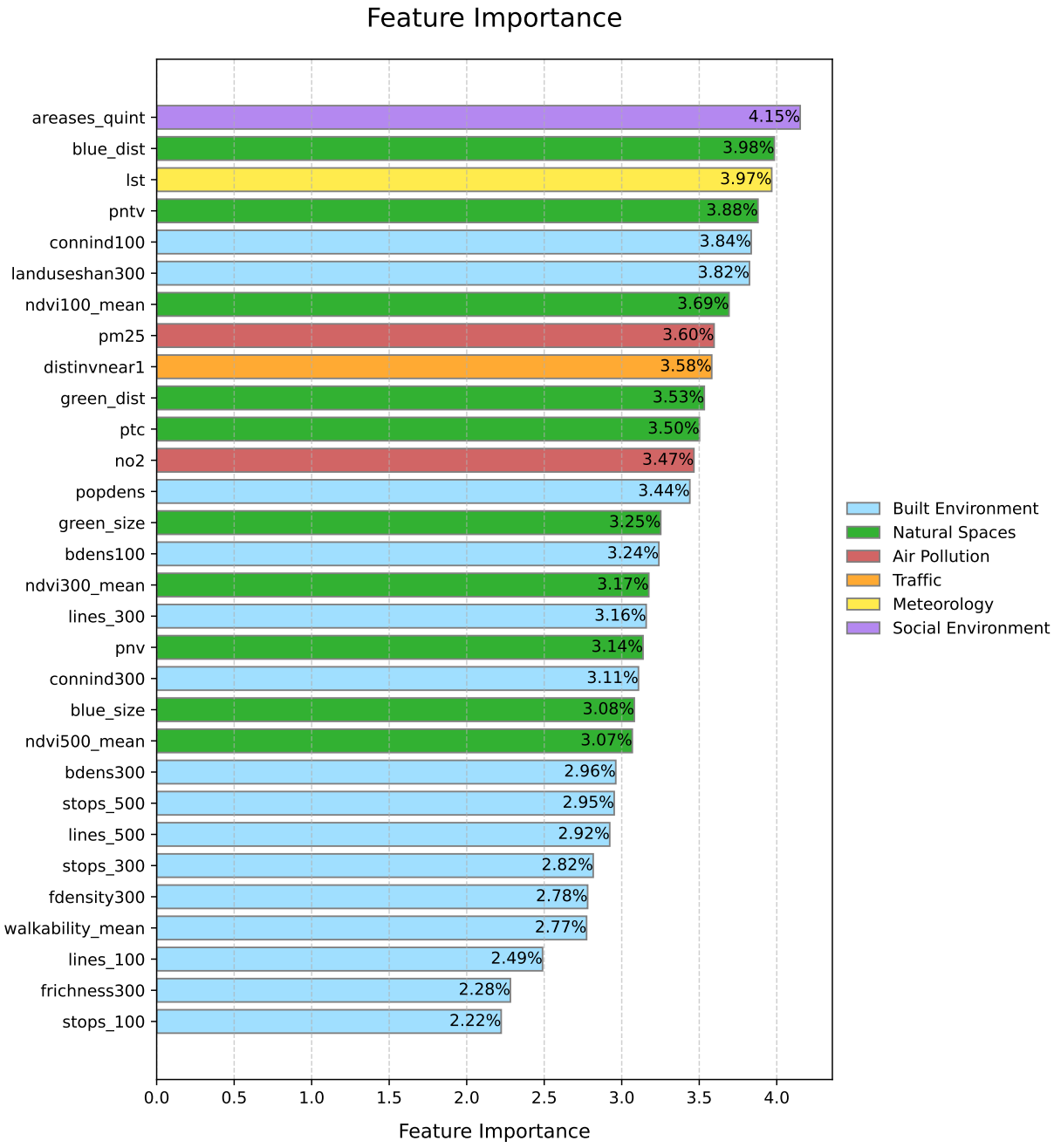


Figure 9.4: Feature importances for the multiSN dataset, colored by feature type.

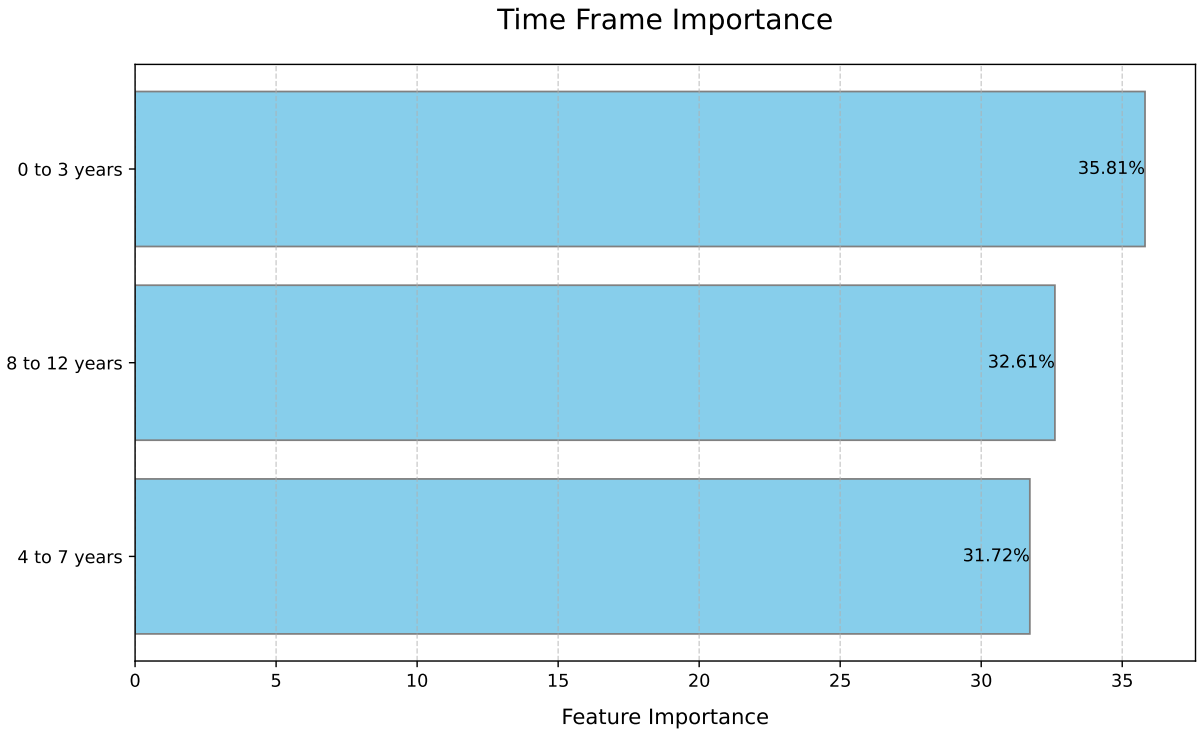


Figure 9.5: Time-frame importances for the multiSN dataset.

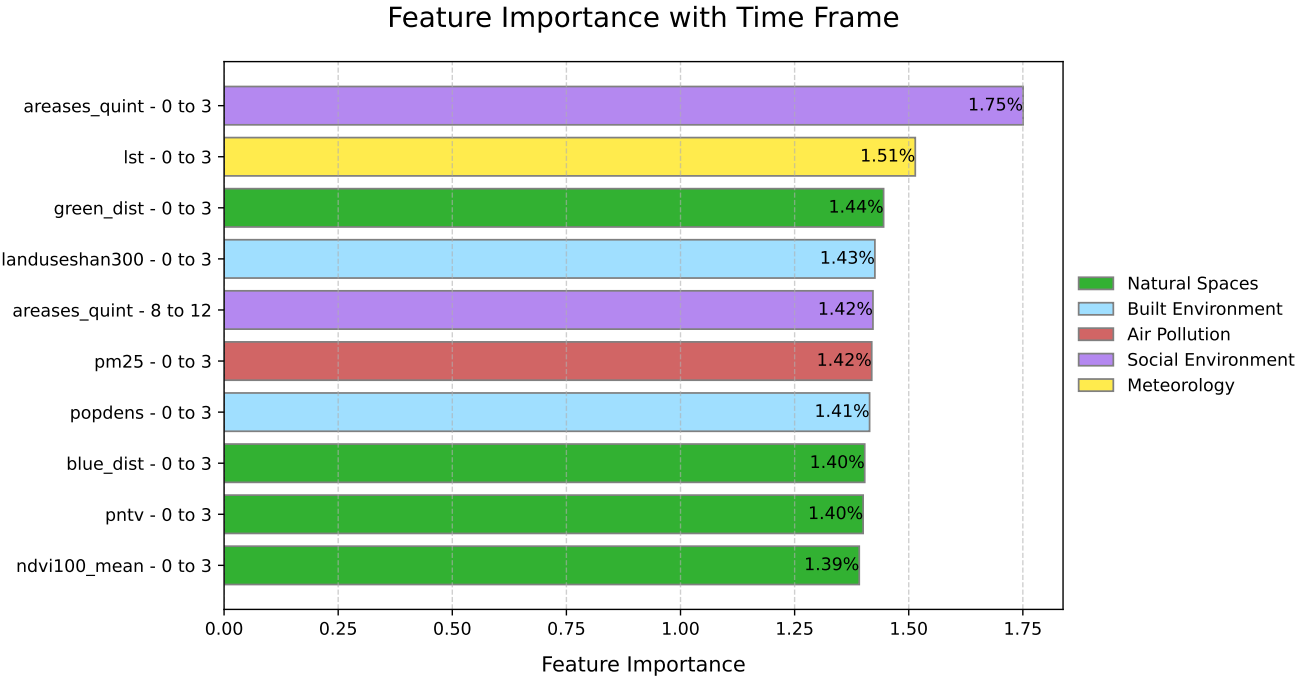


Figure 9.6: Time-frame specific feature importances for the multiSN dataset.

9.2 Specific outcome analysis

9.2.1 bdi

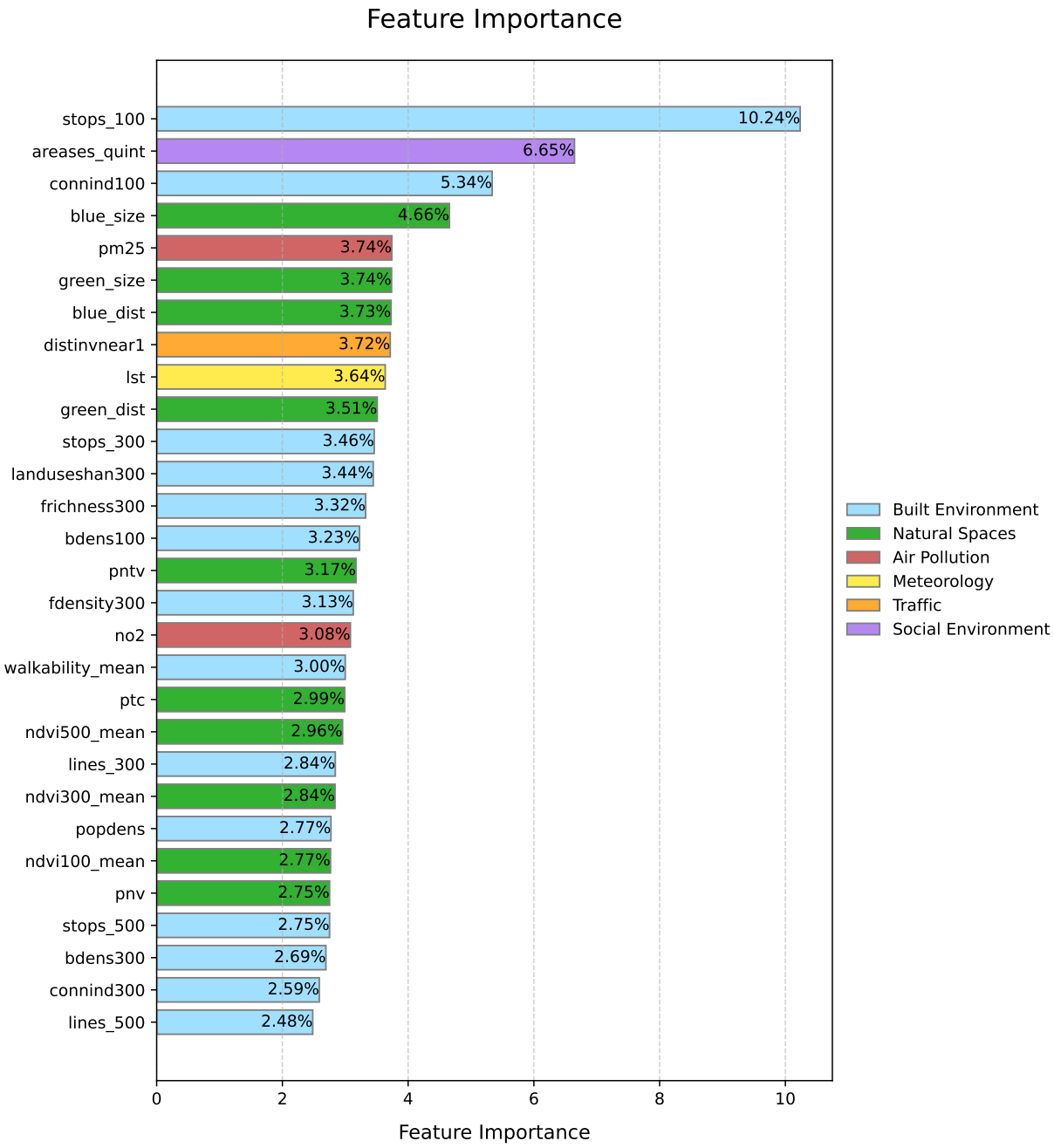


Figure 9.7: Feature importances for the bdi dataset, colored by feature type.

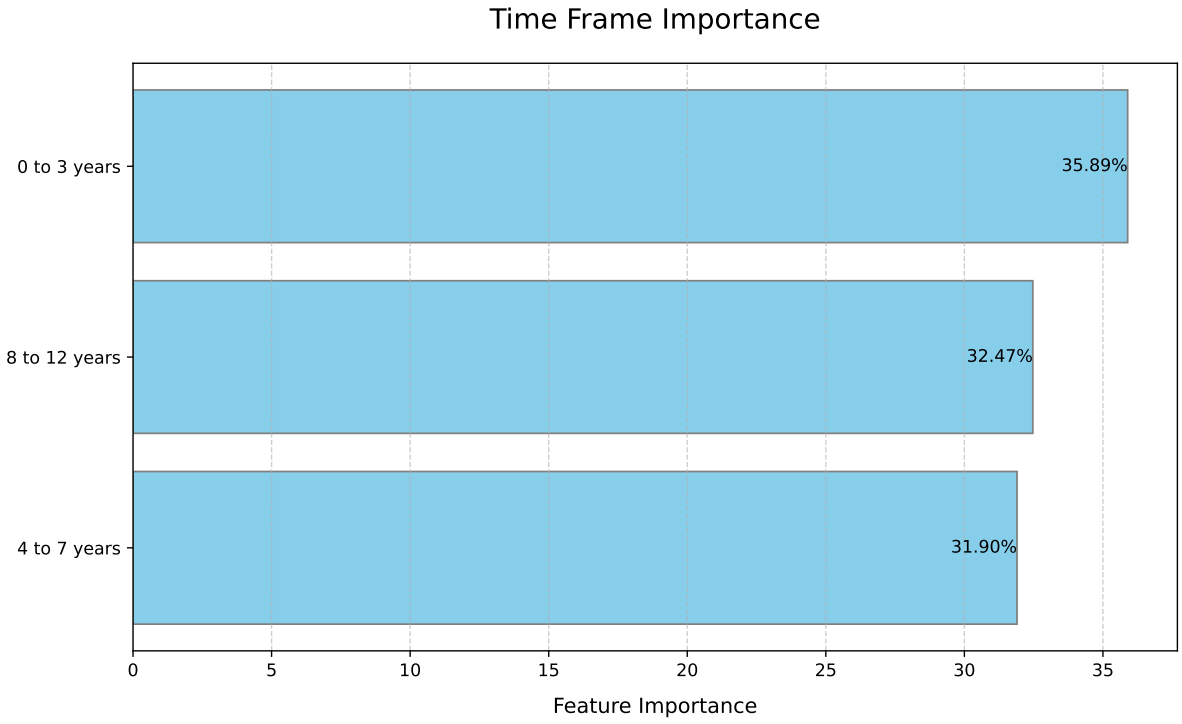


Figure 9.8: Time-frame importances for the bdi dataset.

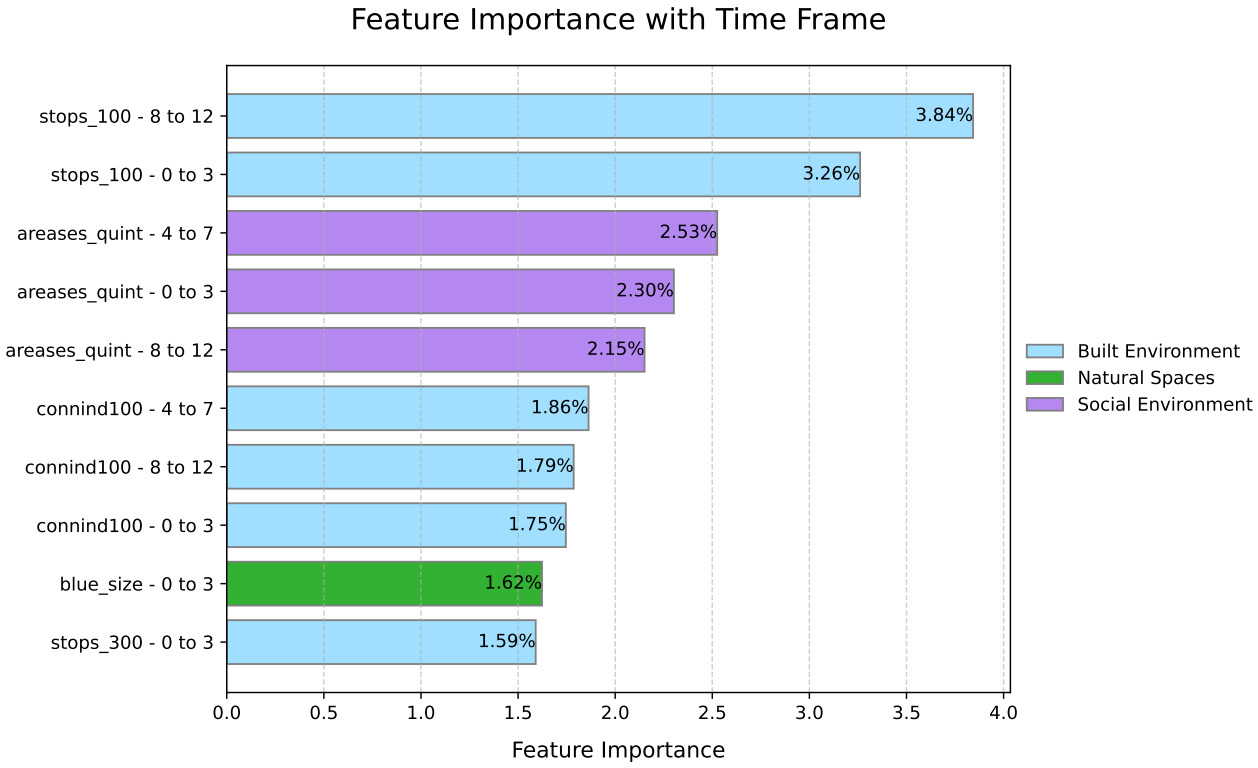


Figure 9.9: Time-frame specific feature importances for the bdi dataset.

9.2.2 diab

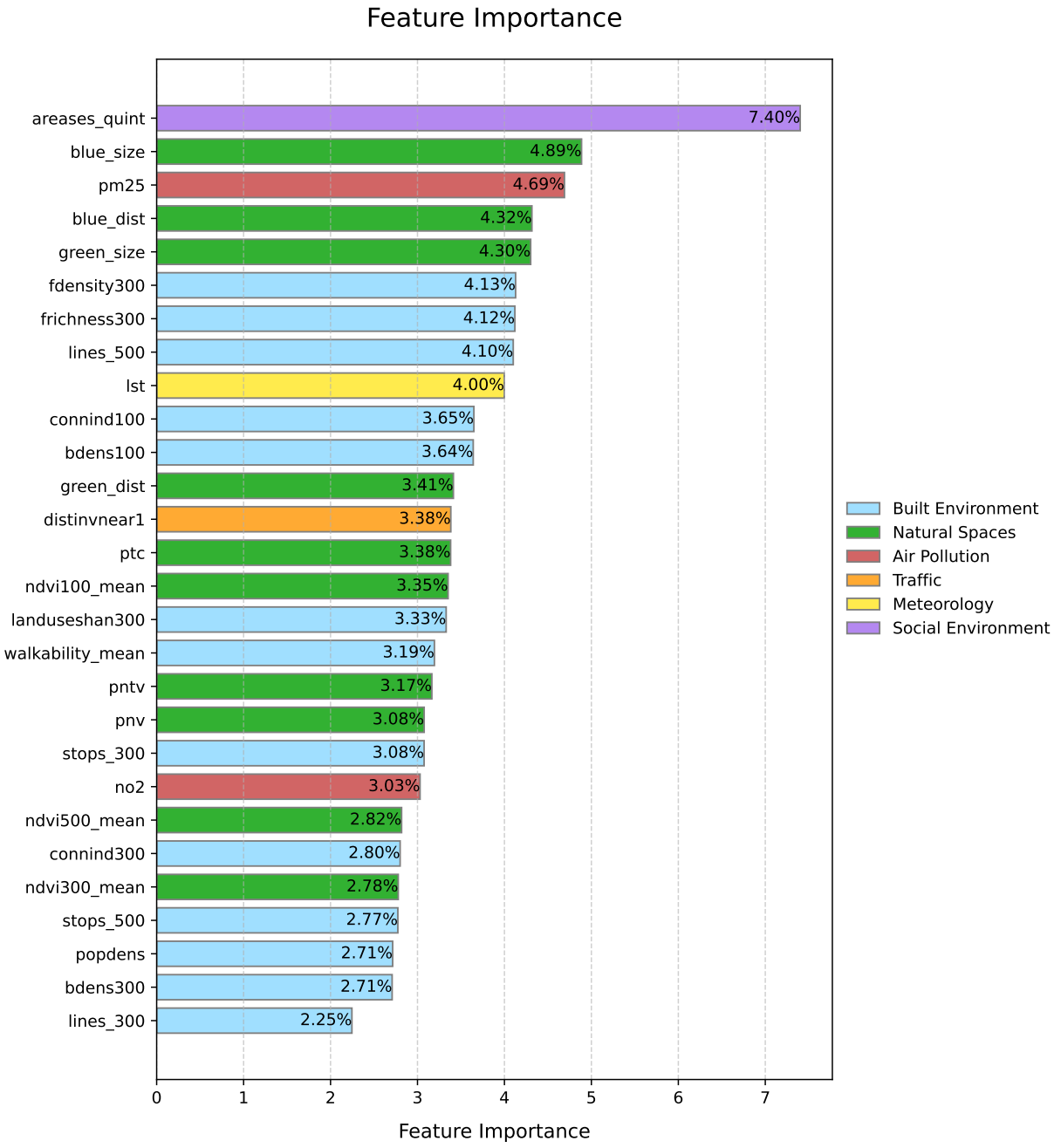


Figure 9.10: Feature importances for the diab dataset, colored by feature type.

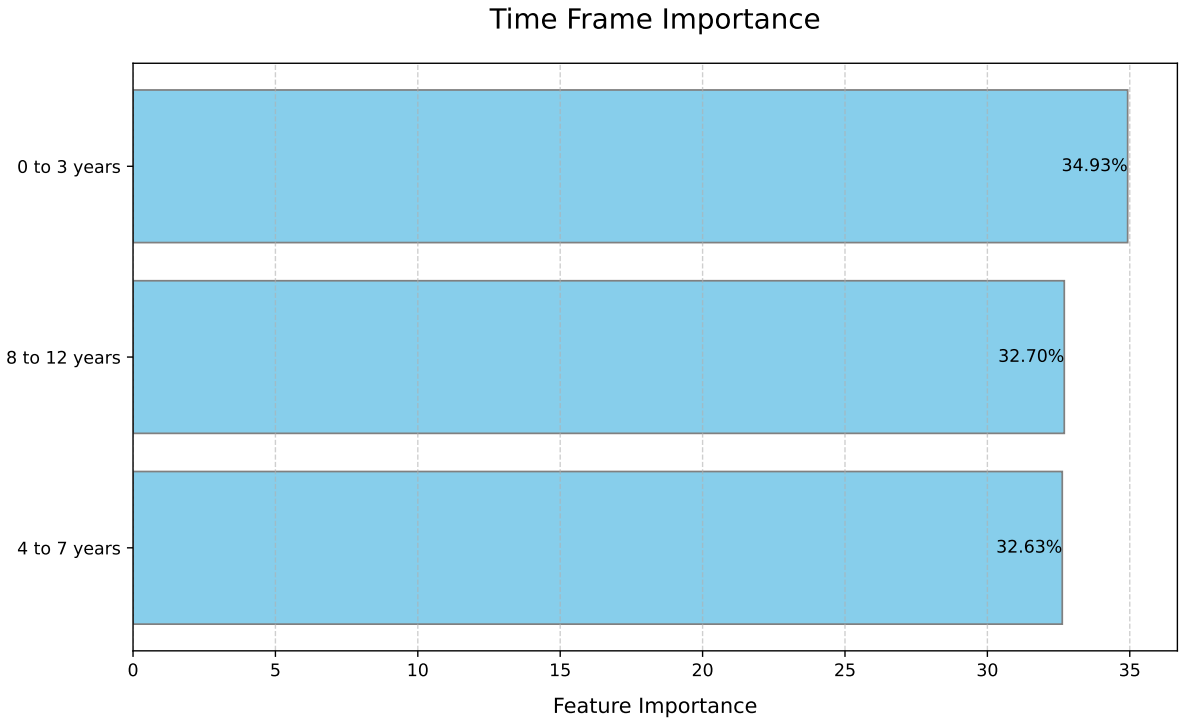


Figure 9.11: Time-frame importances for the diab dataset.

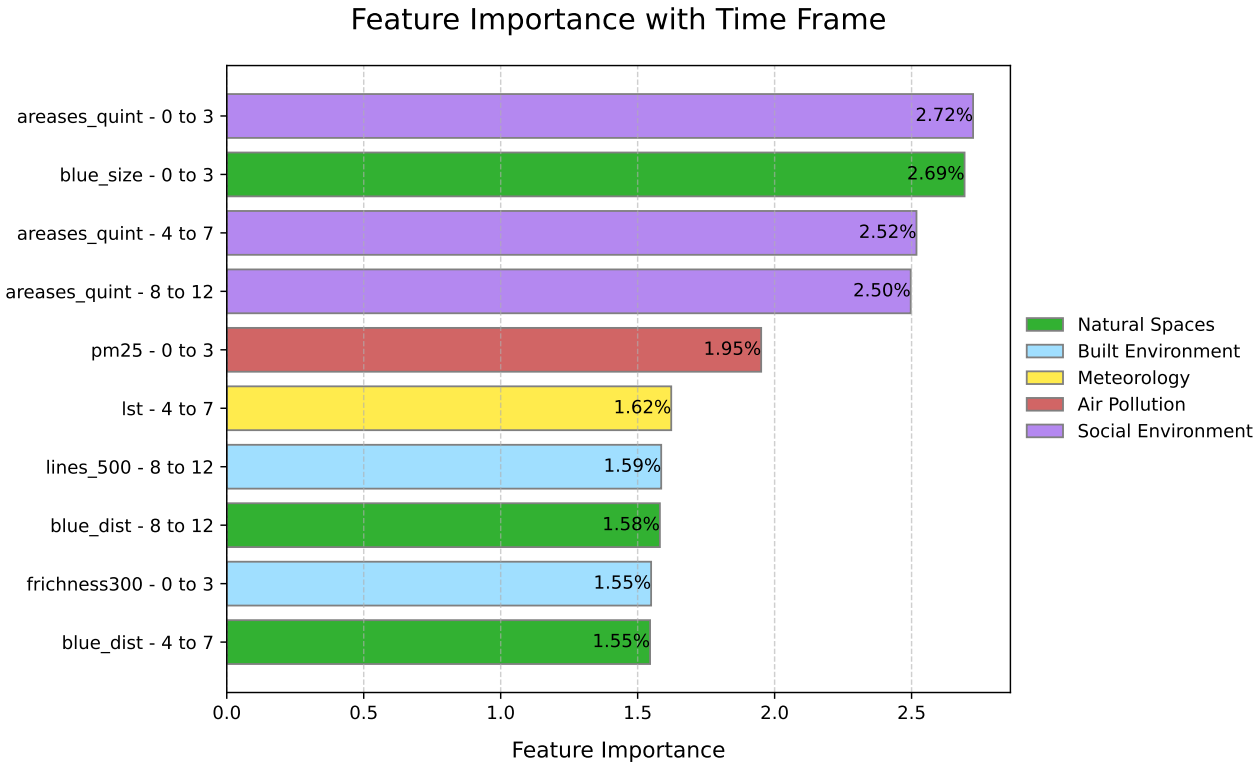


Figure 9.12: Time-frame specific feature importances for the diab dataset.

9.2.3 highbp

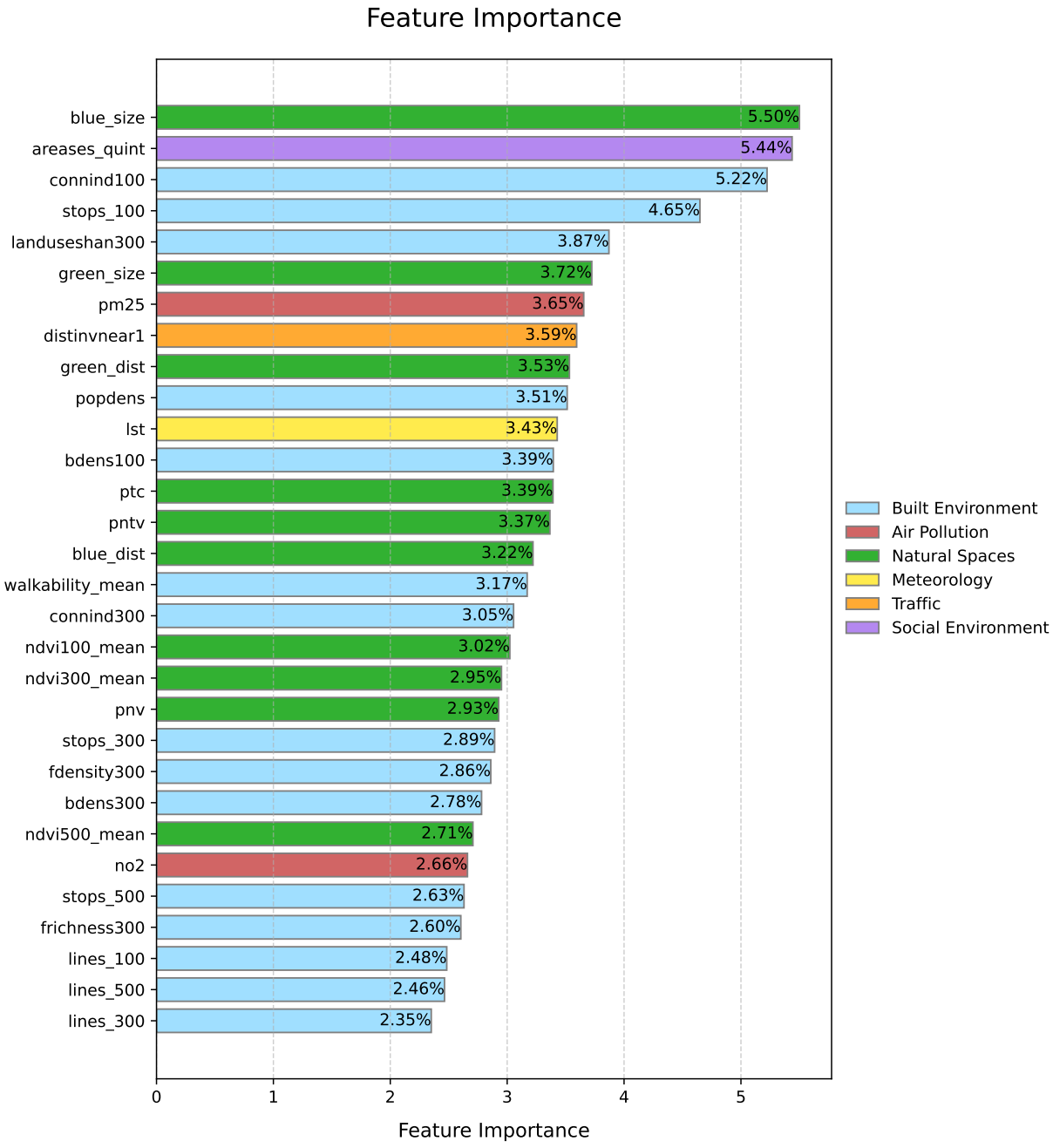


Figure 9.13: Feature importances for the highbp dataset, colored by feature type.

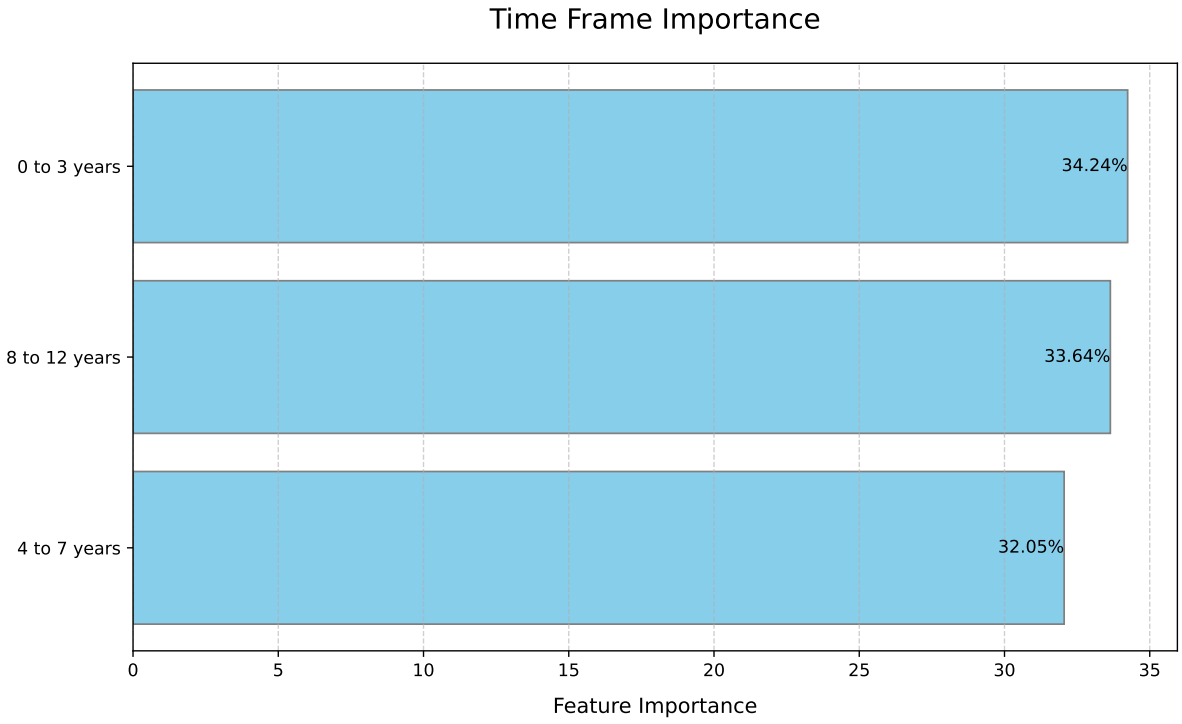


Figure 9.14: Time-frame importances for the highbp dataset.

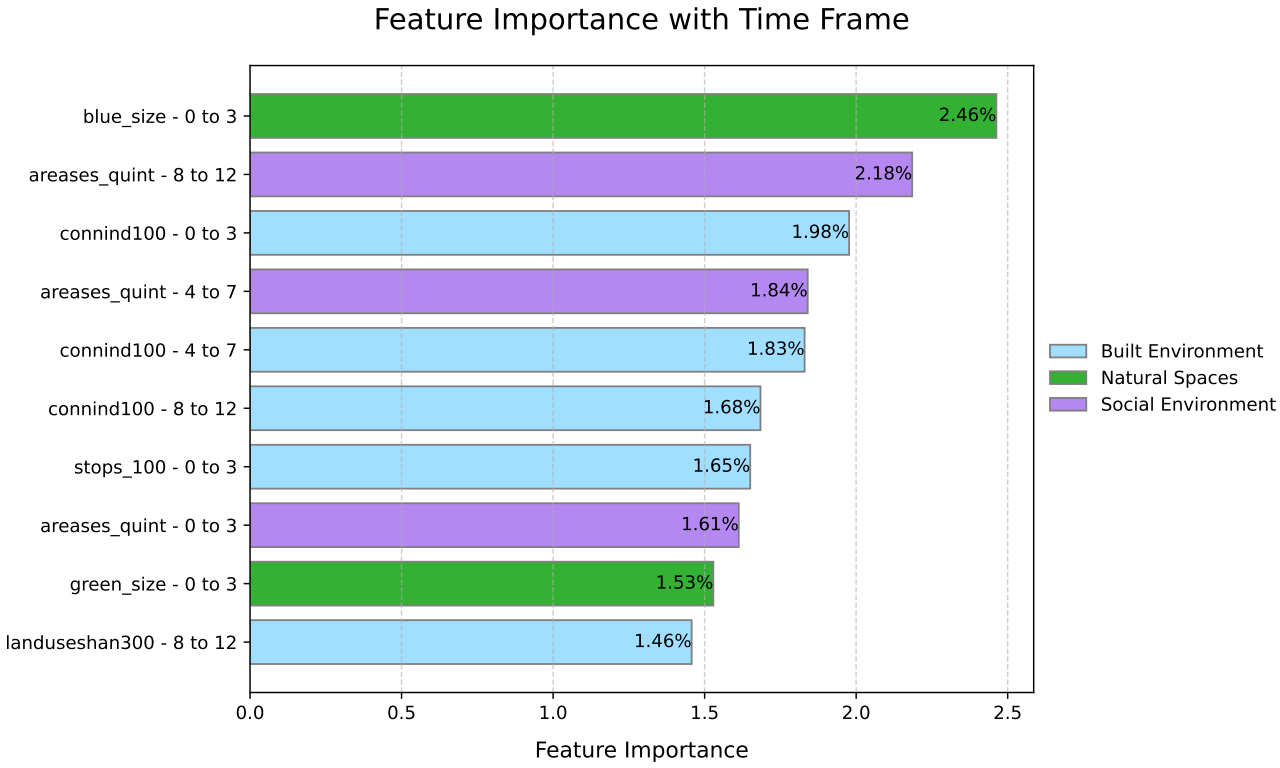


Figure 9.15: Time-frame specific feature importances for the highbp dataset.

9.2.4 overweight

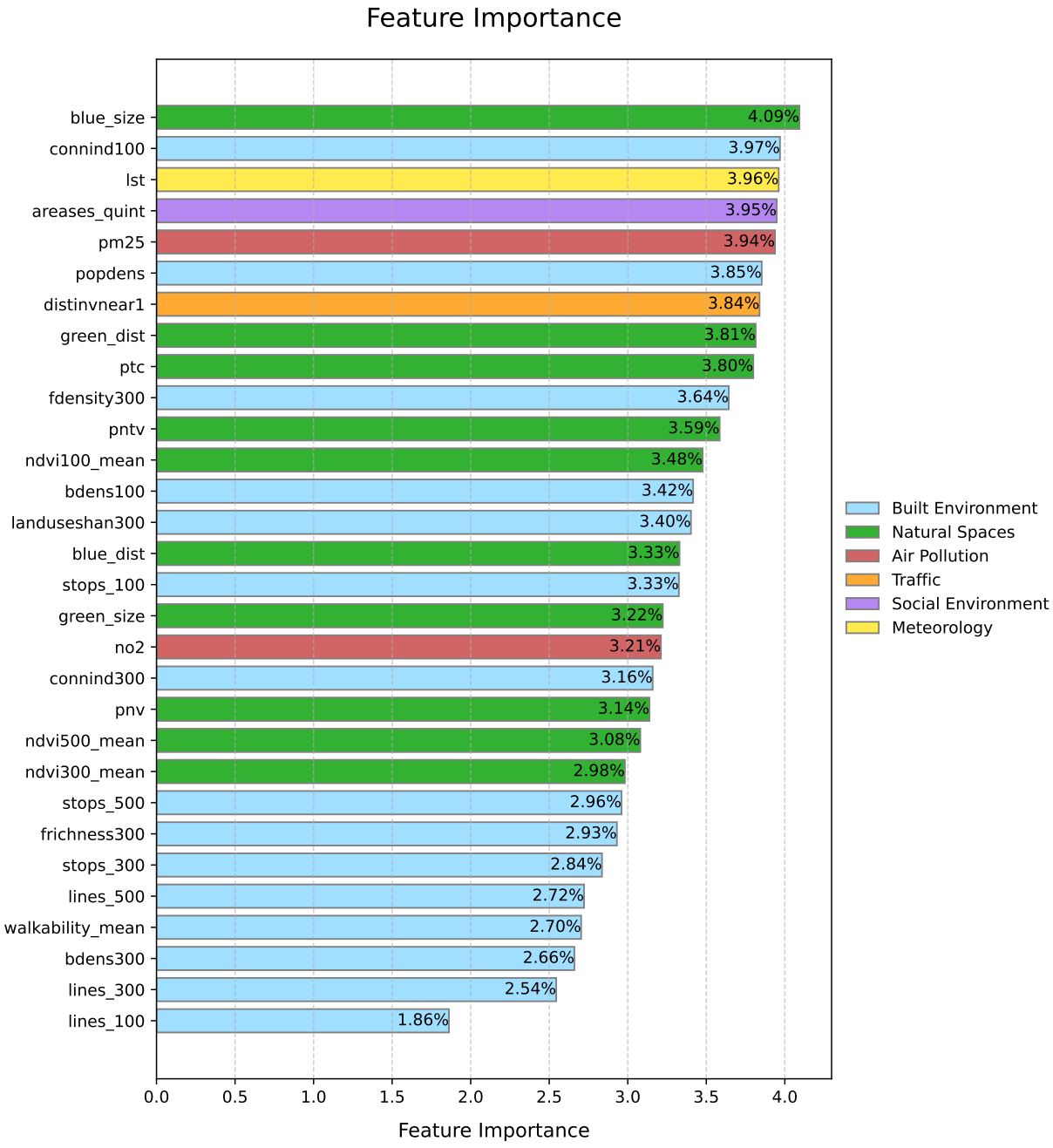


Figure 9.16: Feature importances for the overweight dataset, colored by feature type.

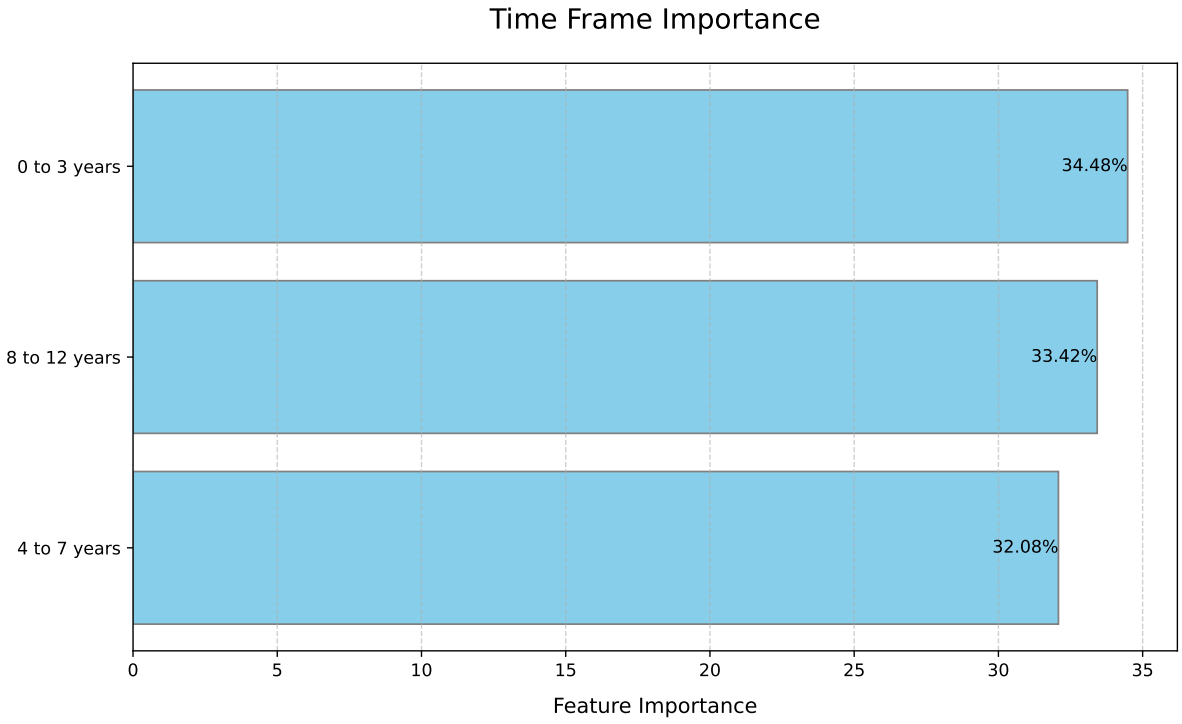


Figure 9.17: Time-frame importances for the overweight dataset.

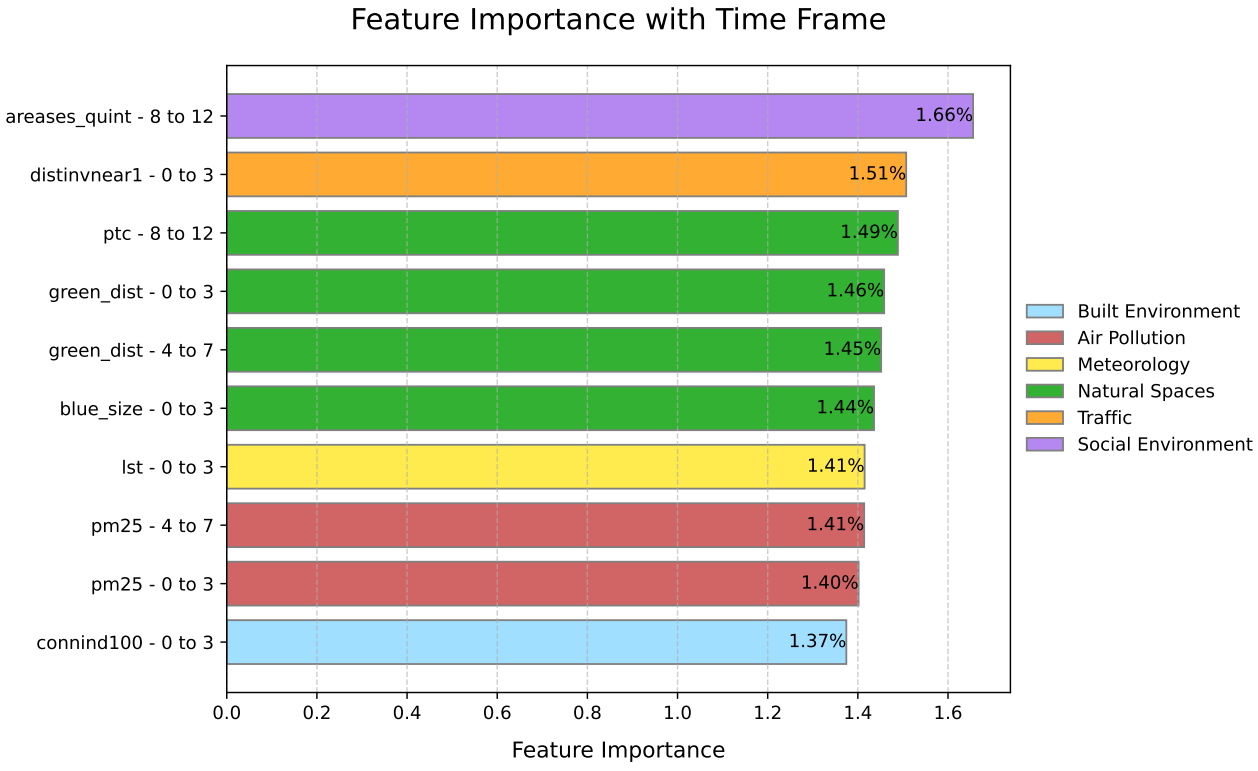


Figure 9.18: Time-frame specific feature importances for the overweight dataset.

9.2.5 perc_dys

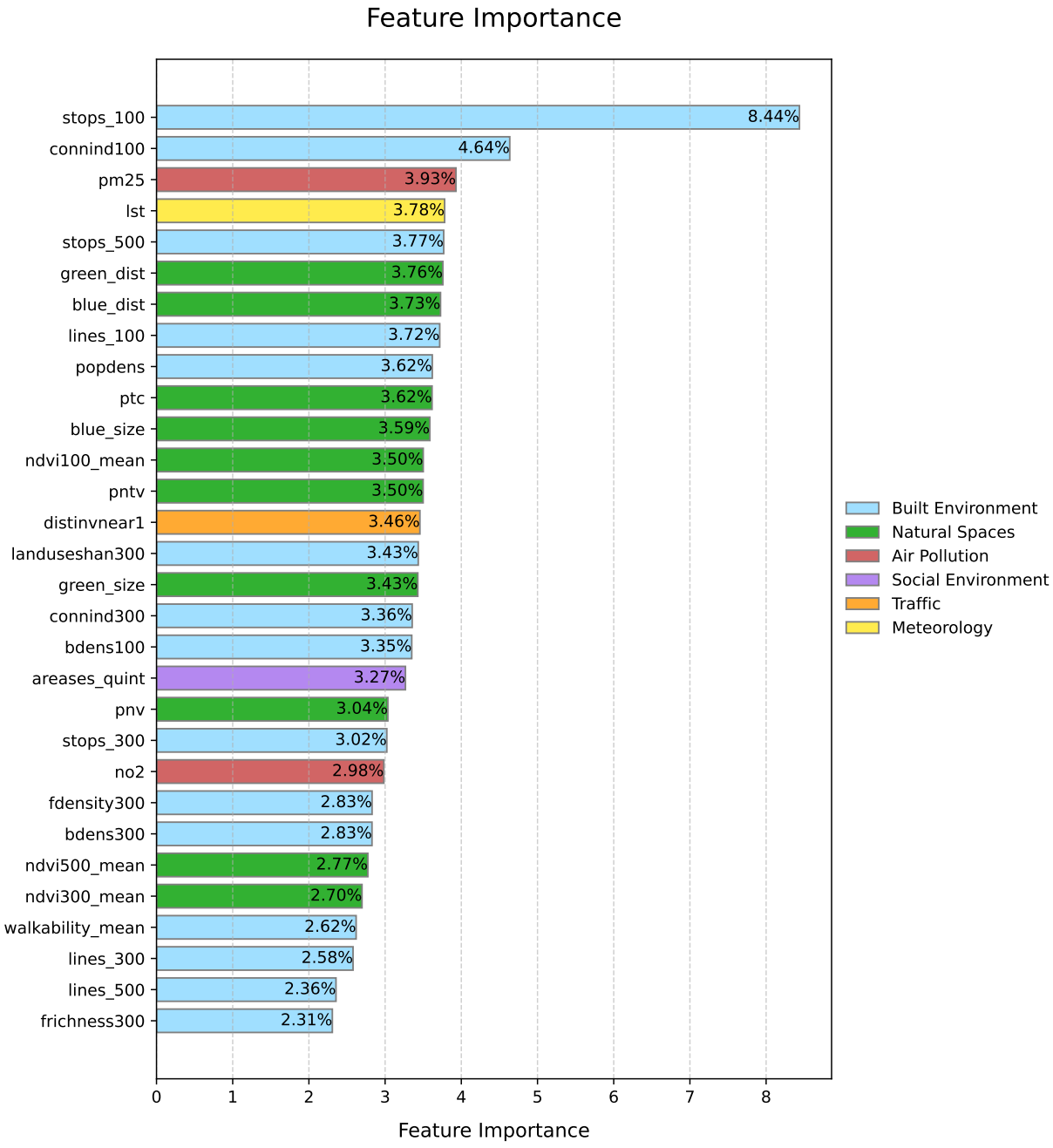


Figure 9.19: Feature importances for the perc_dys dataset, colored by feature type.

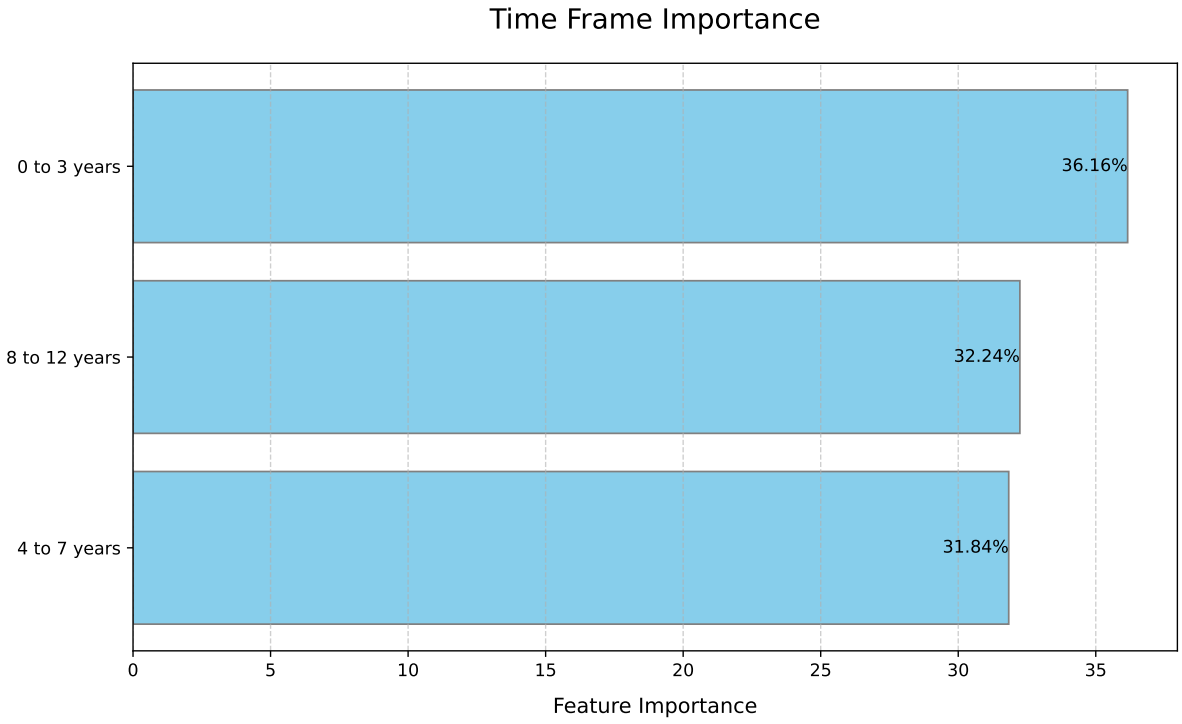


Figure 9.20: Time-frame importances for the perc_dys dataset.

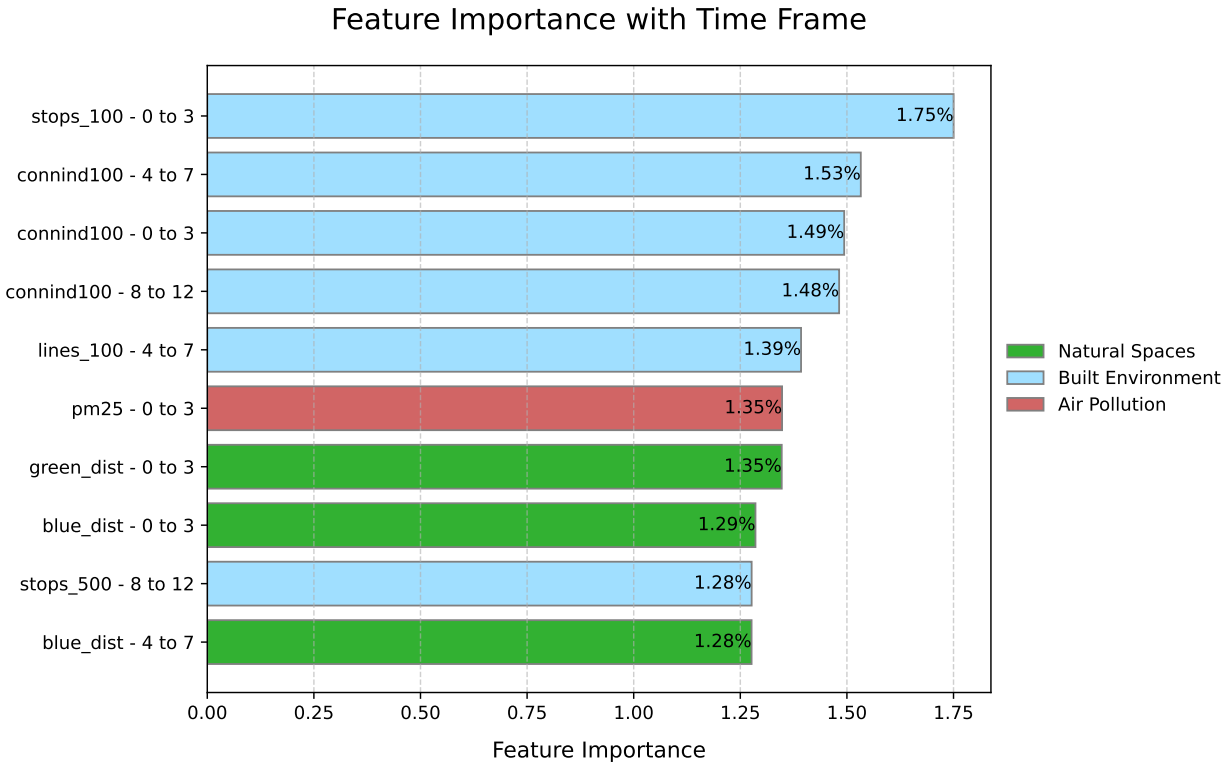


Figure 9.21: Time-frame specific feature importances for the perc_dys dataset.

9.2.6 respir

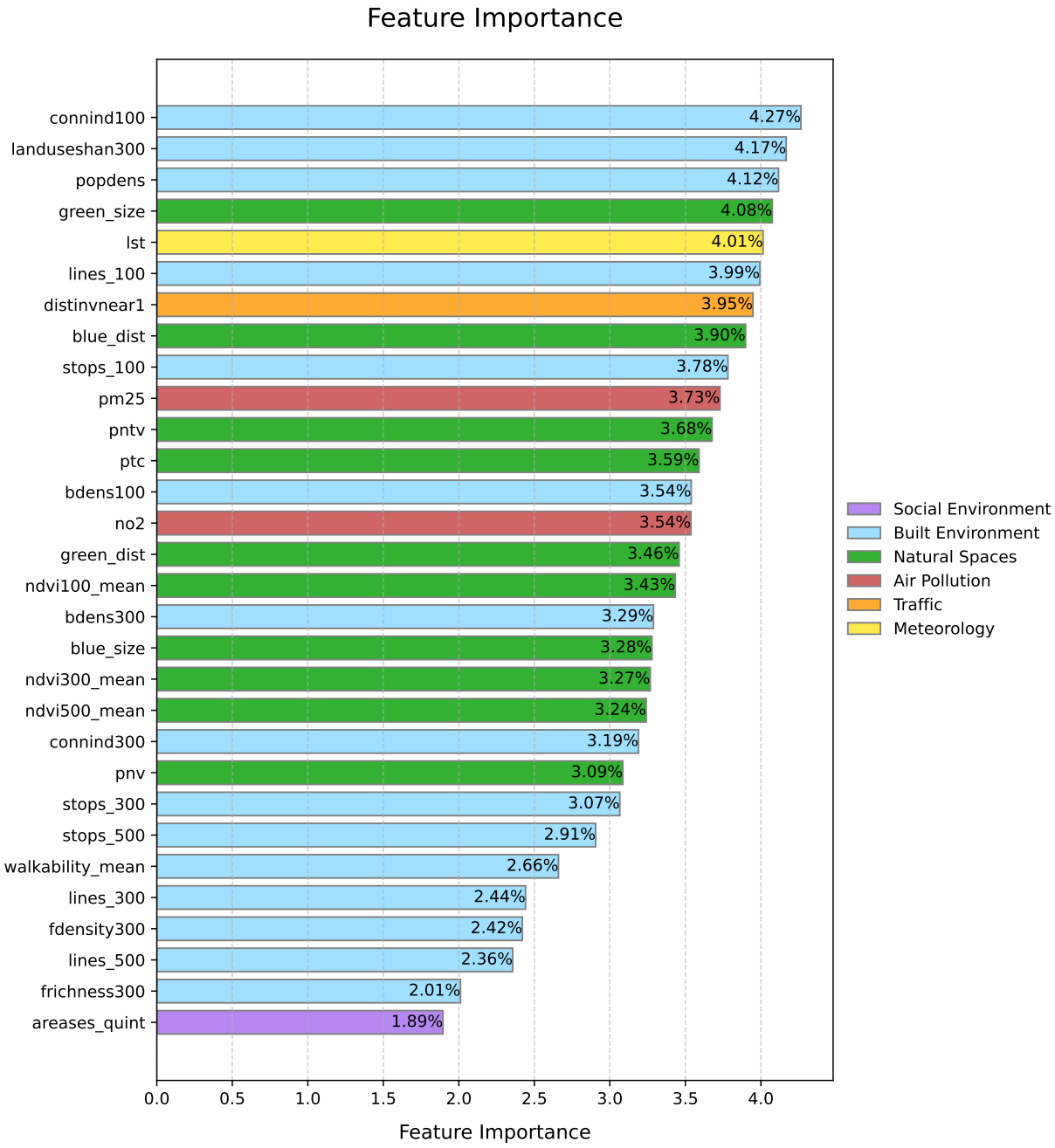


Figure 9.22: Feature importances for the respir dataset, colored by feature type.

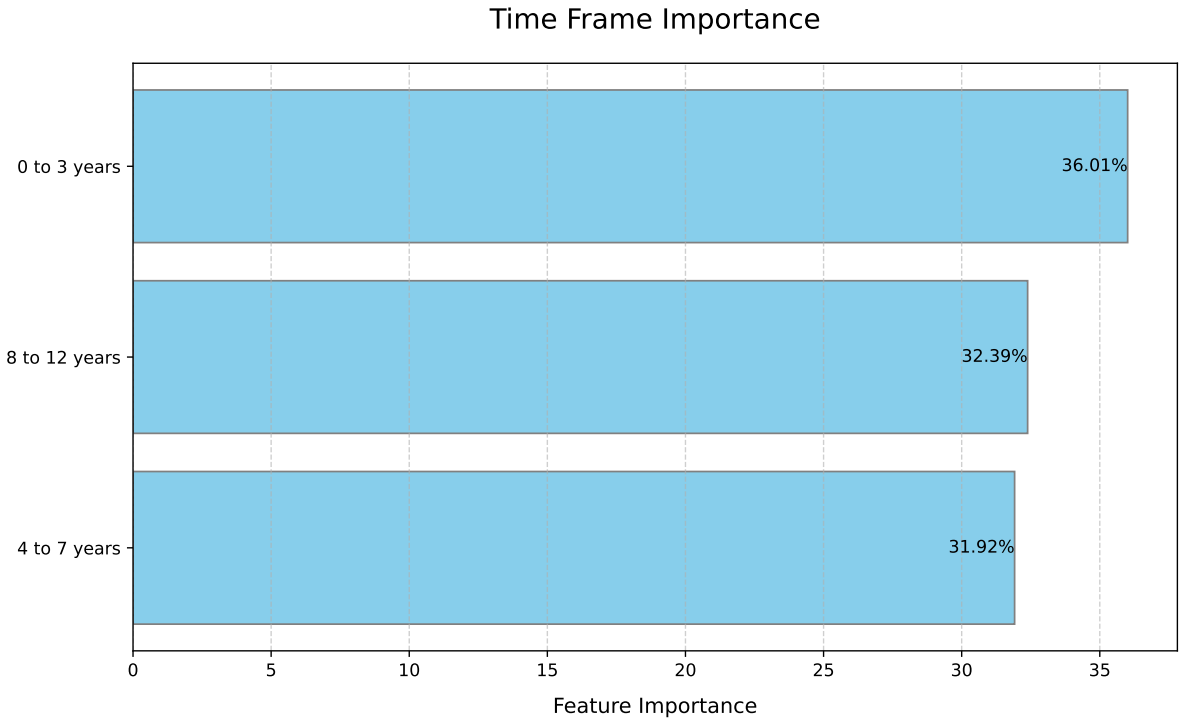


Figure 9.23: Time-frame importances for the respir dataset.

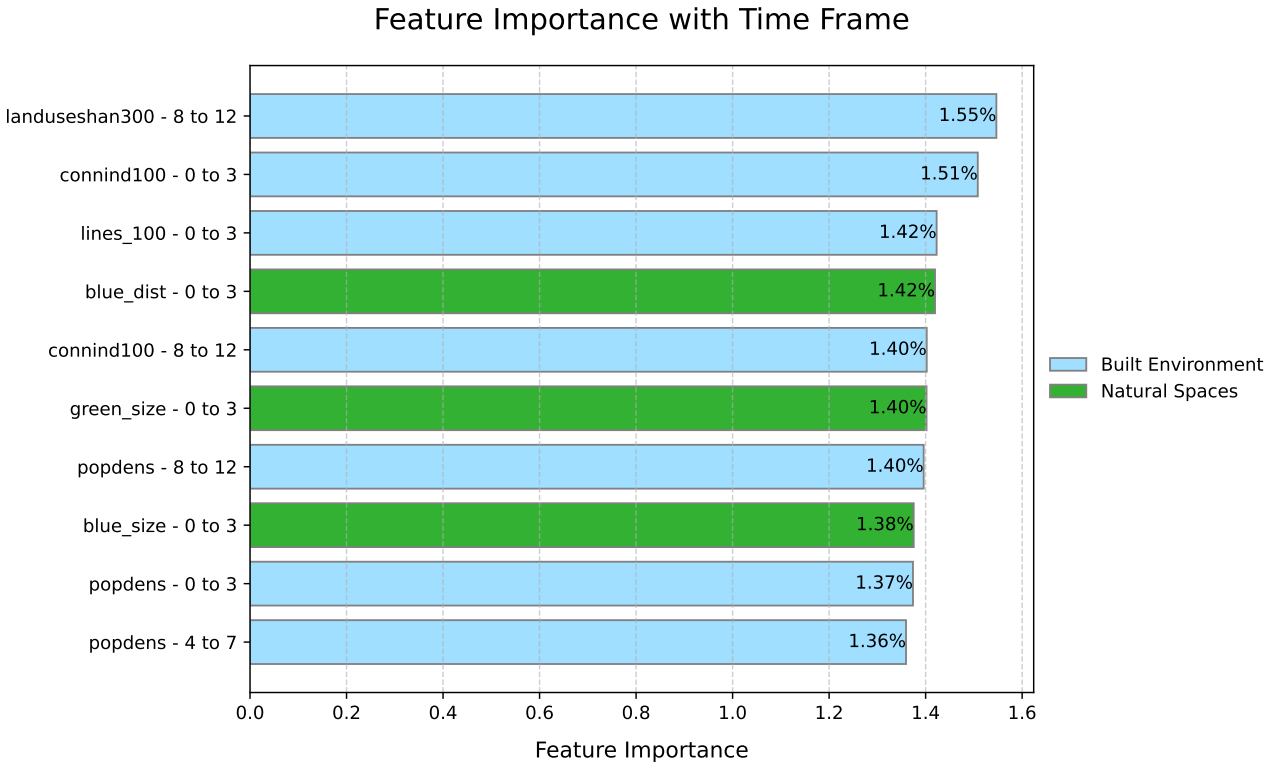


Figure 9.24: Time-frame specific feature importances for the respir dataset.

9.2.7 wisc

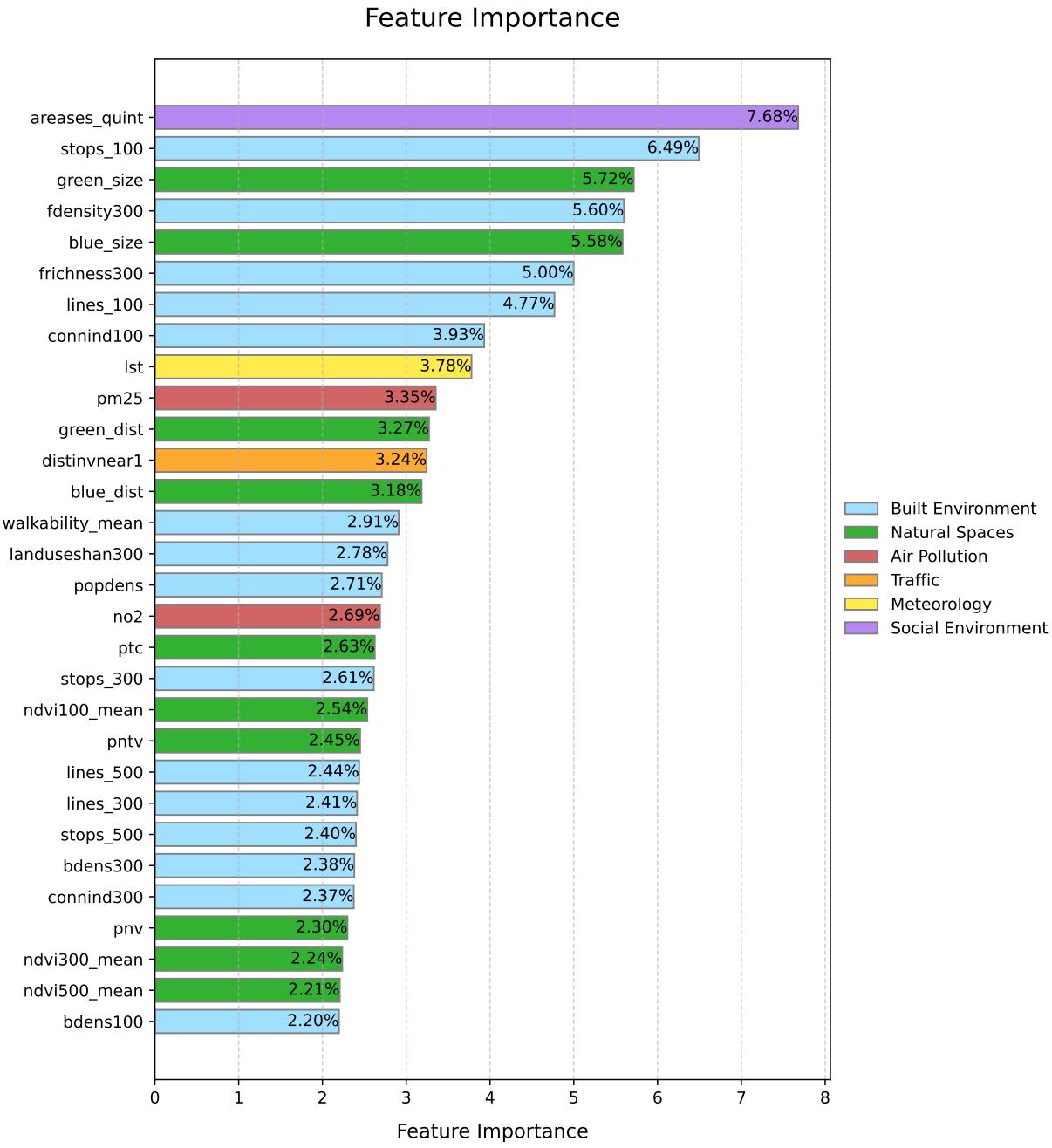


Figure 9.25: Feature importances for the wisc dataset, colored by feature type.

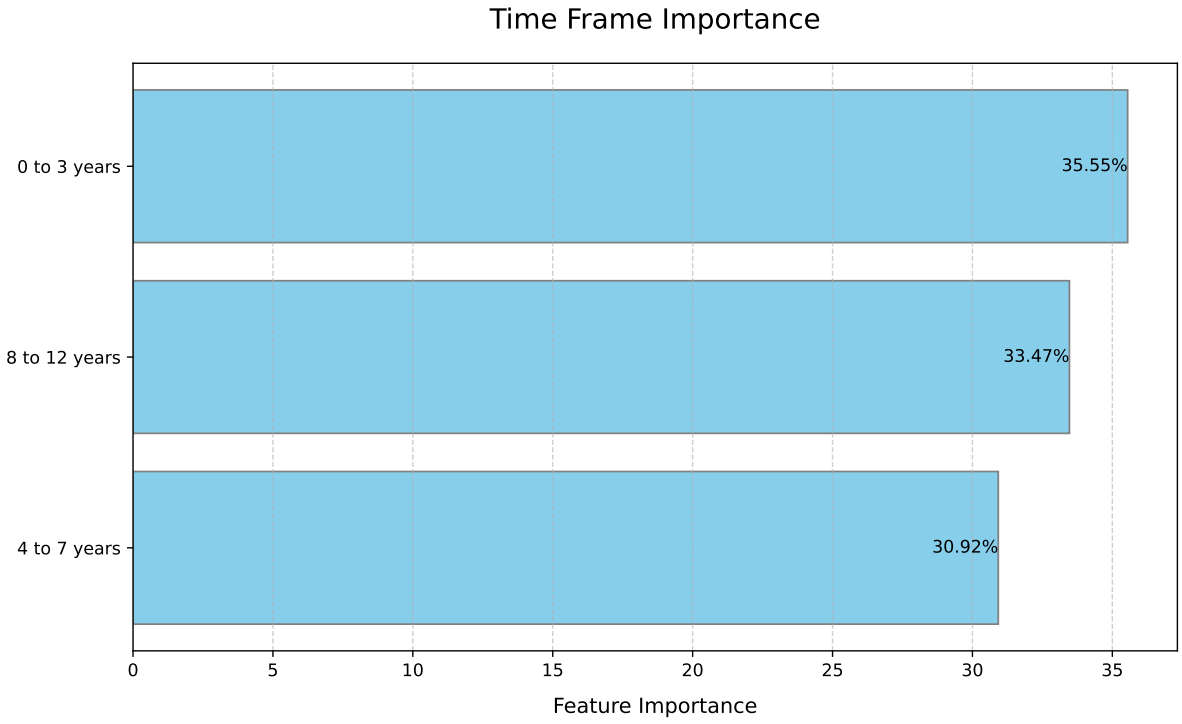


Figure 9.26: Time-frame importances for the wisc dataset.

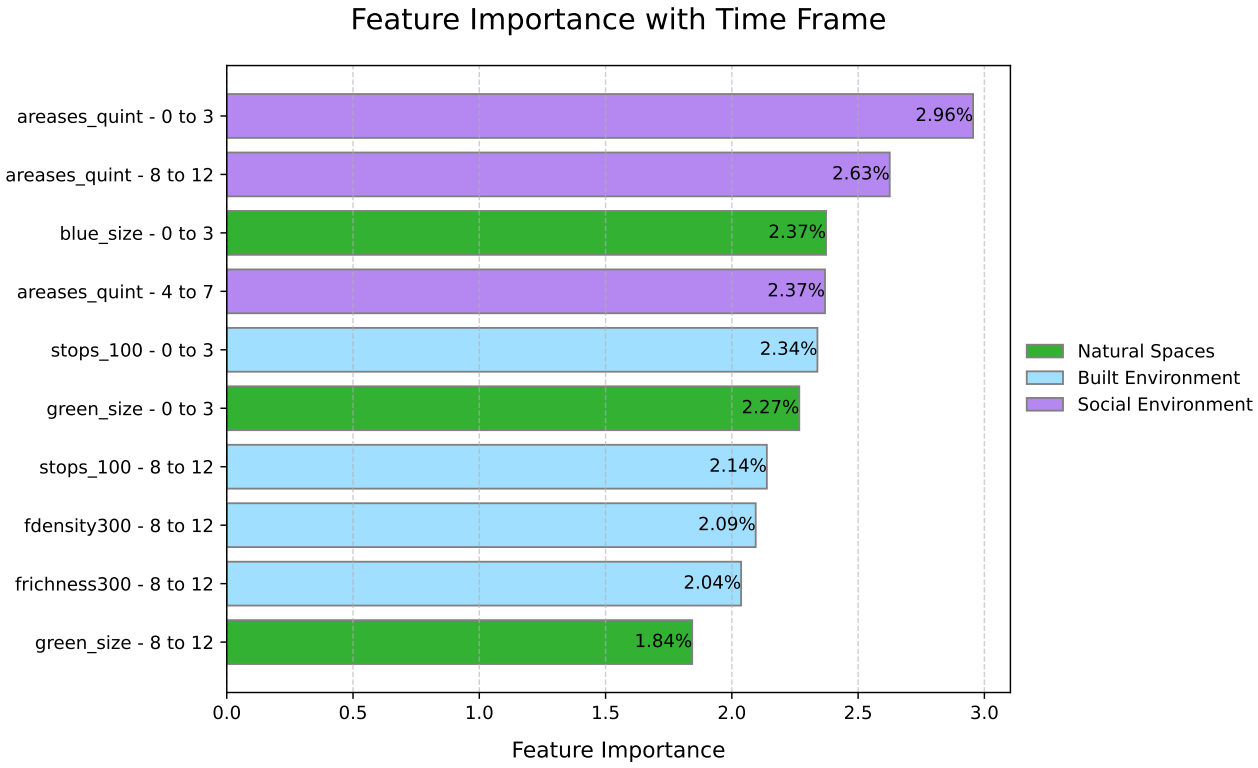


Figure 9.27: Time-frame specific feature importances for the wisc dataset.

9.3 ISPUP specialists commentary

The results were received with enthusiasm by the ISPUP experts, who felt that the results conveyed several interesting observations.

Looking at the feature importance plots, the ISPUP researchers noted the unexpected significance of the *lst* meteorological variable, which measures land surface temperature. Recent studies have linked heat waves with adverse health outcomes, which can explain this result. In contrast, the other meteorological variables were unsurprising in their lack of significance. The experts clarified that their inclusion was primarily aimed at measuring potential impacts on depression.

The importance of the social environment feature was expected, as living in areas with more significant socioeconomic deprivation is associated with a higher risk of disease and subsequent multimorbidity.

Similarly, natural spaces have been linked to health outcomes, so their high importance was also expected. Greater proximity, accessibility, and quality of green spaces (parks, gardens) and blue spaces (seas, rivers) correlate with a reduced risk of adverse outcomes and multimorbidity.

Environmental pollution has also been associated with a higher risk of respiratory and metabolic outcomes, both outcomes of multimorbidity for the scope of this study, so their importance is predictable.

Exposures from the built environment were also relevant predictors. Experts mentioned that it is documented that areas with higher population density, increased traffic, and reduced walkability are associated with a higher risk of disease.

Regarding time frames, the results aligned with what the experts expected. They mentioned that the first few years of life are a period of great plasticity and vulnerability to external factors, so it makes sense that this period has the most impact. The importance of the second highest scoring period, 8-12, must be because it's the closest to the time when the outcomes were measured. In this case, the distance to the least significant time frame is not as high as the first years of life.

Chapter 10

Conclusions

Several key conclusions have been drawn after analyzing the results and listening to the feedback from the ISPUP experts.

For the multimorbidity problem, the features *lst*, *blue_size*, and *areases_quint* were identified as the top three contributors, each accounting for approximately 4% of the importance for both multimorbidity outcomes (*multiSN* and *multi3cat*). Among the ten meteorological features in the raw dataset, *lst* was the only one consistently used across all nine analyses performed for each label.

The socioeconomic level of patients, indicated by the feature *areases_quint*, was a critical multimorbidity influencer, highlighting the impact of financial conditions on adverse health outcomes. Features related to *Natural Spaces* frequently appeared in the upper half of the importance rankings. *Traffic* and *Air Pollution* also showed substantial importance, generally appearing around the upper half of the plots. In the *Built Environment* category, features such as *popdens*, *connind100*, and *landuseshan300* were consistently ranked at the top. In contrast, other features from this category failed to achieve the same importance levels.

Examining specific health outcomes revealed similar observations to those in multimorbidity analyses. However, the outcomes *Perc_dys* and *respir* deviated a bit, as none of the top three features from the multimorbidity analyses appeared in their respective top three features.

In terms of underperforming features, *lines*, *frichness*, *bdens*, and *walkability_mean* were frequently observed in the lower part of the feature importance plots.

Regarding time frames, the exposure window from birth until three years old was identified as the most critical period across all analyses. While the 8 to 12-year window was also always the second most important time frame in all analyses, the difference in importance compared to the 4 to 7-year time frame was smaller. The higher importance of the 8 to 12-year window may be attributed to its proximity to the time of outcome measurement.

The observations associated with these conclusions will require time for the ISPUP experts to digest and plan the subsequent steps for the study.

10.1 Reflection on the investigative work

The first challenge faced was the coordination with ISPUP experts. The conflicting working hours between all members involved made scheduling meetings with all relevant stakeholders difficult, which led to unexpected delays. This limited our ability to improve and expand feature engineering and limited the opportunities to explore and refine investigative pathways due to a lack of time.

However, the most significant challenge was the lack of necessary data points to achieve a satisfactory classification model performance for multimorbidity problems. This limited the confidence with which we could draw conclusions from the results. Despite efforts to improve and generate high-quality datasets and high-performing models, the raw data provided was, in most cases, insufficient to address the problems at hand.

After discussing with the ISPUP experts, and for the reasons already highlighted in this thesis, we felt it was important to highlight that the limited predictive ability can also attributed to the exclusive consideration of exposures from the urban environment. There are many other variables, such as lifestyle choices, and clinical conditions, that are associated with the appearance of multimorbidity. Incorporating these factors would undoubtedly improve the predictive capacity of the data mining models, but due to the scope of this research, those were intentionally left out.

The performance of the data mining models reflected these difficulties. Most models we experimented with performed just slightly better than a random guesser, which was not the desired behavior. Despite extensive experimentation with different modeling techniques, we struggled to improve their performance, as these ended up bottlenecked by the lack of sufficient information in the data.

Despite this, the research yielded some interesting observations about the effects of different exposures on the appearance of multimorbidity, as well as their contribution throughout the different time frames established. These findings were received with enthusiasm by the ISPUP experts, who felt that this was a good step forward in discovering what environmental exposures are behind the appearance of these adverse health conditions.

Experimenting with various data preparation and modeling techniques was also a valuable and considerable part of this work. Although most of them did not yield significant improvements to the classifier's performance, it was a way to demonstrate that the limiting factor of this work was the lack of quality data. Additionally, the lack of enough collaborative feature engineering with the ISPUP experts also left opportunities for improvement unexplored. While we experimented with many data preparation and modeling techniques, the lack of extensive domain-specific feature engineering may have limited our ability to extract better classifier performance from the datasets. Health specialists possess this knowledge which can improve the data by identifying significant patterns that can help the models generalize the typical fragmented information found in health-related datasets.

Our limited domain knowledge of diseases or environmental exposures limited my ability to contribute to domain-specific improvements. Understanding the complexities of diseases and environmental factors is needed to develop meaningful insights.

The problem also handled real-life data, which is inherently complex to handle, which increased the complexity of generating and improving the data mining models.

10.2 Future improvements

For future research, we will prioritize collaboration between the two parts to refine feature engineering strategies. This collaborative approach will likely lead to better-performing models and, consequently, better conclusions as to what drives the appearance of these adverse health outcomes.

To do so, it would be beneficial to gather a bigger collection of individuals with as many different outcome profiles of multimorbidity as possible. Doing this would give data mining models enough information to learn and identify the complete set of patterns that constitute this problem.

Another improvement would be the establishment of both environmental and outcome taxonomies. As mentioned before, health-related data is often partitioned into very specific cases, resulting in unnecessary added complexity for the models to learn. By using domain-specific knowledge, we could create taxonomies that would aggregate exposures or outcomes frequently seen together in the medical context, generalizing the information being fed to the algorithms and facilitating their decision-making process.

Most of the investigative efforts conducted within the context of this thesis were dedicated to exploration and experimentation work to the detriment of refinement and improvement of data transformation strategies and data mining models that were producing promising results.

During the final stages of the investigation, an oversampling technique was identified that demonstrated potential for improving model performance in regard to the multimorbidity problems. However, this technique requires further development and optimization. Future work should focus on refining this oversampling method to ensure it effectively balances the dataset without introducing bias or overfitting. Additionally, subsequent model tuning is necessary to improve the reliability and the performance metrics.

Together with the ISPUP experts, we will examine, discuss, and assemble the complete set of subsequent steps to take moving forward, after thoroughly analyzing the results.

References

[1] Gustavo Batista and Maria-Carolina Monard. A study of k-nearest neighbour as an imputation method. volume 30, pages 251–260, 01 2002.

[2] Andrew L. Beam and Isaac S. Kohane. Big Data and Machine Learning in Health Care. *JAMA*, 319(13):1317–1318, 04 2018.

[3] Daniel Berrar. Cross-validation. In Shoba Ranganathan, Michael Gribskov, Kenta Nakai, and Christian Schönbach, editors, *Encyclopedia of Bioinformatics and Computational Biology*, pages 542–545. Academic Press, Oxford, 2019.

[4] Leo Breiman. Random forests. *Machine Learning*, 45(1), October 2001.

[5] Leo Breiman, Jerome Friedman, Charles J. Stone, and R.A. Olshen. *Classification and Regression Trees*. Chapman and Hall/CRC, 1984.

[6] Peter Bühlmann. Bagging, boosting and ensemble methods. *Handbook of Computational Statistics*, 01 2012.

[7] Nitesh V. Chawla, Kevin W. Bowyer, Lawrence O. Hall, and W. Philip Kegelmeyer. Smote: synthetic minority over-sampling technique. *J. Artif. Int. Res.*, 16(1), jun 2002.

[8] Angela Hsiang-Ling Chen and Sebastian Gunawan. Enhancing retail transactions: A data-driven recommendation using modified rfm analysis and association rules mining. *Applied Sciences*, 13(18), 2023.

[9] Tianqi Chen and Carlos Guestrin. Xgboost: A scalable tree boosting system. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 785–794, New York, NY, USA, 2016. ACM.

[10] Frans Coenen. Data mining: Past, present and future. *Knowledge Eng. Review*, 26:25–29, 03 2011.

[11] Corinna Cortes and Vladimir Vapnik. Support-vector networks. *Machine learning*, 20(3):273–297, 1995.

[12] John W. Graham. Missing data analysis: Making it work in the real world. *Annual Review of Psychology*, 60(Volume 60, 2009):549–576, 2009.

[13] Vinutha H P, B. Poornima, and B. Sagar. *Detection of Outliers Using Interquartile Range Technique from Intrusion Dataset*, pages 511–518. 01 2018.

[14] Jack Hogan and Niall M. Adams. On averaging ROC curves. *Transactions on Machine Learning Research*, 2023. Survey Certification.

[15] Mohammad Hossin and Sulaiman M.N. A review on evaluation metrics for data classification evaluations. *International Journal of Data Mining & Knowledge Management Process*, 5:01–11, 03 2015.

[16] Micheline J Haniawei, Kamber and Jian Pei. *Data mining concepts and techniques*. Morgan Kaufmann, Third edition, 2012.

[17] Tomás Horvath João Moreira, André Carvalho. *A General Introduction to Data Analytics*. John Wiley & Sons, Inc, First edition, 2018.

[18] Christoph Kern, Thomas Klausch, and Frauke Kreuter. Tree-based machine learning methods for survey research. *Survey Research Methods*, 13(1):73–93, Apr. 2019.

[19] H. H. König and S. Riedel-Heller. Prävention aus dem blickwinkel der gesundheitsökonomie [economic aspects of prevention]. *Internist (Berl)*, 49(2):146–153, February 2008.

[20] Muthukrishnan R and R Rohini. Lasso: A feature selection technique in predictive modeling for machine learning. pages 18–20, 10 2016.

[21] P. Ravisankar, V. Ravi, G. Raghava Rao, and I. Bose. Detection of financial statement fraud and feature selection using data mining techniques. *Decision Support Systems*, 50(2):491–500, 2011.

[22] Debora Rizzuto, René J. F. Melis, Sara Angleman, Chengxuan Qiu, and Alessandra Marengoni. Effect of chronic diseases and multimorbidity on survival and functioning in elderly adults. *Journal of the American Geriatrics Society*, 65(5):1056–1060, 2017.

[23] Robert E. Schapire. *Explaining AdaBoost*, pages 37–52. Springer Berlin Heidelberg, Berlin, Heidelberg, 2013.

[24] Dalwinder Singh and Birmohan Singh. Investigating the impact of data normalization on classification performance. *Applied Soft Computing*, 97:105524, 2020.

[25] K. Suresh, S. V. Thomas, and G. Suresh. Design, data analysis and sampling techniques for clinical research. *Annals of Indian Academy of Neurology*, 14(4):287–290, 2011.

[26] R. Wirth and Jochen Hipp. Crisp-dm: Towards a standard process model for data mining. *Proceedings of the 4th International Conference on the Practical Applications of Knowledge Discovery and Data Mining*, 01 2000.

Appendix A

Appendix

A.1 Environmental features

Here, we detail a complete list of all environmental exposure names, respective categories and descriptions, and the number of years with recorded measurements.

Table A.1: Environmental variables discrimination

Variable name	**Category**	**Description**	**Years measured**
stops_100	Built environment	Density of public transport stops within a buffer of 100 m	12 years
stops_300	Built environment	Density of public transport stops within a buffer of 300 m	12 years
stops_500	Built environment	Density of public transport stops within a buffer of 500 m	12 years
lines_100	Built environment	Length of public transport stops within a buffer of 100 m	12 years
lines_300	Built environment	Length of public transport stops within a buffer of 300 m	12 years
lines_500	Built environment	Length of public transport stops within a buffer of 500 m	12 years

bdens100	Built environment	Building density within a buffer of 100 m	13 years
bdens300	Built environment	Building density within a buffer of 300 m	13 years
hdres	Built environment	Percentage of hdres (continuous urban fabric) land use within a buffer of 300 m	13 years
ldres	Built environment	Percentage of ldres (discontinuous dense/medium density/low density urban fabric) land use within a buffer of 300 m	13 years
vldres	Built environment	Percentage of vldres (discontinuous very low density urban fabric) land use within a buffer of 300 m	13 years
indtr	Built environment	Percentage of indtr (industrial, commercial, public, military and private units) land use within a buffer of 300 m	13 years
trans	Built environment	Percentage of trans (road and rail network and associated land, fast transit roads and associated land, other roads and associated land, railways and associated land) land use within a buffer of 300 m	13 years
port	Built environment	Percentage of port (port areas) land use within a buffer of 300 m	13 years
airpt	Built environment	Percentage of airpt (airports) land use within a buffer of 300 m	13 years

other	Built environment	Percentage of other (mineral extraction and dump sites, construction sites, land without current use) land use within a buffer of 300 m	13 years
urbgr	Built environment	Percentage of urbgr (green urban areas, sports and leisure facilities) land use within a buffer of 300 m	13 years
agrgr	Built environment	Percentage of agrgr (agricultural areas, semi-natural areas and wetlands) land use within a buffer of 300 m	13 years
natgr	Built environment	Percentage of natgr (forests) land use within a buffer of 300 m	13 years
water	Built environment	Percentage of water land use within a buffer of 300 m	13 years
landuseshan300	Built environment	Description	13 years
connind100	Built environment	Connectivity density within a buffer of 100 m	13 years
connind300	Built environment	Connectivity density within a buffer of 300 m	13 years
walkability_mean	Built environment	Walkability index (as mean of deciles of facility richness index, landuse shannon's Evenness Index, population density, connectivity density) within a buffer of 300 m	13 years
frichness300	Built environment	Number of different facility types present divided by the maximum potential number of facility types within a buffer of 300 m	13 years
fdensity300	Built environment	Number of facilities present within a buffer of 300 m at birth	13 years

popdens	Built environment	Population density	13 years
green_dist	Natural Spaces	Straight line distance to nearest green space > 5,000 m2	13 years
blue_dist	Natural Spaces	Straight line distance to nearest blue space > 5,000 m2	13 years
green_size	Natural Spaces	Area of closest green space > 5,000m2	13 years
blue_size	Natural Spaces	Area of closest blue space > 5,000m2	13 years
green_yn300	Natural Spaces	Is there a green space > 5,000 m2 in a distance of 300m	13 years
blue_yn300	Natural Spaces	Is there a blue space > 5,000 m2 in a distance of 300m	13 years
ndvi100_mean	Natural Spaces	NDVI within a 100 meter buffer around geocode using all available images during the greenest period	13 years
ndvi300_mean	Natural Spaces	NDVI within a 300 meter buffer around geocode using all available images during the greenest period	13 years
ndvi500_mean	Natural Spaces	NDVI within a 500 meter buffer around geocode using all available images during the greenest period	13 years
pntv	Natural Spaces	Average of percent of nontree vegetation within a buffer of 300 m	13 years
pnv	Natural Spaces	Average of percent of non-vegetated within a buffer of 300 m	13 years
ptc	Natural Spaces	Average of percent of tree covered within a buffer of 300 m	13 years

hu	Meteorology	Average of relative humidity from E-OBS data during first year	13 years
lst	Meteorology	Land surface temperature	13 years
rr	Meteorology	Average of precipitation sum from E-OBS data	13 years
pp	Meteorology	Average of sea level pressure from E-OBS data	13 years
qq	Meteorology	Average of daily mean global radiation from E-OBS data	13 years
tx	Meteorology	Average of maximum temperature from E-OBS data	13 years
tg	Meteorology	Average of mean temperature from E-OBS data	13 years
tn	Meteorology	Average of minimum temperature from E-OBS	13 years
uvddc	Meteorology	Average of DNA-damage UV dose	13 years
uvdec	Meteorology	Average of erythemal UV dose	13 years
uvdvc	Meteorology	Average of vitamin-d UV dose	13 years
no2	Air pollution	no2 elapse average value (back extrapolated in time using ratio method)	13 years
pm25	Air pollution	pm25 elapse average value (back extrapolated in time using ratio method)	13 years
distinvnear1	Traffic/Noise	Inverse distance to nearest road	13 years
areases_quint	Social Environment	Area-level SES indicator (deprivation index in quintiles) at birth	13 years
areases_tert	Social Environment	Area-level SES indicator (deprivation index in tertiles) at 12 years of age	13 years

A.2 Multimorbidity outcomes

Table A.2: Multimorbidity variables discrimination

Variable name	Description	Coding
respir	Respiratory or allergic health outcomes. Physician-diagnosed asthma OR allergies (excluding allergies to medication) OR rhinitis.	0 or 1
perc_dys	Cardiometabolic outcomes - dyslipidemia. Use of age and sex-specific percentiles: triglycerides, total cholesterol, and low-density lipoprotein (>= 90th percentile) and high-density lipoprotein (<= 10th).	0 or 1
diab	Cardiometabolic outcomes - (pre)diabetes. Use of age and sex-specific percentiles for Glucose and HbA1c (>= 90th percentile).	0 or 1
highbp	Cardiometabolic outcomes - high blood pressure and Systolic and/or diastolic blood pressure >=90th percentile, or reported physician-diagnosed hypertension.	0 or 1
overweight	Cardiometabolic outcomes - overweight/obesity. Overweight/obesity (>1 SD) and Normal weight (reference category, between >= -2 and <= 1SD) were calculated according to the WHO Growth Reference BMI age and sex-adjusted z-scores.	0 or 1
bdi	Neurodevelopmental health outcomes - depressive symptomatology. Depressive symptomatology assessed by the Beck's Depression Inventory (BDI-II); cut-off for depressive symptomatology >= 13 points.	0 or 1
wisc	Neurodevelopmental health outcomes - cognitive impairment. Cognitive impairment is assessed through the Intelligence coefficient, obtained using the Wechsler Intelligence Scale for Children (WISC-III).	0 or 1
multicont	Number of adverse health outcomes in an individual.	0 to 7

multiSN	Presence of multimorbidity (>= 2 outcomes).	0 or 1
multi3cat	Number of adverse health outcomes (0 outcomes, 1 outcome, >= 2 outcomes)	0 to 2

A.3 Environmental categorical variables statistics

Table A.3: Categorical variable statistics

Variable Name	Statistics
blue_yn300_12	- Value 0: 6899 individuals - 93.89% - Value 1: 449 individuals - 6.11%
green_yn300_12	- Value 0: 403 individuals - 5.48% - Value 1: 6945 individuals - 94.52%
areases_tert_12	- Value 1: 2350 individuals - 31.98% - Value 2: 2403 individuals - 32.7% - Value 3: 2595 individuals - 35.32%
areases_quint_12	- Value 1: 1405 individuals - 19.12% - Value 2: 1410 individuals - 19.19% - Value 3: 1444 individuals - 19.65% - Value 4: 1499 individuals - 20.4% - Value 5: 1590 individuals - 21.64%

A.4 Environmental symmetric continuous features statistics

Table A.4: Symmetric continuous variable statistics

Adverse Health Outcome	Mean	Standard Deviation
bdens300_12	452328.24	167868.35
hu_12	77.82	2.08
ldres_12	23.57	11.5
lst_12	22.45	1.92
ndvi100_mean_12	0.21	0.06
ndvi300_mean_12	0.24	0.06
ndvi500_mean_12	0.24	0.05
pntv_12	61.06	10.46
pnv_12	30.28	13.54
rr_12	3.55	0.7
pp_12	1018.81	0.39
qq_12	165.51	7.19
tx_12	22.0	0.61
tg_12	15.29	0.27
tn_12	9.55	0.17
uvddc_12	1.21	0.02
uvdec_12	2.64	0.02
uvdvc_12	4.64	0.05
walkability_mean_12	5.51	1.61
no2_12	28.77	6.01

A.5 Environmental asymmetrical continuous feature statistics

Table A.5: Asymmetric continuous variable statistics

Variable name	Median	25th quartile	75th quartile
stops_100_12	0.0	0.0	32.36
stops_300_12	14.38	7.19	25.17
stops_500_12	14.24	7.77	22.01
lines_100_12	0.0	0.0	16510.21
lines_300_12	7160.35	0.0	18638.57
lines_500_12	7805.14	1808.08	17356.44
bdens100_12	595454.42	446590.19	721676.59
green_dist_12	74.53	34.42	147.13
blue_dist_12	1511.82	851.16	2268.63
green_size_12	22865.67	13739.91	47450.56
blue_size_12	24933.03	14425.32	3992634.51
hdres_12	18.86	7.62	31.38
vldres_12	0.0	0.0	0.51
indtr_12	10.13	5.41	16.85
trans_12	9.32	6.57	12.19
port_12	0.0	0.0	0.0
airpt_12	0.0	0.0	0.0
other_12	2.02	0.88	3.56
urbgr_12	2.96	0.9	6.84
agrgr_12	11.83	4.69	21.83
natgr_12	3.83	0.0	13.46
water_12	0.0	0.0	0.0
distinvnear1_12	0.11	0.07	0.21
landuseshan300_12	0.57	0.52	0.61
connind100_12	226.53	97.09	355.98
connind300_12	190.58	125.85	273.28
ptc_12	7.0	4.8	10.5
pm25_12	14.1	13.7	14.54
frichness300_12	0.03	0.02	0.08
fdensity300_12	10.79	3.6	28.77
popdens_12	6662.96	3079.79	12724.2

A.6 ROC-AUC curves

A.6.1 Random Forest

A.6.1.1 multi3cat dataset

The following two ROC curves are relative to the dataset used to extract the final results (oversampling of rare outcome configurations).

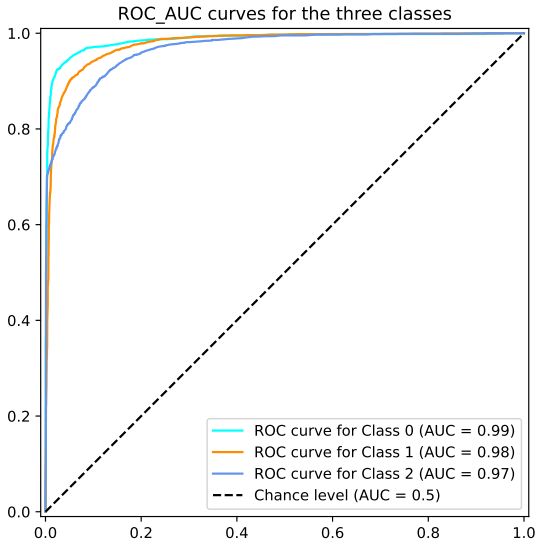


Figure A.1: ROC-AUC curve for the multi3cat dataset.

A.6.1.2 multiSN dataset

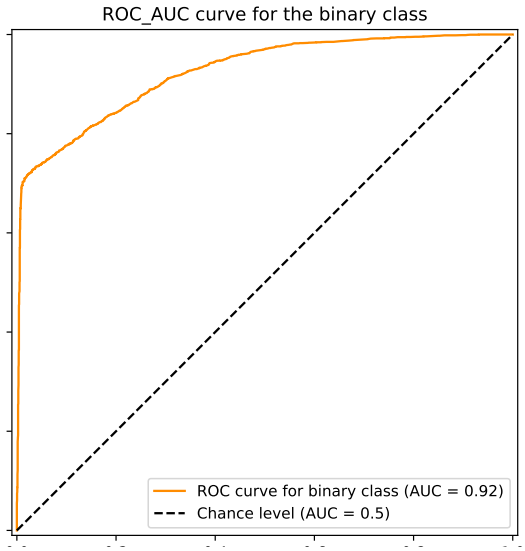


Figure A.2: ROC-AUC curve for the multiSN dataset.

A.6.2 Neural Network

A.6.2.1 multi3cat dataset

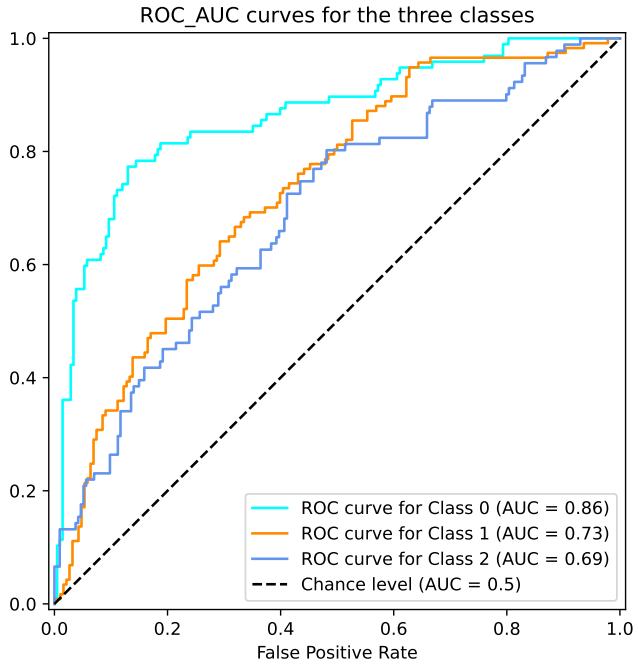


Figure A.3: ROC-AUC curve for the multi3cat dataset.

A.6.2.2 multiSN dataset

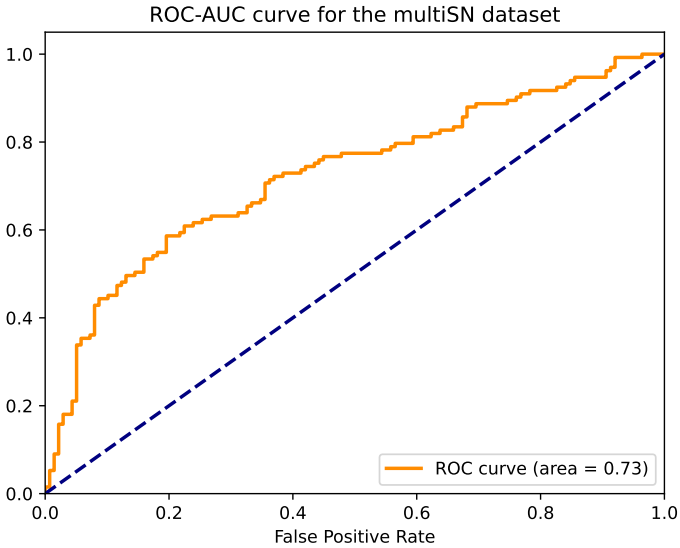


Figure A.4: ROC-AUC curve for the multiSN dataset.

A.6.3 Support vector machines

A.6.3.1 multi3cat dataset

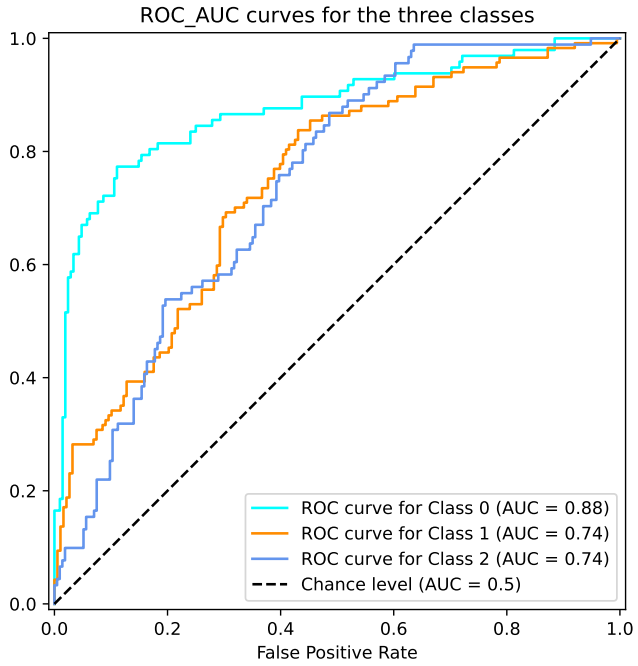


Figure A.5: ROC-AUC curve for the multi3cat dataset.

A.6.3.2 multiSN dataset

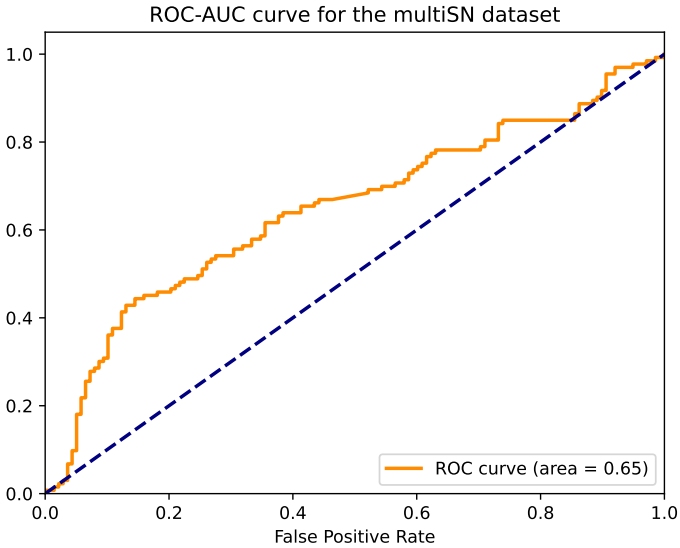


Figure A.6: ROC-AUC curve for the multiSN dataset.

A.6.4 CART

A.6.4.1 multi3cat dataset

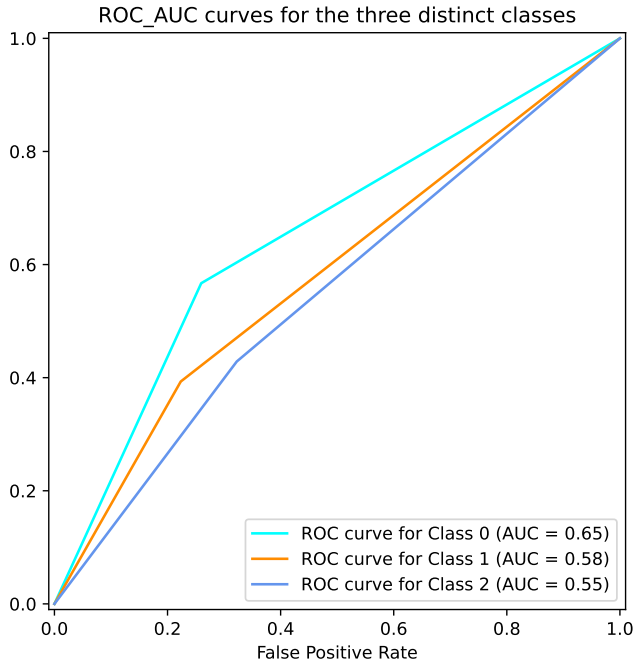


Figure A.7: ROC-AUC curve for the multi3cat dataset.

A.6.4.2 multiSN dataset

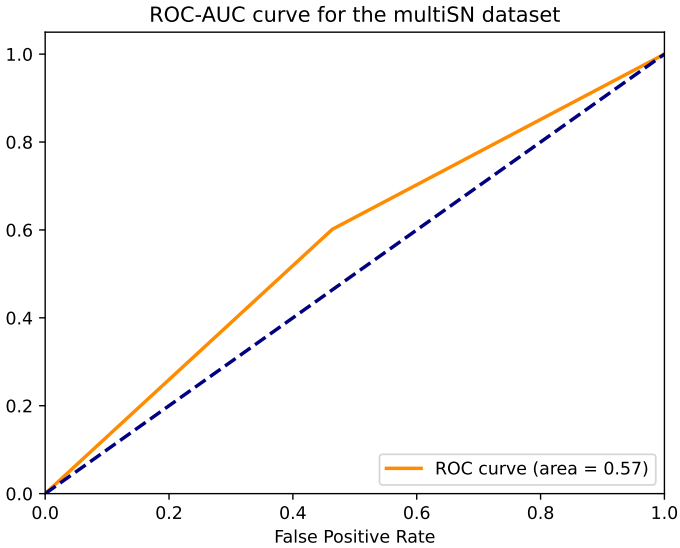


Figure A.8: ROC-AUC curve for the multiSN dataset.

A.6.5 AdaBoost with Decision Trees

A.6.5.1 multi3cat dataset

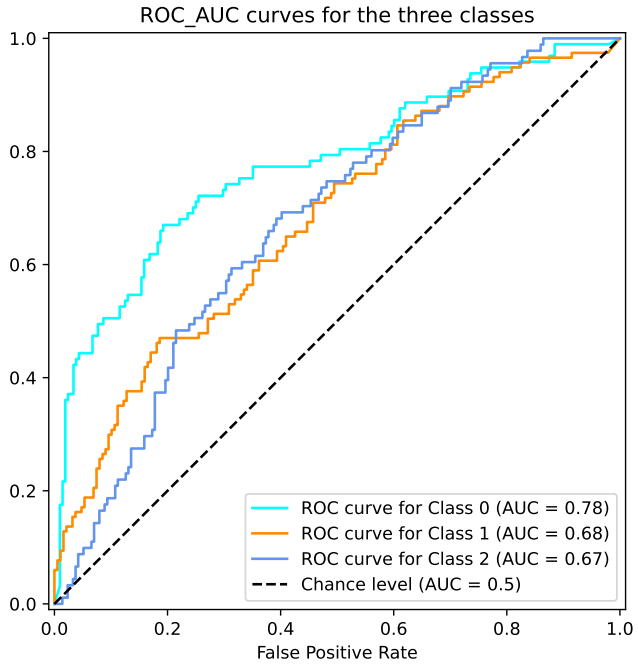


Figure A.9: ROC-AUC curve for the multi3cat dataset.

A.6.5.2 multiSN dataset

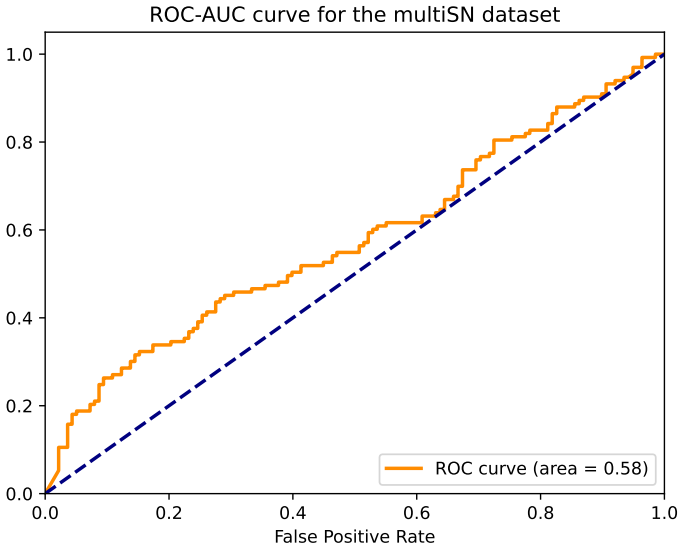


Figure A.10: ROC-AUC curve for the multiSN dataset.

A.6.6 XGBoost

A.6.6.1 multi3cat dataset

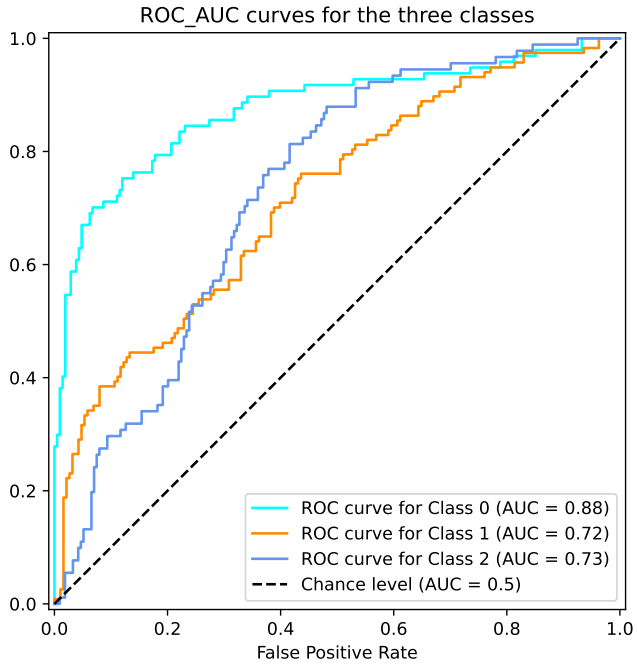


Figure A.11: ROC-AUC curve for the multi3cat dataset.

A.6.6.2 multiSN dataset

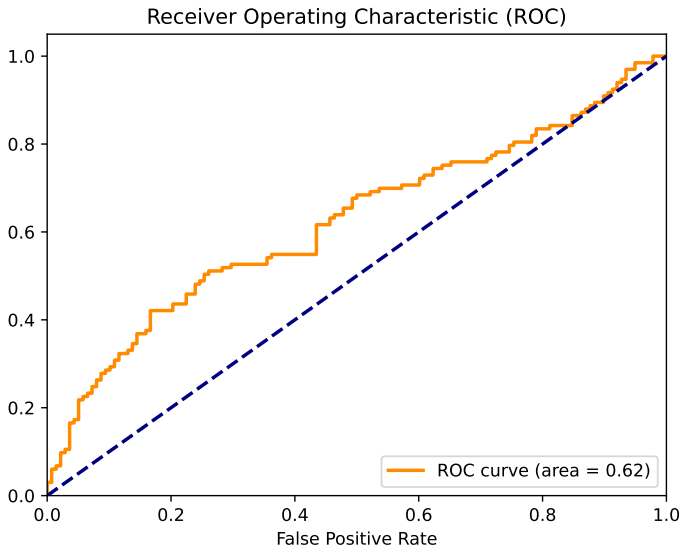


Figure A.12: ROC-AUC curve for the multiSN dataset.

A.7 Multi3cat feature cutoffs

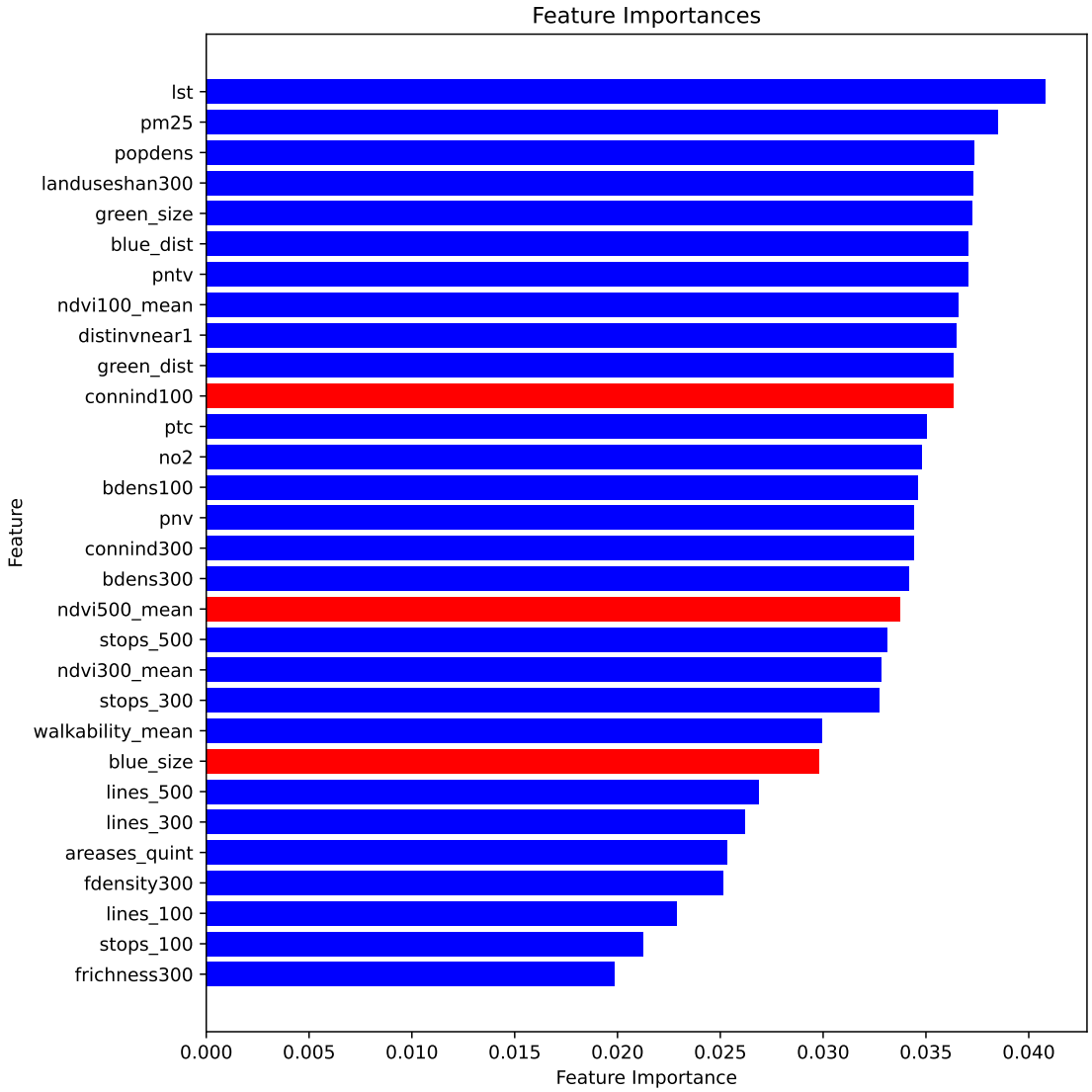


Figure A.13: Feature importances to be used to determine the cutoff points in the multi3cat dataset.

A.8 MultiSN feature cutoffs

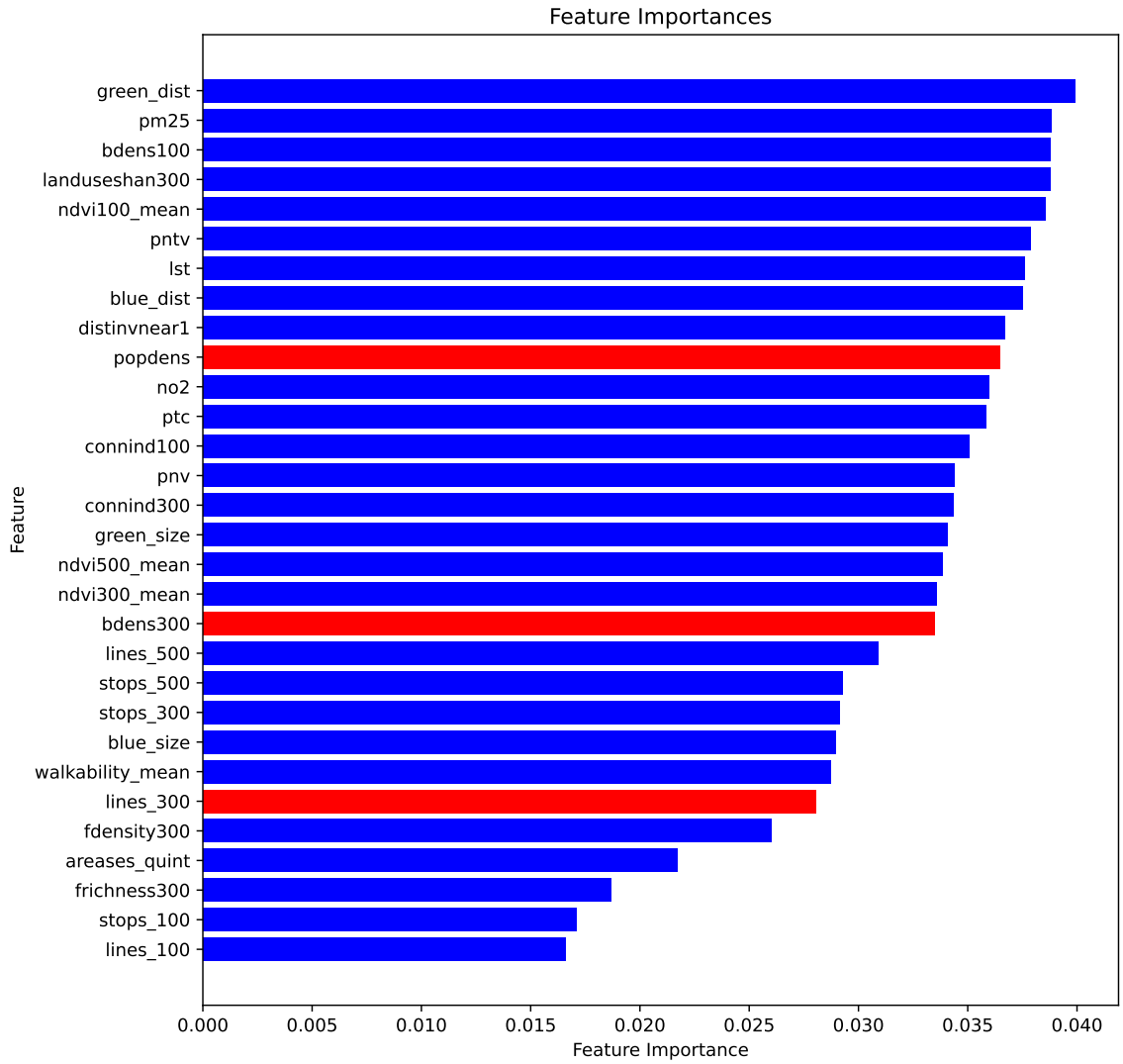


Figure A.14: Feature importances to be used to determine the cutoff points in the multiSN dataset.