



Resource Planning for Construction Projects

Group 18

Student ID:

11356488, 11334675

11306181, 11332444

May 2024

Applying Data Science

DATA70202

Contents

1	Introduction	1
2	Project Goals	3
2.1	Resource optimization:	3
2.2	Real-time Adjustment:	3
2.3	Decision-making Support:	4
3	Project Management	5
4	Data Pre-processing and EDA	7
4.1	Data Pre-processing	7
4.1.1	Merging datasets	7
4.1.2	Handling Duplicates	8
4.1.3	Feature Engineering	9
4.1.4	Data Oversampling	10
4.2	Exploratory Data Analysis	10
5	Methodology	16
5.1	Process flow Diagram	16
5.2	Modelling	17
5.2.1	One Hot Encoding	17
5.2.2	Random Over Sampler	18
5.2.3	Standard Scalar	19

5.2.4	Random Forest Classifier	19
5.2.5	Random Forest Regressor	21
6	Results and Outcomes	24
6.1	Model Results	26
6.1.1	Model 1 Results: Upcoming Project Department Count Prediction . . .	26
6.1.2	Model 2 Results: Upcoming Project Department Names Classification .	29
6.1.3	Model 3 Results: Upcoming Project Total Staff Count Prediction	32
6.1.4	Model 4 Results: Multi-Output Classifier for Department-wise Em- ployee Allocation	34
6.2	Model Deployment	36
7	Challenges and Limitations	41
8	Recommendation and Future work	43
9	Conclusion	45
	References	47
A	Appendix	51
A.1	Meeting Reports	51
A.2	User Guide	55
A.3	Code and Data Repository Links	57

Chapter 1

Introduction

With the continuous development and intensified competition in the global construction industry, traditional experiences and intuitions are no longer sufficient to cope with the increasingly complex project demands and variations. Therefore, by introducing advanced technologies such as data analytics and machine learning into the field of construction management, enterprises can be provided with more accurate and effective decision support, thereby improving project efficiency, reducing costs, and ultimately achieving sustainable growth.

This project is a collaboration with Equans, a well-known construction company operating in the United Kingdom, to develop a human resource allocation management system. Equans aims to utilize existing project data to forecast future project resource allocation, with the objective of providing efficient resource planning for the company's construction projects. The main focus of the project includes establishing a comprehensive resource planning framework for construction projects in Wales, England, and Scotland, including the allocation of staff such as engineers, project managers and mechanical supervisors.

The existing project data provided by Equans was initially analysed, followed by a step-by-step prediction of the results using a variety of classification and regression models, including models Random Forest Classifier and Random Forest Regression. The prediction consisted of a first step of determining which departments would be involved, a second step of predicting the names of the participating departments, a third step of predicting the total number of people

who would be involved in the project, and a fourth step of predicting the number of people involved in the project from each department. After completing the development of the model, we continued to work on developing a user-friendly and intuitive interface designed to allow people with no-technical background to easily use the prediction model.

This collaboration will help Equans to achieve optimization of human resource management and streamlining of decision-making processes, which will lead to increased project efficiency and productivity. In addition, the project represents a pioneering endeavor at the intersection of construction management and data science.

Chapter 2

Project Goals

For this project we worked closely with Equans. As a construction company operating in the UK, they aim to achieve effective resource planning in their construction projects. Thus, the following are the main goals of this project:

2.1 Resource optimization:

Develop a system that can predict the amount of employee resources needed for a project. By taking into account key variables such as hours of work, order intake, etc., the system will assist us in forecasting project manpower requirements for each department. Thereby, it will help to optimise resource allocation, cut down on waste, and boost efficiency.

2.2 Real-time Adjustment:

There will be four successive prediction steps implemented by the system. In addition to direct data input, the outcomes of the preceding forecasting stage will be used as a guide for the next level of forecasting. Furthermore, at any stage of the project, we can adjust the forecast result of the previous stage according to the latest project situation and external changes. This mechanism ensures that the prediction is flexible and adaptable. Moreover, the final prediction will also more closely represent actual needs.

2.3 Decision-making Support:

With our forecasting system, project managers will receive information on resource allocation, project progress, and other issues. These insights will be converted into workable plans that assist decision-makers in choosing more wisely and optimising project procedures.

Chapter 3

Project Management

In collaboration with Equans, the development of a human resource allocation management system was meticulously managed using the Waterfall methodology, segmented into a series of well-defined stages (Figure 1).

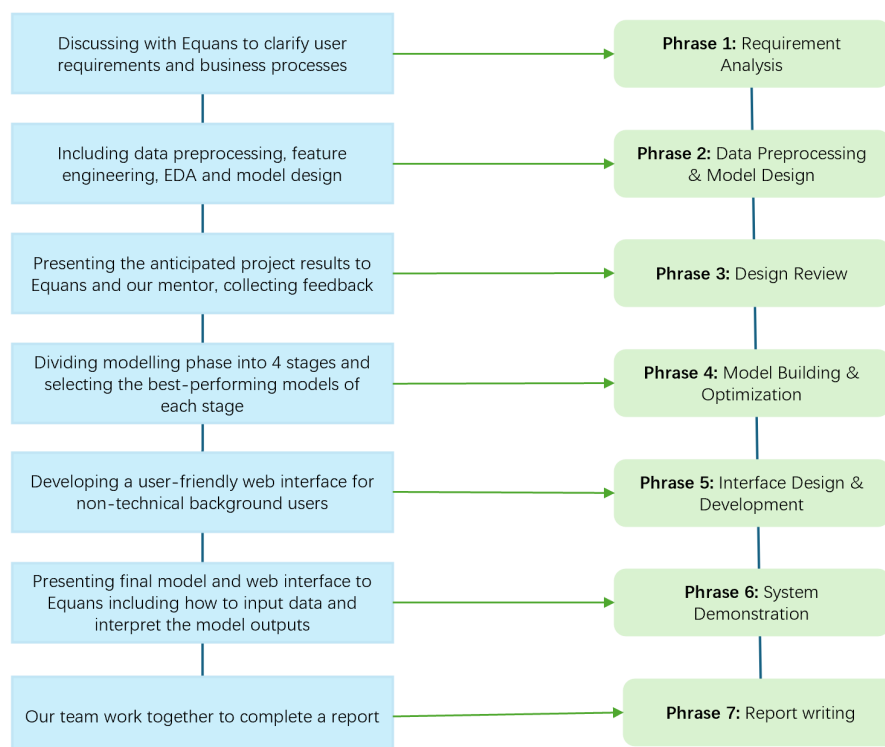


Figure 1: Project Management Diagram

In the division of labor within our team, each member participated in the data pre-processing and exploratory data analysis phases. The results of the study were then summarized and the model design strategy was developed through discussion. During the model building phase, while team members were each primarily responsible for one model, collaborative problem-solving sessions were held to effectively address challenges as they arose.

Chapter 4

Data Pre-processing and EDA

4.1 Data Pre-processing

4.1.1 Merging datasets

Before beginning any data analysis or model building in this project, we combined the two datasets that the industry partner provided us. Here's a description of the two original datasets' features:

2021-2023 Orders Report_Client Level

This dataset primarily records the order intake of different customers, comprising two sheets named Synthesis and data, respectively. The Synthesis sheet includes Clients' names and the order intake in euros for each month and year from 2021 to 2023. The data sheet serves as an expansion of the Synthesis sheet, containing a total of 1011 data samples (4 columns × 1011 rows). Additionally, each customer may correspond to multiple orders. Hence, it can be utilized to analyze the total number of orders for each customer.

TimesheetPortal-Report

This dataset includes a comprehensive work record for each employee, consisting of the date of the work record, employee name, department, job title, project name related to the work record,

specific task name, and the amount of work completed (in hours or days). In total, there are 240,433 data samples (9 columns \times 240,433 rows). Consequently, it can be utilized to analyze the workload of each department and the allocation of human resources to projects.

As shown in Figure 2, we matched and merged the two datasets using 'Project_name' to obtain a new dataset that combines all the information. It includes the total order intake required for the project, the involvement of each department, the duration of the project, etc. Through the analysis of this merged dataset, we were able to gain a deeper understanding of how each project's resources were allocated. Furthermore, the results of these analyses will also provide data support for the subsequent training of project resource planning models.

	Date	Employee_name	Department	Job Title	Project_name	Task_name	Work quantity	Rate code	Units	Client Name	Total Order Intake
0	2021-01-01	Bruce Murray	DATA & SECURITY	SENIOR PROJECT ENGINEER	Beta Top Corporation	XVDSM-OCA	1.0	STD_DAILY	Days	Beta Top Corporation	339309.3941
1	2021-01-01	Alison Good	QUALITY, SAFETY & ENVIRONMENT	H&S MANAGER	Beta Top Corporation	XVL	8.0	STD	Hours	Beta Top Corporation	339309.3941
2	2021-01-01	Davian Dudley	COMMERCIAL	SENIOR COMMERCIAL MANAGER	Beta Top Corporation	XVL	8.0	STD	Hours	Beta Top Corporation	339309.3941
3	2021-01-01	Brecken Butler	INDUSTRIAL MAINTENANCE	MECHANICAL SUPERVISOR	Lambda Keystone Concepts	XE-H-OM	8.0	STD	Hours	Lambda Keystone Concepts	-39220.7417
4	2021-01-01	Athena Griffith	INDUSTRIAL MAINTENANCE	SHIFT TEAM LEADER	Lambda Keystone Concepts	XE-H-OM	12.5	STD	Hours	Lambda Keystone Concepts	-39220.7417

Figure 2: Merged Dataset

4.1.2 Handling Duplicates

Since there were multiple work records for the same employee on the same project in the original dataset, we took the following steps to deal with them:

First, we used the 'Project_name' to group all of the projects together. Next, the total number of hours worked for each project was calculated by summing the hours contributed by each employee working on the project. As a result, the issue of duplicate counting data is avoided. Even if someone records work hours for a project more than once, only their overall contribution will be taken into account.

After that, the total amount of time spent on each project was merged into another data frame, with only one record per project. In this way, we created a new dataset that contains only unique entries and no duplicates for each project. It includes the duration of each project, the number of departments, the total order intake, and other relevant information. Moreover, it provides

accurate base data for subsequent data analysis and resource planning.

By using the above approach, we effectively dealt with the duplicates in the original timesheets and ensured the accuracy and validity of the data.

	Project_ID	Project_name	Duration_hours	Total_order_intake	Num_of_Departments	min	max	Duration_days	Number_of_JobTitles	Number_of_Tasks
0	1	Alpha Gateway Networks	6040.5	2116.7104	12	2022-04-11	2024-02-14	674	22	3
1	2	Beta Top Corporation	284691.0	339309.3941	15	2021-01-01	2024-02-21	1146	72	8
2	3	Chi Gateway Consultants	126.0	15.8268	2	2021-01-19	2024-02-01	1108	3	2
3	4	Chi Summit Technologies	9532.0	2785.9621	9	2021-01-04	2024-02-21	1143	17	12
4	5	Epsilon Apex Partners	4907.0	602.0486	6	2021-01-04	2022-10-14	648	11	1

Figure 3: Dataset With No Duplicates

4.1.3 Feature Engineering

In this section, we extended and enriched the original dataset by deriving several new features from the existing data, allowing us to gain more comprehensive insights into project management and resource allocation.

First, we added up the workloads of all the jobs that were recorded in a single project to calculate the total working hours for every project. Next, in order to better understand the complexity and diversity of project resource allocation, we computed the number of distinct jobs and tasks associated with each project. We also counted the number of employees in each project to illustrate how human resources are distributed throughout projects.

Furthermore, we calculated the overall project duration by counting the days from the start to the finish of each project. Based on this, to better understand the relationship between time resources and human resources, we created the 'Duration to Employees Ratio,' which is the project's duration divided by the total number of employees.

To further analyse the resource allocation efficiency of the project, we generated a number of efficiency and diversity indicators. 'Work Intensity' is a measure of the project's workload. Moreover, diversity in the project's staff is measured by 'Department diversity' and 'JobTitle diversity', which indicate the percentage of departments and jobs involved, respectively.

We also calculated the average amount of work per department and the average amount of

work per task. These metrics help us assess whether the workload distribution is balanced across different departments and tasks.

Meanwhile, the variable 'Department' in the original dataset has a string data type. However, most machine learning algorithms require the input data type to be numeric. In light of this, we use One-Hot encoding for the 'Department' field. It converts each department into a new binary (0 or 1) feature. This method of coding enables the model to better handle the categorical data. At the same time, it retains all the details of departmental involvement.

With feature engineering, we not only enriched the dataset and provided a more comprehensive perspective to analyse project resources and management, but also laid the groundwork for future model training.

4.1.4 Data Oversampling

Due to the issue of unbalanced data in the original dataset, we utilised Data Oversampling as a solution. It is a common method to improve the balance of the data distribution by randomly copying samples from a few classes. In doing so, the model gains a deeper understanding of the patterns of a few classes during the training process. Moreover, it is able to prevent overfitting or performance degradation due to the sample distribution bias.

In our research, we used RandomOverSampler, which is a simple and effective oversampling method. With RandomOverSampler, we separated the features and target variables from the original dataset and oversampled them. Then, we obtained new balanced features and target variables. Finally, we recombined them into a new data frame.

Following the oversampling procedure, the dataset grew in size and became more evenly distributed across all categories. Meanwhile, the generalisation ability and accuracy of the model were also improved.

4.2 Exploratory Data Analysis

After data preprocessing, Currently there are 13 variables in total, which are shown in table 1:

Table 1: Description of variables

Features	Description
Duration_hours	The total number of hours allocated to a project
Total_order_intake	The total cost of a project.
Num_of_Departments	The number of different departments involved in a project.
Duration_days	The total number of days from the start to the completion of a project.
Number_of_JobTitles	The total number of job titles
Number_of_Tasks	The count of distinct job titles associated with a project.
Work_Intensity	The intensity of work per day.
Department_diversity	The variety of departments within a project.
JobTitle_diversity	The diversity of job titles within the project.
Average_work_quantity_per_department	The average hours assigned to each department within a project.
Average_Work_Quantity_per_Task	The average amount of work (in hours) expected for each task within a project.
Num_of_Employees	The total number of employees allocated to a project.
Duration_to_Employees_Ratio	The ratio of the project's duration in days to the number of employees involved.

Initially, an examination was conducted on the Total_order_intake distribution across projects (Figure 4). Given the substantial variance observed in Total_order_intake values across projects, Minmax scaling was applied for data normalization, thereby rescaling Total_order_intake values of each project to a range between 0 and 1. As depicted in Figure 4, it is notable that the majority of observations are concentrated below 0.2, indicating relatively lower project costs. Additionally, a limited number of projects (Beta Top Corporation, Phi Crown Partners, Omega Zenith Futures) exhibit values nearing 1.0, indicates uneven distribution of data with extreme values.

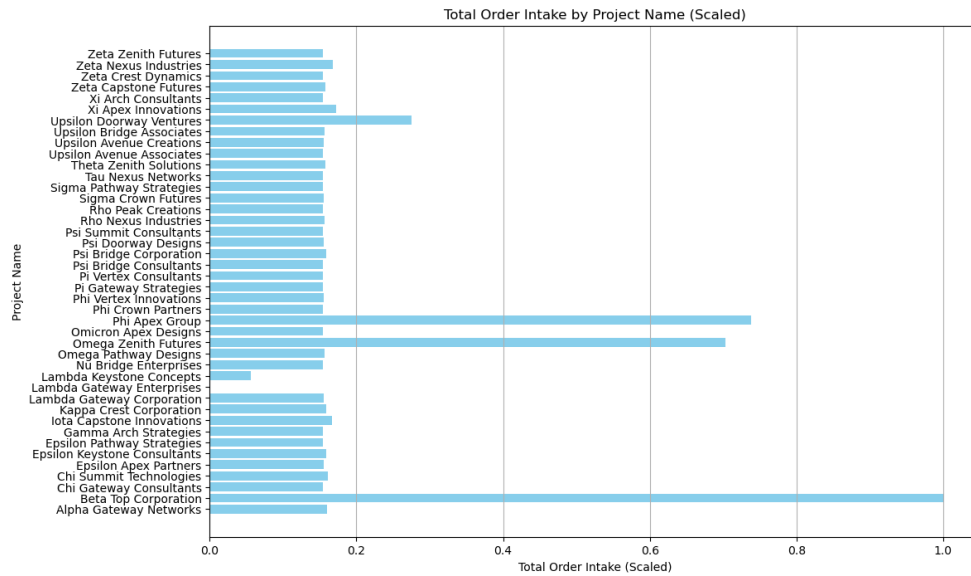


Figure 4: Total_order_intake distribution across projects

Next, a box-plot analysis was conducted to investigate the presence of outliers in variables. As shown in Figure 5, except for Num_of_Department and Duration_days, all variables have a large number of outliers.

- **Number_of_JobTitles:** Indicates the total number of different job titles involved in projects. The majority of projects involve a limited variety of job titles.
- **Number_of_Tasks:** Illustrates the quantity of tasks in each project. The majority of projects have a small number of tasks.
- **Work_Intensity:** Depicts the distribution of daily work intensity for projects. While most projects exhibit low daily work intensity, a few projects demonstrate higher intensity.
- **Department_diversity** and **JobTitle_diversity:** Display the diversity of personnel involved in each project. It can be observed that the majority of projects involve a limited number of departments and job titles.
- **Average_work_quantity_per_department** and **per_task:** Indicates that the workload of most departments and tasks is relatively low, although there are a few departments with high workloads.

In summary, all variables exhibit outliers, necessitating the consideration of standardization and oversampling techniques in subsequent model development to mitigate their impact.

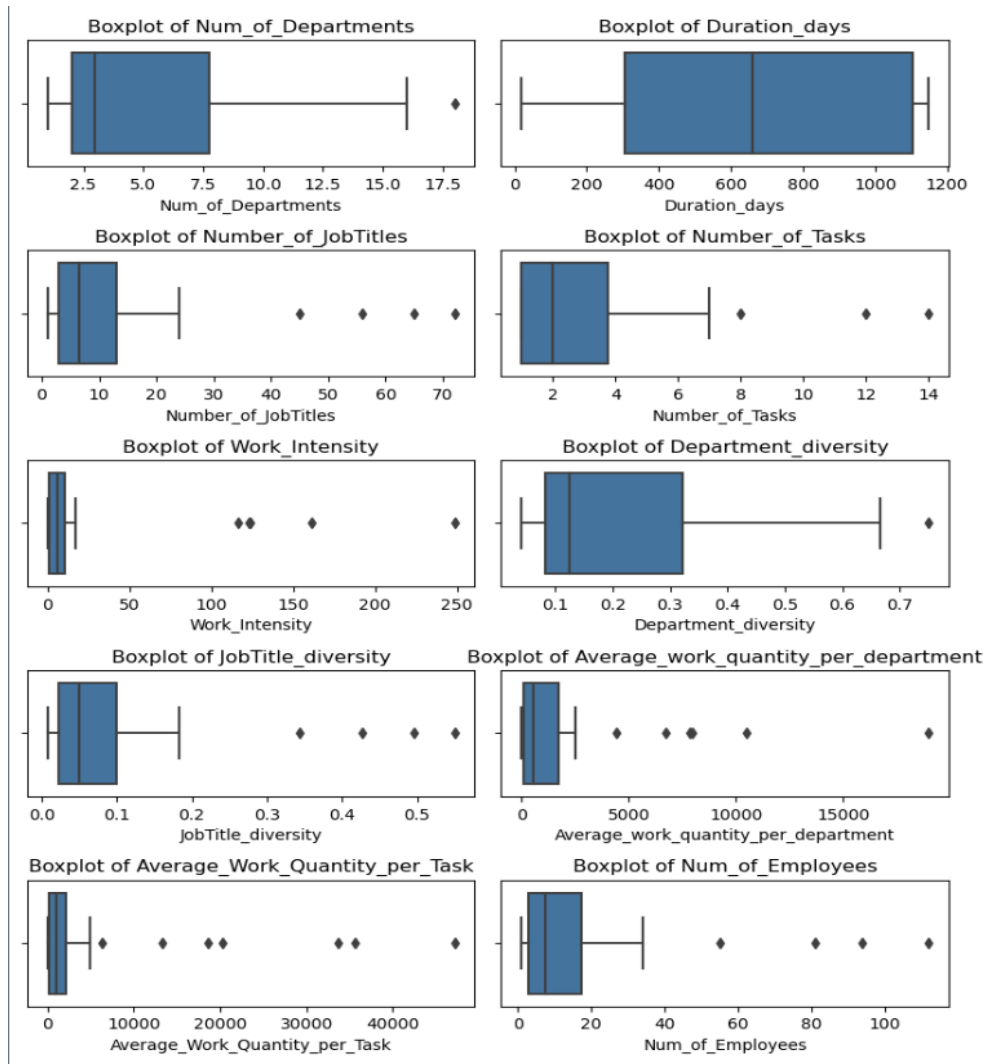


Figure 5: Boxplot of variables

The duration of each project was then explored (Figure 6), with the longest lasting project being Upsilon Doorway Ventures (1,148 days) and the shortest lasting project, Theta Zenith Solutions (19 days).

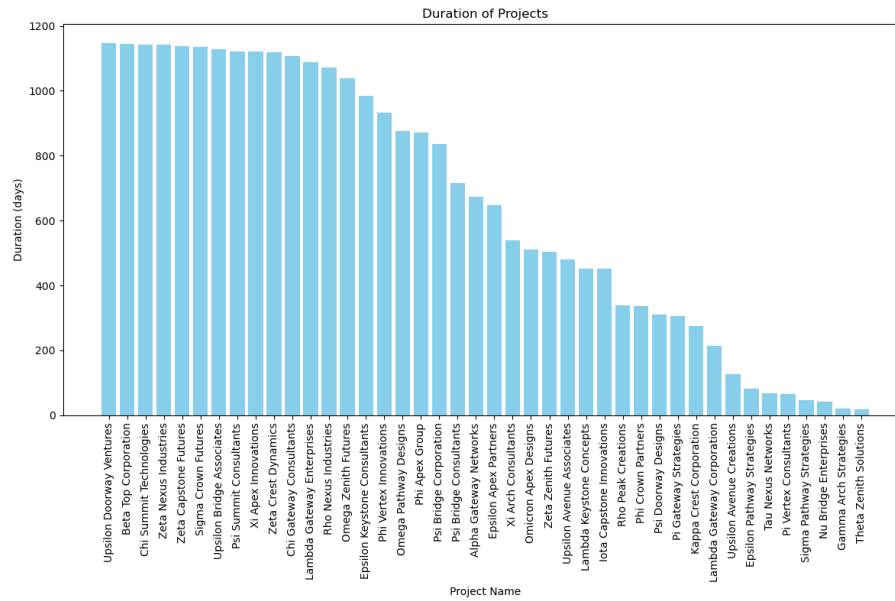


Figure 6: Duration of projects

In addition, We explored how often each department appeared in the project (Figure 7), the department that appeared most frequently was “project management” and the department that appeared least frequently was “procurement”.

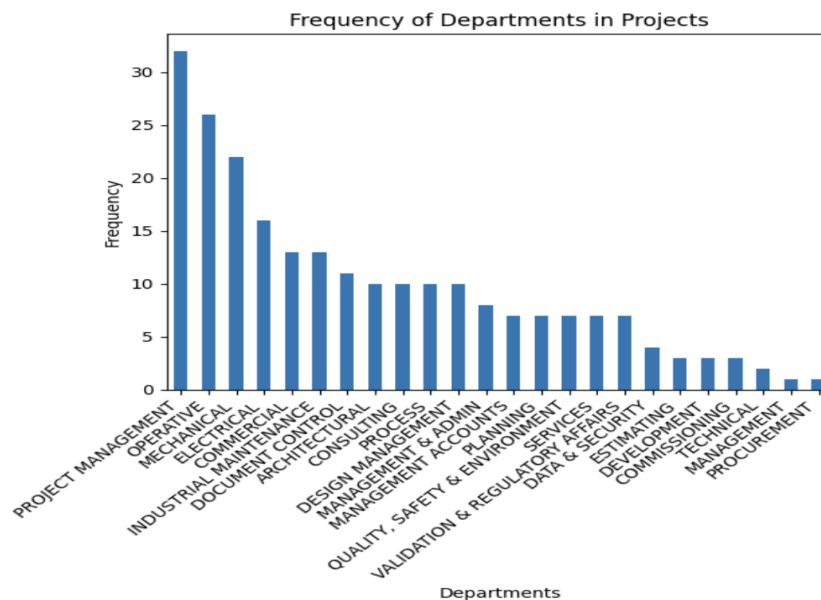


Figure 7: Frequency of Departments in Projects

Moreover, as can be seen from the correlation matrix (Figure 8), only the variables Duration_days and Number_of_Tasks do not have a strong correlation ($r > 0.6$) with the target variable of model 1 (Num_of_Departments). The rest of the variables are significantly and

positively correlated with Num_of_Departments, so these variables were chosen as features for the first model.

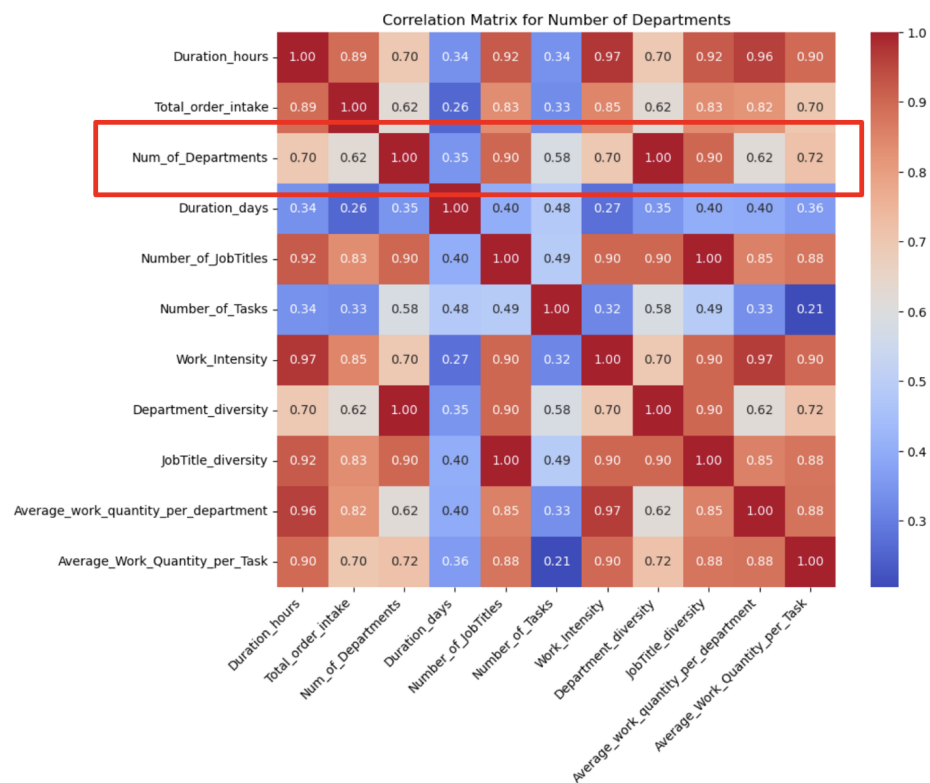


Figure 8: Correlation matrix

Chapter 5

Methodology

5.1 Process flow Diagram

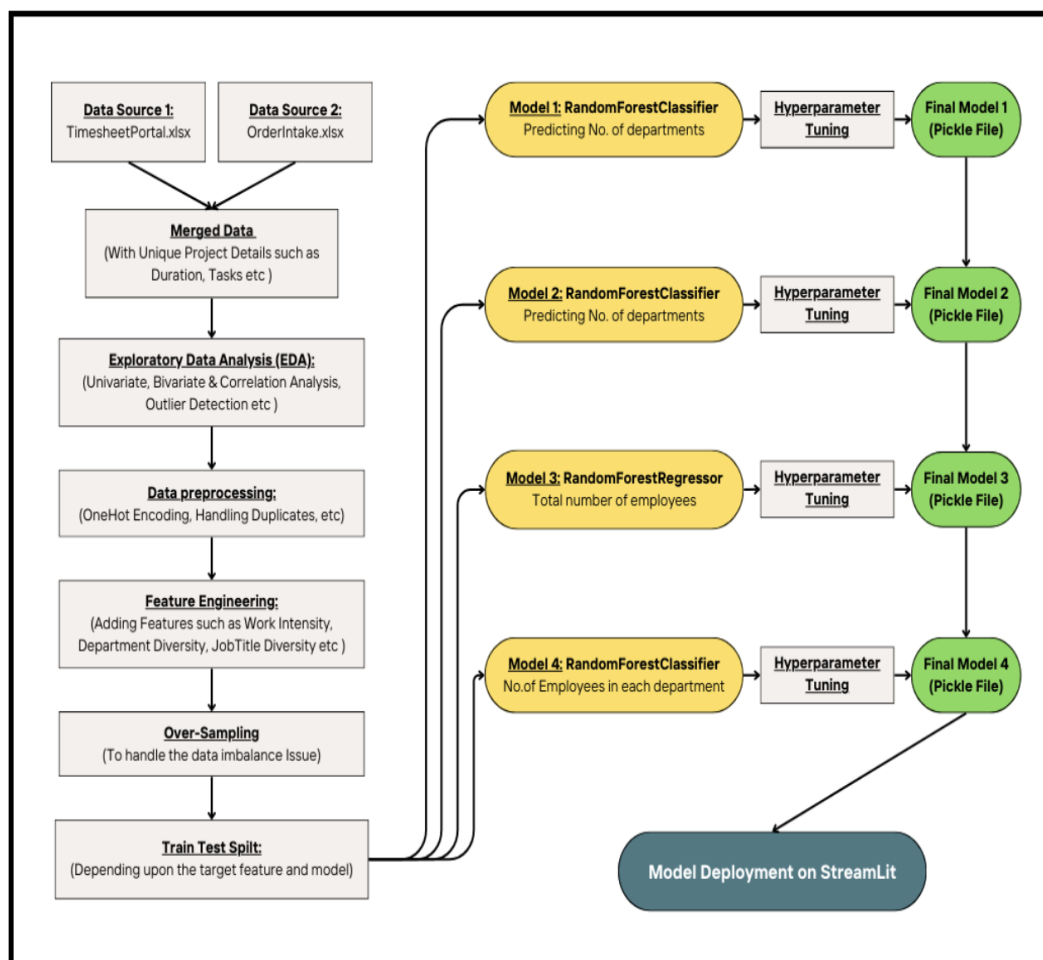


Figure 9: Process Flow Diagram

5.2 Modelling

5.2.1 One Hot Encoding

One Hot Encoding is a machine learning and data preprocessing technique that converts categorical variables into numerical representations that algorithms can understand. It is especially useful when working with categorical variables that lack an intrinsic numerical order or ranking.

The idea behind One Hot Encoding is to create a binary column for each unique category in the categorical variable. Each row in the dataset is then assigned a value of 1 in the column corresponding to its category, and 0 in all other columns. This process effectively transforms the categorical variable into a set of binary variables, with each category represented by a separate column.

Mathematically, if we have a categorical variable X with n unique categories, One Hot Encoding creates n new binary columns, X_1, X_2, \dots, X_n . For each instance i in the dataset, the value of X_j is set to 1 if the instance belongs to category j , and 0 otherwise. This can be represented as follows:

$$X_j^{(i)} = \begin{cases} 1 & \text{if } X^{(i)} = j, \\ 0 & \text{otherwise.} \end{cases}$$

Where $X^{(i)}$ represents the value of the categorical variable for instance i , and $X_j^{(i)}$ represents the value of the binary column corresponding to category j for instance i .

One Hot Encoding was applied to the department names to convert them into a numerical representation suitable for the machine learning algorithm. For example, if the dataset contained departments like "DATA & SECURITY", "DESIGN MANAGEMENT", "DEVELOPMENT", etc., One Hot Encoding would create a binary column for each unique department name.

After applying One Hot Encoding, the dataset would have additional columns representing each department, with values of 1 or 0 indicating the presence or absence of that department for each instance. This numerical representation allowed the machine learning algorithm to understand

and learn the patterns associated with each department, enabling accurate predictions.

5.2.2 Random Over Sampler

In our project, we encountered a class imbalance problem, where the number of instances for different departments varied significantly. This imbalance could lead to biased predictions, as the machine learning model would tend to favor the majority class (departments with more instances) during training, while neglecting the minority classes (departments with fewer instances).

To address this issue, we employed the RandomOverSampler technique from the imbalanced-learn library in Python. The RandomOverSampler is an oversampling method that aims to balance the class distribution by randomly duplicating instances from the minority class(es) until their representation matches that of the majority class.

The working principle of the RandomOverSampler is straightforward. It identifies the minority class(es) in the dataset and randomly selects instances from these classes to be duplicated. This process continues until the number of instances for each class is equal to the number of instances in the majority class. By doing so, the RandomOverSampler creates a synthetic dataset with a balanced class distribution, allowing the machine learning model to learn patterns from all classes without favoring the majority class.

In our project, we had a dataset containing information about various projects and their associated departments. We noticed that some departments, such as "DATA & SECURITY" and "DESIGN MANAGEMENT," had significantly more instances than others, like "DEVELOPMENT" and "TESTING." This imbalance could have led to biased predictions, where the model performed well for the majority departments but poorly for the minority ones.

To mitigate this issue, we applied the RandomOverSampler to our dataset. By training our machine learning model on the oversampled dataset, we ensured that it learned patterns and relationships specific to each department without being biased towards the majority classes. This approach helped us achieve more generalized and better predictions across all departments, reducing the potential for biased or skewed results.

5.2.3 Standard Scalar

We encountered a challenge with the numeric features in our dataset. Features such as `Duration_hours`, `Duration_days`, `Total_order_intake`, and `Work_Intensity` had different scales and units, which could potentially cause issues during the modeling process. To address this challenge, we employed the `StandardScaler` technique from the `scikit-learn` library in Python.

The `StandardScaler` is a popular method for scaling numeric features in machine learning. It standardizes the features by subtracting the mean and dividing by the standard deviation of each feature. This transformation ensures that all features are on a similar scale, centered around zero, with a standard deviation of one. Scaling features is crucial because many machine learning algorithms assume that the input features have a similar range and scale, which can significantly impact the model's performance and convergence.

The `StandardScaler` works by calculating the mean and standard deviation of each numeric feature independently across the entire dataset. It then applies the following transformation to each feature value:

$$x_{\text{scaled}} = \frac{x - \mu}{\sigma}$$

Where x is the original feature value, μ is the mean of the feature, and σ is the standard deviation of the feature.

By applying this transformation, the `StandardScaler` ensures that the scaled features have a mean of zero and a standard deviation of one. This scaling process helps to prevent features with larger ranges from dominating the model's learning process and ensures that all features contribute equally to the final predictions.

5.2.4 Random Forest Classifier

In our project, we employed the powerful Random Forest Classifier algorithm, which is an ensemble learning technique that proved to be particularly useful in our second model, where we aimed to predict the names of the departments that would be involved in a given project.

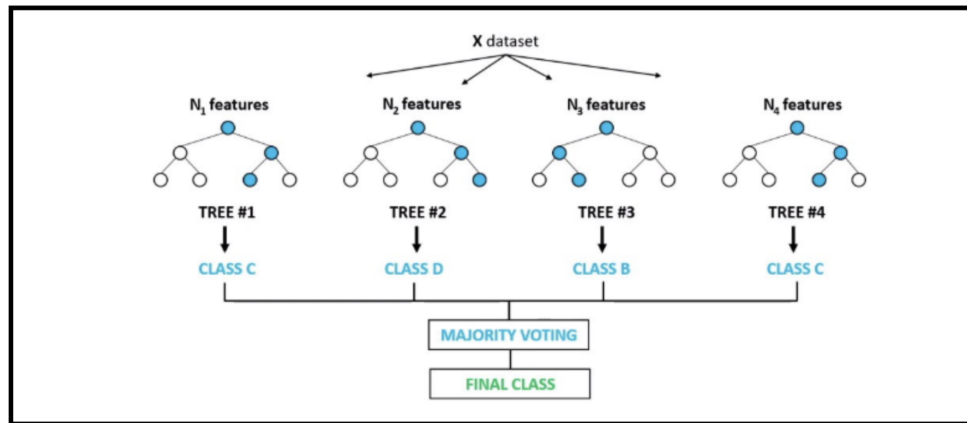


Figure 10: Schematic Diagram of Random Forest Classifier

Overview

The Random Forest Classifier is an ensemble learning method that constructs multiple decision trees during the training phase and combines their predictions to make a final decision. Each individual decision tree in the ensemble is trained on a randomly selected subset of the training data and a random subset of features, introducing randomness and diversity into the model.

The fundamental concept behind the Random Forest Classifier is the “wisdom of crowds” principle, where the collective decision of multiple diverse models often outperforms any single model. By combining the predictions of numerous decision trees, the Random Forest Classifier can capture complex patterns and relationships in the data, leading to improved accuracy and robustness compared to individual decision trees.

Working

The Random Forest Classifier operates by constructing multiple decision trees, each trained on a bootstrap sample of the training data. During the training process, at each node of a decision tree, a random subset of features is considered for splitting the data. This randomization process helps to reduce the correlation between individual trees, promoting diversity within the ensemble.

Once the decision trees are trained, the Random Forest Classifier makes predictions by aggregating the predictions of all the individual trees. For classification tasks, such as predicting the

names of departments, the final prediction is determined by majority voting, where the class (department name) predicted by the highest number of trees is selected as the output.

Application

In our second model, we leveraged the output from the first model, which predicted the number of departments for a given project, along with other relevant features. We then utilized the Random Forest Classifier to predict the specific names of the departments that would be involved in that project.

By employing the Random Forest Classifier in our second model, we were able to leverage its ability to handle complex relationships and interactions between the input features. The ensemble nature of the algorithm allowed us to capture the intricate patterns and dependencies that govern the assignment of specific departments to projects, leading to more accurate and reliable predictions.

Advantages

The Random Forest Classifier is known for its robustness to overfitting, as the ensemble approach and the randomization techniques used during training help to reduce the risk of individual decision trees overfitting to the training data.

5.2.5 Random Forest Regressor

The Random Forest Regressor is a variation of the Random Forest algorithm that is specifically developed for regression problems that require predicting a continuous numerical value rather than a categorical class. The Random Forest Regressor, like its classification counterpart, works by creating an ensemble of decision trees, each trained on a random subset of the training data and features.

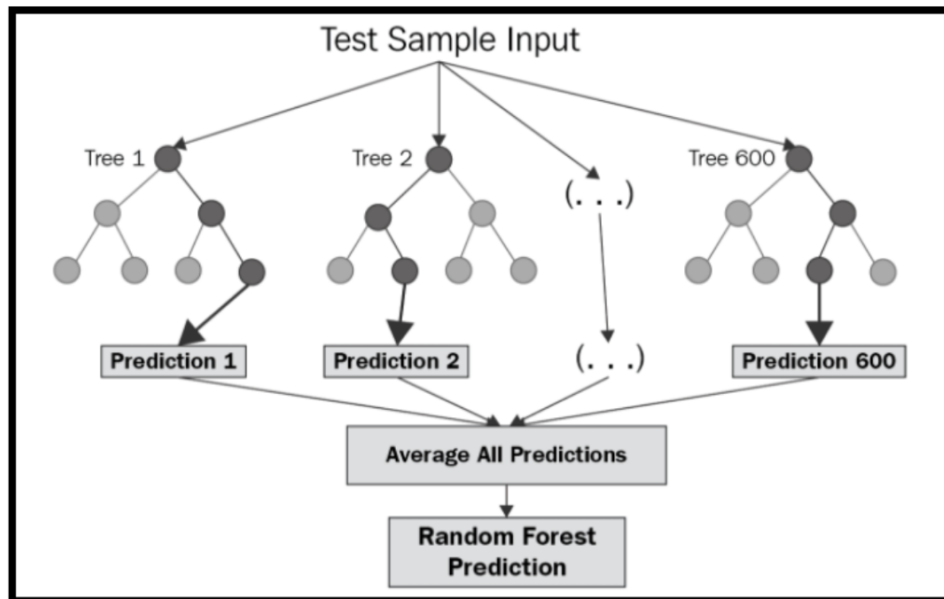


Figure 11: Schematic Diagram of Random Forest Regressor

Overview

The Random Forest Regressor constructs multiple decision trees, each trained on a bootstrap sample of the training data. During the training process, at each node of a decision tree, a random subset of features is considered for splitting the data. This randomization process helps to reduce the correlation between individual trees, promoting diversity within the ensemble and preventing overfitting.

Once the decision trees are trained, the Random Forest Regressor makes predictions by aggregating the predictions of all the individual trees. For regression tasks, the final prediction is determined by taking the average or mean of the predictions made by all the trees in the ensemble.

Application

In our first model, we utilized the Random Forest Regressor to predict the number of departments required for each project. We fed the algorithm with a set of relevant features, such as Duration_hours, Total_order_intake, Duration_days, and other project-related variables. By training the Random Forest Regressor on this data, the algorithm learned to identify patterns

and relationships between these features and the target variable, which was the number of departments.

In our third model, we leveraged the Random Forest Regressor once again, this time to predict the total number of employees required for a project. We utilized the output from our second model, which provided the names of the predicted departments, along with other relevant features. By training the Random Forest Regressor on this data, we were able to accurately forecast the overall headcount required for the upcoming project.

Furthermore, in our fourth model, we employed the Random Forest Regressor to predict the number of employees required for each individual department predicted by our second model. We used the output from our third model, which provided the total number of employees, along with other features, to train the algorithm and obtain granular predictions for the staffing requirements of each department.

Advantages

The Random Forest Regressor's ability to handle complex non-linear relationships and its robustness to outliers and noise in the data made it an ideal choice for our project. By leveraging the ensemble nature of the algorithm, we were able to capture intricate patterns and dependencies, leading to more accurate and reliable predictions across all stages of our analysis.

Chapter 6

Results and Outcomes

We will examine the outcomes of all 4 models utilizing diverse metrics appropriate for classification problems and other metrics suitable for regression scenarios.

METRICS FOR CLASSIFICATION PROBLEMS

- **Confusion Matrix:** A tabular representation displaying the counts of true positive, false positive, true negative, and false negative predictions.

Predicted Values	Actual Values	
	Positive (1)	Negative (0)
Positive (1)	TP	FP
Negative (0)	FN	TN

- **Accuracy:** The proportion of accurate predictions to the overall number of predictions made.

$$\text{Accuracy} = \frac{tp + tn}{tp + tn + fp + fn}$$

- **F1 Score:** The harmonic mean of precision and recall, offering a balance between the two.

$$F1 = 2 \cdot \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}$$

- **Precision:** The proportion of correct positive forecasts to the total number of positive forecasts.

$$\text{Precision} = \frac{\text{TruePositive}}{\text{TruePositive} + \text{FalsePositive}}$$

- **Recall (Sensitivity):** The proportion of correct positive forecasts to the total number of real positives.

$$\text{Recall} = \frac{\text{TruePositive}}{\text{TruePositive} + \text{FalseNegative}}$$

- **ROC Curve and AUC:** The trade-off between true positive rate and false positive rate is displayed by the Receiver Operating Characteristic curve (ROC) and Area under the curve (AUC).
- **Classification Report:** provides the summary of precision, recall, F1-score and support for each class in the dataset.

METRICS FOR REGRESSION SCENARIOS

- **Mean Absolute Error (MAE):** The mean of the absolute variances between predictions and real values.

$$MAE = \frac{1}{n} \sum_{i=1}^n |y_i - x_i|$$

- **Root Mean Squared Error (RMSE):** Measures the average error magnitude between predicted and real values.

$$RMSE = \sqrt{MSE}$$

$$RMSE = \sqrt{\frac{1}{\text{total samples}} \sum_{i=1}^{\text{total samples}} (\text{true}_i - \text{predicted}_i)^2}$$

$$RMSE = \sqrt{\frac{1}{n} \sum_{i=1}^n (y_i - f(x_i))^2}$$

- **R² Score (Coefficient of determination):** Determines the predictability proportion of the dependent variable from independent variables. (1= perfect score, 0 or less = worse model performance)

- **Adjusted R^2 :** It is a modified version of R^2 considering the number of predictors in the model and it penalizes the inclusion of redundant predictors in the model.

$$\text{Adjusted } R^2 = 1 - (1 - R^2) \frac{(N - 1)}{N - p - 1}$$

where R^2 is the sample R-Squared, N is the total sample size, and p is the number of independent variables.

6.1 Model Results

6.1.1 Model 1 Results: Upcoming Project Department Count Prediction

The Random Forest Classifier utilized in Model 1 aimed to predict the number of departments for a future project, we initially evaluated the random forest classifier model on original dataset with imbalanced classes and subsequently evaluated the performance of model 1 on oversampled data which handles the class imbalances.

Model 1 struggled to generalize to original imbalanced dataset with low test accuracy of 33.33% (Figure 12), highlighting the challenge of skewed class distributions. To address this, we transitioned to an oversampled dataset to mitigate class imbalance, leading to a substantial enhancement in testing accuracy to **94.44%**.

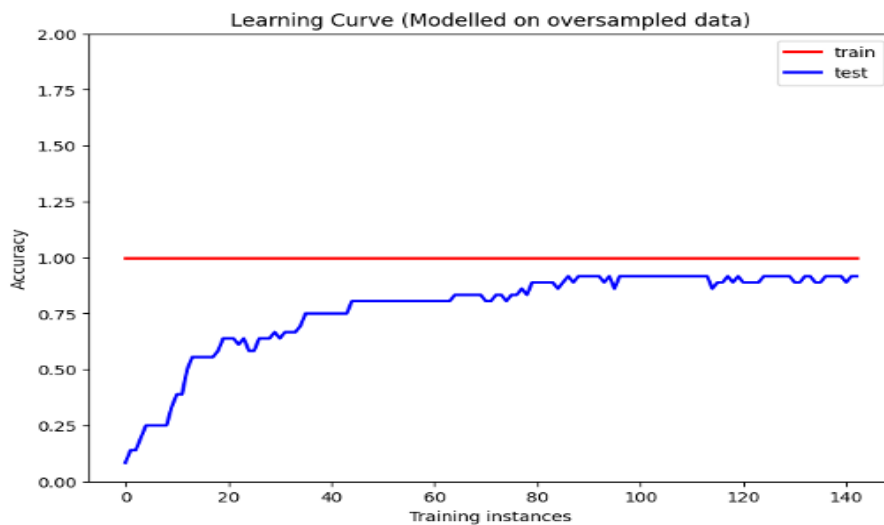


Figure 12: Learning Curve of Model 1 on Original Dataset

Cross-validation confirmed model's robustness and reliability, with scores ranging from 86% to 100%. Classification report demonstrated improved precision, recall and F1-scores across all classes, indicating enhanced predictive power for department counts (Table 2). The ROC-AUC curve (Figure 13) exhibited an L-shaped-fit pattern, affirming the model's exceptional discriminatory power and its capacity to accurately classify department counts.

Table 2: Classification Report for Model 1 (Department Counts Classification)

Department Count (Classes)	Precision	Recall	F1-score
1	1.00	1.00	1.00
2	0.67	0.67	0.67
3	0.33	0.50	0.40
5	1.00	0.50	0.67
6	1.00	1.00	1.00
7	1.00	1.00	1.00
8	1.00	1.00	1.00
9	1.00	1.00	1.00
10	1.00	1.00	1.00
11	1.00	1.00	1.00
12	1.00	1.00	1.00
13	1.00	1.00	1.00
15	1.00	1.00	1.00
16	1.00	1.00	1.00
18	1.00	1.00	1.00
<i>Accuracy</i>			0.92
<i>Macro Avg</i>	0.93	0.91	0.92
<i>Weighted Avg</i>	0.94	0.92	0.92

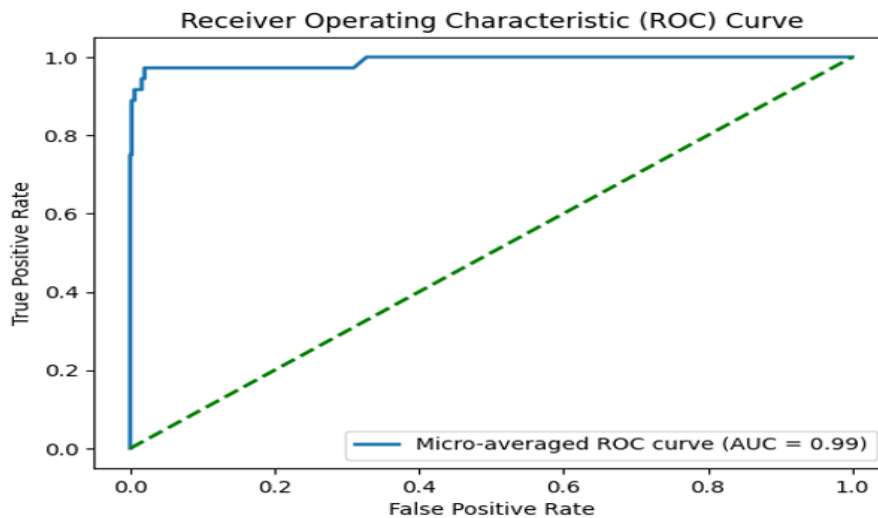


Figure 13: ROC AUC Curve for Tuned Model 1

Hyperparameter Tuning

The process of hyperparameter optimization was carried out using Optuna to boost the model's efficacy and mitigate issues related to overfitting. The refined model attained a training accuracy of **97.92%** and a testing accuracy of **97.22%**, demonstrating its proficiency in generalizing effectively to unseen data (Table 3).

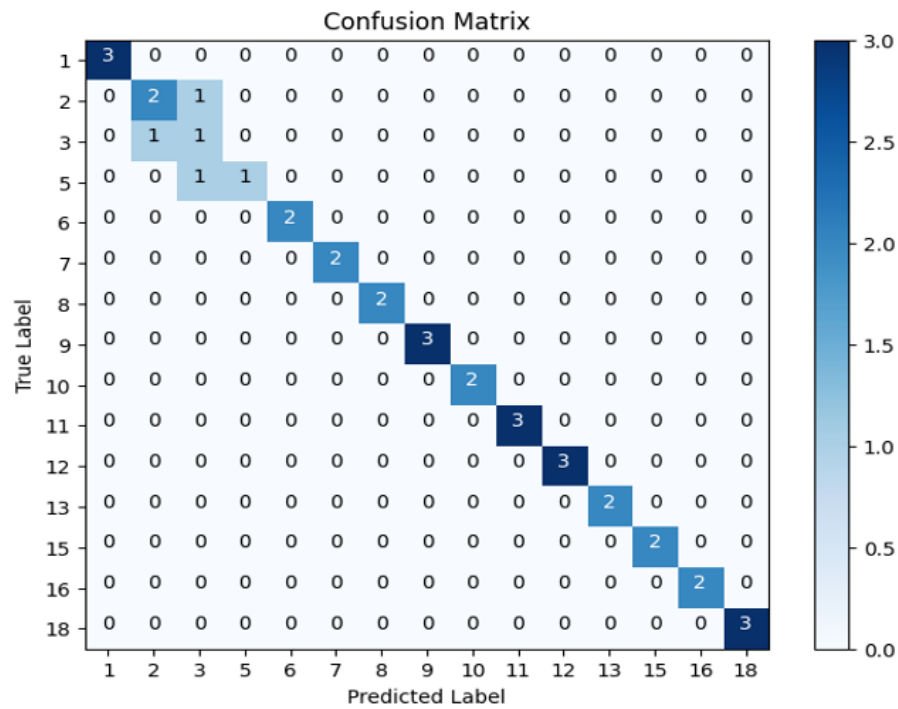


Figure 14: Confusion Matrix for Hyper Parameter Tuned Model 1

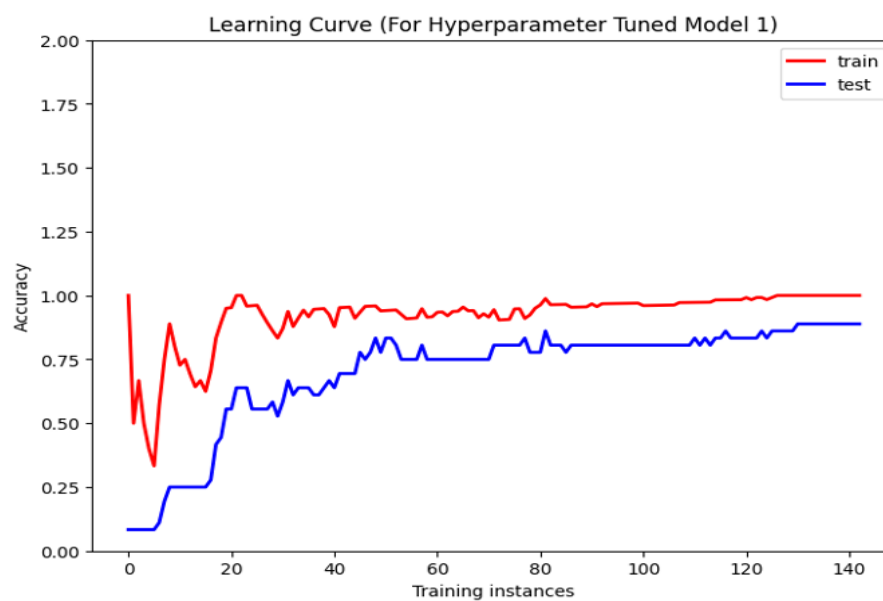


Figure 15: Improved Learning Curve of Model 1 After Hyperparameter Tuning

Table 3: Model 1 Performance Evaluation and Results Comparison

MODEL 1 APPROACH	TEST ACCURACY	TRAIN ACCURACY	F1 SCORE
Base Model 1 on Original Data	33.33%	100%	0.33
Model 1 on Oversampled Data	97.22%	100%	97.22
Hyperparameter Tuned Model 1	91.66%	97.91%	91.66

6.1.2 Model 2 Results: Upcoming Project Department Names Classification

Model 2, which integrates Multioutput Classifier with Random Forest Classifier as the base estimator(with default parameters), displayed robust performance in predicting the department names for upcoming project. Through cross-validation, an average accuracy of 89.10% was detected, demonstrating stable performance across diverse folds. The training accuracy and F1 score both reached 100% showcasing a perfect fit to the training dataset, this is a case of overfitting since the dataset is very small. After examining the test set, the model obtained an accuracy of **92%** and F1 of **99%**, showcasing reliable predictive prowess(Table 4, 5). The micro-averaged ROC AUC of 1.00 further emphasises the model's outstanding discriminatory capability across various departments. Despite potential overfitting Model 2 remains a dependable method for selecting the departments for the new project.

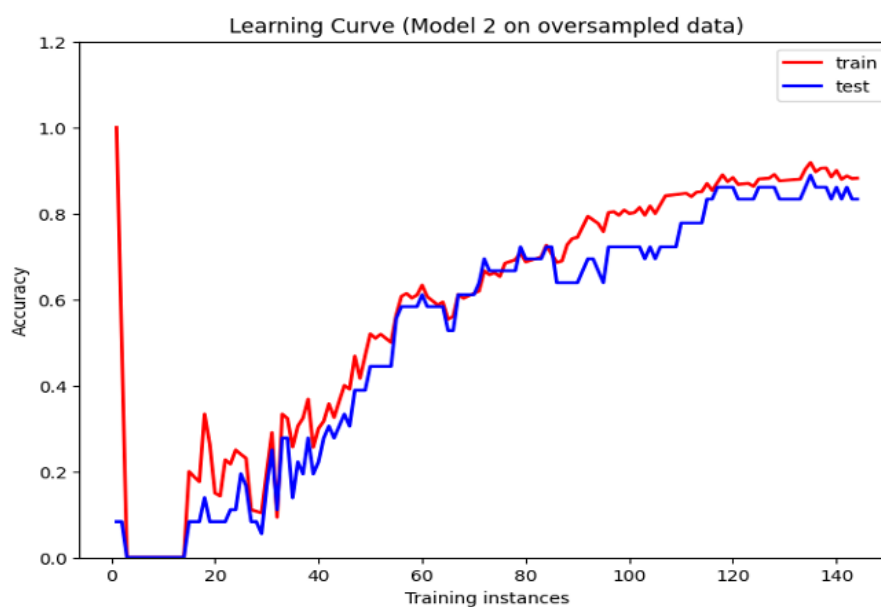


Figure 16: Learning curve of oversampled dataset on Model 2

We handled the overfitting by **tuning the hyperparameters** of our model which results in marginally improved performance(Figure 19).

Table 4: Classification report metrics of Tuned Model 2

Department Names (Classes)	Precision	Recall	F1-score
ARCHITECTURAL	1.00	1.00	1.00
COMMERCIAL	1.00	1.00	1.00
COMMISSIONING	1.00	1.00	1.00
CONSULTING	1.00	1.00	1.00
ATA & SECURITY	1.00	1.00	1.00
IGN MANAGEMENT	1.00	1.00	1.00
DEVELOPMENT	1.00	1.00	1.00
CUMENT CONTROL	1.00	1.00	1.00
ELECTRICAL	1.00	1.00	1.00
ESTIMATING	1.00	1.00	1.00
AL MAINTENANCE	0.94	1.00	0.97
MANAGEMENT	1.00	1.00	1.00
GEMENT & ADMIN	1.00	1.00	1.00
EMENT ACCOUNTS	1.00	1.00	1.00
MECHANICAL	1.00	1.00	1.00
OPERATIVE	0.93	1.00	0.96
PLANNING	1.00	1.00	1.00
PROCESS	1.00	1.00	1.00
PROCUREMENT	1.00	1.00	1.00
ECT MANAGEMENT & ENVIRONMENT	0.96	1.00	0.98
SERVICES	1.00	1.00	1.00
TECHNICAL	0.50	1.00	0.67
LATORY AFFAIRS	0.62	1.00	0.77
Micro Avg	0.99	0.99	0.99
Macro Avg	0.96	0.99	0.97
Weighted Avg	0.99	0.99	0.99
Samples Avg	0.94	0.94	0.94

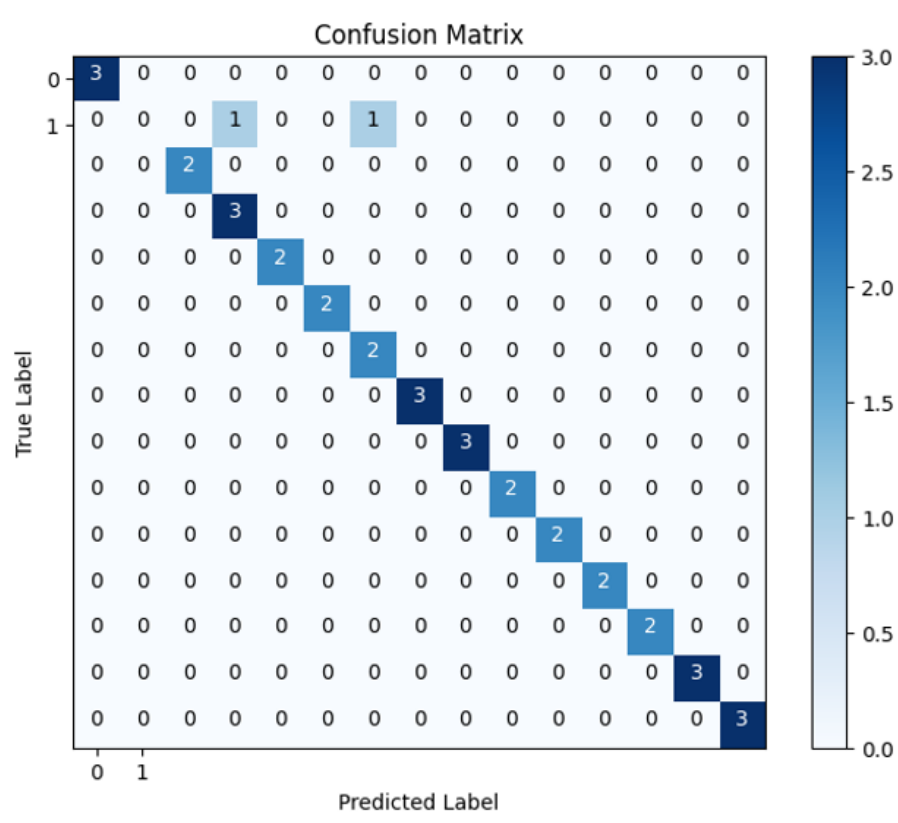


Figure 17: Confusion Metrics of Model 2: Diagonal depicts accurate numbers

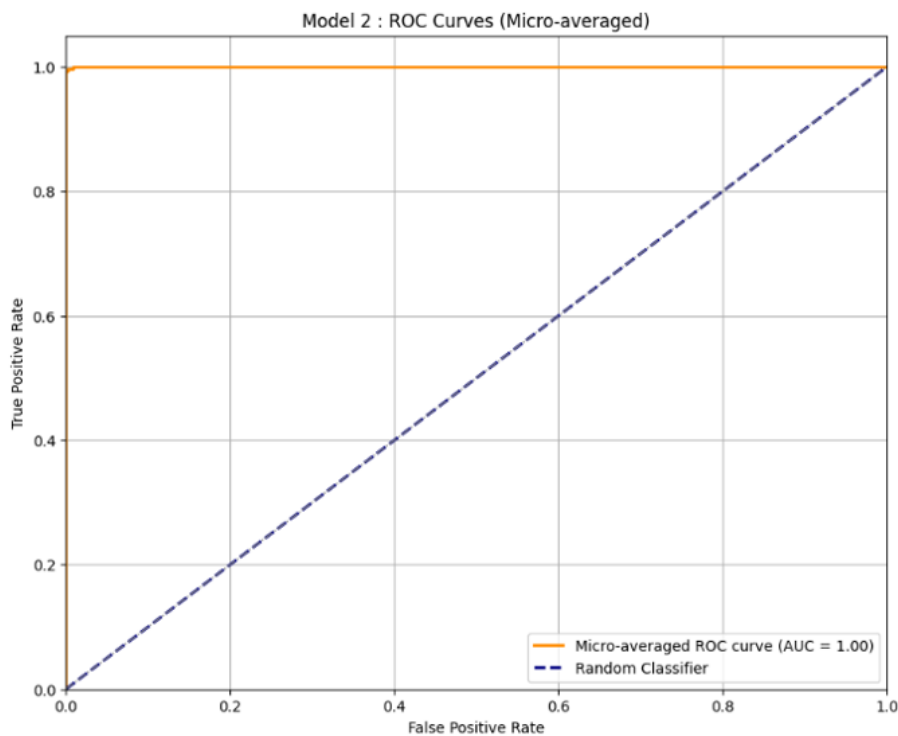


Figure 18: ROC AUC Curve for Tuned Model 2(L-shaped-good fit)

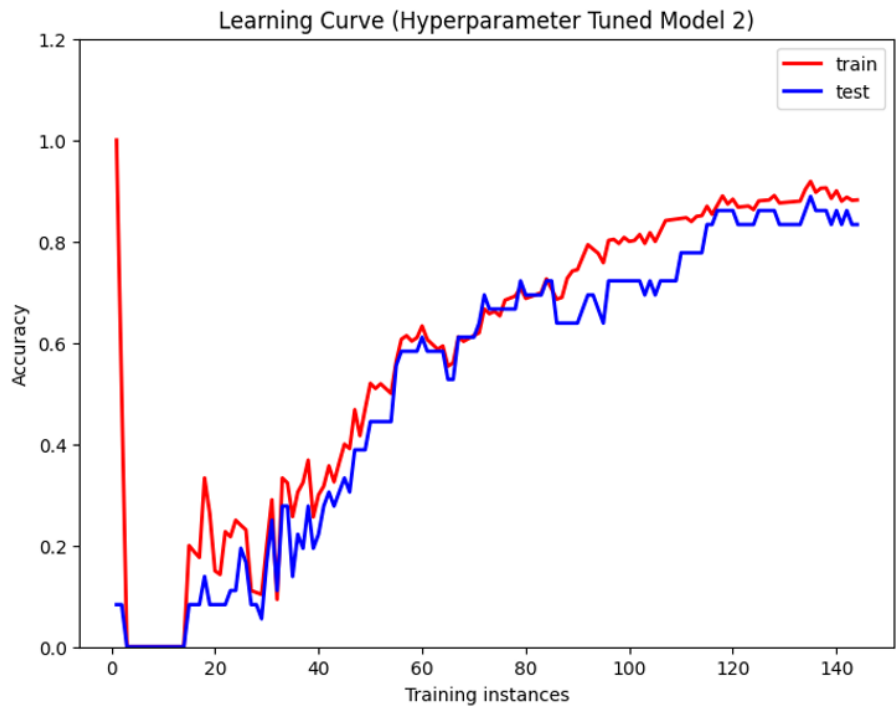


Figure 19: Well-fit Learning curve of Dataset on Tuned Model 2

Table 5: Model 2 Performance Evaluation and Results Comparison

MODEL 2 APPROACH (MultiOutputClassifier + RandomForestClassifier)		TEST ACCURACY	TRAIN ACCURACY	F1 SCORE
Model 2 on Oversampled Data		92%	100%	0.99
Hyperparameter Tuned Model 2		92%	100%	0.99

6.1.3 Model 3 Results: Upcoming Project Total Staff Count Prediction

Random Forest regressor model 3, designed to predict the total staff numbers for forthcoming projects, demonstrated remarkable performance. The mean squared error(MSE) of -1.75 obtained from cross-validation, indicated consistent precision when dealing with unseen data. Training phase findings minimal discrepancies with MSE of 0.23, RMSE of 0.48 and mean absolute error(MAE) of 0.15. The notably high R-squared value of 0.99 highlighted the model’s capacity to elucidate almost all variations in the overall staff counts. Evaluation of the tuned model’s performance on test data showed resilience, with an MSE of 0.11, RMSE of **0.33**, and

MAE of 0.11, coupled with an exceptional R-squared value of **0.99** indicating good correlation between predicted and real values (Table 6). Both scatter and residual plots depicted the precision and impartiality of the model, as the forecasts were closely grouped around the diagonal line and residuals/error were near zero(Figure 20).

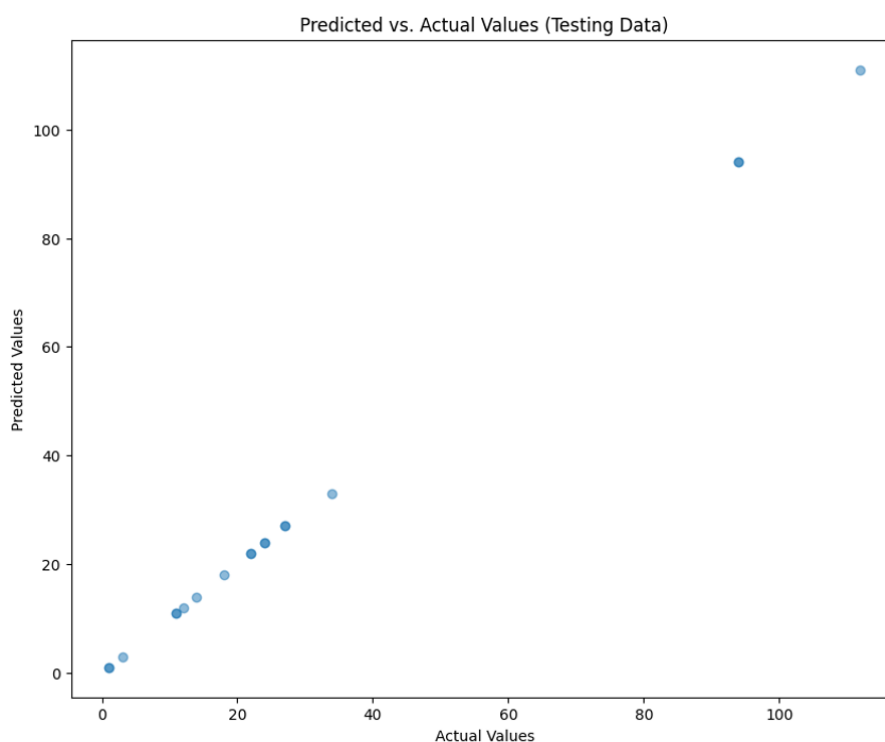


Figure 20: Scatter plot for Predicted vs Actual Data on Model 3

The learning curve for train and validation R-squared scores converged towards 1.00, indicating minimal errors and good fitting on train data(Figure 21). The trend also signifies that the model benefits from increasing sample data size and generalises well to new samples, suggesting a well-balanced fitted model with no trace of underfitting or overfitting issue. Model 3's robustness makes it a good predictor for new project's required staff count.

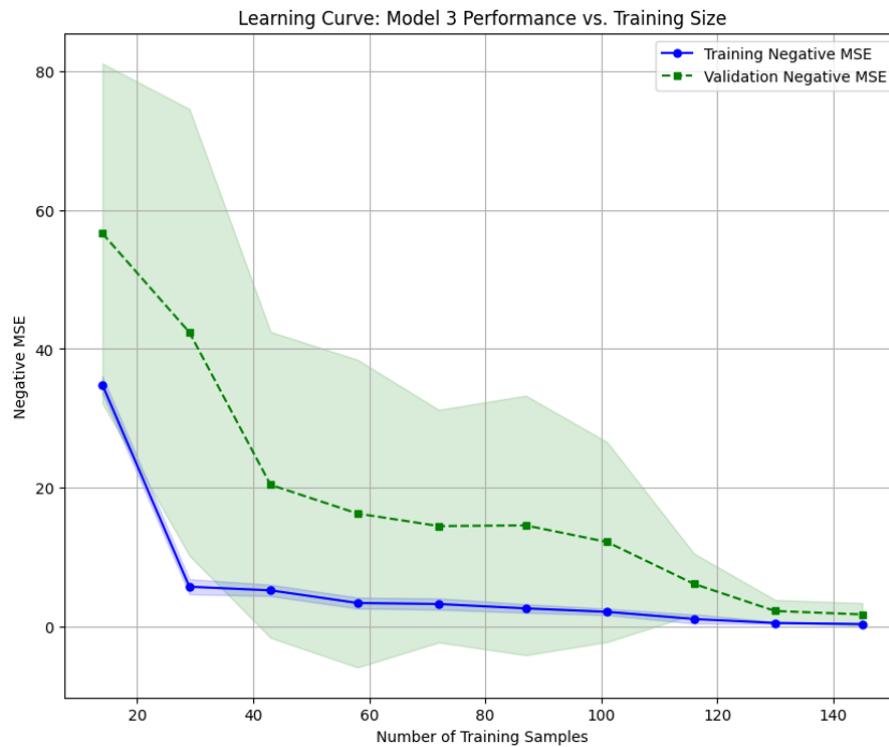


Figure 21: Learning curve of oversampled dataset on Tuned Model 3

Table 6: Model 3 Performance Evaluation and Results Comparison

MODEL 3 AP- PROACH	RMSE (Test)	MAE (Test)	Adjusted R-Squared (Test)	RMSE (Train)	MAE (Train)	Adjusted R-Squared (Train)
Model 3 on Over- sampled Data	0.33	0.11	1.00	0.48	0.14	0.99
Hyperparameter Tuned Model 3	0.23	0.55	1.00	0.47	0.13	0.99

6.1.4 Model 4 Results: Multi-Output Classifier for Department-wise Employee Allocation

Model 4 utilizes a multi-output classifier, employing a Random Forest Classifier, to allocate employees among different departments for an upcoming project. The model achieved an outstanding training accuracy of **100%**, indicating robust generalization. Notably, the accuracy of classifying the 'n_ARCHITECTURAL' department staff was highlighted in the confusion matrix (Figure 22). While the testing accuracy reached **99%** (Table 7), it suggested good performance with minor over-fitting attributed to limited department figures.

Examination of the learning curve revealed flawless training accuracy, with validation accuracy gradually improving as the sample size increased, starting from a lower point (Figure 23). This trend indicates the model's capacity to learn more effectively with additional data. Nonetheless, the need for more department-specific data is evident to enhance the overall performance of the model. Despite choosing not to oversample the data further to prevent data manipulation, acquiring additional data would offer deeper insights into the model's behavior and improve its effectiveness.

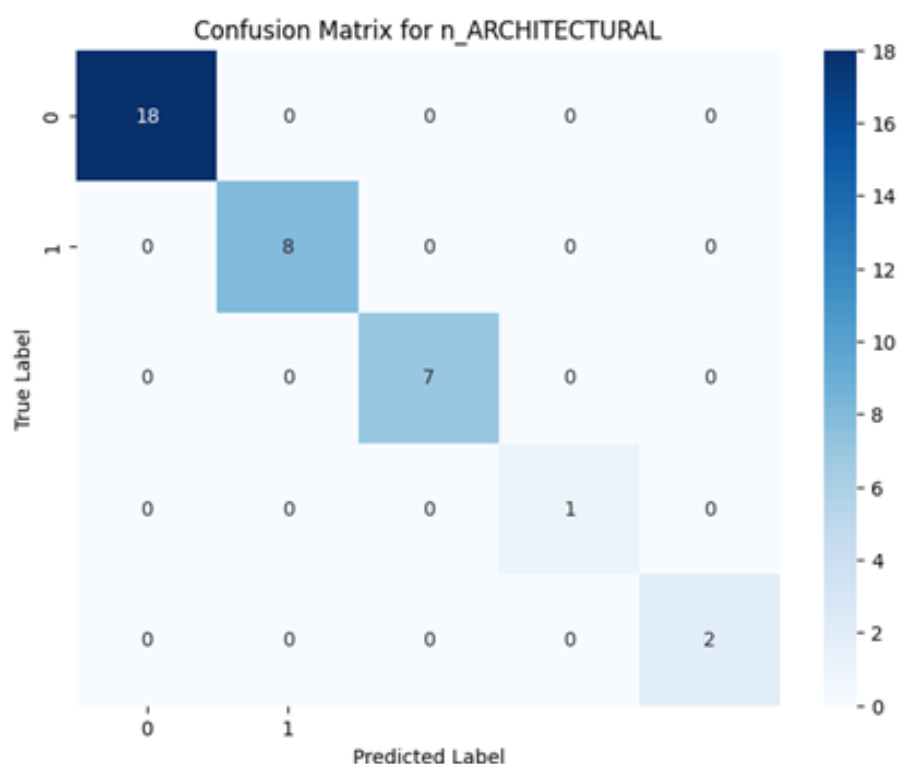


Figure 22: Confusion Matrix for one of the department-level employee counts

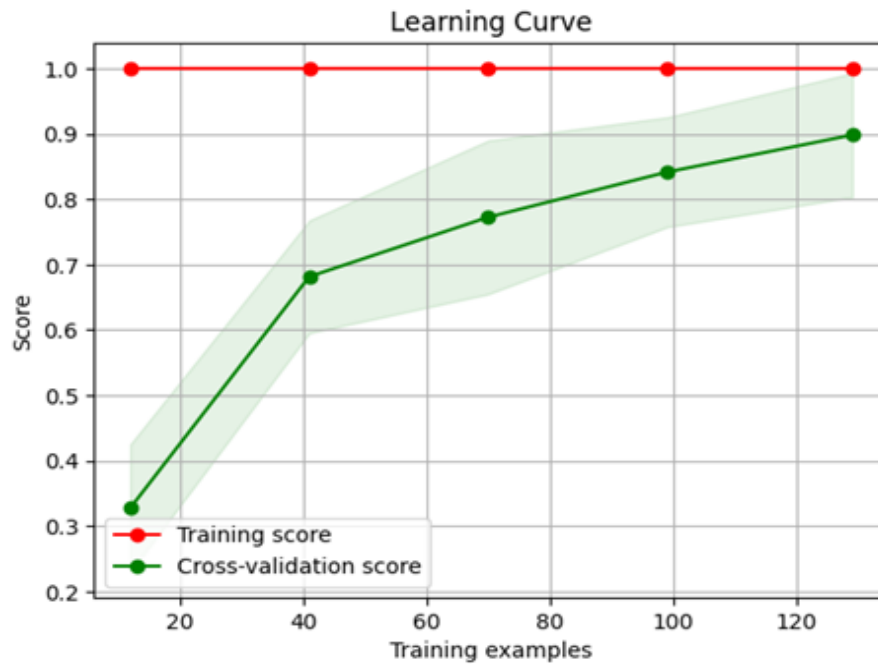


Figure 23: Improved Learning curve of oversampled dataset on Tuned Model 4

Table 7: Model 4 Performance Evaluation and Results Comparison

MODEL 4 APPROACH (MultiOutputClassifier + RandomForestClassifier)	TEST ACCURACY	TRAIN ACCURACY	F1 SCORE
Model 2 on Oversampled Data	99.4%	100%	0.99
Hyperparameter Tuned Model 2	99.5%	100%	0.99

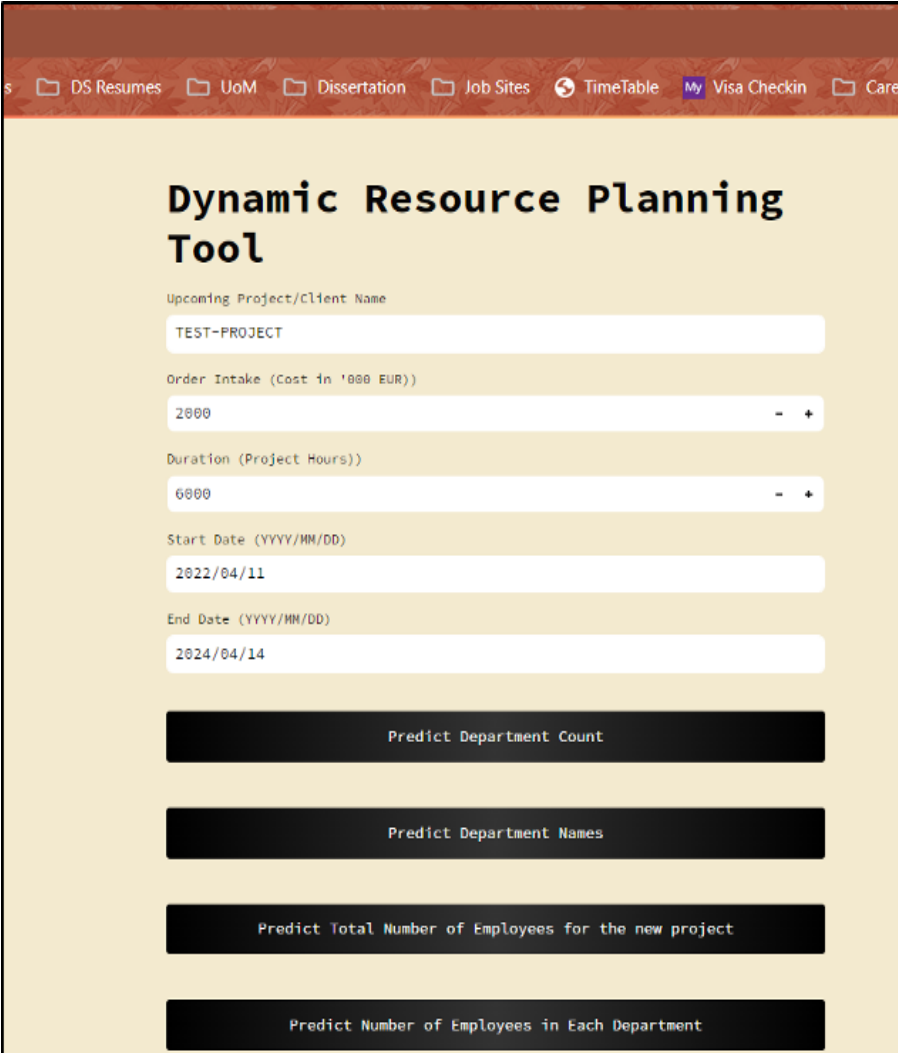
6.2 Model Deployment

The pivotal deliverable of our project involved the development of a resource planning tool (Figure 24) enhanced by Streamlit for deploying models. By employing thorough design and execution, we effectively incorporated machine learning models into a user-friendly interface, thereby improving accessibility and user interaction. The backend logic of the tool was encapsulated in the streamlit.pt file, enabling smooth integration of our models into an intuitive interface.

Our resource planning tool/UI empowers users to interact effortlessly with predictive insights by inserting project details into the input fields. Various buttons incorporated in the dashboard

trigger a distinct functionality regarding the 4 models, ranging from forecasting department count to department names and total employee figures and forest for a future construction client/project.

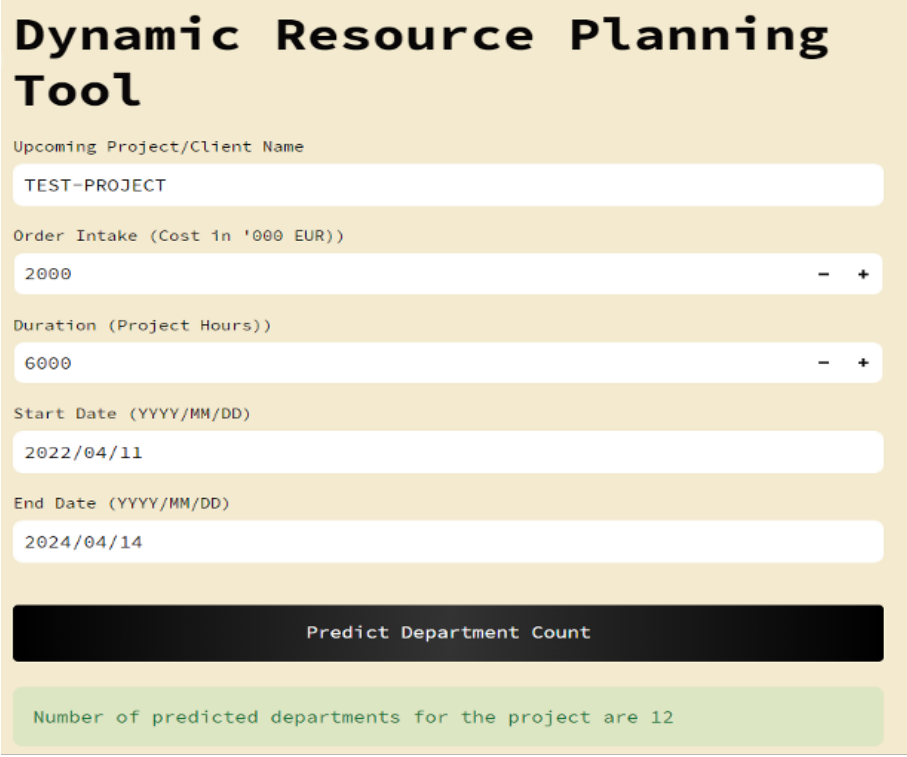
Streamlit's straightforward deployment process facilitates easy sharing of our application, whether locally or on cloud platforms. Overall, User Interface enhanced the usability of our resource planning tool.



The screenshot shows a web browser window with a red title bar. The browser's address bar and tabs are visible at the top. The main content area has a light beige background and is titled "Dynamic Resource Planning Tool" in bold black text. Below the title, there are five input fields with labels: "Upcoming Project/Client Name" (containing "TEST-PROJECT"), "Order Intake (Cost in '000 EUR)" (containing "2000"), "Duration (Project Hours)" (containing "6000"), "Start Date (YYYY/MM/DD)" (containing "2022/04/11"), and "End Date (YYYY/MM/DD)" (containing "2024/04/14"). Each of the last three input fields has a minus sign on the left and a plus sign on the right. Below the input fields are four black buttons with white text, stacked vertically: "Predict Department Count", "Predict Department Names", "Predict Total Number of Employees for the new project", and "Predict Number of Employees in Each Department".

Figure 24: Deployed Resource Planning Tool Running on Localhost

1. Predict Department Count Button



Dynamic Resource Planning Tool

Upcoming Project/Client Name
TEST-PROJECT

Order Intake (Cost in '000 EUR)
2000 - +

Duration (Project Hours)
6000 - +

Start Date (YYYY/MM/DD)
2022/04/11


End Date (YYYY/MM/DD)
2024/04/14

Predict Department Count

Number of predicted departments for the project are 12

Figure 25: Input Fields and Button-1 with Model-1 Prediction Display

2. Predict Department Names Button



Predict Department Names

Predicted department names are:

1. ARCHITECTURAL
2. COMMERCIAL
3. CONSULTING
4. DESIGN MANAGEMENT
5. DOCUMENT CONTROL
6. ELECTRICAL
7. INDUSTRIAL MAINTENANCE
8. MANAGEMENT ACCOUNTS
9. MECHANICAL
10. OPERATIVE
11. PROJECT MANAGEMENT
12. QUALITY, SAFETY & ENVIRONMENT

Figure 26: Button-2 with Model-2 Results Display

3. Predict Employee count for the new project button.

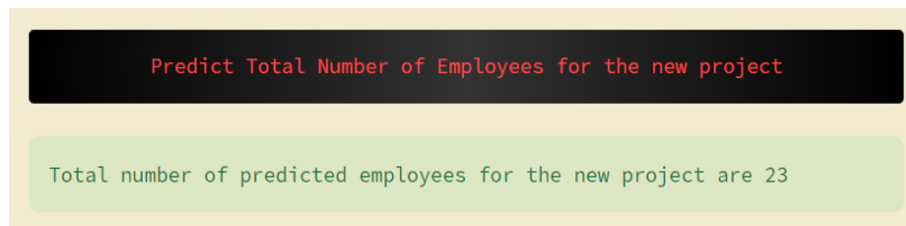


Figure 27: Button-3 with Model-3 Results Display

4. Predict Number of Employees per department button

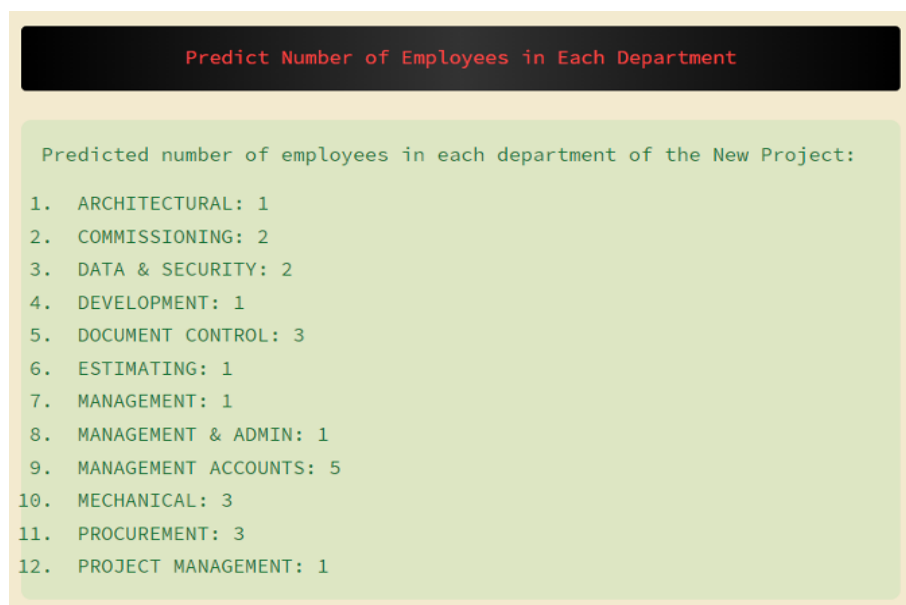


Figure 28: Button-3 with Model-3 Results Display

5. Display Resource Planning Summary of Project

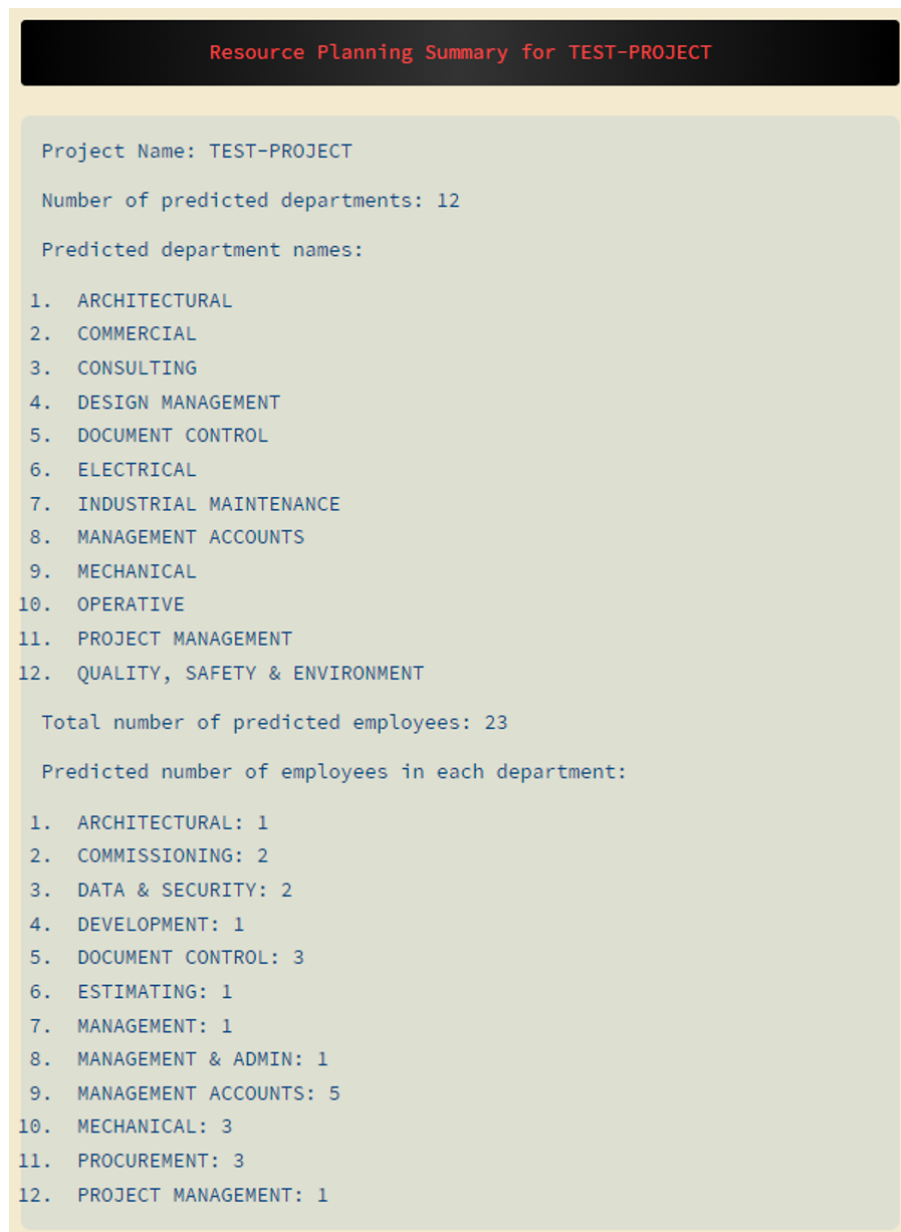


Figure 29: Button-4 Displays the Resource Planning Summary of New Project

Chapter 7

Challenges and Limitations

This project encountered several challenges and constraints that has a substantial influence on its methodology and outcomes.

Scarcity of Data: One prominent challenge faced during the project revolved around the insufficiency of accessible data. Despite being able to access employee timesheet data spanning three years from 2021 to 2023, only a maximum of 42 unique projects could be retrieved. This scarcity significantly diminished the pool of training data and impeded the model's ability to generalize.

Limited Data Features: Another significant constraint arose from the lack of essential data attributes that could enhance the model's predictive capacities. Attributes like project location indicators, onsite and offsite staff indicators, subcontractor information were missing from the dataset. The inclusion of these attributes could have offered valuable insights into staffing dynamics and project resource allocation.

Data Imbalance and Risk of Overfitting: Handling the data imbalance through oversampling techniques heightened the risk of overfitting, especially on a small dataset. Despite attempts to alleviate the risk, the models encountered difficulties during training phase. Further fine-tuning of the model hyperparameters were imperative to tackle overfitting and elevate performance.

Despite these challenges, optimizing the hyperparameters proved to be essential in addressing

overfitting and enhancing model performance. Moving forward, addressing these limitations will involve expanding the dataset size, integrating additional features, and refining the models to enhance the predictive precision.

Chapter 8

Recommendation and Future work

1. Enhancing Model Generalization:

Given the constraints of limited data, it's imperative to acknowledge the potential limitations in the model's ability to generalize predictions on unseen data. The current dataset encompasses specific ranges of project attributes such as cost, duration, and hours. To bolster the model's predictive accuracy and robustness, it's recommended to augment the dataset with a more diverse range of values. This can be achieved through acquiring additional data from various sources or by implementing data augmentation techniques.

2. Expansion into Predicting Job Titles and Departments:

To enrich the project's scope and utility, there's a promising opportunity to expand the model's capabilities beyond its current focus. In addition to predicting project attributes, consideration can be given to predicting job titles and department allocations associated with each project. This extension would provide stakeholders with more comprehensive insights into project dynamics and resource allocation.

3. Incorporation of Employee Allocation System:

A notable avenue for further refinement involves the integration of an employee allocation system within the project framework. Such a system would facilitate the efficient allocation of existing employees to new projects based on their schedules, availability, and skill sets. By

closely monitoring employee schedules and project demands, this system can optimize resource allocation, streamline project execution, and enhance overall organizational efficiency.

These recommendations and future directions aim to not only address existing limitations but also to propel the project towards greater effectiveness and utility. Continued efforts in these areas will contribute to the refinement and evolution of the project, ultimately fostering its continued success and impact within the organization.

Chapter 9

Conclusion

The collaborative endeavour involving Equans, a prominent construction company, and our project team P18 has resulted in the development of a comprehensive resource planning framework that is poised to revolutionize operational dynamics within the organisation. By incorporating sophisticated predictive models and user-friendly interfaces, Equans is anticipated to gain substantial advantages across various aspects of project management and operations. The initial phase of project planning established the foundation for subsequent data collection and analysis efforts, ensuring that all analytical work was firmly rooted in a thorough understanding of Equans' operational environment.

In addressing challenges such as data scarcity, limited features, and data imbalance directly, our project navigated intricate issues to provide robust solutions such as handling imbalance with oversampling techniques like RandomOverSampler . Despite facing constraints, fine-tuning random forest regression and classification model hyperparameters proved instrumental in enhancing predictive accuracy and mitigating overfitting risks. Moving forward, expansion of the dataset, incorporation of additional features, and refinement of models will be pivotal in further bolstering model generalization and predictive precision.

Impact on Industry partner operations Through the culmination of our modelling efforts bore witness to the unveiling of results that reverberated with implications for Equans' operational landscape. Armed with predictive models exhibiting robustness and accuracy, Equans

now possesses an arsenal of tools for informed decision-making in resource allocation and project management. Foremost impact is cost optimization, where data-driven resource allocation is projected to yield notable savings and enhance financial performance. Additionally, the project promises improvements in project timelines through enhanced staff allocation strategies, leading to reduced delays in project schedules. Furthermore, the automation of resource planning and allocation tasks is expected to alleviate manual workloads, thereby enhancing operational efficiency and workforce productivity.

Moreover, the intuitive Resource Planning Tool UI developed, promises to enhance user experience and amplifies the impact of our data driven analysis. By providing stakeholders with a user-friendly interface, we ensure that the insights gleaned from the models can be readily translated into actionable strategies, thereby maximizing their utility and relevance in real-world scenarios.

Looking ahead, our recommendations for future work offer a roadmap for further refinement and innovation. By expanding the dataset, augmenting predictive capabilities, and integrating additional features, Equans can unlock new avenues for efficiency, agility, and sustainable growth in resource planning and project management.

In conclusion, the collaborative resource planning project between Equans and our team is poised to drive significant operational enhancements and resource allocation strategies. By leveraging dynamic resource allocation strategy, Equans projects to achieve cost optimisation, improved project timelines and reduction in manual workload through interactive user-interface. The holistic solutions provided through this project position Equans for sustained growth and competitive advantage in the dynamic landscape of construction-based project management.

References

- Al-Alawi, Adel Ismail and Albuainain, Muneera Salem (2024). “Machine Learning in Human Resource Analytics: Promotion Classification using Data Balancing Techniques”. In: *2024 ASU International Conference in Emerging Technologies for Sustainability and Intelligent Systems (ICETSYS)*, pp. 1–5. DOI: 10.1109/ICETSYS61505.2024.10459566.
- Almajid, Adi Sakti (2021). “Multilayer Perceptron Optimization on Imbalanced Data Using SVM-SMOTE and One-Hot Encoding for Credit Card Default Prediction”. In: *Journal of Advances in Information Systems and Technology* 3.2. DOI: 10.15294/jaist.v3i2.57061. URL: <https://doi.org/10.15294/jaist.v3i2.57061>.
- Chaudhari, Archana, Khandelwal, Harshada, Khan, Ali, Kurade, Omkar, and Kolekar, Amit (2023). “Mineral Prediction Using Random Forest Classifier”. In: *2023 14th International Conference on Computing Communication and Networking Technologies (ICCCNT)*, pp. 1–6. DOI: 10.1109/ICCCNT56998.2023.10306952.
- Chiang, Hui Yi and Lin, Bertrand M. T. (2020). “A Decision Model for Human Resource Allocation in Project Management of Software Development”. In: *IEEE Access* 8, pp. 38073–38081. DOI: 10.1109/ACCESS.2020.2975829.
- Gao, Xiang, Wen, Junhao, Zhang, Cheng, et al. (2019). “An improved random forest algorithm for predicting employee turnover”. In: *Mathematical Problems in Engineering* 2019.
- Jackins, V., Vimal, S., Kaliappan, M., et al. (2021). “AI-based smart prediction of clinical disease using random forest classifier and Naive Bayes”. In: *Journal of Supercomputing* 77,

- pp. 5198–5219. DOI: 10 . 1007 / s11227 – 020 – 03481 – x. URL: [https://doi.org/10 . 1007/s11227-020-03481-x](https://doi.org/10.1007/s11227-020-03481-x).
- Jaffar, Zarmina, Noor, Waheed, and Kanwal, Zartash (2019). “Predictive human resource analytics using data mining classification techniques”. In: *Int. J. Comput* 32.1, pp. 9–20.
- Lee, Hansoo, Jung, Seunghyan, Kim, Minseok, and Kim, Sungshin (2017). “Synthetic minority over-sampling technique based on fuzzy c-means clustering for imbalanced data”. In: *2017 International Conference on Fuzzy Theory and Its Applications (iFUZZY)*. IEEE, pp. 1–6.
- Lu, Wei (2022). “Human Resource Optimization Allocation Technology Based on Neural Network and Its Algorithm Implementation”. In: *2022 2nd International Conference on Networking, Communications and Information Technology (NetCIT)*, pp. 467–470. DOI: 10 . 1109/NetCIT57419.2022.00116.
- Ma, Chenxiang, Zhang, Shouxin, Zhuo, Jihuan, Liu, Yang, and Zhou, Yuewu (2022). “Research on Project Group Human Resource Allocation of Construction Enterprises Based on Decision Tree Algorithm”. In: *2022 2nd International Conference on Networking, Communications and Information Technology (NetCIT)*, pp. 193–196. DOI: 10 . 1109/NetCIT57419 . 2022 . 00055.
- Mundra, Shikha, Vijay, Shounak, Mundra, Ankit, Gupta, Punit, Goyal, Mayank Kumar, Kaur, Mandeep, Khaitan, Supriya, and Rajpoot, Abha Kiran (2022). “Classification of Imbalanced Medical Data: An Empirical Study of Machine Learning Approaches”. In: *Journal of Intelligent Fuzzy Systems* 43.2, pp. 1933–1946. DOI: 10 . 3233/JIFS-219294.
- Muntu, D, Setyawati, R, Riantini, LS, and Ichsan, M (2021). “Effect of human resources management and advances to improve construction project performance”. In: *Physics and Chemistry of the Earth, Parts A/B/C* 122, p. 103000.
- Nakashima, Hayato, Arai, Ismail, and Fujikawa, Kazutoshi (2019). “Passenger Counter Based on Random Forest Regressor Using Drive Recorder and Sensors in Buses”. In: *2019 IEEE In-*

- ternational Conference on Pervasive Computing and Communications Workshops (PerCom Workshops)*, pp. 561–566. DOI: 10.1109/PERCOMW.2019.8730761.
- Nhita, Fhira, Adiwijaya, and Kurniawan, Isman (2023). “Performance and Statistical Evaluation of Three Sampling Approaches in Handling Binary Imbalanced Data Sets”. In: *2023 International Conference on Data Science and Its Applications (ICoDSA)*, pp. 420–425. DOI: 10.1109/ICoDSA58501.2023.10276805.
- Rao, N. Sri Sai Venkata Subba and Thangaraj, S. John Justin (2023). “Flight Ticket Prediction using Random Forest Regressor Compared with Decision Tree Regressor”. In: *2023 Eighth International Conference on Science Technology Engineering and Mathematics (ICONSTEM)*, pp. 1–5. DOI: 10.1109/ICONSTEM56934.2023.10142260.
- Suparyati, Utami, Ema, and Muhammad, Alva Hendi (2022). “Lumpy Skin Disease Prediction Based on Meteorological and Geospatial Features using Random Forest Algorithm with Hyperparameter Tuning”. In: *2022 5th International Conference on Information and Communications Technology (ICOIACT)*, pp. 99–104. DOI: 10.1109/ICOIACT55506.2022.9971807.
- Vaidyanathan, S., Arumugam, C., Jaganathan, K., et al. (2024). “LeSS agile projects: a machine learning driven empirical model to predict the human resource allocation”. In: *International Journal of Information Technology* 16, pp. 861–870. DOI: 10.1007/s41870-023-01513-2.
- Wang, Zhouhou (2022). “Enterprise Human Resource Allocation Optimization Model Based on Improved Particle Swarm Optimization Algorithm”. In: *Explorations in Pattern Recognition and Computer Vision for Industry 4.0* 2022. DOI: 10.1155/2022/1789276. URL: <https://doi.org/10.1155/2022/1789276>.
- Xu, Yanlan (2022). “Design of human resource allocation algorithm based on improved random forest”. In: *ICASIT 2021: 2021 International Conference on Aviation Safety and Information Technology*, pp. 656–661. DOI: 10.1145/3510858.3511353.

Xu, Yuan, Park, Yongshin, Park, Ju Dong, and Sun, Bora (2023). “Predicting Nurse Turnover for Highly Imbalanced Data Using the Synthetic Minority Over-Sampling Technique and Machine Learning Algorithms”. In: *Healthcare*. Vol. 11. 24. MDPI, p. 3173.

Appendix A

Appendix

A.1 Meeting Reports

1st Feb 2024: Project Kick-off Meeting

- Conducted an initial face-to-face meeting with industry partners to establish rapport and align on project goals.
- Provided an overview of the company's background, its operations, and its role within the project.
- Delivered a comprehensive briefing on the project scope, objectives, and anticipated outcomes.
- Analyzed sample data to gain insights into its format and structure, laying the groundwork for subsequent data analysis.

7th Feb 2024: Group Member Meeting

- Facilitated introductions among team members to foster collaboration and camaraderie.
- Reviewed the project brief to ensure a shared understanding among team members.
- Clarified expected outcomes and milestones to guide project progress.
- Analyzed the dataset structure and identified key data types to inform subsequent analysis.
- Assigned tasks for exploratory data analysis (EDA) and modeling on sample data to maximize efficiency and collaboration.

14th Feb 2024: Task Follow-up and Model Discussion

- Conducted a follow-up session to review progress on assigned tasks and discuss individual outcomes.
- Engaged in a detailed discussion on the strengths and weaknesses of various machine learning models.
- Prepared for presenting initial findings to industry partners, ensuring alignment and coherence in the presentation of results.

21st Feb 2024: Online Meeting with Industry Partners

- Conducted a virtual presentation of initial exploratory data analysis (EDA) findings to industry partners.
- Presented and discussed potential workflows for subsequent project phases.
- Explored the feasibility of obtaining real-world data at the earliest opportunity.
- Facilitated a discussion on potential challenges and corresponding solutions to ensure project success.

27th Feb 2024: Dataset Acquisition and Initial Exploration

- Procured the actual dataset from industry partners for further analysis.
- Analyzed the format and structure of the acquired data to inform subsequent steps.
- Assigned tasks for exploring the initial dataset to uncover insights and patterns.
- Prepared materials for the upcoming poster presentation to showcase project progress and initial findings.

4th March 2024: Poster Presentation Preparation

- Finalized content to be included in the poster presentation, incorporating conclusions from various analyses.
- Designed the poster to effectively communicate project objectives, methodologies, and key results.
- Discussed and refined the proposed workflow based on insights gained from data exploration.

7th March 2024: Poster Presentation and Mentor Feedback

- Participated in the poster presentation event, engaging with industry and project mentors to discuss project details.
- Received valuable feedback from mentors to refine project direction and improve outcomes.
- Leveraged interactions with other teams and their mentors to gather insights and exchange knowledge for mutual benefit.

15th March 2024: Research and Task Allocation

- Initiated research into the breakdown of the structure and procedural flow of data through different modeling stages.
- Commenced exploration of potential models suitable for various stages of the project.
- Facilitated task division among team members to commence work on data preprocessing and modeling.

22nd March 2024: Data Preprocessing and Model Testing

- Collaborated to integrate diverse preprocessing steps contributed by team members to streamline data preparation.
- Consolidated and merged datasets, extracting crucial insights and relevant information.
- Experimented with a range of models to identify optimal solutions for the project's objectives.

2nd April 2024: Model Evaluation and Report Preparation

- Conducted comprehensive assessment to identify strengths and weaknesses of various models under consideration.
- Finalized the dataflow process to ensure smooth progression through different stages of the project.
- Initiated groundwork for report preparation, outlining key sections and structuring content.

10th April 2024: Code Compilation and Report Planning

- Consolidated all code contributions from team members to ensure uniformity and compatibility.
- Conducted rigorous testing and evaluation of the developed models to validate their performance.

- Allocated tasks and responsibilities among team members for efficient progress on report writing.

17th April 2024: Deployment Preparation and Report Refinement

- Commenced work on deploying the proposed system to prepare for real-world implementation.
- Continued refining exploratory data analysis (EDA) and preprocessing steps to enhance the quality of insights for the report.
- Prepared to present findings and results to industry mentors, ensuring alignment with project objectives and stakeholder expectations.

25th April 2024: Results Discussion with Industry Partner

- Conducted a meeting with the industry partner to review project results and finalize outcomes.
- Managed expectations by assessing the feasibility of project outputs and addressing any discrepancies.
- Discussed potential drawbacks of implemented models and proposed viable solutions to mitigate them effectively.

2nd May 2024: Final Report Preparation and Presentation Planning

- Dedicated efforts towards preparing the final project report for the impending submission deadline.
- Developed presentation slides, outlining key findings and insights to be communicated during the presentation.
- Divided responsibilities among team members for seamless coordination and execution during the presentation.
- Prepared final deliverables for submission, ensuring completeness and adherence to project requirements.

8th May 2024: Final Report and Presentation Submission

A.2 User Guide

We have curated a user guide for our industry partners, Equans, to access and setup the resource planning tool.

Environment Setup:

Step 1: Install a python IDE on your local system, preferably Visual Studio Code

Step 2: Open the Google Drive link provided, containing the code and data repository.

Step 3: Open the code repository folder on Visual Studio Code.

Step 4: Locate the “tmenv” file, this is a virtual environment preconfigured with all the necessary Python libraries installed. Set up this virtual environment to ensure compatibility.

Step 5: Open a new terminal by navigating to Terminal ↵ New Terminal in the VS Code menu.

Step 6: Navigate to the directory containing the ”tmenv” virtual environment.

Step 7: Activate the virtual environment by running the following command:

- For Windows:

```
css
```

[Copy code](#)

```
path\to\your\venv\Scripts\activate
```

- For macOS and Linux:

```
bash
```

[Copy code](#)

```
source path/to/your/venv/bin/activate
```

Figure 30: Commands to activate the environment

Step 8: After running the activation command, you will see the name of the virtual environment in your terminal prompt, indicating that it’s active.

Step 9: Now the user can use the virtual environment in VS Code. Any scripts you run will use the dependencies installed in this environment.

Running Resource Planning tool:

Step 10: Now on the same terminal : Run the following command to start the Streamlit application:

```
streamlit run streamlitapi.py
```

```
(tmenv) C:\Users\Kashish\1.Kashish\Manchester\Semester 2\ADS\ADS Code Repository>streamlit run streamlitapi.py
```

Figure 31: Commands to start the Streamlit application

Step 11: Now you will be redirected to the localhost web application.

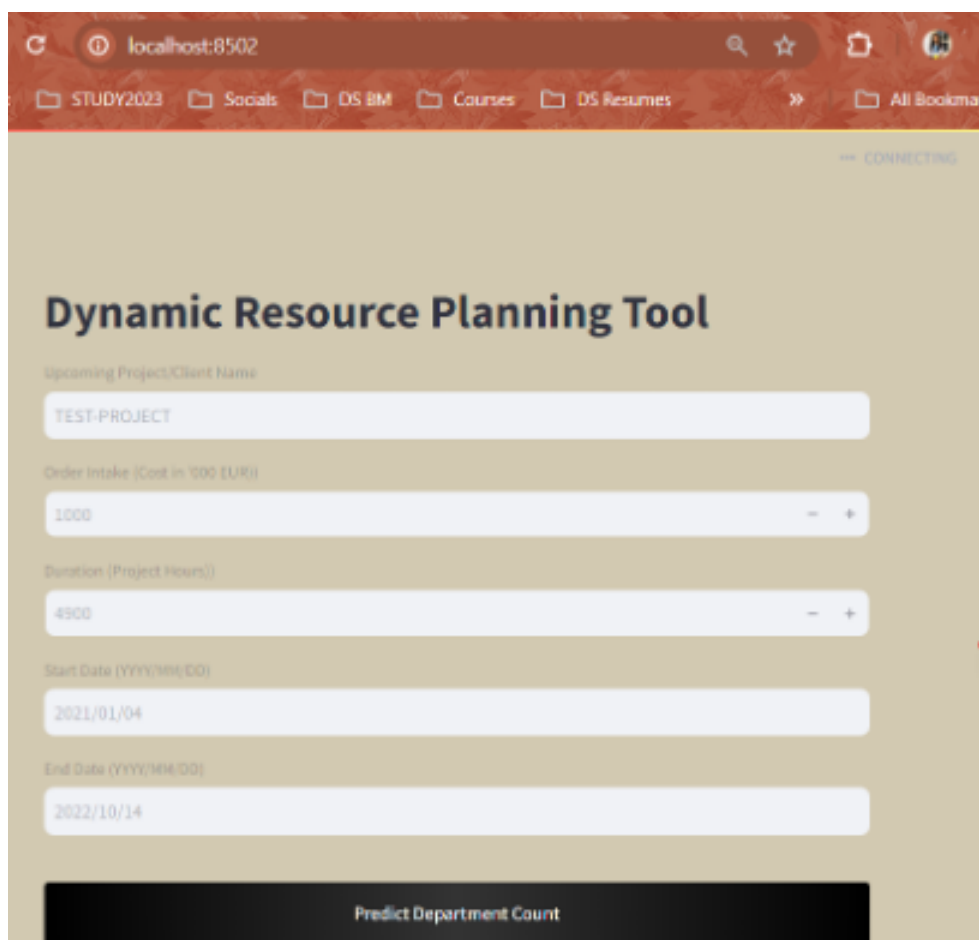


Figure 32: Webpage of the application

Step 12: Access the Resource Planning Tool by opening the provided localhost web application in your web browser.

Step 13: Enter project-specific input details into the designated field and click on each button to generate predictive insights.

Steps to enhance the model training for improved resource allocation predictions:

Populate or add additional data to the provided two excel sheets in the same format as the existing 42 projects. Ensure that both time-sheet data and cost data are included for comprehensive analysis and training of the models.

A.3 Code and Data Repository Links

[Google drive link to the code and data repository](#)

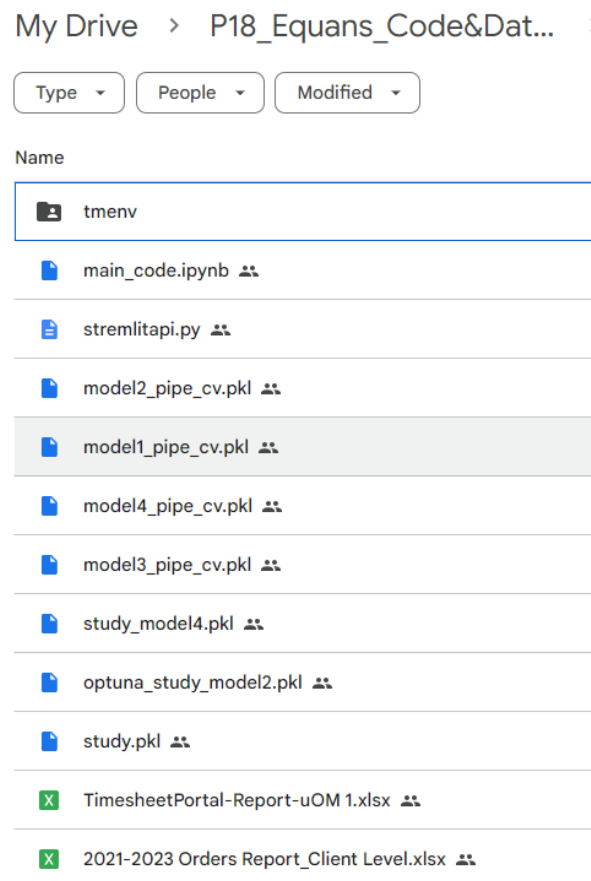


Figure 33: Repository Structure