9th North American Conference on Industrial Engineering and Operations Management
Washington D.C., USA, June 4-6, 2024
Publisher: IEOM Society International, USA DOI: 10.46254/NA09.20240142
Published: June 4, 2024

Predictive and Prescriptive Analytics Models to Support the Internal Audit Capacity Planning System of an IT Company

Patricio Alejandro Zamora Navarro, Rafael Alejandro Ruiz Miller, Roberto Serna Zuazua and José D. Morcillo, Ph.D.
Escuela de Ingeniería y Tecnologías, Universidad de Monterrey, México
patricio.zamora@udem.edu, rafael.ruizm@udem.edu , roberto.serna@udem.edu, jose.morcillo@udem.edu.mx

Abstract

The internal audits of a company are a fundamental process for quality assurance and proper compliance with different processes within a company. For this reason, it is also important that companies are able to adequately plan their implementation; But it often happens that companies do not have the necessary tools to make these adequate plans. Within the topic of planning, in this context, the factors of effort of internal auditors in work hours and capacity (number of audits) are taken into account. These aspects are vital for the organization of times and audits over a period of time, since, if an adequate calculation is not made, it can significantly affect planning and trigger different consequences for the company and its processes. Given the above, it was proposed to develop and implement predictive and prescriptive analytics models to facilitate the planning and optimization of internal audits in terms of hours and number of audits (2 models); improving an (existing) digital planning tool. Historical audit information was obtained, worked and processed within an IT company. Different Machine Learning Python libraries were applied to carry out the modeling process, which include: PyCaret, H20, FLAML, TPOT, LazyRegressor, among others.Various evaluation metrics such as coefficient of determination (R2), root mean square error (RMSE), mean square error (MSE), and mean absolute percentage error (MAPE) were used to evaluate the models. The best results were obtained by Random Forest models, where R2 metrics of 94.40% and a MAPE of 6.26% were obtained for the hours of effort model; An R2 of 90.49% and MAPE of 6.77% were obtained for the capacity model.

Keywords

Audit Planning, Effort, Prediction Model, Machine Learning, Prescriptive Analytics, Capacity, Optimization .

1. Introduction

The aspect of quality has been a highly relevant topic for companies in recent years. It has been found that adequate quality management in companies leads to better success indicators, such as: product or service quality, customer satisfaction, operational and economic performance (Sila, et al., 2020). Internal audits are one of the greatest tools that help ensure quality in the processes of a company.

An audit is defined as “the process of reviewing and verifying the operation of a company, either comprehensively or in a specific aspect; which is carried out with a precise and detailed methodology to identify incidents, eventualities, diagnoses, panoramas and points of opportunity” (Auren México, 2021). Internal audits help ensure that a company's processes are done correctly and appropriately, therefore, ensuring better quality in them and in the results that come from them.

From the above lies the importance of a company's internal audit process being effective and efficient. One of the key aspects for a successful execution of internal audits is their planning. According to Selvarajah (2019), proper internal audit planning ensures the efficiency, effectiveness and quality of internal audit processes. Otherwise, audits do not adequately fulfill their purpose and end up slowing down a company's improvement processes.

In this project, the current internal audit planning system of a technological and digital solutions company in Mexico will be analyzed and improved. The planning techniques and tools analyzed and developed along this project are quantified in terms of the effort of the internal auditors in working hours, internally referenced as "Audit Effort", and in capacity (number of audits), which is referenced as "Total Projects".

It is worth mentioning that this project provides continuity to the previous project: Improvement of the internal audit planning system in an IT company through predictive analytics. The most relevant results of this project include:

- A forecasting model was developed that is capable of predicting the number of hours required (Actual Effort) to perform a certain number of audits in a two-week period (referred to as Sprint). This model has an average margin of error (MAPE) of 8.27% and R^2 of 96%.
- A digital automatic Actual Effort forecasting tool was developed that serves as support for the company's audit planning system.

1.1 Objectives

Develop and implement predictive and prescriptive analytics models to facilitate the planning and optimization of internal audits in terms of hours ("Actual Effort") and number of audits ("Total Projects"); in such a way that the digital management tool is improved and a data visualization system is generated.

To achieve this main objective, the following particular objectives must be met:

- Implement prediction models that are capable of predicting team capacity ("Total Projects") per Sprint, with a maximum margin of error (MAPE) of 15% and a minimum R^2 of 85%.
- Re-train the "Actual Effort" prediction models with more updated information, keeping the MAPE below 15%.
- Develop and implement prescriptive analytics models that are capable of maximizing "Total Projects" per Sprint for the auditor team.
- Application of Machine Learning to improve and extend the digital tool by adding short and long-term predictive models, and optimization models for "Total Projects".
- Develop a data visualization system that allows monitoring KPIs and statistical information with historical and real-time data.

2. Literature Review

Calculating the capacity required to perform certain audits is essential for effective planning and implementation of the internal audit system. The main objective of our literature research was to learn from the experience and practices of other researchers and authors, as well as to become familiar with the algorithms and metrics derived from previous studies similar to ours. The research phase comprehensively covered two main aspects related to capacity and capacity planning: its interaction with optimization models and prescriptive analysis, and its connection with the characteristics and factors of team members or workers. This project aims to support the planning system by prescriptively estimating the capacity of the audit team, as well as the hours required to complete a specific number of audits in a work cycle (sprint). One of the main challenges we faced was the lack of information available in professional articles that specifically address the prediction of hours of effort in audits. Some of the highlights identified in certain studies include prescriptive analytics in capacity planning, equipment capacity planning, integration of prescriptive and predictive analytics, among others.

The first aspect of the review focused on capacity planning and prescriptive analysis. Despite initial challenges in finding specific literature on capacity in audits using prescriptive analytics, expansion of search terms provided promising results. The definition of "capacity" was highlighted and the prevalence of topics such as production capacity planning and equipment capacity optimization were illustrated (Tektas et al. 2022; Nasser and Mitschele-Thiel 2022; Geng and Jiang 2007; Chang 2019; Ghoniem et al. 2016). In addition, the relevance of prescriptive analysis algorithms was emphasized, with mixed and integer programming models, stochastic programming and heuristic optimization being the most prominent among the reviewed documents.

In the current era, characterized by complexity and uncertainty, prescriptive analytics emerges as an indispensable tool for capacity optimization in various sectors. This discipline, which integrates machine learning models with advanced optimization techniques, allows organizations to not only understand and predict complex dynamics, but also formulate optimal and adaptive strategies.

Integer mixed linear programming is a pillar in this approach, providing a framework for making optimal decisions in scenarios that require a combination of integer and continuous variables. This technique is observed in energy capacity planning, where investments, maintenance and operations must be managed in volatile contexts (Tektas et al. 2022). It allows modeling and solving multifaceted problems, simultaneously considering multiple factors and constraints.

On the other hand, heuristic optimization offers fast and effective solutions for large-scale problems, typical in the optimization of coverage and capacity in telecommunications. These techniques allow efficient and real-time adjustments, essential in an environment where speed and adaptability are critical (Chang, L. 2022)

Furthermore, Integer Programming and Stochastic Programming Models are fundamental in human resource planning and operations management. These models are especially useful for making sequential decisions under uncertainty, as seen in capacity planning for semiconductor manufacturing (Geng and Jiang, 2007) and (Ghoniem et al. 2016).

The integration of these optimization models with machine learning techniques, such as neural networks and decision trees, represents a significant advance. These machine learning models feed optimization systems with accurate forecasts and relevant variables, improving the quality and efficiency of strategic decisions. This combined approach is particularly powerful in contexts such as energy capacity planning and resource allocation in hospitals, where decisions must be both strategic and adaptive.

The second aspect involves the analysis of capacity with the characteristics of the workers. Although obstacles were encountered in locating studies that specifically addressed both topics, the adaptation of search terms allowed for a successful collection of relevant literature. A correlation was observed between “ability” and “performance,” with a significant focus on worker performance. The articles reviewed not only addressed individual and group performance, but also relevant characteristics and factors of workers that predict and evaluate performance using mainly Machine Learning techniques (McComb et al. 2017; GururamaSenthilvel et al. 2023; Prasad et al. 2023; Banu et al. 2020; Rivaldo and Nabella 2023; Srividya Prathiba 2023; van Geffen 2020). In addition, the variables used in these models were evaluated, highlighting those related to education, intelligence, work experience and age, and the metrics used to evaluate the effectiveness of the models were reviewed, with "accuracy" being the most prevalent.

Methods and models used in the studies were identified and analyzed, with regression analysis and the Decision Tree models predominating. In addition, the variables used in these models were evaluated, emphasizing those related to education, intelligence, work experience and age, and the metrics used to evaluate the effectiveness of the models were reviewed, with "accuracy" being the most prevalent.

There are various evaluation methods and algorithms used to make these estimates. These include neural networks, decision trees, reinforcement learning and algorithms such as Support Vector Machines (SVM), which have become essential tools for solving complex problems in various sectors.

Neural networks are particularly effective at capturing complex, non-linear relationships between variables. Goodfellow et al. (2016) mentions that these models are inspired by the human brain, composed of interconnected units called neurons, organized in layers. These networks can adapt to different contexts, learning from patterns and trends in historical data. However, its effectiveness depends on the quality and quantity of available data, and its implementation may require advanced technical knowledge. In the area of capacity planning, neural networks have been used to forecast electricity demand and optimize investments in energy capacity (Thammaboosadee and Wongpitak, 2018). In evaluating employee performance, they allow a range of variables to be analyzed to predict performance.

Decision trees are supervised learning algorithms used for both classification and regression. They are based on a hierarchical tree structure, where each internal node represents a characteristic and each branch a possible decision. They are particularly useful for understanding the logic behind decisions and for handling non-linear data (Fang and Jen-Hwa 2020). In design equipment optimization, decision trees help evaluate equipment characteristics and their relationship to equipment performance (Geffen, 2020).

Reinforcement learning, used in coverage and capacity optimization in telecommunications, is based on the idea of rewards and penalties to direct decision making (Naseer and Mitschele-Thiel, 2012). It is effective for dynamically adjusting configurations in environments. changing. On the other hand, SVM is a powerful method to find optimal

hyperplanes in high-dimensional spaces, maximizing the sparsity between classes (MathWorks, 2023). This technique has proven valuable in predicting audit outcomes and evaluating employee performance (GururamaSenthilvel, et al. 2023).

3. Methods

In this project, the ASUM-DM methodology will be implemented to carry out the prescriptive model and develop the tool within the planning of internal audits. ASUM-DM, an extension and refinement of CRISP-DM developed by IBM, is an iterative process for implementing analytics projects. This methodology allows us to remove phases from it depending on the nature of the project.This methodology is detailed in a more elaborate way below (only phases or stages that were applied to this particular project are elaborated):

1. Analyze-Design-Configure-Build

- Preparation level assessment

This stage involves assessing the organization's readiness for analytics, including data availability, technology, skills, and culture. It is verified whether the organization is ready to undertake the project and whether the prerequisites have been met.

- Business understanding: It consists of understanding the needs of the business and how the analytics project can satisfy those needs.
- Data understanding: It involves reviewing and understanding the available data, its quality, structure and relevance to the business problem.
- Infrastructure design and validation: This stage refers to the design of the technical infrastructure necessary to implement the analytical solution and its validation.
- Planning and establishment of environments: In this phase, the different environments necessary for development and implementation are planned and established, such as the development, testing and production environment.
- Data preparation: Here data cleaning, transformation, engineering, formatting and integration activities are performed to prepare it for modeling.
- Modeling: Selection, application and fine-tuning of algorithms and analytical techniques to generate models that answer business questions.
- Assessment: Review of the models created to ensure that they satisfy the business objectives.
- Analytical knowledge transfer: Share analysis results and models with interested parties.
- Define deployment plan: Here you define the plan to implement the analytical solution in the production environment.
- Operational test strategy design: Plan how models will be tested in an operational environment.
- Validation and testing: Test models and infrastructure before deployment.

2. Deployment

- Deployment: Implementation of the models in a production environment.
- Launch: This activity marks the transition of the project to a functioning service. The solution is made available to end users.

3. Operate and Optimize

- Monitoring: Continuous monitoring of model and infrastructure performance. This phase involves continually monitoring the performance and usefulness of the analytical solution.
- Operate, Optimize and Improve: This activity refers to the maintenance and continuous improvement of models and solutions based on feedback and observed performance.

4. Data Collection

Throughout this project, several versions of databases were granted to work with; This is because it was found in different cases that these databases had certain defects such as incorrect, incomplete information, etc. Eventually, a final version was obtained that had the most complete and correct information possible. Certain processes and adjustments were made to the database for use in subsequent stages. First, the database was filtered where only the projects or audits of auditors that belonged to the Mexico team would be left. Afterwards, a standard data cleaning and/or engineering process was carried out. By the end of the adjustments, the database had 3840 records.

Once the previous adjustments were made, it was decided that two sub-versions of the database would be generated: one for the Actual Effort model and another for the capacity model, since each one occupies different columns or variables of the database. These versions were obtained through the data grouping process. This technique generally helps to reduce high volatility problems of a database, which is the present situation of the database. In this case, the database records were grouped based on the Sprint variable; In other words, all records that had the same Sprint were grouped (the Sprint variable takes into consideration the current year and semester).

With the grouping of the database for the Actual Effort model, a new database was generated, which contains 78 records. The database variables were also reworked due to the grouping; the nature of the records changed and the original variables were no longer suitable for these records. Previously, each of the records represented a project or audit; currently, each record represents a Sprint. Different new variables were worked on and generated, which include:

- Number of projects or audits carried out (or "Total Projects")
- Sprint (number)
- Semester (number)
- Count of projects carried out by each type of WBS Category (internal categorization of projects of the organization); 11 variables resulted (each category has a different column or represents a variable)
- Number of hours of effort it took to perform the Sprint audits (new dependent variable)

Feature Importance (FI) and Permutation Feature Importance (PI) tests were carried out, which are tests to determine to what degree the variables explain and/or influence the behavior of the dependent variable and how effective they are all together. According to the FI test, the variable of greatest importance for the model (and with a very considerable weight) is Total Projects, followed by the Sprint variable and several variables related to the WBS Category.. Although most of the variables have very little weight, the combination of these is very effective. The above is concluded because a metric called score was calculated, which represents what percentage of the dependent variable (Actual Effort) can be explained with the same variables. This percentage resulted in 96.39%, which is very favorable. The results of the PI are very similar (score of 95.69%) to those of the FI and the same can be inferred.

In the case of the database for the capacity model, it also contains 78 records and was also grouped based on the Sprint variable. Different variables were generated for this version; in total 7. These variables are detailed below:

- Number of projects or audits carried out (Total Projects; dependent variable)
- Sprint (number)
- Semester (number)
- 4 variables were developed that show a count of the number of auditors segregated by the percentage of their work time they dedicated exclusively to audits along the Sprint. Four approximate occupancy percentages were established: 25%, 50%, 75%, and 100% (each percentage represents a variable or column).

The FI and Pi tests were also applied to this database to analyze the variables in more detail.

The auditor count variable by percentage of occupancy in 50% audits is the one that most influences the dependent variable; followed by the variables 25% and Sprint. The score received by the test is 90.93%, which means that, together, the variables effectively explain the dependent variable. These same observations are also true for the case of the PI.

The PI score is 89.57%, which is very close to that of the FI. As very favorable results were obtained for both tests, it can be concluded that the variables for the capacity model are highly effective (except for the 100% variable, which shows a very low or even null significance in both tests).

5. Results and Discussion

5.1 Initial Results

Different Machine Learning Python libraries were applied to carry out the modeling process, which include: PyCaret, H20, FLAML, TPOT, LazyRegressor, among others. After some initial tests, the number of tools was reduced to just PyCaret, H20 and LazyRegressor, because they were the ones that gave the best results (generally). These tools specialize in training and optimizing prediction models of many types and have a wide variety of customization and flexibility for model development and show great detail of the results obtained. Various metrics were applied for

evaluation of the models, these being: coefficient of determination (R^2), root mean square error (RMSE), mean square error (MSE), and mean absolute percent error (MAPE) which was the most prioritized metric.

During the first stages of this project, the models obtained did not show very favorable results, which developed the need to apply certain tools and techniques to try to obtain better results. This includes the use of PCAs, ensemble models (predictive models that combine several models into one), among others. These models (overall) slightly did not reach the established goal of the MAPE metric (maximum 15%) and had a very low R^2 metric (a minimum of 85% was sought).

Eventually, better results began to be obtained once the final or latest version of the database was obtained. Throughout the modeling process with this database, a significant progress was made that generated very satisfactory results in the modeling.

5.2 Improved Results

Within the modeling process, there is a stage where a division of the database is made; this with the purpose of having a portion of the database that serves to train the model (represents a high percentage of the records in the database) and another portion that serves to evaluate the models obtained (represents a low percentage of the database). In practice, randomness is used to determine these portions of data. During modeling of the latest version of the database, it was discovered that the random division of training and evaluation data significantly affected the results of the models (even affecting which types of models were the best). Seeing this, it was decided to carry out a large-scale analysis of how the results of the models varied based on different training and evaluation data. This is known as a Cross-Validation test.

First, using some of the Machine Learning libraries, a series of automatic modelings were performed with different instances of test and evaluation data. For each of these tests (only 10, because these Machine Learning libraries would have taken a lot of time with more tests), the best 3 models were retrieved and their results were recorded in a table. Once the previous table was obtained, a general analysis of the results was carried out to determine which types of models were the most appropriate. We analyzed which types of models appear most frequently as the best, which had the most appropriate means, minimums and maximums for different metrics, among other observations. At the end of this analysis, it was concluded which were the best types of models for the data. Once the best types of models were obtained, an analysis was performed to see how the results of each type of model vary depending on the training and evaluation data set. For this, a code was applied that generated 1000 different models of the same type of model but with different training and evaluation data sets (each one); For each modeling, the results and metrics obtained were recorded. Once the previous was obtained, a general analysis of the results for the type of model was carried out, where the behavior of the R^2 metric was mainly analyzed, which is the metric that generated the most conflict in the modeling (the MAPE metric generally met with the established goal). If the majority of the R^2 results obtained throughout the 1000 tests showed data above 70%, the type of model implemented was considered appropriate and effective (in addition to meeting the established goal of the MAPE, a low RMSE and did not have overfitting). Histograms, averages, and record counts were used for these observations. Finally, if a type of model passed the Cross-Validation test successfully, we proceed to obtain (within the 1000 tests of different training and evaluation data) the best version of the model (and its respective data set of training and evaluation)l. These optimal models were registered to later make a selection of the best models. If any of these models did not give favorable results, they were discarded. Table 1 shows the best models obtained for the Actual Effort Model.

After the modeling stage, the models with the most outstanding metrics for both the Actual Effort and Capacity models were compiled, such as the R2 test, the difference between R2 train and test, MAPE (in %) and RMSE.

Model: Actual Effort

Table 1. Results of the best models for Audit Effort

Model	Seed (train y test)	R2 test	Difference	MAPE	RMSE
RandomForestRegressor(max_features=0.9, max_samples=0.9)	600	91.40%	3.11%	6.21%	16.05
LassoLarsCV(max_n_alphas=500, precompute=False)	892	86.27%	9.93%	6.27%	16.47
AdaBoostRegressor(random_state=5711)	768	88.86%	5.84%	6.61%	16.13
LassoCV(random_state=0)	223	86.39%	9.21%	6.31%	20.75
linear_model.BayesianRidge()	768	85.88%	7.16%	6.58%	18.16
RandomForestRegressor(random_state=0)	717	89.06%	6.72%	6.64%	18.49
BaggingRegressor()	717	88.30%	6.55%	7.02%	19.12
LassoCV(random_state=0)	35	85.12%	9.28%	7.18%	17.36

It can be seen, at first glance, that the models meet the proposed objectives, where all have an R2 Test greater than or equal to 85%, as well as the MAPE ranges between 6.21 and 7.18% (results lower than the 15% established in particular objectives). Furthermore, the RMSE (in hours) ranges between 16.05 and 20.75 hours (Table 2).
Below the best three models are shown:

Table 2. Three best models for Audit Effort.

RandomForestRegressor (max_features=0.9, max_samples=0.9)	
METRIC	RESULT
R^2 Train	91.29%
R^2 Test	94.40%
Diferencia:	3.11%
MAPE Test:	6.26%
RMSE Test:	16.05
Random Seed	600.00

LassoLarsCV (max_n_alphas=500, precompute=False)	
METRIC	RESULT
R^2 Train	76.34%
R^2 Test	86.27%
Diferencia:	9.93%
MAPE Test:	6.32%
RMSE Test:	16.47
Random Seed	892.00

AdaBoostRegressor (random_state=5711)	
METRIC	RESULT
R^2 Train	83.02%
R^2 Test	88.86%
Diferencia:	5.84%
MAPE Test:	6.61%
RMSE Test:	16.13
Random Seed	892.00

The Random Forest Regressor model with optimized parameters stands out, obtaining an R2 Test of 94.40%, a MAPE of 6.26% and RMSE of 16.05 hours (Table 2).

Model: Capacity

Table 3. Results of the best models for Capacity

Model	Seed (train y test)	R2 test	Difference	MAPE	RMSE
RandomForestRegressor()	551	90.49%	5.66%	6.77%	4.57
PoissonRegressor()	13	75.76%	7.45%	9.37%	5.95
PoissonRegressor()	773	75.32%	7.39%	10.10%	5.43
PoissonRegressor()	945	77.14%	8.91%	10.12%	5.55
SGDRegressor(alpha=0.1, max_iter=500)	805	70.42%	8.40%	10.78%	6.42

It can be seen, at first instance, that the models meet the proposed objectives, where all have an R2 Test greater than or equal to 85%, as well as the MAPE ranges between 5.74 and 6.11% (results lower than the 15% established in particular objectives). Furthermore, the RMSE (in hours) ranges between 3.16 and 3.99 projects (Table 3).
Below is the best model (Table 4):

Table 4. Best model for Capacity

Random Forest Regressor	
METRIC	RESULT
R^2 Train	96.15%
R^2 Test	90.49%
Difference:	5.66%
MAPE Test:	6.77%
RMSE Test:	4.57
Random Seed	551

The Random Forest Regressor model stands out with the default parameters with an R2 Test of 90.49%, a MAPE of 6.77% and RMSE of 4.57 projects.

5.3 Prescriptive Analytics Results

Our project effectively employed heuristic optimization via iterative searching, a method where a loop systematically generates and evaluates possible solutions within set constraints to find an optimal solution. Initially, we imported and trained a database with a predictive model, performed an initial forecast, and established constraints for variables including the desired range of forecast outcomes. The code iteratively generates variable combinations, forecasts outcomes with a pre-defined model (Random Forest), and updates the best-found solution if it enhances the current one and adheres to constraints. Although not guaranteed to find the universally optimal solution, it efficiently identifies a sufficiently optimal solution within computational bounds. The resultant optimization, showcased with an example in Figure 1, adheres closely to the maximum established value, complying with constraints and presenting a logical variable combination.

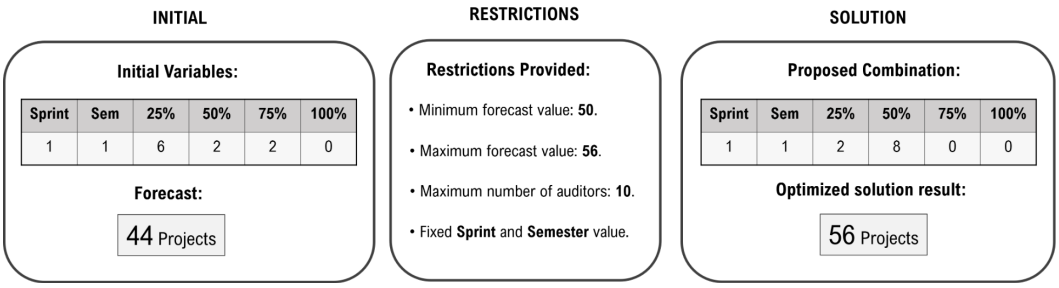


Figure 1. Comparison between initial forecast and the optimization solution *(Falta cambiar la figura a inglés)*

This optimization method was compared to other optimization options (for instance, differential evolution) and stood out as the best option in terms of computation, practicality and effectiveness.

5.4 Validation

It is worth mentioning that the web application contains several points where data and user inputs are validated before certain operations are performed. The previous is done because, in these parts of the application, if an invalid operation is done (for example, writing a character in an user input that should be numeric), the application would crash. Code was implemented so these crashes wouldn't happen and, instead, display a personalized error message which would let the user know what was done wrong and to try again. More of this is explained in Section 5.5.

A preliminary version of the web application was made available for use to certain auditors. This was mainly done for the purpose of getting some preliminary comments and suggestions from the auditors. The auditors had an overall positive feedback. They were satisfied by the web application and its intuitive framework. Also, they liked all the detail and extensiveness of the results and information that is presented on it.

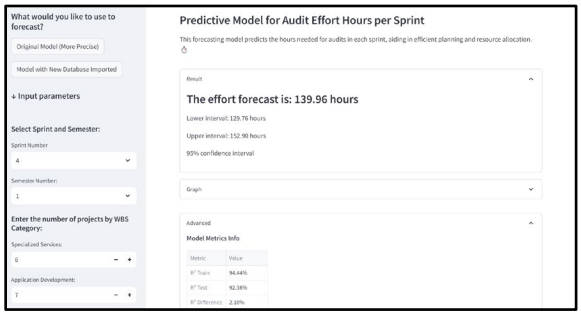
As already mentioned in previous sections, many metrics were evaluated to determine the best Actual Effort and Capacity models. Many methods were also applied to make a more detailed and complete analysis to evaluate the same metrics, such as Cross Validation and the Law of the Great Numbers. All of the previous led to a complete analysis of the results and it can be concluded that all metrics and methods showed satisfactory results; being aligned with the project's objectives.

5.5 Deployment

In the deployment phase of our project, we introduced a web application tailored for audit planning and analysis, leveraging the Python-based Streamlit framework to facilitate user interaction with predictive and prescriptive models, as well as data visualizations. This application integrates Plotly for dynamic charting, alongside pandas for data manipulation and sklearn for model training, embodying a minimalist and intuitive user interface.

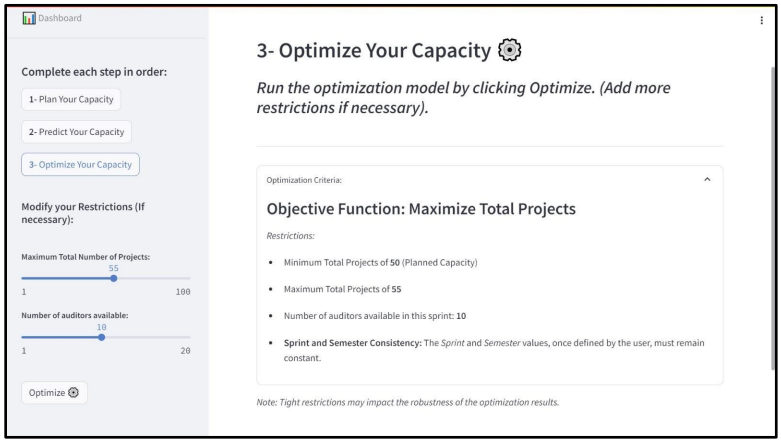
The application's "Import Data" feature allows users to upload updated CSV files, ensuring the models remain relevant over time by reflecting the latest "Actual Effort" data. This functionality is critical for maintaining the accuracy of the "Audit Effort Model," which offers two model types: a pre-trained model and one retrained with newly imported data, utilizing the PyCaret library for model selection and Cross Validation for optimal prediction accuracy. Predictions are presented with confidence intervals to account for variability, enhancing reliability (Figure 2).

Figure 2. Audit Planning Tools - Audit Effort Models



In the "Audit Capacity Models" section, users navigate through a three-step process to plan, predict, and optimize audit capacity. This involves inputting variables, comparing planned vs. potential audits, and using an objective function with customizable constraints for optimization. The process utilizes stored variables from previous steps to align the optimization with user-defined parameters (Figure 3).

Figure 3. Audit Planning Tools - Audit Capacity Models



The "Dashboard" tab offers extensive data visualization options, allowing users to analyze various aspects of the audit process, from overall database metrics to specific analytical views like time series and histograms, leveraging the interactivity provided by Plotly. This dashboard enhances user engagement and understanding of the data through a variety of visual representations, including box plots, pie charts, bar charts, bubble plots, and area plots, each offering different insights into the audit data (Figure 4).

Figure 4. Dashboard - Audit Planning Tools.

Overall, the web application serves as a comprehensive tool for audit planning and analysis, enabling users to import data, select and train models, and visualize outcomes through a highly interactive and user-friendly platform. This deployment highlights the application's role in facilitating efficient audit management and decision-making processes.

6. Conclusion and Recommendations

Through the use of different automatic modeling libraries in the Python programming language, hundreds of models were generated and trained in order to be able to predict the number of hours of effort (Actual Effort) needed to perform a certain number of audits in a Sprint and the number of projects (Total Projects) that can be performed in a Sprint based on certain input data. The best results or models were obtained with the Lazy Regressor and Pycaret libraries. For the Actual Effort model, a Random Forest Regressor was obtained as the best model, with optimized parameters, obtaining an R^2 Test of 94.40%, a MAPE of 6.26% and RMSE of 16.05 hours. For the capacity model, a Random Forest Regressor model was also obtained, with optimized parameters, obtaining an R^2 Test of 90.49%, a MAPE of 6.77% and RMSE of 4.57 projects.

Different optimization methods were explored to optimize the number of projects that can be audited in a period of time. The heuristic optimization method based on iterative search turned out to be the best method in computational, practical and effective terms.

Finally, visualization and Machine Learning libraries were applied to complement and/or improve the digital management tool. These libraries allow you to monitor KPIs and statistical information with historical and updated data from the audits carried out by the auditors. Additionally, there are now more complete and precise methods and processes for forecasting Actual Effort and Total Projects. Very positive feedback was received from auditors about the digital tool in general.

These methods and solutions to the objectives of this project are recommended for other projects or solutions that have a similar nature or theme to that of this project. Additionally, it is recommended that this study be carried out from time to time because, over time, the optimal models found at the time this project was carried out may not adequately represent how the nature of the data behaves in future times.

References

Banu, S., Singh, A., Agarwal, N., Shaik, S. and Nikitha, P. S., Machine Learning Algorithm to predict and improve efficiency of employee performance in organizations, *Proceedings of the 2nd International Conference on Advances in Computing & Information Technology*, pp. 6-11, Bangalore, India, 20-30 , 2020.

Carter, K. M., Mead, B. A., Stewart, G., Nielsen, J. and Solimeo, S., Reviewing Work Team Design Characteristics Across Industries: Combining Meta-Analysis and Comprehensive Synthesis, *Small Group Research*, pp. 138-188, Pennsylvania, United States, 2019.

Chang, L. Capacity Planning and Scheduling of a Nursing Resource Team (NRT). Toronto, Canada, November, 2019.

Fang, X., Gao, Y. and Hu, P., A Prescriptive Analytics Method for Cost Reduction in Clinical Decision Making. United States. 2021.

Geng, N. and Jiang, Z. Capacity Planning for Semiconductor Wafer Fabrication with Uncertain Demand and Capacity, *Proceedings of the 2007 IEEE International Conference on Automation Science and Engineering*, pp. 100-105, Arizona, United States, 22-25 September, 2007.

Ghoniem, A., Ali, A. I., Al-Salem, M. and Khallouli, W. Prescriptive analytics for FIFA World Cup lodging capacity planning, *Journal of the Operational Research Society*, pp. 1183–1194, United States, 15 December 2016.

GururamaSenthil, P., Duraimutharasan, N., Esther, H. R., Periyasamy, P. N. and Kumar, R. Artificial Intelligence enabled Employee Performance Prediction using Comprehensive Learning Metrics, Proceedings of the 2023 International Conference on Advances in Computing, Communication and Applied Informatics (ACCAI), pp. 1-7, Chennai, India, 25-26 May 2023.

IBM. (2023). Predictive Analytics. https://www.ibm.com/mx-es/analytics/predictive-analytics. September 1, 2023.

IBM. (2023). Prescriptive Analytics. https://www.ibm.com/mx-es/prescriptive-analytics. September 2, 2023.

Kenton, W. (2021). What Is Capacity and How Does a Company Maximize Output? Investopedia. https://www.investopedia.com/terms/c/capacity.asp#:~:text=Key%20Takeaways,or%20several%20other%20related%20concepts

LinkedIn. (2023). How do you define and measure team performance? https://www.linkedin.com/advice/0/how-do-you-define-measure-team-performance

MathWorks. (2023). Support Vector Machine (SVM) https://www.mathworks.com/discovery/support-vector-machine.html, November 25, 2023.

McComb, C., Cagan, J., & Kotovsky, K. (2017). Optimizing Design Teams Based on Problem Properties: Computational Team Simulations and an Applied Empirical Test, *Journal of Mechanical Design*, vol. 139, April 2017.

Moura, I., Dominguez, C., & Varajão, J. (2021). Information systems project team members: factors for high performance. ProQuest. https://www.proquest.com/docview/2586669222/fulltextPDF/ABD7189D7F4241B7PQ/28?accountid=17236.

Prasad, U., Chakravarty, S., Bisht, Y. S., Prusty, A., Nijhawan, G., & Lourens, M. (2023). Using Natural Language Processing and Blockchain for Employee Performance Evaluation. IEEE Xplore. https://ieeexplore.udemproxy.elogim.com/document/10182524.

Rivaldo, Y. and Diana, N. S, Employee Performance: Education, Training, Experience and Work Discipline, *Quality - Access to Success*, 2023.

Srividya, C. S. and Prathiba, S, Impact of Employee Experience on Employee Performance - A Competitive Advantage, *Journal of Strategic Human Resource Management,* vol. 12, India, 2023.

Tektas, B., Turan, H. H., Kasap, N., Çebi, F., & Delen, D., A Fuzzy Prescriptive Analytics Approach to Power Generation Capacity Planning, *Energies 2022*, Turkey, 2022.

Thammaboosadee, S. and Wongpitak, P., An Integration of Requirement Forecasting and Customer Segmentation Models towards Prescriptive Analytics for Electrical Devices Production, *Proceedings of the 2018 International Conference on Information Technology (InCIT)*, pp. 1-5, Khon Kaen, Thailand, 24-26 October 2018.

Naseer, M. and Mitschele-Thiel, A., Reinforcement Learning Strategies for Self-Organized Coverage and Capacity Optimization, *Proceedings of the 2012 IEEE Wireless Communications and Networking Conference (WCNC),* pp. 2818-2823, Paris, France, 01-04 April 2012.

Valerie. How to Use Individual Performance To Advance Team Performance. 2023. https://blog.vantagecircle.com/individual-performance-into-team-performance/#:~:text=Individual%20performance%2C%20whether%20good%20or,affected%20by%20an%20individual's%20contribution. September 15, 2023.

Van Geffen, C. Optimizing Team Effectiveness and Performance by Using the Cycle of Team Development, pp. 298-314, Netherlands, 8 September, 2020.

Biographies

Patricio Alejando Zamora Navarro is a current student of Business Management Engineering at Universidad de Monterrey. His current passion and development of skills and knowledge involve topics related to predictive analysis, programming languages, web development and machine learning. He is currently working as an intern at an information technology services company, where he is getting professional experience and development with programming skills; he is also getting programming knowledge through the online learning platform: DataCamp.

Rafael Alejandro Ruiz Miller is an 8th semester student of Business Management Engineering at Universidad de Monterrey. Currently he is working as an analyst intern at a global information technology services company, where he is developing skills such as: data engineering, data analysis, web applications development, programming with Python, R, and more. He continues to expand his knowledge through the learning platform of DataCamp. In addition to his academic achievements, he has shown entrepreneurial interest by co-founding and serving as the director of a digital marketing company for two years. Presently, he is also pursuing a diploma in business ethics to further enhance his understanding of responsible corporate practices.

Roberto Serna Zuazua is a Business Administration Engineering student at the Universidad de Monterrey. He currently works as a data analyst in a global information technology services company, where he develops knowledge related to predictive and descriptive analysis, programming languages such as Python, R and ACL, web development with Sharepoint and Javascript, among others. In addition, he graduated with honorable mention of excellence from high school, as well as completing the International Baccalaureate Diploma Programme. Roberto has been a member of the Líderes Plus group from the Universidad de Monterrey with the purpose of achieving excellence through leadership. On the other hand, he obtained second place in the Cisco-UDEM Hackathon with the *Twinky* project focused on Smart Cities and pollution. And he served as Secretary in the Student Chapter of the American Society for Quality (ASQ) at the Universidad de Monterrey.

José D. Morcillo received the Electronic Engineer degree, and the M.Sc. degree in industrial automation from the National University of Colombia, Manizales, in 2010 and 2012, respectively. He got his Ph.D degree in Computer Science from National University of Colombia, Medellín, in 2018. He also worked as a postdoctoral researcher at the National University of Colombia, Manizales, in 2020. In 2010, he joined the Perception and Intelligent Control Group, and in 2014, joined the Systems and Informatics Group, working in different I + D projects. Currently, he is working as a full-time professor at University of Monterrey, Mexico, in the Engineering and Technologies Department. His research interests include modeling, simulation and control of power electronics and electricity markets, applying the theory of dynamical systems, system dynamics and data analytics/machine learning.