Vrije Universiteit Amsterdam

Universiteit van Amsterdam

Master's Thesis

All-Stage Machine Learning-Driven B2B Sales Predictor

Author: Rohit Shaw (VU: 2630119, UvA: 12025666)

Supervisor: Dr Adam S. Z. Belloum

A thesis submitted in fulfillment of the requirements for
the joint UvA-VU Master of Science degree in Computer Science

March 2023

"The most important thing is to try and inspire people so that they can be great in whatever they want to do."

Kobe Bryant

Abstract

Sales forecasting involves predicting future sales for a business that sells products or services to other customers or businesses. The process involves analyzing historical sales data, market trends, and other factors to develop a projection of future sales. Predicting future demand in a B2B (business-to-business) context is critical because the entire manufacturing and supply chain relies on these forecasts. While various traditional (manual) forecasting methods are available, the process could be fully automated end-to-end using machine learning techniques. This thesis is built on the concept of the machine-learning-driven B2B sales predictor developed by K. M. Kasinathan. Instead of looking at Won and Lost sales opportunities, this thesis considers all Sales Stages that an opportunity goes through during its lifecycle. The result was a significant improvement in prediction accuracy, comparable to that of the conventional sales predictor in the organization. Of all models tested, CatBoost produced the best predictions, albeit only by a minuscule margin. The final model was deployed in production on Tableau, and a user-friendly dashboard was built to visualize the predictions. It considerably reduced the time required to prepare weekly updates of Sales Pipeline predictions for the sales leadership.

Acknowledgements

First and foremost I am extremely grateful to my supervisor, Dr. Adam S. Z. Belloum, for his invaluable advice, continuous support, and patience throughout my thesis. I would also like to thank my best friend for being selfless and supporting me in a myriad of ways. I would like to thank all the members of the EMEA & LATAM Sales Operations team at NetApp for their support with the conventional sales predictor. It is their kind help and support that have made my thesis and its outcome a success. Finally, I would like to express my gratitude to my parents for their tremendous understanding and encouragement over the past few years.

Contents

List of Figures

List of Tables

1

Introduction

B2B sales (business-to-business sales) is the process of selling products or services to other businesses rather than to individual consumers. This can involve a variety of industries and products, from office supplies and equipment to consulting services and software. B2B sales can be complex, as the purchasing decisions are often made by multiple people within an organization, and the sales process may involve a longer decision-making timeline than B2C sales (business-to-consumer sales).

Sales forecasting is the process of predicting future sales for a business that sells products or services to other businesses. This can involve analyzing historical sales data, market trends, and other factors to develop a projection of future sales (1). B2B sales forecasting is important for a variety of reasons, such as helping businesses plan for future growth and allocate resources, identify potential challenges and opportunities, and set sales targets for teams or individual salespeople. B2B sales forecasting can be complex, as it involves taking into account many different variables and can be impacted by a variety of internal and external factors.

Predicting future demand in a B2B context is critical because the entire manufacturing and supply chain is dependent on these forecasts. There are various traditional forecasting methods, the majority of which are based on previous sales figures. Companies now have more data than ever before, which can be mined for useful insights and used for advanced analytics applications with machine learning.

While forecasting sales based on open and closed opportunities has proven to offer quality forecasts, the stages an opportunity goes through from the moment it is open to the moment it is marked as closed is yet to be explored. Including all stages in forecasts could potentially produce higher quality and more reliable predictions as it would consider other factors that could influence the booking of sales opportunities.

This thesis builds upon the B2B sales predictor developed by K. M. Kasinathan (2), by considering all stages a sales opportunity goes through at NetApp and developing an all-stage sales predictor. At NetApp, Sales Operations is responsible for sales forecasting, and the predictor will operate from the Sales Operations' perspective, as opposed to the Finance department's as in K. M. Kasinathan's model.

1.1 Research Questions

To structure the direction of this thesis, research questions were formulated along with their rationales.

RQ.1 **Does the introduction of all sales stages of opportunities have a positive impact in terms of prediction accuracy?**
The goal is to identify the degree to which the inclusion of all sales stages of opportunities helps improve predictions.

RQ.2 **Does CatBoost continue to produce the best predictions with the added sales stages, or is there a machine learning model that performs even better?**
CatBoost and other machine learning algorithms will be put to test based on findings from related work. The aim is to find the algorithm that produces the most accurate results.

RQ.3 **In what ways can the results produced by the predictor be utilized for business-related analytics?**
The predictor's main task is to forecast future sales of the company, but the data behind it could be used for additional analytics too. The aim is to find alternative use cases for the predictor model.

1.2 Research Overview

This thesis is organized into eight chapters.

- Chapter 1 describes the problem statement of this thesis and outlines the research questions to be answered. It also includes the structure of this thesis report.
- Chapter 2 explores available related work on machine-learning-driven B2B Sales Forecasting, providing insight into the best practices in the industry.
- Chapter 3 breaks down the conventional method of predicting sales in the organization and expands upon each metric that it takes into consideration to drive predictions.
- Chapter 4 discusses the methods used to extract, analyze, and transform the data used for experimenting with machine learning models.
- Chapter 5 discusses the machine learning approaches taken to build the automated version of the conventional sales predictor.
- Chapter 6 discusses the deployment of the machine-learning-driven sales predictor model on Tableau using TabPy, Jupyter Notebook, and Microsoft SQL Server and building a user-friendly dashboard to visualize the model's predictions.
- Chapter 7 discusses the challenges and limitations of using machine learning in B2B Sales Forecasting in the organization.
- Chapter 8 concludes this thesis by answering the research questions formulated at the beginning when outlining the problem statement.

2

Related Work

This chapter explores available related work on machine-learning-driven B2B Sales Forecasting, providing insight into the best practices in the B2B space. While there are numerous studies on B2C sales forecasting available, the same methodologies cannot be applied to B2B sales forecasting. Research on machine-learning-driven B2B sales forecasting may be limited, but it offers insight into common practices and paves the way for this thesis.

In the study by Kailainathan Muthiah Kasinathan, a machine-learning-based predictor model was developed to classify NetApp Inc.'s sales opportunities into Won and Lost (2). The predictor was modeled using the company's worldwide sales data. Random Forest, XGBoost, CatBoost, and Decision Tree classifiers were chosen for machine learning tasks due to their superior classification abilities. CatBoost was reported as the best-performing model and achieved 77% classification accuracy. However, the model suffered from data drift and its classification accuracy reportedly fell to 75% as a result. While not a replacement, the model developed served as a good alternative to conventional forecasting techniques being used in the organization.

Alireza Rezazadeh proposed a machine learning-driven workflow using Microsoft Azure Machine Learning Service for predicting the likelihood of winning sales opportunities in (3). The study uses data from a B2B consulting firm from many industries, including healthcare, energy, and finance. XGBoost and LightGBM classifiers were selected due to their higher classification accuracy for the problem. Predictions were made using the voting ensemble on the test set to infer the probability of winning for each sales opportunity and an accuracy of 83% was observed. The machine-learning-driven approach was then deployed at a multi-business B2B consultancy firm to test its efficacy over time and was compared to predictions made by salespeople. The findings of this study indicate that a machine-learning-driven strategy for forecasting sales is a more practical method than salespeople's subjective forecasts. The author also points out that the proposed approach should not rule out salespeople's justifiable opinions when analyzing sales opportunities.

In the study by Tiemo Thiess et al., an explainable two-level win-propensity prediction system was proposed as a solution to improve MAN Energy Solutions' hit rate of quotations (4). It used LightGBM and a conditional probability model for predicting quotation age. An action design research process was used to assess and create the field problem of improving the after-sales hit rate at the company. Following that, a win-propensity scoring system was developed and integrated into the organization's existing IT architecture. The system's core consisted of a LightGBM-based model that generated the base win-propensity probabilities for sales quotations, and a second-level conditional probability model that took into consideration the diminishing influence of quotation age on the base win-propensity probabilities. LightGBM was selected because, after being trained on a dataset of 3 million records of quote positions and 15 carefully chosen features, it was the most effective tree-based ensemble approach achieving an average accuracy of 76% and an AUC of 0.74. While the proposed solution addresses challenges prevalent in the B2B domain, the design principles could be transferable to the B2C domain too.

Marko Bohanec et al. proposed a novel method for B2B sales forecasting using machine learning to aid the decision-making process of forecasting via transparent reasoning (5). Random Forest, Naive Bayes, and Decision Tree were selected to be experimented with. Upon testing, Random Forest was observed to be performing the best, achieving a classification accuracy of 96% and an AUC of 0.98. The research was limited by the number of training instances it had, many of which were generated artificially, which may have contributed to the high levels of classification accuracy attained using this method. The authors also point out that the major obstacle to obtaining well-structured tabular data with attributes defining B2B sales opportunities is that companies and salespeople pay little attention to critical attributes required to build a domain knowledge architecture.

The study by Stephen Mortensen et al. analyzes and compares a number of popular techniques for categorizing and rating propensities, the bulk of which fall under the umbrella of decision tree modeling (6). To forecast win probabilities for sales prospects, several models were developed. The model with the best predictive performance and insight-generating capacity was chosen to serve as the framework for a client tool. Structured and unstructured data from the company's customer relationship management system was used to achieve this. Several methods, including boosting with gradient boost and Random Forest, as well as binomial logit and different decision tree algorithms, were tested. Individual customer characteristics, opportunities, and internal documentation procedures were found as having the biggest influence on sales success. The best model had an accuracy of 80% when predicting win propensity, with precision and recall of 86% and 77%, respectively.

Junchi Yan et al. proposed a unified machine-learning-driven framework for predicting the win propensities of sales opportunities (7). The models for the framework were built on training samples from historical sales data. Logistic Regression was the model of choice as it could produce a probability number, making it easy for business-related interpretation. While accuracy metrics were not reported, the authors elaborated on propensity scores and lead conversions being dependent on the synergy between the sales and marketing teams.

In the study by D Rohaan et al., a technique for using request for quotation (RFQ) data as advance demand information for B2B sales forecasting was proposed (8). To evaluate and learn from RFQs, supervised machine learning techniques were used. Gradient Boosting Classifier, Random Forest, and Logistic Regression classifiers were chosen for carrying out experiments. Random Forest was reported as the best-performing model and achieved 54% classification accuracy.

Machine Learning Algorithm	No. of Mentions
Random Forest	6
Decision Tree	3
XGBoost	2
LightGBM	2
CatBoost	1
Naive Bayes	1
Logistic Regression	1

Table 2.1: Machine Learning Algorithms & No. of Mentions in Related Work

Looking at the related work in the field of B2B sales forecasting, a common trend was observed - all studies performed the task of classification using machine learning. One study performed regression, but that wasn't the main focus of it. This thesis solely focuses on regression as the goal is to predict future sales, making it the first of its kind. Furthermore, the most popular machine learning algorithms were also identified (Table 2.1), offering insight into what to expect from each.

3

Conventional Sales Forecasting

B2B sales forecasting is a crucial component of running and expanding a company. It can assist in identifying possibilities and obstacles, setting practical targets, and making better-informed decisions.

In traditional B2B sales forecasting, a model for predicting future sales is often developed utilizing historical sales data, market trends, and other criteria. It can be difficult to account for all of the factors that could affect sales in this lengthy and complex procedure. Furthermore, modern approaches to forecasting, like machine learning, may be more accurate than conventional ones.

Machine learning-based B2B sales forecasting on the other hand entails employing algorithms and data analysis to create forecasts about upcoming sales. This entails using historic sales data as well as other pertinent information, such as market trends and client demographics, to train a machine learning model (9). Based on a wider range of factors and data, machine learning enables organizations to anticipate future sales with better efficiency and accuracy. The ability of machine learning models to learn from new data makes them more flexible and adaptable. Machine learning-based approaches have been elaborated on in the next chapters.

This chapter breaks down the existing (conventional) method of predicting sales in the organization and expands upon each metric that it takes into consideration to derive sales predictions.

3.1 Sales Stages

The various stages that prospective customers experience as they think about and, eventually, make a purchase are referred to as Sales Stages. To organize and monitor the advancement of opportunities, these stages are frequently employed in the sales process. NetApp uses 7 Sales Stages in all its sales-related tools and systems which can be categorized into open and closed stages.

- Open Stages
  - **Prospecting** - A lead or potential and genuine piece of business (from account planning, marketing, 3rd party, etc.) which is being reviewed by the sales team before moving into a full qualification.
  - **Qualification** - The (customer) account team has met with the customer and qualified that there is a business requirement and a budget (or potential budget) to solve the business need.
  - **Proposal** - A solution has been formulated for the business problem with a proposed budget that makes business sense when stacked up against the problem. The solution and business case have been socialized with the customer, and the customer has challenged/discussed and workshopped out the solution and costs associated with implementing the solution.
  - **Acceptance** - All information, proposals, and solution documentation sent to the customer for internal reviews, budget approvals, and business acceptance.
  - **Negotiation** - Customer engaging in next steps, raising objections and commercial discussions in order to progress the deal. Sales teams are actively looking to close this deal. The customer has agreed to the terms and is finalizing the paperwork to invest in the solution.
- Closed Stages
  - **Won** - Purchase Order has been received and agreements have been signed.
  - **Closed/Lost** - Opportunity not won with the identified customer on the deal.

The conventional sales predictor in the organization only looks at the Open Stages to derive predictions.

3.2 Sales Organization Hierarchy

A sales organization hierarchy refers to the way a company organizes and manages its sales teams. At NetApp, GARD (Geo, Area, Region, District) is an acronym that is often used to represent the sales hierarchy. It is common practice at the organization to change the hierarchy at the end of every fiscal year.

The organization was in fiscal year 2023 at the time of writing this thesis. The GARD during the fiscal year comprised of 3 Geos:

- Asia Pacific
- EMEA & LATAM
- North America

Due to geographic data access restrictions, the conventional predictor only forecasted sales in EMEA & LATAM. The machine learning-based predictor for this thesis was also built for the same Geo due to similar access restrictions for the author. The expanded GARD for EMEA & LATAM fiscal year 2023 is shown in Appendix A.

3.3 Sales Metrics

Sales metrics are measurements that are used to monitor and assess a sales team's or an individual salesperson's performance. These indicators assist in identifying areas of strength and weakness in the sales process and assist sales leaders in making data-driven decisions to enhance performance. The metrics the conventional predictor takes into consideration are AOP, Bookings, Pipeline, and Forecast.

3.3.1 AOP

The AOP (Annual Operating Plan) aids sales teams in future planning and target achievement. It describes the objectives and tasks a sales team intends to carry out over the course of a year. The plan often outlines explicit objectives for important metrics like revenue, bookings, and client acquisition in addition to the approaches and tools required to meet those objectives (10). Additionally, it could contain information about the sales team's key performance indicators (KPIs), timeframes, and budgets.

3.3.2 Bookings

Bookings refer to the total value of sales that have been agreed to but not yet completed. Bookings are generally used to describe huge, complex transactions that necessitate a lengthy decision-making process, such as enterprise-level software or consulting services (11). Bookings are distinct from revenue, which is the sum of money that a company has made from completed sales.

Bookings are a crucial business statistic since they can predict future revenue and assist managers in planning for expansion. For instance, a lot of bookings in a single quarter can indicate that a company is doing well and that future revenue would be healthy. In order to set goals and monitor success, sales teams can also benefit from bookings.

3.3.3 Pipeline

Pipeline describes the procedure that takes a potential consumer from the initial point of contact to the final point of sale. The objective is to transfer potential clients through the pipeline as quickly and effectively as possible. Each stage of the pipeline reflects a distinct level of the sales process.

The sales pipeline is a crucial tool for sales teams since it enables them to comprehend the state of their sales and spot any potential issues or possibilities (12). An indication that a salesperson is succeeding with their sales pitch, for instance, would be if they have plenty of potential clients in the pipeline at the proposal stage. On the other hand, if there aren't enough prospective clients in the closing stage, there may be an issue with the sales process.

3.3.4 Forecast

Forecast is the expected or estimated sales figure that a corporation uses to make future plans. These figures can be based on a number of factors, including previous sales statistics, market trends, and the general growth strategy of the company (1). Sales forecast data can be used to allocate resources, define targets for the sales staff, and decide on the future course of the company.

For organizations, forecasting is crucial since it can offer insightful data about the company's future. For instance, a company's high sales projection for consecutive quarters may be a sign that the business is performing well and that future sales may be robust. Conversely, a poor sales projection may indicate that the company needs to adjust its sales strategy or operational procedures.

3.4 Sales Pipeline Hygiene

The term "sales pipeline hygiene" in the field of sales is the practice of maintaining and organizing the various stages of the sales pipeline on a regular basis to make sure it is operating effectively. This may entail activities like routinely evaluating and updating the pipeline, determining which leads or opportunities are no longer active and eliminating them, keeping accurate records of all contacts with leads, and making sure that each lead is in the appropriate stage of the pipeline. A well-maintained pipeline allows the sales staff to concentrate on closing agreements and increasing productivity.

To maintain sales hygiene, there are specific sales-related fields that are crucial for sales staff to monitor and update within the CRM (Customer Relationship Management) tool. The Sales Operations department within the organization relies on these being accurate and then later uses them to derive predictions of a given Sales Region in a particular Quarter. The critical fields for the conventional predictor to work effectively are:

- Sales Stage
- Opportunity Value
- Opportunity Close Date

3.5 Conventional Predictor Workflow

The conventional method of forecasting sales, which is currently being used in the organization, comprises an Excel file with several formulas and some manual input from the person updating it. The workflow of the conventional predictor for sales in EMEA & LATAM at NetApp can be broken down into five steps.

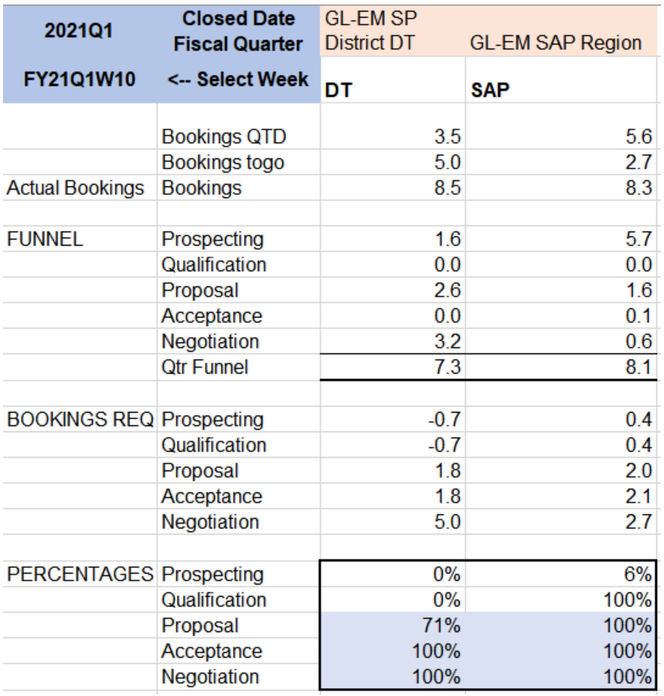
1. **Fetch the required data:** All the data required for the predictor to work - AOP, Bookings, Pipeline, Forecast, is extracted from the respective sales tools.

2. **Consolidate the data as per the predictor file's format:** Since different data metrics reside in different tools, they need to be consolidated and transformed into the format that the predictor file takes as input.

3. **Update the data in the predictor file and refresh:** The data in the predictor file is updated/replaced with the most recent data that was fetched and prepared in the previous steps. Then the data model and formulas are refreshed in the file, revealing the updated predictor numbers.

4. **Make manual adjustments wherever necessary:** Metrics stated in Step 1 may be manually adjusted based on information received from the sales teams. This is done to factor in numbers that might not be in the sales tools at the time of the predictor data refresh.

5. **Derive predictions and share the results with sales leaders:** When the predictor is presented to the sales leadership, it is generally used to provide a high-level overview of what the predictability looks like with the current weekly trend with Bookings coming in, and comparing the predicted figure against the AOP and Forecast assigned to each sales team.



2021Q1	Closed Date Fiscal Quarter	GL-EM SP District DT	GL-EM SAP Region
FY21Q1W10	<-- Select Week	DT	SAP
	Bookings QTD	3.5	5.6
	Bookings togo	5.0	2.7
Actual Bookings	Bookings	8.5	8.3
FUNNEL	Prospecting	1.6	5.7
	Qualification	0.0	0.0
	Proposal	2.6	1.6
	Acceptance	0.0	0.1
	Negotiation	3.2	0.6
	Qtr Funnel	7.3	8.1
BOOKINGS REQ	Prospecting	-0.7	0.4
	Qualification	-0.7	0.4
	Proposal	1.8	2.0
	Acceptance	1.8	2.1
	Negotiation	5.0	2.7
PERCENTAGES	Prospecting	0%	6%
	Qualification	0%	100%
	Proposal	71%	100%
	Acceptance	100%	100%
	Negotiation	100%	100%

Figure 3.1: Preview of the Conventional Predictor's Workings

3.6 Manual Adjustments & Prediction Accuracy

Step 4 of the predictor workflow in section 3.5 involves making manual adjustments based on information received from the sales staff, to incorporate upcoming large changes in one or more metrics used in the predictor. However, the data extracted in Step 1 may also include “Whale" opportunities that have been a part of historical data for sales/revenue and perhaps could influence the predictability of future quarter sales.

A “Whale” opportunity is described as a larger-than-normal sales deal size, which may be uncommon to see in a typical business environment. This could be a huge contracted deal that would occur once in a while. Such deals have a big impact on the sales pipeline and should be treated differently.

The team managing the sales predictor in the organization reported that it has a prediction accuracy in the range of 95-105% with manual adjustments. The near-perfect prediction accuracy set a high bar for the machine learning-based predictor developed as a part of this thesis in the next chapters.

4

Data Preparation

This chapter discusses the methods used to extract, analyze, and transform the data used for experimenting with machine learning models.

4.1 Data Extraction

To prepare the data for machine learning tasks, historic sales-related data had to be extracted. The data extraction phase involved two steps - fetching sales data for the current fiscal year and obtaining weekly snapshots of historic sales data.

4.1.1 Data for Current Fiscal Year

The organization hosts different kinds of data across various tools, each having its own set of (GARD) access requirements. For the predictor, data for the ongoing fiscal year had to be extracted from four different tools - Ascend, eBI, Tableau, and SharePoint, the process for each of which is described below.

1. Fetch Sales Pipeline data from Ascend (CRM tool) – Live changes or updates are made regularly on the tool; refreshed weekly for predictor use.

2. Fetch Sales Bookings data from eBI (business application bookings tool) – Live changes or updates are made daily on the tool; refreshed weekly for predictor use.

3. Fetch AOP data from Tableau (data visualization tool) – Live changes or updates related to AOP are uploaded once every six months; refreshed once every Fiscal Quarter for predictor use, as the predictor focuses on the ongoing quarter.

4. Fetch Sales Forecast data from SharePoint (organization intranet site) – Published by the EMEA & LATAM finance team once a month; refreshed once a month also for predictor use.

   For example: When the finance team publishes Month 2 (M2) Forecast of the Fiscal Quarter, it includes the actuals of Month 1 (M1), and Forecasts of Month 2 (M2) and Month 3 (M3).

The columns (features) chosen for all four sales metrics were the same as those used in the conventional sales predictor in the organization. Consulting the team managing the conventional predictor regarding the relevance of including other available features on the tools yielded that the predictor was only driven by Sales Stages, GARD, and the Fiscal Year calendar. Doing a deep dive at the account/customer level would be tedious. Moreover, the metrics AOP and Sales Forecast didn't exist at a level that granular.

4.1.2 Historic Data Snapshots

Continuing on Steps 1 and 2 in section 4.1.1, the weekly snapshots of Sales Pipeline data and Sales Bookings data for previous fiscal years weren't available on any tool and had to be obtained from the team managing the conventional predictor. The historic data obtained was for FY2020 and later as the conventional predictor was first deployed during that fiscal year. The historic data was then remapped to the current GARD hierarchy (as of writing this report) to obtain conversion rates (%), which acted as the foundation of the predictor.

4.2 Data Preprocessing

After obtaining historic data and realigning it to the current GARD hierarchy in section 4.1, the raw data files were transformed, reduced, and combined to produce a single file. Table 4.1 lists all features that were present in the raw data file of each sales metric. The features for the GARD hierarchy and the Fiscal Year calendar were present in all four files, making them ideal for consolidation.

Sales Pipeline data had 5 instances (one for each open Sales Stage) per Sales Region in each week's snapshot, whereas the files for all remaining metrics had 1 instance per weekly snapshot. To have the Sales Pipeline data in the format of the rest of the data, it was transformed. 5 new features (one for each open Sales Stage) were created - Prospecting

Sales Metric	Feature	Data Type
Sales Pipeline	`Sales Multi Area`	Categorical (object)
	`Sales Area`	Categorical (object)
	`Sales Multi Region`	Categorical (object)
	`Sales Region`	Categorical (object)
	`Opportunity Close Fiscal Year`	Categorical (object)
	`Opportunity Close Fiscal Quarter`	Categorical (object)
	`Sales Stage`	Categorical (object)
	`Snapshot Week`	Categorical (object)
	`Pipeline Amount`	Numerical (float64)
Sales Bookings	`Sales Multi Area`	Categorical (object)
	`Sales Area`	Categorical (object)
	`Sales Multi Region`	Categorical (object)
	`Sales Region`	Categorical (object)
	`Fiscal Year`	Categorical (object)
	`Fiscal Quarter`	Categorical (object)
	`Snapshot Week`	Categorical (object)
	`Booking Amount`	Numerical (float64)
	`Bookings Quarter-To-Date`	Numerical (float64)
	`Bookings To Go`	Numerical (float64)
AOP	`Sales Multi Area`	Categorical (object)
	`Sales Area`	Categorical (object)
	`Sales Multi Region`	Categorical (object)
	`Sales Region`	Categorical (object)
	`Fiscal Year`	Categorical (object)
	`Fiscal Quarter`	Categorical (object)
	`AOP Amount`	Numerical (float64)
Sales Forecast	`Sales Multi Area`	Categorical (object)
	`Sales Area`	Categorical (object)
	`Sales Multi Region`	Categorical (object)
	`Sales Region`	Categorical (object)
	`Fiscal Year`	Categorical (object)
	`Fiscal Quarter`	Categorical (object)
	`Forecast Amount`	Numerical (float64)

Table 4.1: Features in the Extracted Data of each Sales Metric

Amount, Qualification Amount, Proposal Amount, Acceptance Amount, and Negotiation Amount. The data was then transposed, which resulted in the Sales Pipeline data being in the same format as the raw data for the rest of the sales metrics.

To join Sales Bookings data and Sales Pipeline data, a unique key comprising of 3 features (`[Sales Region]_[Fiscal Quarter]_[Snapshot Week]`) was created. The raw data of both metrics were then joined, resulting in 1 instance per weekly snapshot for each Sales Region's bookings and pipeline.

For joining AOP and Sales Forecast, a new unique key comprising of 3 features (`[Sales Region]_[Fiscal Year]_[Fiscal Quarter]`) was created. The data for both metrics were then joined with the previously combined Sales Pipeline and Sales Bookings data. The result of this was a file having all the necessary metrics as features and 1 instance for every weekly snapshot. Table 4.2 lists all features and their data types that were obtained as a result of preprocessing the raw data.

Feature	**Data Type**
`Sales Multi Area`	Categorical (object)
`Sales Area`	Categorical (object)
`Sales Multi Region`	Categorical (object)
`Sales Region`	Categorical (object)
`Fiscal Year`	Categorical (object)
`Fiscal Quarter`	Categorical (object)
`Snapshot Week`	Categorical (object)
`Prospecting Amount`	Numerical (float64)
`Qualification Amount`	Numerical (float64)
`Proposal Amount`	Numerical (float64)
`Acceptance Amount`	Numerical (float64)
`Negotiation Amount`	Numerical (float64)
`Booking Amount`	Numerical (float64)
`Bookings Quarter-To-Date`	Numerical (float64)
`Bookings To Go`	Numerical (float64)
`AOP Amount`	Numerical (float64)
`Forecast Amount`	Numerical (float64)

Table 4.2: Features After Preprocessing the Data

4.3 Exploratory Data Analysis

Exploratory data analysis was performed on the preprocessed data using various statistical and graphical analysis methods. As the data was obtained from the team managing the conventional predictor, it was assumed that it had already been cleaned up. Performing checks for NaNs, duplicates, and nulls in the data validated the assumption.

Drawing graphs using graphical analysis methods provided a visual overview of the preprocessed data. Figure 4.1 shows the number of instances of data in each fiscal year and Figure 4.2 shows the correlation between all numerical features in the dataset. Looking at the correlation matrix, a trend was observed that the higher the value of any of the Acceptance and Negotiation sales stages, the better the likelihood of having a higher conversion rate. If more than 30% of Proposal deals (general rule of thumb) come through, this improves the predictability of a Sales (Multi) Area/Region.

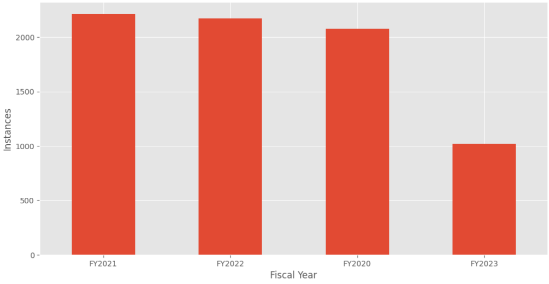


Figure 4.1: Instances per Fiscal Year

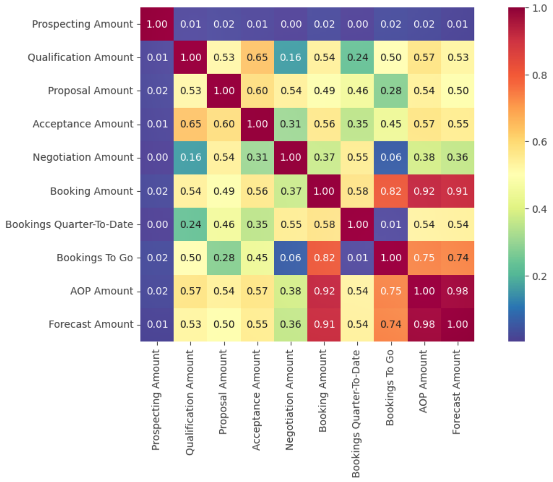


Figure 4.2: Correlation Matrix

Bookings Amount, AOP Amount, and Forecast Amount show the strongest correlation, but this was expected as the three are inter-dependent. Figure 4.3, Figure 4.4, and Figure 4.5 visualize their correlation and spreads. Finally, Figure 4.6 visualizes all numerical feature pairs in a pair plot.

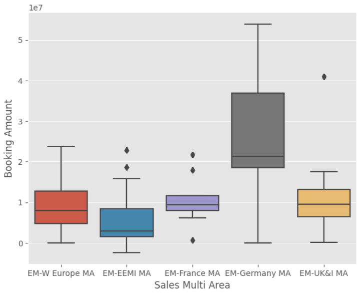


Figure 4.3: Bookings Boxplot

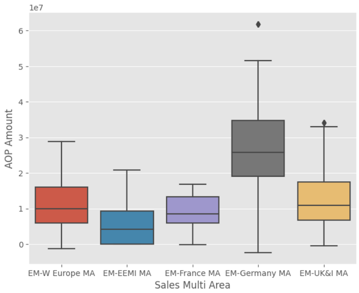


Figure 4.4: AOP Boxplot

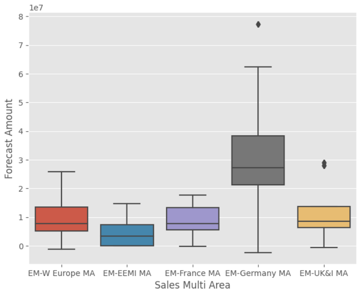


Figure 4.5: Forecast Boxplot

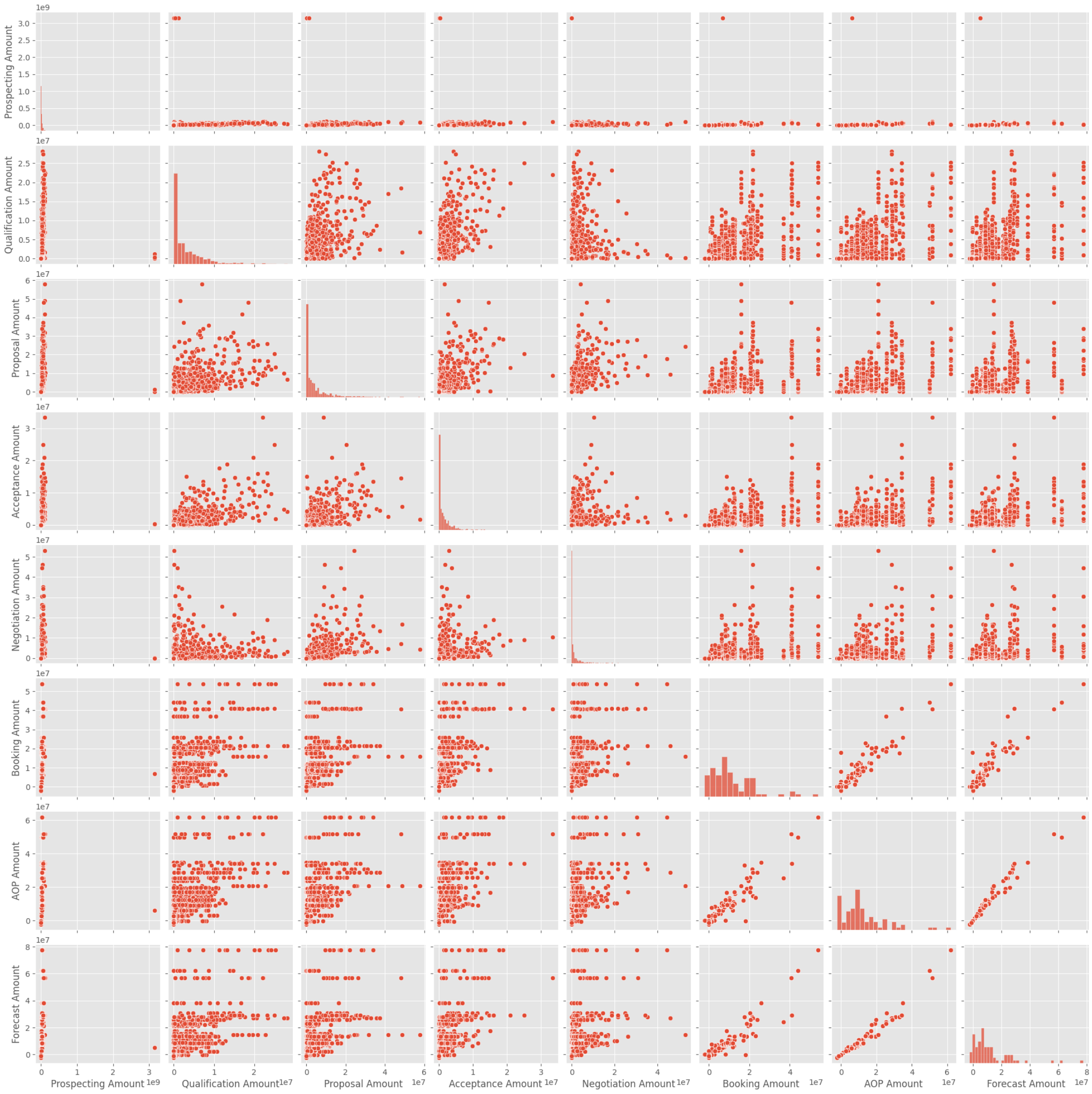


Figure 4.6: Pair Plot of Numerical Features

4.4 Feature Engineering

This thesis aimed to create an automated machine learning-driven version of the conventional sales predictor used in the organization. All manual and formula-driven calculations in the conventional predictor were created as new features in the preprocessed dataset to achieve this. Doing this was critical as it offered machine learning models (being tested) a better context of the data and the target variables.

Feature	**Data Type**
`Sales Multi Area`	Categorical (object)
`Sales Area`	Categorical (object)
`Sales Multi Region`	Categorical (object)
`Sales Region`	Categorical (object)
`Fiscal Year`	Categorical (object)
`Fiscal Quarter`	Categorical (object)
`Snapshot Week`	Categorical (object)
`Prospecting Amount`	Numerical (float64)
`Qualification Amount`	Numerical (float64)
`Proposal Amount`	Numerical (float64)
`Acceptance Amount`	Numerical (float64)
`Negotiation Amount`	Numerical (float64)
`Booking Amount`	Numerical (float64)
`Bookings Quarter-To-Date`	Numerical (float64)
`Bookings To Go`	Numerical (float64)
`AOP Amount`	Numerical (float64)
`Forecast Amount`	Numerical (float64)
`Prospecting Required`	Numerical (float64)
`Qualification Required`	Numerical (float64)
`Proposal Required`	Numerical (float64)
`Acceptance Required`	Numerical (float64)
`Negotiation Required`	Numerical (float64)
`Prospecting Probability`	Numerical (float64)
`Qualification Probability`	Numerical (float64)
`Proposal Probability`	Numerical (float64)
`Acceptance Probability`	Numerical (float64)
`Negotiation Probability`	Numerical (float64)

Table 4.3: List of Features After Feature Engineering

The 10 new features comprised calculated fields for required Sales Bookings and the probability of achieving it for each open Sales Stage. Table 4.3 lists all features and their data types after performing feature engineering on the preprocessed dataset.

4.5 Train and Test Data

With the new (critical) features in place, the dataset was ready for machine learning tasks. While the standard practice involves using 70-80% of the entire dataset for training and the remainder for testing, this approach wasn't followed. Instead, the training set was composed of data for the last 3 fiscal years i.e., FY2020 - FY2022, and the test set comprised data for the ongoing fiscal year (as of writing this report) i.e., FY2023. The train:test split ratio was approximately 84:16.

5

Machine Learning

This chapter discusses the machine learning approaches taken to build the automated version of the conventional sales predictor. CatBoost was chosen for the final deployment as it achieved the best prediction accuracy across all machine learning models tested.

5.1 Model Selection

The models to be experimented with were chosen based on findings from related work in Chapter 2 (Table 2.1). The 4 most popular machine learning algorithms for regression (based on performance and relevance) were picked, which have been listed below.

1. CatBoost
2. Random Forest
3. LightGBM
4. XGBoost

5.2 Model Training

The next step was training all the models and noting their behavior and performances. A key difference between the machine learning setup for this thesis and all related work was identified - this thesis involved multiple target variables whereas all others had one target variable. This suggested that all models being tested had to perform multi-output regression.

While Feature Selection is standard practice in machine learning, it was skipped entirely as all critical features were hand-picked at the beginning of the data preparation process. This resulted in the dataset being completely free of irrelevant features.

5.2.1 CatBoost Regressor

CatBoost is based on gradient-boosted decision trees where a set of decision trees is built consecutively during training. Each successive tree is built with reduced loss compared to the previous trees (13).

CatBoost supports the use of categorical features, eliminating the need for manual categorical encoding. All categorical features in the final dataset were passed to the `Pool()` class which converted them into CatBoost's special Pool datatype. Another feature of CatBoost is that it produces great results without the need for parameter tuning. However, tuning the parameters offered even better results. The model was trained on the final dataset using the parameters listed in Table 5.1 and its performance metrics were noted.

Parameter	**Value**
`learning_rate`	0.1
`depth`	3
`l2_leaf_reg`	3
`loss_function`	MultiRMSE
`eval_metric`	MultiRMSE
`od_type`	Iter
`bootstrap_type`	Bernoulli
`allow_const_label`	True
`early_stopping_rounds`	10
`use_best_model`	True

Table 5.1: CatBoost Regressor Training Parameters

5.2.2 Random Forest Regressor

Random Forest uses ensemble methods and creates numerous decision trees during training. The resulting output is the mean of means/modes of all individual trees (14).

Since Random Forest doesn't support categorical features, they were encoded using the Label Encoder in the scikit-learn library. The model was then trained on the encoded dataset to determine the baseline performance metrics. Next, a randomized search was performed to find the optimal hyperparameters using scikit-learn's `RandomizedSearchCV` method. The model was then trained again with the optimal parameters, listed in Table 5.2, to get the optimal performance metrics.

Parameter	**Value**
`max_depth`	50
`min_samples_leaf`	2
`min_samples_split`	5
`n_estimators`	2000
`random_state`	42

Table 5.2: Random Forest Regressor Training Parameters

5.2.3 LightGBM Regressor

LightGBM is a gradient-boosting ensemble method that is also based on decision trees. It generates decision trees that develop leaf-wise. Depending on the gain, just one leaf is split for each tree (15).

LightGBM can handle categorical features provided their datatype is `category`. Since the datatype of all categorical features in the dataset was `object`, they were all converted to `category` type to get them ready for use. The LightGBM model was encapsulated in scikit-learn's `MultiOutputRegressor` method to enable multi-output regression. The model was then trained on the transformed dataset using the optimal parameters listed in Table 5.3 and its performance metrics were noted.

Parameter	**Value**
`learning_rate`	0.11
`random_state`	42
`n_jobs`	-1

Table 5.3: LightGBM Regressor Training Parameters

5.2.4 XGBoost Regressor

XGBoost is an enhanced version of the GBM algorithm and uses a more regularized model which prevents overfitting. It works by training several decision trees where each tree is trained on a subset of the data. The resulting prediction is the combination of all individual trees (16).

XGBoost cannot handle categorical features, so they were encoded using the Label Encoder in the scikit-learn library. The model was then trained on the encoded dataset to determine the baseline performance metrics. Next, a randomized search was performed to find the optimal hyperparameters using scikit-learn's `RandomizedSearchCV` method. The model was then trained again with the optimal parameters, listed in Table 5.4, to get the optimal performance metrics.

Parameter	**Value**
`base_score`	0.5
`booster`	gbtree
`colsample_bylevel`	1
`colsample_bynode`	1
`colsample_bytree`	1
`grow_policy`	depthwise
`learning_rate`	0.1
`max_bin`	256
`max_cat_threshold`	64
`max_cat_to_onehot`	4
`max_depth`	6
`min_child_weight`	1
`n_estimators`	2000
`num_parallel_tree`	1
`predictor`	auto
`random_state`	42

Table 5.4: XGBoost Regressor Training Parameters

5.3 Model Evaluation

To evaluate the performances of all machine-learning models, six different metrics were recorded for each. Table 5.5 lists the performance numbers of each model tested.

1. **Train Score:** Scikit-learn's `score()` method was used to calculate this score based on X_train and y_train datasets (17).

2. **Test Score:** Scikit-learn's `score()` method was used to calculate this score based on X_test and y_test datasets (17).

3. **MAE:** The Mean Absolute Error indicates the magnitude of the difference between the prediction of an observation and the true value of that observation (18). It was calculated using scikit-learn's `mean_absolute_error` method on y_test and y_predicted datasets.

4. **RMSE:** The Root Mean Squared Error refers to the absolute fit of the model to the data (19). It was calculated using scikit-learn's `mean_squared_error` method on y_test and y_predicted datasets, taking the square root of the result.

5. **$\mathbf{R^2}$:** The Coefficient of Determination measures the goodness of fit of a regression model (20). It was calculated using scikit-learn's `r2_score` method on y_test and y_predicted datasets.

6. **Adj. $\mathbf{R^2}$:** Adjusted R^2 measures the variation in the target features explained only by the features which are helpful in making predictions. It was calculated using equation 5.1 where R^2 is the Coefficient of Determination, n is the number of data points, and p is the number of features in the model excluding the dependent features.

$$R^2_{adj} = 1 - \frac{(1 - R^2)(n-1)}{(n-p-1)} \tag{5.1}$$

Model	Train Score	Test Score	MAE	RMSE	$\mathbf{R^2}$	Adj. $\mathbf{R^2}$
CatBoost	0.9991	**0.9950**	0.0095	0.0266	**0.9929**	**0.9928**
Random Forest	0.9986	0.9919	0.0036	**0.0192**	0.9919	0.9919
LightGBM	0.9992	0.9914	0.0057	0.0313	0.9914	0.9913
XGBoost	**0.9999**	0.9852	**0.0033**	0.0203	0.9852	0.9846

Table 5.5: Model Comparison

5.4 Overfit Detection

The high training and testing scores of all models raised the concern of them overfitting the training data. To test this, various overfit detection and prevention methods were carried out. CatBoost's built-in overfitting detector methods `l2_leaf_reg` and `od_type` were set to values that would prevent overfitting (21). Figure 5.1 illustrates its Train vs Test RMSE Curve, visually indicating that it did not overfit.

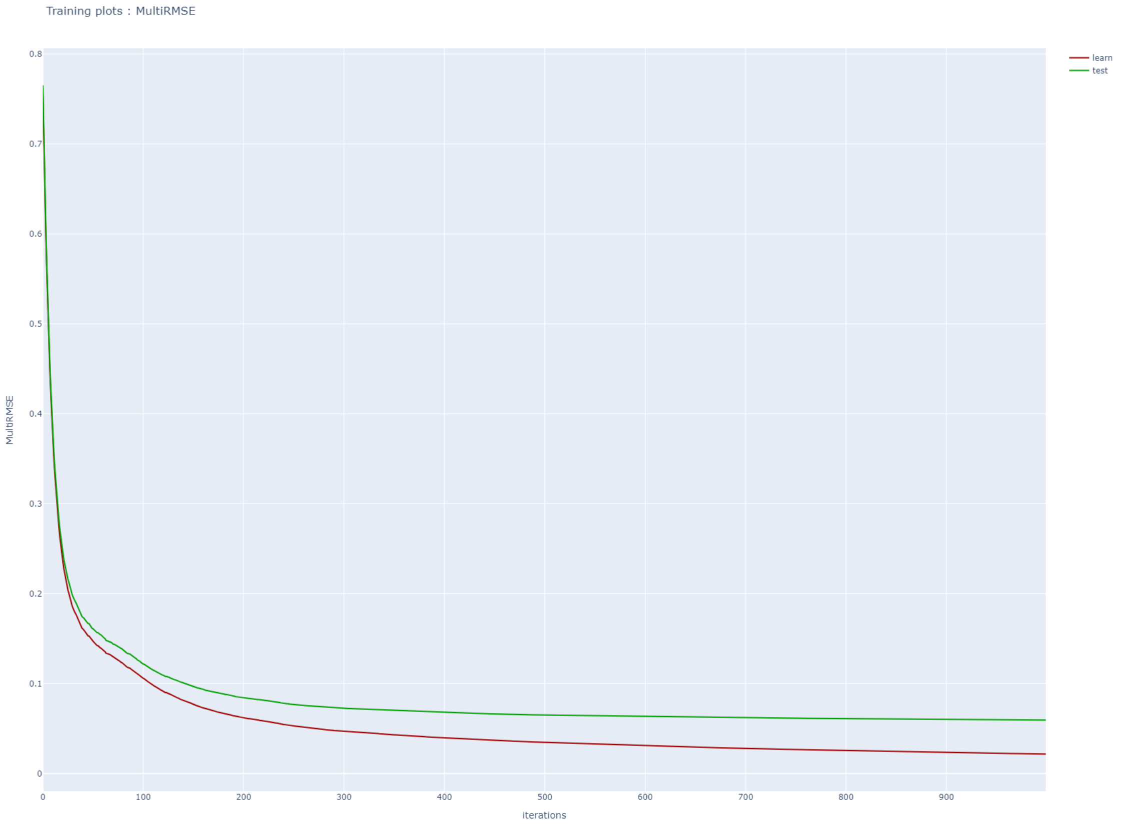


Figure 5.1: CatBoost Train vs Test RMSE Curve

Scikit-learn's `cross_val_score` function with a *cv* of 10 and *scoring* type 'r2' was used to calculate k-fold cross-validation scores for Random Forest, LightGBM, and XGBoost models (22). The resulting accuracy for each model was over 0.992, concluding that none of them were overfitting.

6

Model Deployment & Visualization

Tableau is a popular application that is particularly efficient at producing stunning interactive data visualizations. It is used in a variety of ways, such as creating charts, graphs, and maps, to analyze and visualize data. It is widely used at NetApp for visualizing all sorts of data as it reduces the time for analysis, encouraging everyone in the organization to be more data-driven.

Tableau's extension "TabPy" makes the process of deploying machine learning models and visualizing their inferences quick and easy. This chapter discusses the deployment of the CatBoost-based sales predictor model on Tableau using TabPy, Jupyter Notebook, and Microsoft SQL Server and building a user-friendly dashboard to visualize the model's predictions (inferences).

6.1 Deploying the Model & Connecting to Live Data

In order to make the final dashboard visualize the results of the trained CatBoost model, it had to be made fully automated. To achieve full automation, a data pipeline had to be established which would keep the dashboard updated at a set cadence.

6.1.1 Data Pipeline on SQL Server

All the sales metrics required for the dashboard were available on an SQL server, with the data refreshing daily. This made the server a suitable candidate for creating a data pipeline for the final Tableau dashboard. To achieve this, a stored procedure that joined and transformed the required data (sales metrics) was created, and a job was set up which would run on the server at 08:00 UTC every Monday. The resulting output of the job was a table with data in the required format along with the snapshot of the fiscal quarter week in a separate column. The flow of data has been illustrated in Figure 6.1.

6.1.2 CatBoost Model in Tableau

The Analytics Extension "TabPy" (the Tableau Python Server) enhances Tableau's functionality by enabling users to run Python scripts and stored functions through Tableau's table calculations. Tableau can connect to the TabPy server to run Python code instantly and show the results in Tableau visuals. Users can control the data and parameters fed to TabPy by interacting with Tableau worksheets, dashboards, or stories. It also enables the creation of calculated fields and can be used for deploying predictive (machine-learning) algorithms within Tableau (23).

Deploying the trained CatBoost model on Tableau was quick, courtesy of several tutorials available on the internet.

1. The TabPy server was launched and the listening port was noted.

2. In Tableau desktop, an Analytics Extension Connection was added after specifying the server (localhost) and the port from the previous step. Testing the connection confirmed that Tableau was successfully connected to the TabPy server.

3. To deploy the model, a function was defined in Jupyter Notebook with an argument for each variable in the trained CatBoost model. Then the deployment script was run on the TabPy server.

4. After successful model deployment, Tableau was connected to the SQL server and database where the data for the predictor is hosted. Then a live connection to the table in the database was created.

5. To bring the deployed CatBoost model on the TabPy server to the dashboard, a new calculated field named "Predictor" was created.

6. Finally, the rest of the elements of the dashboard were developed and designed to achieve an interactive visualization.

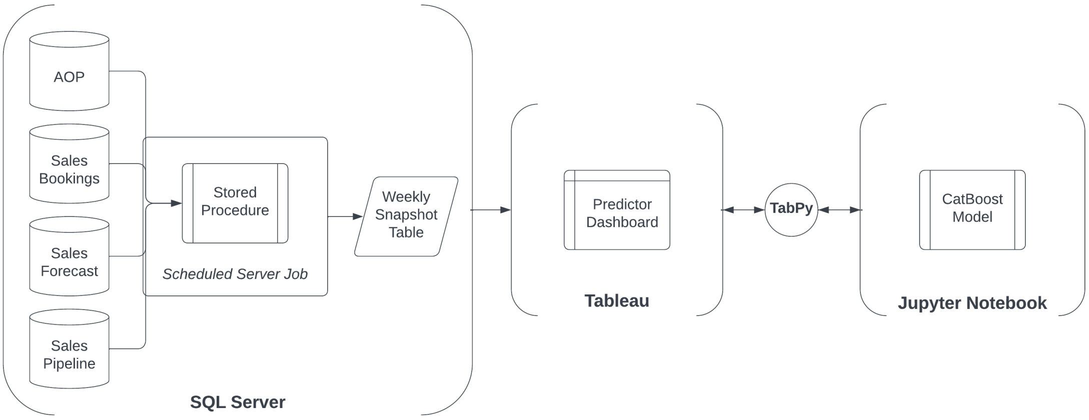
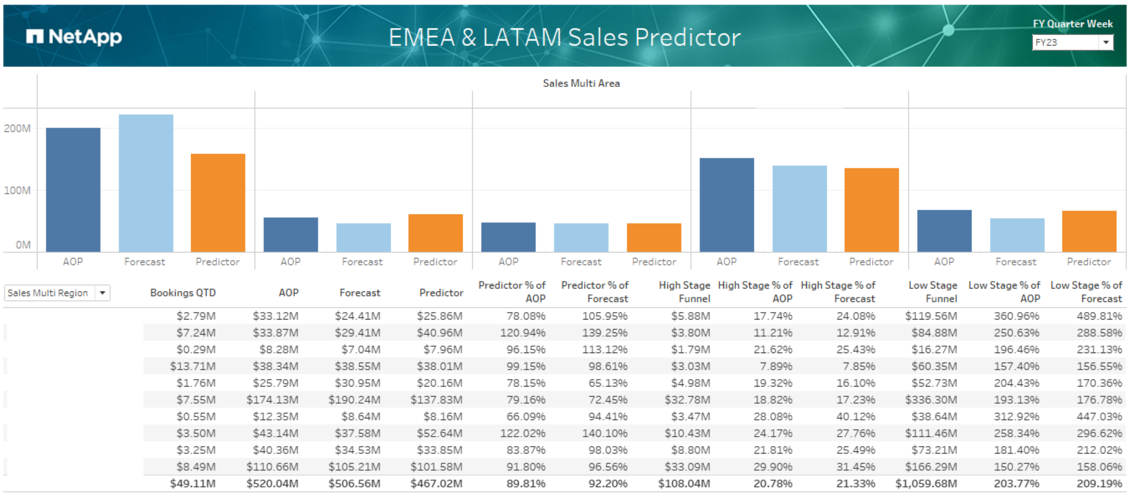


Figure 6.1: Data Flow between SQL Server, Tableau, TabPy, and Jupyter Notebook

6.2 Tableau Dashboard

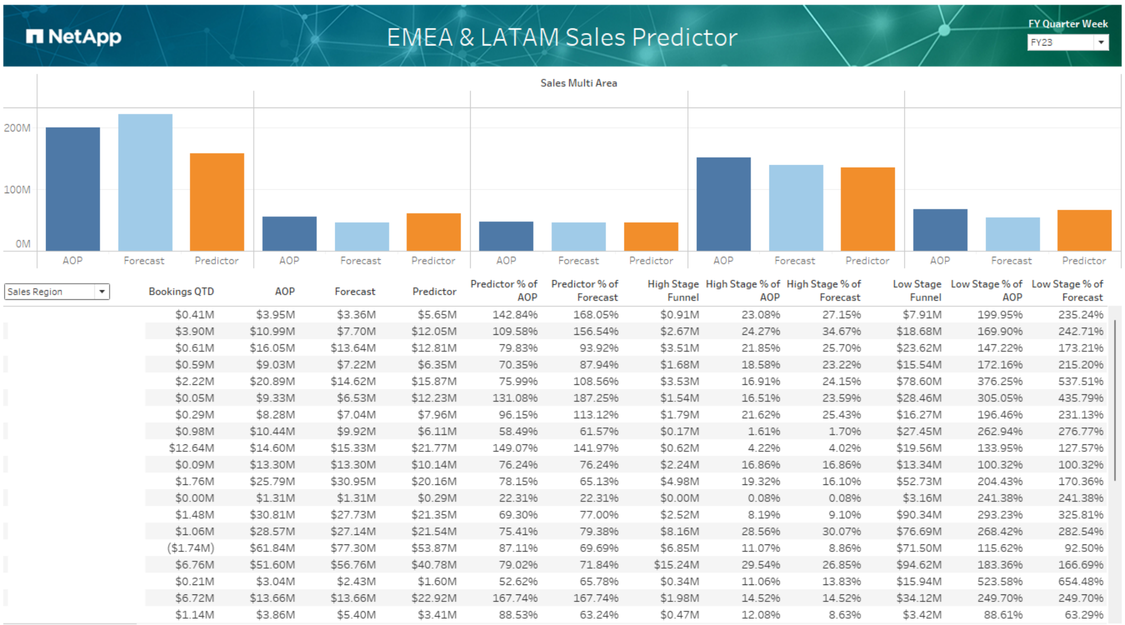
The final Tableau dashboard was designed as per the guidance received from the team managing the conventional sales predictor in the organization. It features:

- **A bar chart:** Shows a visual representation of how the AOP, Forecast, and Predictor numbers compare next to each other, for each Sales Multi Area in EMEA & LATAM.
- **A FY Quarter Week filter:** Allows the user to switch to the predictor dashboard for any week in the past (limited to the current fiscal year).
- **A table with metrics:** Shows the specific numbers for each sales metric. It also shows High and Low Stage Funnels and how they compare to High/Low Stage AOP and Funnel. The granularity of the table can be changed using the drop-down filter for the sales organization hierarchy. Levels of granularity include:
  - Sales Multi Area
  - Sales Area
  - Sales Multi Region (Figure 6.2)
  - Sales Region (Figure 6.3)



Sales Multi Region ▾	Bookings QTD	AOP	Forecast	Predictor	Predictor % of AOP	Predictor % of Forecast	High Stage Funnel	High Stage % of AOP	High Stage % of Forecast	Low Stage Funnel	Low Stage % of AOP	Low Stage % of Forecast
	$2.79M	$33.12M	$24.41M	$25.86M	78.08%	105.95%	$5.88M	17.74%	24.08%	$119.56M	360.96%	489.81%
	$7.24M	$33.87M	$29.41M	$40.96M	120.94%	139.25%	$3.80M	11.21%	12.91%	$84.88M	250.63%	288.58%
	$0.29M	$8.28M	$7.04M	$7.96M	96.15%	113.12%	$1.79M	21.62%	25.43%	$16.27M	196.46%	231.13%
	$13.71M	$38.34M	$38.55M	$38.01M	99.15%	98.61%	$3.03M	7.89%	7.85%	$60.35M	157.40%	156.55%
	$1.76M	$25.79M	$30.95M	$20.16M	78.15%	65.13%	$4.98M	19.32%	16.10%	$52.73M	204.43%	170.36%
	$7.55M	$174.13M	$190.24M	$137.83M	79.16%	72.45%	$32.78M	18.82%	17.23%	$336.30M	193.13%	176.78%
	$0.55M	$12.35M	$8.64M	$8.16M	66.09%	94.41%	$3.47M	28.08%	40.12%	$38.64M	312.92%	447.03%
	$3.50M	$43.14M	$37.58M	$52.64M	122.02%	140.10%	$10.43M	24.17%	27.76%	$111.46M	258.34%	296.62%
	$3.25M	$40.36M	$34.53M	$33.85M	83.87%	98.03%	$8.80M	21.81%	25.49%	$73.21M	181.40%	212.02%
	$8.49M	$110.66M	$105.21M	$101.58M	91.80%	96.56%	$33.09M	29.90%	31.45%	$166.29M	150.27%	158.06%
	$49.11M	$520.04M	$506.56M	$467.02M	89.81%	92.20%	$108.04M	20.78%	21.33%	$1,059.68M	203.77%	209.19%

Figure 6.2: Tableau Dashboard - Sales Multi Region view (Masked)



Sales Region ▾	Bookings QTD	AOP	Forecast	Predictor	Predictor % of AOP	Predictor % of Forecast	High Stage Funnel	High Stage % of AOP	High Stage % of Forecast	Low Stage Funnel	Low Stage % of AOP	Low Stage % of Forecast
	$0.41M	$3.95M	$3.36M	$5.65M	142.84%	168.05%	$0.91M	23.08%	27.15%	$7.91M	199.95%	235.24%
	$3.90M	$10.99M	$7.70M	$12.05M	109.58%	156.54%	$2.67M	24.27%	34.67%	$18.68M	169.90%	242.71%
	$0.61M	$16.05M	$13.64M	$12.81M	79.83%	93.92%	$3.51M	21.85%	25.70%	$23.62M	147.22%	173.21%
	$0.59M	$9.03M	$7.22M	$6.35M	70.35%	87.94%	$1.68M	18.58%	23.22%	$15.54M	172.16%	215.20%
	$2.22M	$20.89M	$14.62M	$15.87M	75.99%	108.56%	$3.53M	16.91%	24.15%	$78.60M	376.25%	537.51%
	$0.05M	$9.33M	$6.53M	$12.23M	131.08%	187.25%	$1.54M	16.51%	23.59%	$28.46M	305.05%	435.79%
	$0.29M	$8.28M	$7.04M	$7.96M	96.15%	113.12%	$1.79M	21.62%	25.43%	$16.27M	196.46%	231.13%
	$0.98M	$10.44M	$9.92M	$6.11M	58.49%	61.57%	$0.17M	1.61%	1.70%	$27.45M	262.94%	276.77%
	$12.64M	$14.60M	$15.33M	$21.77M	149.07%	141.97%	$0.62M	4.22%	4.02%	$19.56M	133.95%	127.57%
	$0.09M	$13.30M	$13.30M	$10.14M	76.24%	76.24%	$2.24M	16.86%	16.86%	$13.34M	100.32%	100.32%
	$1.76M	$25.79M	$30.95M	$20.16M	78.15%	65.13%	$4.98M	19.32%	16.10%	$52.73M	204.43%	170.36%
	$0.00M	$1.31M	$1.31M	$0.29M	22.31%	22.31%	$0.00M	0.08%	0.08%	$3.16M	241.38%	241.38%
	$1.48M	$30.81M	$27.73M	$21.35M	69.30%	77.00%	$2.52M	8.19%	9.10%	$90.34M	293.23%	325.81%
	$1.06M	$28.57M	$27.14M	$21.54M	75.41%	79.38%	$8.16M	28.56%	30.07%	$76.69M	268.42%	282.54%
	($1.74M)	$61.84M	$77.30M	$53.87M	87.11%	69.69%	$6.85M	11.07%	8.86%	$71.50M	115.62%	92.50%
	$6.76M	$51.60M	$56.76M	$40.78M	79.02%	71.84%	$15.24M	29.54%	26.85%	$94.62M	183.36%	166.69%
	$0.21M	$3.04M	$2.43M	$1.60M	52.62%	65.78%	$0.34M	11.06%	13.83%	$15.94M	523.58%	654.48%
	$6.72M	$13.66M	$13.66M	$22.92M	167.74%	167.74%	$1.98M	14.52%	14.52%	$34.12M	249.70%	249.70%
	$1.14M	$3.86M	$5.40M	$3.41M	88.53%	63.24%	$0.47M	12.08%	8.63%	$3.42M	88.61%	63.29%

Figure 6.3: Tableau Dashboard - Sales Region view (Masked)

7

Challenges & Limitations of Machine Learning in B2B Sales Forecasting

This chapter discusses the potential challenges and limitations one would face using the CatBoost-based model over its lifetime. A part of the challenges and limitations of the model faced relate to how the business operates while some directly relate to the model's behavior.

- **GARD Changes:** The GARD structure/sales organization hierarchy is subject to change on a yearly basis. As an example, three different Sales Multi Areas merged into one big Multi Area in FY23 (Table 7.1). This shows that there are substantial changes taking place between fiscal years. The model would not be able to identify these changes and would have to be re-trained according to the new hierarchy (when available) as a result.

Previous (FY22)	**Current (FY23)**
EM-Channel Led MA EM-IberoAmericas MA EM-ENT Focus MA	EM-EEMI MA

Table 7.1: Example of FY22 vs FY23 GARD Change

  Furthermore, data with the new GARD structure is not immediately available at the beginning of the new fiscal year. This could lead to inaccurate predictions by the model (working on the old hierarchy) for up to 1 month of the new fiscal year. This was also the case with the team managing the conventional sales predictor using Excel spreadsheets in the organization.

- **No Manual Intervention:** When looking at the conventional sales forecasting method, the CatBoost-based model would not be able to identify any "Whale" opportunities (ref. section 3.6) in the sales pipeline. This could lead to it producing fluctuating predictions which would invalidate the prediction for the GARD unit having the Whale opportunity.

- **Pipeline Hygiene:** The model relies on the stream of live data being clean and this comes down to how the data is being maintained by the sales organization i.e., Sales Pipeline Hygiene (ref. section 3.4). Good Pipeline Hygiene is imperative to accurate sales forecasts.

- **Model Drift:** Changes in the macroeconomic environment caused by global events could negatively impact the prediction performance of machine-learning-based sales forecasting models. For example, the COVID-19 pandemic affected the sales and revenue of businesses across the globe. Some businesses saw their sales fall while others witnessed sudden exponential growth. Machine-learning-based sales predictors in both cases would be affected negatively due to unusually low/high sales pipeline numbers.

8

Conclusion

This thesis was built upon the concept of the machine-learning-driven B2B sales predictor developed by K. M. Kasinathan (2) for NetApp. Predicting sales based on opportunity status as open or closed offered quality results. However, all the active sales stages an opportunity goes through during its lifecycle remained unexplored.

Instead of looking at Won and Lost sales opportunities, this thesis explored all Sales Stages that an opportunity goes through during its lifecycle in the organization. As a result, an all-stage sales predictor was developed for this thesis. The rationale was that including all stages in the predictor could produce better sales forecasts. Furthermore, the predictor model in this thesis was built on the logic of the existing (conventional) sales predictor in the organization to obtain the best possible predictions.

The experiments carried out had favorable outcomes, which helped answer the research questions formulated in the beginning.

RQ.1 **Does the introduction of all sales stages of opportunities have a positive impact in terms of prediction accuracy?**

The goal was to identify the degree to which the inclusion of all Sales Stages of opportunities helped improve the prediction accuracy. As weekly snapshots of data required (with all sales stages) was not available on any of the tools in the organization, it was obtained from the team managing the conventional sales predictor.

All machine learning models tested achieved over 98% regression accuracy, a significant improvement (+120%) over K. M. Kasinathan's model which was 83% accurate.

The results were also comparable to that of the conventional sales predictor which has prediction accuracy in the range of 95%-105%.

The results made it evident that the inclusion of all Sales Stages of opportunities had a positive impact on prediction accuracy to a great degree.

RQ.2 **Does CatBoost continue to produce the best predictions with the added sales stages, or is there a machine learning model that performs even better?**
During experimentation, CatBoost was tested alongside Random Forest, LightGBM, and XGBoost, as the aim was to identify which model performed the best in terms of prediction accuracy. The results were fascinating as all hyperparameter-tuned models performed equally well, with the difference in test accuracy being <2% between the best and worst performers. All other model evaluation metrics recorded were also comparable across all four.

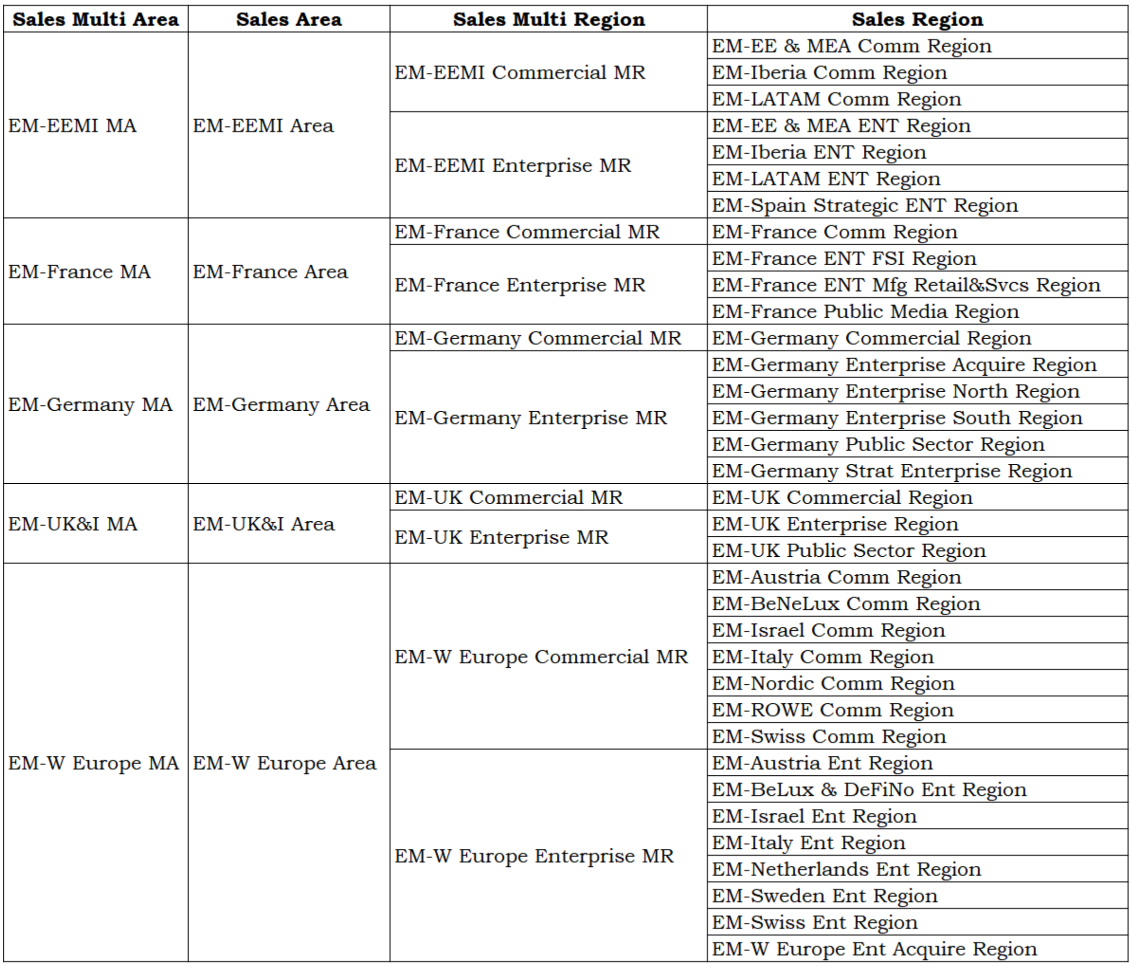
CatBoost performed the best of the four models and achieved 99.5% accuracy on the test set, which was composed of sales data from the recent past (as of writing this report). Therefore, it can be concluded that CatBoost continues to produce the best predictions even with added sales stages, albeit only by a minuscule margin.

RQ.3 **In what ways can the results produced by the predictor be utilized for business-related analytics?**
The predictor's main task is to forecast future sales of the company. From internal knowledge and experience, this data could also be used by the senior management of the Sales Operations and Finance departments to generate AOP, Forecast, and Quota/Target for the GARD and the Sales Management (Sales District Managers, Sales Regional Managers, etc.). Specifically, AOP and Forecast details rely on various factors, which also consider the conversion rates (%) produced by the predictor.

The results of this thesis were promising as a near-identical-performing version of the conventional sales predictor in the organization was built using machine learning techniques. It could be argued that the CatBoost-based predictor model could replace the conventional predictor, but its limitations prevent that from happening. There are certain things that are possible to do with the conventional predictor, which simply are not with machine-learning-based models i.e., manually handling anomalies such as Whale opportunities (ref. Section 3.6) in Sales Pipeline data. As part of future work, the model can be trained to identify Whale opportunities and omit them to improve the reliability of the model and its output.

Appendix A

FY23 EMEA & LATAM GARD



Sales Multi Area	Sales Area	Sales Multi Region	Sales Region
EM-EEMI MA	EM-EEMI Area	EM-EEMI Commercial MR	EM-EE & MEA Comm Region
			EM-Iberia Comm Region
			EM-LATAM Comm Region
		EM-EEMI Enterprise MR	EM-EE & MEA ENT Region
			EM-Iberia ENT Region
			EM-LATAM ENT Region
			EM-Spain Strategic ENT Region
EM-France MA	EM-France Area	EM-France Commercial MR	EM-France Comm Region
		EM-France Enterprise MR	EM-France ENT FSI Region
			EM-France ENT Mfg Retail&Svcs Region
			EM-France Public Media Region
EM-Germany MA	EM-Germany Area	EM-Germany Commercial MR	EM-Germany Commercial Region
		EM-Germany Enterprise MR	EM-Germany Enterprise Acquire Region
			EM-Germany Enterprise North Region
			EM-Germany Enterprise South Region
			EM-Germany Public Sector Region
			EM-Germany Strat Enterprise Region
EM-UK&I MA	EM-UK&I Area	EM-UK Commercial MR	EM-UK Commercial Region
		EM-UK Enterprise MR	EM-UK Enterprise Region
			EM-UK Public Sector Region
EM-W Europe MA	EM-W Europe Area	EM-W Europe Commercial MR	EM-Austria Comm Region
			EM-BeNeLux Comm Region
			EM-Israel Comm Region
			EM-Italy Comm Region
			EM-Nordic Comm Region
			EM-ROWE Comm Region
			EM-Swiss Comm Region
		EM-W Europe Enterprise MR	EM-Austria Ent Region
			EM-BeLux & DeFiNo Ent Region
			EM-Israel Ent Region
			EM-Italy Ent Region
			EM-Netherlands Ent Region
			EM-Sweden Ent Region
			EM-Swiss Ent Region
			EM-W Europe Ent Acquire Region

References

[1] **Beginner's guide to sales forecasting methodology**. 1, 12

[2] Kailainathan Muthiah Kasinathan. *Infer - A Full Scale B2B Sales Predictor*. Master's thesis, Vrije Universiteit Amsterdam | Universiteit van Amsterdam, 2021. 2, 5, 37

[3] Alireza Rezazadeh. **A Generalized Flow for B2B Sales Predictive Modeling: An Azure Machine-Learning Approach**. *Feature Papers of Forecasting*, **2(3)**:267–283, 2020. 5

[4] Tiemo Thiess, Oliver Müller, and Lorenzo Tonelli. **Design Principles for Explainable Sales Win-Propensity Prediction Systems.** In *Wirtschaftsinformatik (Zentrale Tracks)*, pages 326–340, 2020. 6

[5] Marko Bohanec, Mirjana Kljajić Borštnar, and Marko Robnik-Šikonja. **Integration of machine learning insights into organizational learning**. *Proceedings of 28th Bled eConference, Bled, Slovenia*, 2015. 6

[6] Stephen Mortensen, Michael Christison, BoChao Li, AiLun Zhu, and Rajkumar Venkatesan. **Predicting and defining B2B sales success with machine learning**. In *2019 Systems and Information Engineering Design Symposium (SIEDS)*, pages 1–5. IEEE, 2019. 6

[7] Junchi Yan, Min Gong, Changhua Sun, Jin Huang, and Stephen M Chu. **Sales pipeline win propensity prediction: A regression approach**. In *2015 IFIP/IEEE International Symposium on Integrated Network Management (IM)*, pages 854–857. IEEE, 2015. 7

[8] D Rohaan, E Topan, and Catharina GM Groothuis-Oudshoorn. **Using supervised machine learning for B2B sales forecasting: A case study of**

spare parts sales forecasting at an after-sales service provider. *Expert systems with applications*, **188**:115925, 2022. 7

[9] M. Priya Alagu Dharshini and S. Antelin Vijila. **Survey of Machine Learning and Deep Learning Approaches on Sales Forecasting**. In *2021 4th International Conference on Computing and Communications Technologies (ICCCT)*, pages 59–64, 2021. 9

[10] **The Complete Guide to Writing an Annual Operating Plan**. 11

[11] **Bookings vs. Revenue in Sales**. 12

[12] **What are the Stages of a Sales Pipeline?** 12

[13] **CatBoost - How training is performed**. 26

[14] **A Quick Guide to Random Forest Regression**. 26

[15] **How LightGBM algorithm works**. 27

[16] **How XGBoost Works**. 28

[17] **Metrics and scoring: quantifying the quality of predictions**. 29

[18] **Metrics and scoring: sklearn.metrics.mean_absolute_error**. 29

[19] **Metrics and scoring: sklearn.metrics.mean_squared_error**. 29

[20] **Metrics and scoring: sklearn.metrics.r2_score**. 29

[21] **CatBoost: Using the overfitting detector**. 30

[22] **Metrics and scoring: sklearn.model_selection.cross_val_score**. 30

[23] **TabPy**. 32