Politecnico
di Torino

Corso di Laurea magistrale in Ingegneria Gestionale

percorso Supply Chain Design

Tesi di Laurea Magistrale

Optimizing Project Cost and Duration Forecasting through Machine Learning

Relatore

Alberto De Marco

Correlatore

Filippo Maria Ottaviani

Candidato

Stefano Francios

2025

“Without data, you are just

another person with an opinion”

W. Edwards Deming

Table of Contents

1. Abstract

The ability to accurately predict the duration and final costs of a project is essential to implement timely corrective actions during its execution. Estimates at Completion (EAC) are the key tool to support project control. In recent years, the integration of Machine Learning (ML) techniques with traditional methodologies such as Earned Value Management (EVM) and Earned Schedule (ES) has improved the accuracy of forecasting. However, many studies have focused exclusively on static data, neglecting both the influence of dynamic data and topological indicators and the issues of underfitting and overfitting that compromise model robustness. This thesis aims to address both issues by proposing a structured ML pipeline, including automated preprocessing, feature engineering, and cross-validation procedures, with the goal of improving the generalizability of predictive models. The pipeline is designed to exploit both static data and dynamic data and topological network indicators by analyzing their impact on predictive performance. The pipeline was tested using 30 machine learning algorithms on a dataset of 90 real projects by evaluating their effectiveness with respect to mean square error, project progress stage, and via SHAP analysis for interpretability.

The findings indicate that ML algorithms outperform classical approaches concerning accuracy and precision particularly during the initial and mid phases of a project. Moreover, feature analysis with SHAP underscored the tremendous value of dynamic data and project network attributes concerning model prediction capability.

In conclusion the study demonstrates the effectiveness of the proposed pipeline and suggest that further integration of ML in project management practices could lead to improved project outcomes, especially as ML techniques continue to evolve.

2. Introduction

Effective project management is often hindered by uncertainties and variations that arise in the planning and execution phases. In particular, the ability to accurately estimate project duration and costs at completion (EACs) is a crucial element in activating corrective strategies and ensuring alignment with contractual objectives.

The EAC is typically calculated through methodologies based on Earned Value Analysis (EVA) that focus on the project's monetary and temporal achievements, including its valuation through methods such as Earned Value Management (EVM) or scheduling through Earned Schedule (ES). However, both the EVM and ES techniques rely on rigid and sequential formulas which do not accommodate the inherent fluidity and intricacy of actual working projects. Their ability to adjust to new situational frameworks and deal with unexpected changes in project activity or related circumstances is limited.

These methods' shortcomings have open to the opportunity to employ Machine Learning (ML) techniques. Because ML models can learn complex non-linear patterns, heterogeneous data, and more precise solutions to a given problem, they provide more flexibility than traditional methods. However, there is still a gap in research concerning the use of static data, dynamic data, simulating actual project progress over time, and topological data detailing the project's network configuration.

Dynamic project data reflect the actual progress of activities, including deviations from initial time and cost estimates. Unlike static parameters generated by standardized networks and resources, dynamic data must represent realistic scenarios, which makes them more complex to model. Indeed, the actual execution of a project is characterized by uncertain events and unforeseen variations that are difficult to capture by simplified assumptions. The use of unrealistic, or overly idealized, data compromises the quality of the results and their practical applicability.

This thesis proposes an ML pipeline that aims to fill these gaps by integrating both static and dynamic data, along with topological network indicators (e.g., Serial/Parallel Index and Regularity Index). The objective is twofold: on the one hand, to evaluate the effectiveness of ML models in predicting EAC and TEAC (Time Estimate at Completion) on real projects; on the other hand, to analyze by SHAP analysis the information impact of different features to understand which variables contribute most to the accuracy of the predictions.

The thesis is structured as follows: Chapter 3 presents a literature review that introduces the project management framework and analyzes in detail the monitoring and control practices, the use of Machine Learning in the sector and the different categories of data (static, dynamic and topological) considered in this thesis. In Chapter 4, the adopted methodology is provided along with the developed ML pipeline, pinpointing the causes of underfitting and overfitting and how they were attempted to be solved. In Chapter 5, the performance of the ML model is compared with the performance of traditional methodologies (EVM and ESM) and in Chapter 6, the focus is on the theoretical and practical implications of the main results discussed. Finally, in Chapter 7, the work's contributions are summarized, followed by its limitations and proposed future research directions.

3. Literature Overview

3.1. Basic of Project Management

Project Management (PM) is the discipline that involves planning, organizing, and managing resources to ensure the effective achievement of a project's goals and objectives. As stated by the Project Management Institute (PMI), PM "involves the application of knowledge, skills, tools, and techniques to project activities to meet the project requirements. Project management refers to guiding the project work to deliver the intended outcomes. Projects are temporary endeavors with a defined beginning and end, undertaken to create a unique product, service, or result" [1].

Always according to the PMI a project is "a temporary endeavor undertaken to create a unique product, service, or result". The temporary nature of projects implies that each project, or phase within a project, has a clearly defined start and finish. A project may exist independently or be integrated within a broader program or portfolio.

Project management is particularly effective when the entire project architecture is oriented toward deliverables. A deliverable represents an item, tangible or intangible, produced as a result of an activity or the entire project. It may refer to the overall output of the project, a single phase, or even a single activity [2].

It is essential that, from the earliest stages, the project manager guide the team in breaking down the work into activities each of which produces clear and well-defined deliverables. These deliverables, in addition to representing the tangible result of an activity, also serve as inputs for subsequent activities. A project that does not precisely define its deliverables risks running into ambiguities, operational inefficiencies and monitoring difficulties.

Although the term "output" is often used synonymously with deliverables, it is useful to distinguish them from the results ("outcomes") of a project. Outcome, according to the PMI is "An end result or consequence of a process or project. Outcomes can include outputs and artifacts but have a broader intent by focusing on the benefits and value that the project was undertaken to deliver" [1]. Thus, the results include deliverables but extend beyond the formal conclusion of the project, encompassing the benefits and value generated over time.

This distinction reflects two different project approaches: the deliverable-oriented one, typical of industrial projects or linked to clearly identifiable physical products, and the result-oriented one, more suited to organizational change projects or, in general, agile contexts. In the former case, the focus is on the control and production of tangible elements; in the latter, the emphasis is on the value generated, although less immediately visible or measurable.

Although it is now considered a partially outdated model and has been gradually abandoned in newer project management texts and courses, the so-called Triangle of Constraint continues to be used in some professional practices to facilitate reflection and framing of project constraints. [3]

The triangle graphically represents the interdependence among three fundamental dimensions of a project: time, cost and resources, and quality [Figure 3-1]. The inner area of the triangle symbolizes the purpose of the project, while the sides indicate the three constraints that must be managed in a coordinated manner. The principle behind the model is that a variation in one of the elements will inevitably affect the other two. For example, to reduce time while keeping quality the same, it is generally necessary to increase resources, and thus costs. Conversely, if a client requests a reduction in time without increasing the budget, a downward revision of expected quality may be necessary.

The triangle has come under criticism for its limited view of the variables involved in managing a project. However, in operational reality many organizations still evaluate the success of a project based on the consistency between the time, cost and quality constraints defined at the outset and those actually achieved. The model can still be a useful decision-support tool, especially in situations where the project manager must act promptly to manage critical issues emerging during project execution.

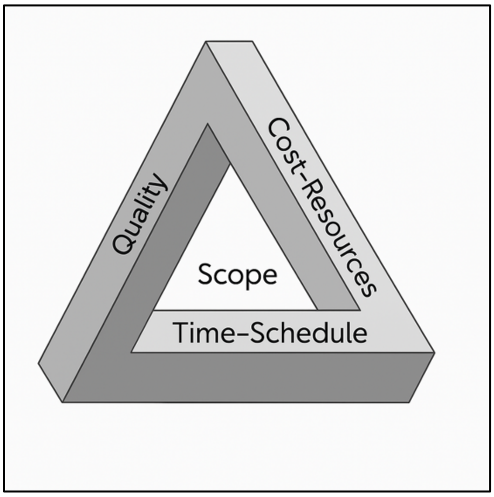

Figure 3-1, Project Management Triangle of Constraints

3.1.1. Historical Evolution of Project Management

Project management practices historical development highlights a significant progression from informal practices to structured methodologies. Project management has accompanied humanity since its origins, as evidenced by the great works of antiquity: the Egyptian pyramids, the Great Wall of China, Roman infrastructure and medieval cathedrals. However, in these contexts there were no codified methods or dedicated management figures: leadership was entrusted to technical experts or small groups with absolute decision-making power.

The Gantt chart is the first formally recognized tool of project management. It was created in 1917 by engineer Henry Gantt. The chart displays sequential activities, showing the time within which each is expected to be completed. In fact, Adamiecki developed the “harmonic” version of the chart called an armonogram in 1896. Adamiecki published his work in Russian and Polish, which made it difficult for most English speakers to access the information. Gantt popularized a similar graph in the US in 1910. This graph was intended to show the time spent by employees on specific tasks. These two systems have since merged into what is regarded today as the modern Gantt chart.

Its wide adoption during the World War I, particularly for industrial and construction projects, emphasized its significance in the management of resource-intensive projects. The initial techniques showcased an increased understanding regarding the need for timeliness and responsibilities in project control which prepared the ground for advancement in methodologies. These techniques, however, did not provide a suitable solution for complex and ambiguous project situations, hence highlighting the requirement for more powerful and flexible techniques.

The expression “project management” was first used in 1953 in association with the aerospace industry in America. These years saw the emergence of several models and techniques that are still basic today; in 1957 the DuPont Corporation created the Critical Path Method (CPM) and in 1958 the U.S. Navy as a part of Polaris program developed Program Evaluation Review Technique (PERT), followed by the Department of Defense introducing Work Breakdown Structure (WBS) in 1962.

Although these three methods had their evolution over the years, they still stand useful in modern-day project management.

The WBS represents the tree-like, top-down decomposition of the project scope down to the atomic level of activities. The PERT highlights the logical links between project activities, between predecessors and successors, and is a very powerful tool for reasoning about execution strategies. The critical path method (CPM), on the other hand, focuses on chains of “critical” activities that, having no margin, cause a delay equal to their own on the completion of the entire project.

By the late 1950s, the figure of the project manager was emerging, at least among those in the industry, as a professional with a specific role of integration and coordination in complex initiatives. This concept was first formalized in an article published by the Harvard Business Review in 1959 [4].

The second half of the 1960s marked the birth of the major professional associations dedicated to the discipline: the International Project Management Association (IPMA) in 1965 and the Project Management Institute (PMI) in 1969. Even today, these two institutions represent the main references for the international project management community.

In the late 1970s, the need emerged for professional certification to attest to practitioners' competencies and, consequently, the need for a reference text that would systematically compile the fundamental knowledge of the discipline. This led to the publication in 1983 of the first PMI Body of Knowledge, compiled by PMI, which initially identified six areas of knowledge. The work took its final name of Project Management Body of Knowledge (PMBOK) with the 1987 edition, to which new areas were added, including risk management and procurement. Later, the area of integration is introduced in the 1996 edition, while soft skills and stakeholder management become an integral part of the corpus only in the 2000s.

In 1989, Earned Value Analysis and consequently the method of Earned Value Management, systems for monitoring project progress, was formalized. It makes it possible to compare the planned baseline with actual progress in percentage terms and provides tools for forecasting future performance. A dedicated chapter will further explore its principles and applications.

The late 1990s witnessed a significant conceptual evolution with the emergence of the Agile movement, which challenged the universal effectiveness of traditional methods, up to that time collected in the various Bodies of Knowledge. Indeed, it is recognized that for certain types of projects, characterized by high uncertainty - such as software development or research and development - predictive methods are unsuitable. Therefore, the need to also adopt adaptive, more flexible and dynamic approaches, capable of reacting quickly to changing objectives or context, is affirmed.

The late 1990s witnessed a significant conceptual evolution with the emergence of the Agile movement, which challenged the universal effectiveness of traditional methods, up to that time collected in the various Bodies of Knowledge. It became evident that for certain types of projects characterized by high uncertainty because traditional project management methods, although highly structured and widely practiced, often struggled to cope with these kinds of projects especially those involving multidisciplinary teams and uncertain environments [5]. These traditional methods were typically based on deterministic assumptions, which led to rigid planning frameworks ideated to address unforeseen challenges, such as resource shortages or market fluctuations

Consequently, the need to adopt more adaptive, flexible, and dynamic methodologies, capable of responding rapidly to shifting objectives or contextual changes, gained increasing recognition.

The publication of the Agile Manifesto in 2001 represents a landmark moment in the project management debate. However, it has also generated some misinterpretations: people often associate "agile" with a lack of planning or organizational disorder, when in fact it is a highly structured approach, albeit planned over shorter time horizons than traditional methods.

In the first two decades of the 21st century, traditional and agile project management were often perceived as two opposing schools: more orthodox practitioners tended to favor predictive approaches, while others, sometimes with a more superficial approach, adopted agile methods. In recent years, however, an integrated and more balanced view is emerging. As of 2021, PMI will in fact require PMP certification candidates to have in-depth knowledge of both traditional and agile methodologies.

This exploration of how project management techniques have changed over time demonstrates a continual shift toward methodologies that prioritize flexibility and adaptability.

3.1.2. Structural and Visual Tools for Managing Projects

The effectiveness of a project's planning and control depends significantly on the ability to represent in a clear, structured and integrated manner both the work to be done and its temporal articulation and progress. In this context, graphical and methodological tools assume a key role in supporting the project manager in decision-making and day-to-day management of activities.

Among the main tools used, three stand out for their widespread use and versatility: the Work Breakdown Structure (WBS), the Gantt chart and the S-curve. Each responds to a specific project management need: the WBS permits breaking down the work to manageable levels, assignment of the scope is easier along with controlling the responsibilities; in the Gantt chart the temporal sequence of activities and their execution can be visualized and monitored; and alongside performing basic calculations to evaluate the performance accurately, those estimates aid analyzing finances and evaluating the overall project progress through the S curve, which represents the accumulated project progress.

The integrated adoption of these tools, in both traditional and agile or hybrid contexts, represents an established best practice for structured, transparent and effective project management. The following sections delve into the features, applications and benefits of each of these tools.

Work Breakdown Strucuture (WBS)

In the project planning process, each sector adopts specific tools to represent the product or expected outcome of the project: in software, structured diagrams; in film, storyboards with settings and actors. In a similar way, in project management, the Work Breakdown Structure (WBS) is one of the most employed tools for representing the work to be done in a structured way [Figure 3-2].

Defined by the Project Management Institute as a “ ahierarchical decomposition of the total scope of work to be carried out by the project team to accomplish the project objectives and create the required deliverables” [1] the WBS allows a complex project to be broken down into manageable components, each with an increasing level of detail as you move down the hierarchy.

This progressive decomposition allows the project manager to tackle the project “one bite at a time!” making it easier to control, assign responsibilities and reduce operational ambiguities [6].

While it may appear similar to an organizational chart or structured bill of materials (BOM), the WBS is clearly distinguished by its purpose: to represent the work to be done, not the people involved or the final objects produced: “a good WBS provides a common framework for all project deliverables and for specific tasks within the project” [7].

The creation of a WBS is a simple process that can be initiated even on informal media with notes written on a paper. However, for detailed analyses to be effective, there is a need to incorporate arrangement at multiple levels within the project’s hierarchy. Text processors enable the creation of WBSs in an easy manner due to content WBSs through various levels of indentations and editing options. In office environments, WBSs are created and presented in tree or outline formats using specialized project management software capable of scheduling and generating network diagrams.

It is important to highlight that the entire content of a project must be represented in the WBS, what is not included is not formally part of the project. Any missing activities require explicit permission to be added, either in the form of an approved change or acknowledgement of an oversight.

In fact, the WBS should be constructed to meet the operational needs of the individual project, rather than tailored for accounting purposes. For comparative cost analysis or financial purposes, parallel tools such a coding of work packages or dedicated structures such as Financial Breakdown Structure (FBS) or Component Breakdown Structure (CBS) can be adopted [6].

A WBS should focus on the work that needs to be done to complete the project. Each level of the hierarchy should represent a deliverable: level 0 corresponds to the final goal, while subsequent levels may contain either physical product components or documentary elements, such as reports or analysis. Some activities, such as project management, may find a place in elements of the WBS that do not produce deliverables but are nonetheless essential to overall success.

In addition to its hierarchical structure, the WBS has some key features for effective project management [2]:

- Uniqueness

  There must be only one WBS for the entire project. No separate WBSs are created for each department (e.g., design, purchasing, production): each part of the team manages its own work packages within a single shared structure.
- Exhaustiveness

  The WBS must cover the entire scope of the project, including all expected products, services and outcomes. What is not included is not part of the project.
- Stability

  Once defined, the WBS should not be changed during the project unless contractual changes are made. If it is revised too often, it is a symptom of poor initial planning.
- Simplicity and clarity

  The structure should be easy for everyone to understand, and each activity/package should be assignable to one person responsible, as well as allowing progress to be monitored.

Finally, it should be noted that the WBS does not represent the temporal or logical relationships between activities-these connections are modeled separately within project network diagrams (PND). However, the way the WBS is structured can significantly influence the organization and execution of the work, making it essential to follow precise rules in its elaboration [6].

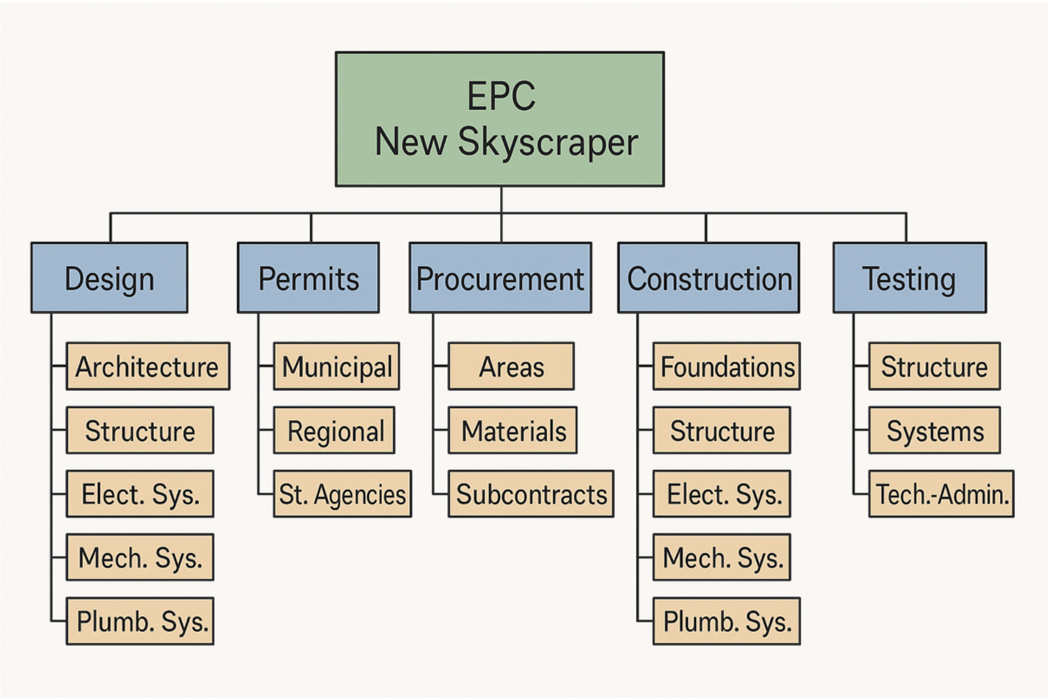

Figure 3-2, high level WBS for Engineering, Procurement and Construction of a skyscraper, taken from [2]

Gantt Chart

As mentioned in the chapter on the historical evolution of PM, the Gantt chart is one of the most enduring and representative tools, developed in the early 20th century and it still is a widely used tool in the planning and time control of projects, even outside purely traditional contexts. Although derived from a deterministic approach, it is also often used in projects with hybrid or, to a more limited extent, agile life cycles.

Defined by the Project Management Institute as a "bar chart provides schedule information where activities are listed on the vertical axis, dates are shown on the horizontal axis, and activity durations are shown as horizontal bars placed according to start and finish dates" [1], takes the form of a time schedule that combines, on the left side, a list of activities and work packages (derived from the WBS) with information on duration and expected dates; on the right side, a calendar displays these activities as time bars proportional to their duration. Relationships between activities are represented by arrows, while milestones, key points in the project, are indicated by specific symbols (usually rhombuses). Higher-level activities that encapsulate multiple sub-activities are displayed with distinctive marks to make the hierarchy easier to read [8].

The employment of Gantt charts focuses on the following objectives:

- Manage complex projects: larger projects involve more tasks to manage. Gantt charts enable project managers to better visualize a project by breaking it down into smaller activities.
- Monitor dependencies between activities: a project can be behind schedule and in these cases Gantt charts help project managers automate dependencies between activities to ensure that the next task does not start until the previous one is finished.
- Monitor project progress: Gantt charts allow to track progress and milestones, in order to easily modify the project plan if necessary.

Once approved, the Gantt chart becomes the project's schedule baseline, the timeline against which progress can be monitored. It is important to consider that scheduling software and tools plan activities "as soon as possible" (ASAP) by default, but this mode is not always optimal. In some cases, it may be preferable to schedule "as late as possible" (ALAP) or to adopt intermediate solutions, carefully weighing margins and risks [Figure 3-3].

If it becomes necessary to compress the project duration, two strategies can be adopted:

- Crashing: it is the allocation of additional resources to speed up the execution of a task, resulting in increased costs.
- Fast-tracking: it entails parallel execution of previously planned activities in sequence, which can increase the risk of failure in case of technical or coordination problems.

The crash strategy results in increased costs; fast-tracking, on the other hand, increases the risk, as executing tasks in parallel eliminates the possibility of catching up with delays accumulated in previous phases.

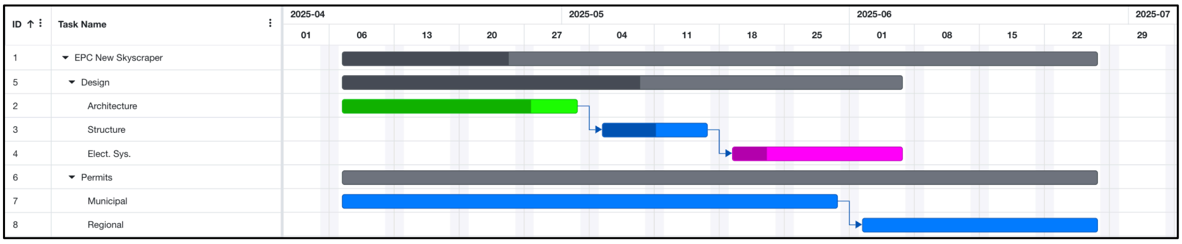

Figure 3-3, Partial view of the Gantt chart related to the Engineering, Procurement and Construction of a skyscraper

S-curve

The S-curve is a graphical representation commonly used in project management to describe the cumulative trend of cost, hours worked, or physical progress over time. [9]

In the context of Earned Value Management (EVM), S-curves take a central role in the visual representation of the three main performance indicators: the Planned Value (PV), Earned Value (EV) and Actual Cost (AC). Plotted on a graph with time on the horizontal axis and cost on the vertical axis, these curves allow us to observe the evolution of the project and provide an immediate snapshot of its progress [10].

The typical profile of the curve follows an “S” pattern: in the early stages of the project, growth is slow, due to preparatory activities such as resource mobilization or initial planning. Then, in the middle phase, operational activities increase rapidly, leading to a steeper growth of the curve, until it reaches the inflection point, which is the time of maximum operational intensity and expenditure. Finally, as closure approaches, the pace slows again, reflecting verification, testing and final delivery activities [9].

However, each design has unique characteristics, and the shape of the S-curve can vary significantly. In the literature, it has been observed that not all S-curves are symmetrical or regular: many have irregularities and local variations due to the uneven distribution of activities [11]. For this reason, a classification of S-curves into three main categories has been proposed, based on the temporal distribution of costs:

- Front-loaded: most of the work and spending is concentrated in the first half of the project.
- Mid-loaded: activities are more symmetrically distributed, peaking in the middle phase.
- Back-loaded: most of the work is concentrated in the second half of the project duration [12].

In addition to supporting progress monitoring, the S-curve is used for cash flow forecasting, performance evaluation, and remaining life estimation, thus being an important tool for planning and monitoring during the execution phase.

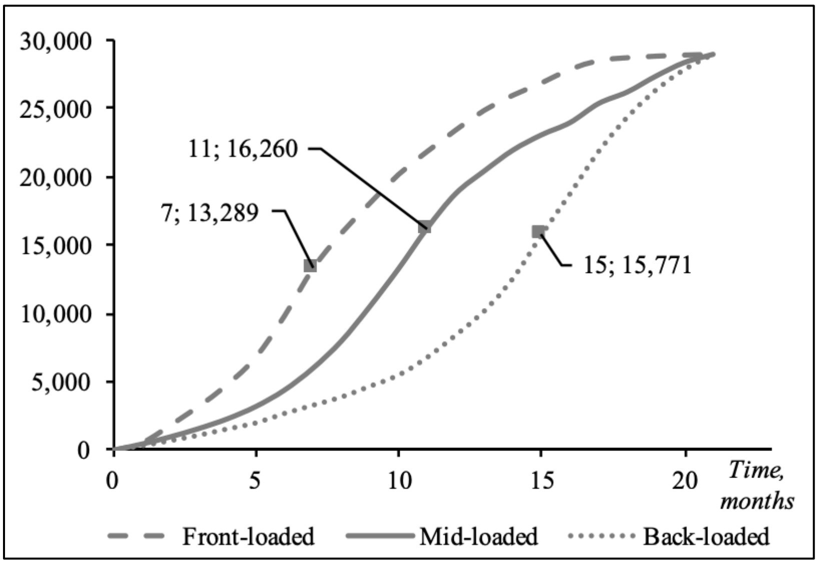

Figure 3-4, An istance of the 3 type of cumulative S-curve, taken from [10]

3.1.3. Project Life Cycle Models: Traditional and Modern Approaches

The life cycle of a project is defined as the set of phases that a project goes through from inception to closure. Each phase is a set of project activities that are logically related and culminate in the realization of one or more deliverables or outcomes. The lifecycle concept takes a central place in the project management discipline, as it provides a framework for organizing, planning, executing and monitoring the project throughout its development.

Five project life cycle models are generally distinguished in the literature, each of which has unique characteristics and lends itself to application in specific contexts and sectors. It is important to highlight that although these models constitute established theoretical references, each project has a unique life cycle, built based on its needs and complexity. The following paragraphs provide an overview of these five main project life cycle models.

Traditional life cycle (Waterfall)

The traditional lifecycle, also often referred Waterfall, is the classic project management model, strongly oriented toward advance planning and sequencing of phases. It is characterized by a predictive, linear and plan-driven approach. It is considered predictive because it assumes that all events and activities of the project can be anticipated with reasonable certainty from the beginning. It is linear because the phases follow one another in a strict and orderly sequence, with each phase starting only after the previous one is completed. Finally, it is plan-driven because the entire project is thoroughly planned upfront, and the execution phase focuses primarily on carrying out that defined plan.

This model is particularly suitable for projects where requirements are well defined and stable over time, and where the risk of change is low. However, the absence of iterations and structured feedback mechanisms makes it less suitable for dynamic and innovative contexts, where there is a frequent need to adapt to new requirements that emerge during the course of the project [Figure 3-5].

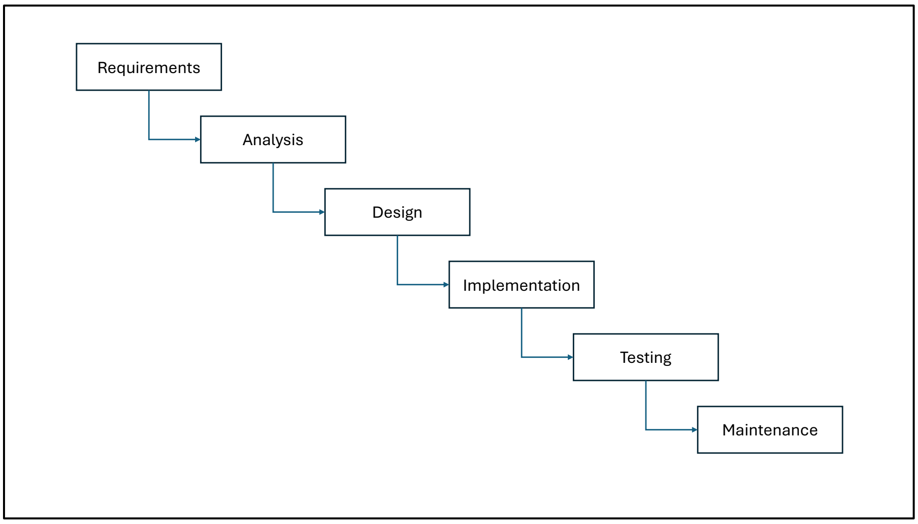

Figure 3-5, traditional life cycle diagram

Iterative life cycle

In the iterative lifecycle project phases do not follow a rigidly sequential flow, but can be retraced multiple times. This approach is typical of contexts in which product knowledge evolves progressively, and deliverables are refined through successive cycles. A prime example is research and development projects, in which it may be necessary to return to the design phase several times as a result of testing, prototyping, or feedback. The hallmark of this model is thus the presence of iterations between phases, which allows new information to be progressively incorporated, improving the product through continuous revisions.

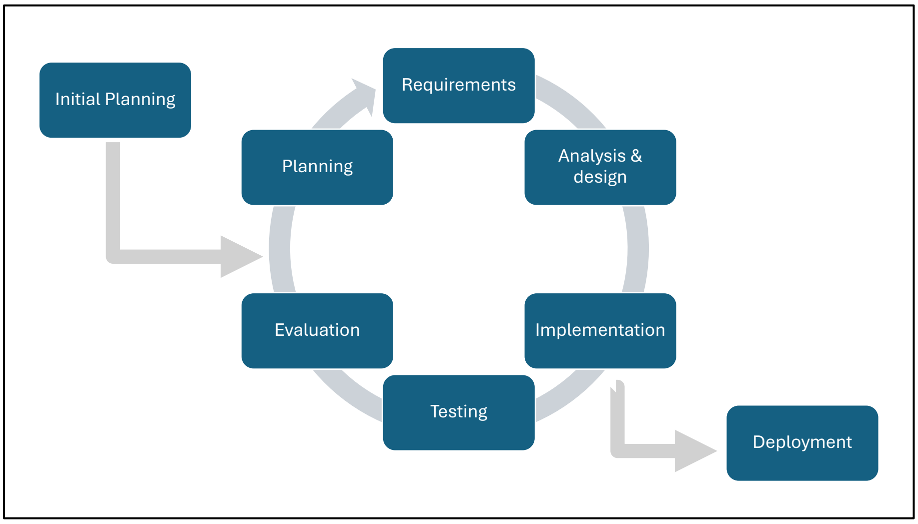

Figure 3-6, iterative life cycle

Incremental life cycle

The incremental life cycle is based on the idea of dividing the project into a series of successive releases, each of which provides an increment of value to the end customer. Each increment produces a complete, usable deliverable that partially contributes to the overall project goals.

From a financial point of view, this approach offers advantages: it allows economic returns to be anticipated and subsequent phases to be partly self-financing. Unlike the traditional model, in which the return on investment becomes evident only at the end of the project, the incremental model promotes greater economic efficiency and faster value generation.

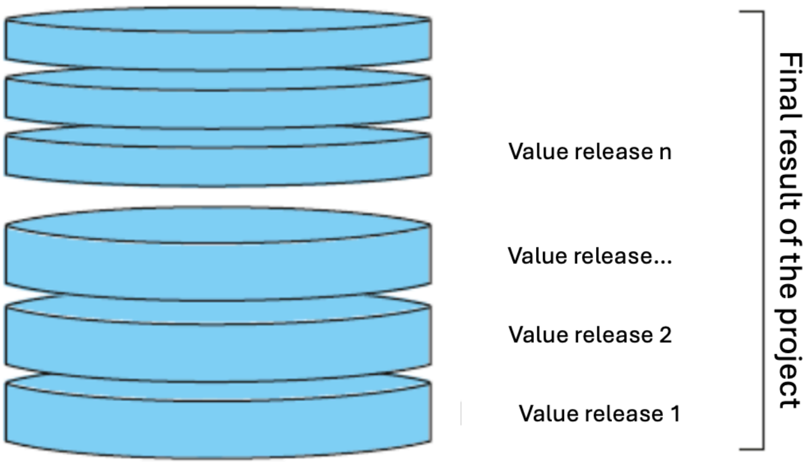

Figure 3-7, incremental life cycle, taken from [2]

Agile life cycle

The agile model represents a synthesis of the iterative and incremental cycles. It is in fact iterative, in that it involves continuous review and revision of work done, and it is incremental, in that each iteration produces a working increment of the product.

It is a model that can adapt to new needs and is flexible enough to do so. It works best for projects that are new and change a lot, where the client can change goals and priorities as the project goes on.

Time-boxed iterations (or sprints) with a set and consistent length are a unique part of this cycle. They help keep a steady and predictable pace of work. Feature-based planning is another important feature. Instead of a Work Breakdown Structure (WBS), the agile model uses a Feature Breakdown Structure (FBS). This organizes the work around the features that the final product needs instead of the tasks that need to be done.

The agile method values teams and stakeholders working together all the time, encourages the delivery of working solutions on a regular basis, and makes it easy to adapt to change..

Hybrid life cycle

The hybrid life cycle represents an intermediate solution, in which elements from different models described above are combined. Typically, a hybrid project adopts a prevalent model, supplemented by practices from other models to meet specific needs.

A practical example might be an industrial project, such as the construction of a manufacturing plant, managed according to traditional logic in terms of planning and execution of work, but with a design phase managed in an agile mode.

In this approach, it is possible to release the design documents in regular iterations (e.g., every two weeks) and after switch to a more rigid management during the construction and procurement stages.

As a strategic option for projects with diverse components, this method balances the flexibility of the agile model with the stability of the waterfall model.

3.1.4. Project Management Principles and Project Performance Domains

As highlighted in the previous section, the discipline of project management has evolved significantly over the years, driven by the increasing complexity of projects and the spread of agile and hybrid approaches. These transformations have necessitated a progressive adaptation of management methodologies to increasingly dynamic, uncertain and rapidly changing operational contexts.

In response to these needs, the Project Management Institute (PMI) has progressively updated its reference standards. Until the publication of the sixth edition of the PMBOK Guide [13], project management was structured according to a strongly process-oriented model, broken down into five process groups, Initiating, Planning, Executing, Monitoring and Controlling, and Closing, and ten knowledge areas. This approach proved particularly effective in predictive and well-structured contexts but showed application limitations in projects characterized by high levels of uncertainty, comlexity and the need for continuous adaptation.

In the seventh edition of the PMBOK, PMI introduced a change: the abandonment of the process-based model in favor of a principles-based, value-oriented approach [1]. The new standard consists of 12 project management principles, which guide the project manager's behavior and thinking, and 8 performance domains, which represent critical areas to focus on to ensure project success, regardless of the method adopted.

As pointed out by Figueiredo [14], this change reflects an attempt to adapt the PMBOK to the contemporary reality of projects. However, the new approach has also been criticized for its abstractness and lack of operationality, as it does not provide concrete practical tools for daily project implementation. In many cases, traditional processes and phases remain a useful reference, especially when it comes to monitoring and control, as in this study, an area where performance forecasting is still based on classical methodologies.

This chapter therefore offers an overview of traditional process groups, followed by an analysis of the new conceptual approach of PMBOK 7^{th}.

As stated in the PMBOK 6^{th} [13], Project management processes are grouped in five Project Management Process Groups:

- Initiating process group: it consists of those processes performed to define a new project or a new phase of an existing project by obtaining authorization to start the project or phase. "The purpose of the Initiating Process Group is to align the stakeholders' expectations and the project purpose, inform stakeholders of the scope and objectives, and discuss how their participation in the project and its associated phases can help to ensure their expectations are met. Within the Initiating processes, the initial scope is defined, and initial financial resources are committed. Stakeholders who will interact and influence the overall outcome of the project are identified. If not already assigned, the project manager is appointed. This information is captured in the project charter and stakeholder register. When the project charter is approved, the project is officially authorized, and the project manager is authorized to apply organizational resources to project activities."
- Planning process group: it consists of those processes that establish the total scope of the effort, define and refine the objectives, and develop the course of action required to attain those objectives. "The processes in the Planning Process Group develop the components of the project management plan and the project documents used to carry out the project. The nature of a project may require the use of repeated feedback loops for additional analysis. As more project information or characteristics are gathered and understood, additional planning will likely be required. Significant changes that occur throughout the project life cycle may initiate a need to revisit one or more of the planning processes and,

possibly, one or both of the Initiating processes. This ongoing refinement of the project management plan is called progressive elaboration, indicating that planning and documentation are iterative or ongoing activities. The key benefit of this Process Group is to define the course of action to successfully complete the project or phase."

- Executing process group: it consists of those processes performed to complete the work defined in the project management plan to satisfy the project requirements. "This Process Group involves coordinating resources, managing stakeholder engagement, and integrating and performing the activities of the project in accordance with the project management plan. The key benefit of this Process Group is that the work needed to meet the project requirements and objectives is performed according to plan. A large portion of the project budget, resources, and time is expended in performing the Executing Process Group processes. The processes in the Executing Process Group may generate change requests. If approved, the change requests may trigger one or more planning processes that result in a modified management plan, project documents, and possibly new baselines."
- Monitoring and controlling process group: it consists of those processes required to track, review, and regulate the progress and performance of the project; identify any areas in which changes to the plan are required; and initiate the corresponding changes. "Monitoring is collecting project performance data, producing performance measures, and reporting and disseminating performance information. Controlling is comparing actual performance with planned performance, analyzing variances, assessing trends to effect process improvements, evaluating possible alternatives, and recommending appropriate corrective action as needed. The key benefit of this Process Group is that project performance is measured and analyzed at regular intervals, appropriate events, or when exception conditions occur in order to identify and correct variances from the project management plan."
- Closing process group: it consists of the processes performed to formally complete or close a project, phase, or contract. "This Process Group verifies that the defined processes are completed within all of the Process Groups to close the project or phase, as appropriate, and formally establishes that the project or project phase is complete. The key benefit of this Process Group is that phases, projects, and contracts are closed out appropriately. While there is only one process in this Process Group, organizations may have their own processes associated with project, phase, or contract closure. Therefore, the term Process Group is maintained."

The principles of a profession serve as fundamental guidelines for strategy, decision-making, and problem-solving. In project management, these principles offer guidance on how people should act within projects, influencing and shaping the performance domains to achieve the desired outcomes. They are broadly defined, allowing for various ways in which individuals and organizations can align with them. While there is conceptual overlap between principles and performance domains, principles primarily direct behavior, whereas performance domains define key areas of focus where this behavior is demonstrated.

As stated in the PMBOK 7th [1] the PM principles are:

- <u>Be a diligent, respectful, and caring steward</u>: stewards must manage resources responsibly, respecting the people involved and taking care of the success of the project while maintaining compliance with internal and external guidelines.
- <u>Create a collaborative project team environment</u>: project teams are made of people who possess different abilities, expertise, and backgrounds. Work environment based on trust and open communication fosters cooperation and the achievement of common goals.
- <u>Effectively engage with stakeholders</u>: proactively involving stakeholders, understanding their expectations and needs, is essential to ensure the success of the project.
- <u>Focus on value</u>: every decision and activity must be driven by the value the project brings to stakeholders and the organization.
- <u>Recognize, evaluate, and respond to system interactions</u>: projects do not exist in isolation, so it is essential to consider interactions with other systems to prevent unintended effects.
- <u>Demonstrate leadership behaviors</u>: a project manager must demonstrate and adapt leadership behaviors to inspire, motivate and lead the team with integrity, empathy and determination.
- <u>Tailor based on context</u>: projects are unique, so it is crucial design the development approach based on the context of the project, its objectives, stakeholders, governance, and the environment to achieve the outcome while maximizing value, managing cost, and enhancing speed.
- <u>Build quality into processes and deliverables</u>: maintain focus on quality, ensuring that deliverables meet project objectives and meet the needs, uses and acceptance requirements established by relevant stakeholders.

- Navigate complexity: managing project complexity requires careful analysis, the ability to simplify complex problems, and a structured approach to enable the project team to successfully manage the project lifecycle.
- Optimize risk responses: risks and threats must be identified, assessed and addressed proactively to maximize positive impacts and minimize negative impacts to the project and its outcomes.
- Embrace adaptability and resiliency: a flexible and resilient environment help the project accommodate change, recover from setbacks, and advance the work of the project.
- Enable change to achieve the envisioned future state: the shift towards the desired future state involves preparing the team for new behaviors and processes essential to the project's success.

Project performance domains, which are correlated with the project management principles [Figure 3-8], are group of related activities that are crucial for effectively delivering project outcomes.

These areas of focus function as interconnected components that operate concurrently throughout the entire project life cycle. Regardless of when value is delivered all domains are engaged from the project's inception through to its completion. Those who manage the project deal with multiple areas, such as stakeholder engagement, team management, and organization of activities. These aspects overlap and intertwine, and their combination varies from project to project.

The breakdown of activities by domain is determined by the context of the organization, the project, deliverables, the project team, stakeholders, and other factors.

As stated in the PMBOK 7th [1] the Project Performance Domains are eight and they are presented below without specific order:

- Stakeholder performance domain: it addresses activities and functions associated with stakeholders, an effective execution of it results in a productive working relationship, stakeholder agreement with project goals, and a positive impact from stakeholders who are satisfied.
- Team performance domain: it focuses on the activities and functions related to the individuals responsible for delivering project outputs that drive business outcomes, it leads to a shared ownership, and a high-performing team guided by the leadership.

- Development approach and life cycle performance domain: it encompasses the activities and functions associated with defining and managing the project's development methodology, rhythm, and life cycle stages. His results are development approaches that are consistent with project deliverables, a project life cycle consisting of phases that connect the delivery of business and stakeholder value from the beginning to the end of the project, and a project life cycle consisting of phases that facilitate the delivery cadence and development approach required to produce the project deliverables.
- Planning performance domain: it addresses activities and functions associated with the initial, ongoing, and evolving organization and coordination necessary for delivering project deliverables and outcomes, an effective execution of it results in organized progresses, holistic approach to delivering the project outcomes, efficient planning information, and in a process for the adaptation of plans throughout the project based on emerging and changing needs or conditions.
- Project work performance domain: it focuses on the activities and functions involved in defining project processes, managing resources, and foster a culture of continuous learning. When effectively executed, this domain leads to high levels of project performance in terms of efficiency, tailored project processes that align with the specific context, effective use and management of resources, and enhanced team capabilities.
- Delivery Performance Domain: it is concerned with the activities and functions required to ensure that the project delivers the intended scope and quality outcomes. Effective execution of this domain leads to projects align with business goals and contribute to strategic progress, outcomes launched and successfully achieved, planned benefits delivered within the expected timeframes, a well-defined understanding of the project's requirements by the project team, and satisfied stakeholder.
- Measurement performance domain: it involves assessing project performance and implementing corrective actions as needed to ensure performance remains within acceptable limits. When applied, this domain supports a clear understanding of the project's current status, informed and timely decision-making, prompt and suitable interventions to keep project performance aligned with expectations, achievement of project goals and delivery of business value through well-informed decisions based on accurate forecasts and evaluations.
- Uncertainty performance domain: it focuses on activities and functions aimed at identifying and managing risks and uncertainties throughout the project lifecycle.

Effective execution of this domain enables solid understanding of the project environment, including technical, social, political, market, and economic factors, proactive identification, exploration, and response to uncertainty, awareness of how multiple variables within the project are interrelated, the ability to anticipate risks and opportunities and evaluate their potential impact, and efficient use of cost and schedule buffers to remain aligned with project goals.

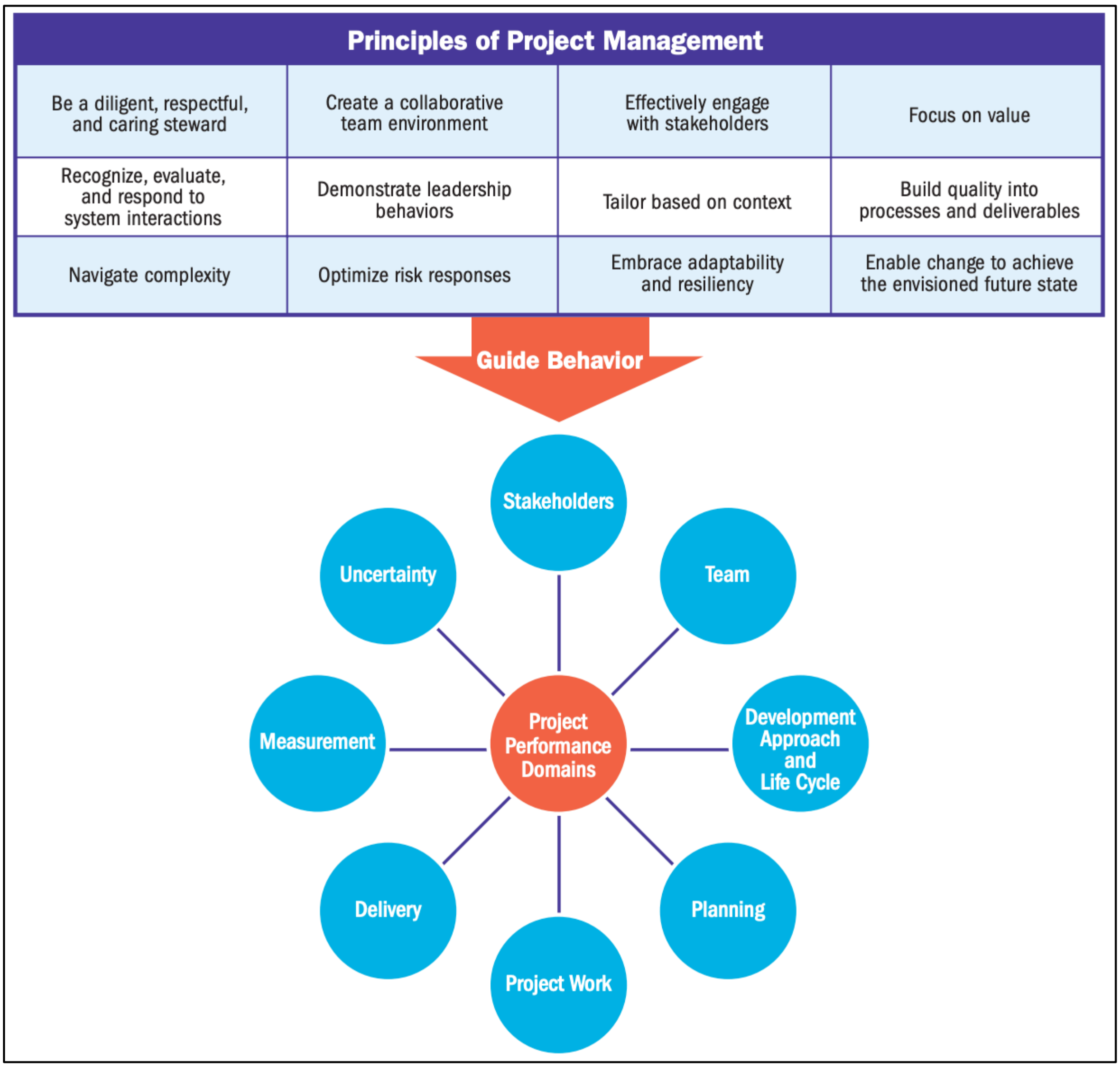

Figure 3-8, relationship between Project Management Principles and Project Performance Domains, taken from [1].

3.2. Project Monitoring and Control

The objective of the study is to optimize the performance measurement of estimates at completion of the project, with the intent of improving the effectiveness of Project Monitoring and Control activities (often shortly called Project Control). The latter can be described as a feedback system with two components: monitoring, aimed at identifying any current deviations from planned objectives, and control, aimed at correcting these deviations through effective corrective actions [15].

Monitoring can be defined as the set of management procedures and practices aimed at systematically gathering information on actual or expected project performance, making use of specific performance metrics. Monitoring, on the other hand, represents the decision-making process aimed at adjusting the project in order to ensure that the initial objectives are achieved. This is done through the analysis of the causes of performance deviations, the design of corrective actions deemed necessary and the subsequent implementation of those actions.

As described in the previous section, in the sixth edition of the PMBOK, the Monitoring and Controlling Process Group constituted one of the five core process groups.

While in the seventh edition of the PMBOK, the Project Management Institute took a novel approach, shifting the focus from processes to a framework based on principles and performance domains. In this new perspective, monitoring and control activities are no longer conceived as a separate phase, but as a cross-cutting and continuous dimension of project management, traceable in particular to the Measurement Performance Domain.

Despite this structural change, the use of established techniques for measuring progress remains valid. Above all in contexts where such methodologies are already integrated or in projects where accurate forecasting and quantitative control play a crucial role, the use of classical performance measurement tools still remains a reliable practice.

Analyzing the content of the PMBOK 7^{th}, it becomes apparent that the Performance Domain includes specific guidance on methods for establishing effective performance measures [1]. According to PMI, establishing effective measures helps ensure that truly relevant information is measured and communicated to stakeholders. These steps make it possible to monitor and report data that shows the project's current status. To put it another way, a well-designed measurement gives the project team the data they need to decide quickly and implement corrective measures.

In this context, PMBOK 7 distinguishes between two main types of Key Performance Indicators (KPIs): leading indicators and lagging indicators.

Leading indicators "predict changes or trends in the project. If the change or trend is unfavorable, the project team evaluates the root cause of the leading indicator measurement and takes actions to reverse the trend. Used in this way, leading indicators can reduce performance risk on a project by identifying potential performance variances before they cross the tolerance threshold."

Lagging indicators "measure project deliverables or events. They provide information after the fact. Lagging indicators reflect past performance or conditions. Lagging indicators are easier to measure than leading indicators. Examples include the number of deliverables completed, the schedule or cost variance, and the amount of resources consumed."

Within the Measurement Performance Domain, PMBOK 7^{th} also provides clear criteria for assessing their quality, proposing the use of the SMART criteria as a guide for defining metrics. According to this approach, a metric is considered effective if it meets the following characteristics:

- Specific: Measurements are specific as to what to measure.
- Meaningful: Measures should be tied to the business case, baselines, or requirements.
- Achievable: The target is achievable given the people, technology, and environment.
- Relevant: Measures should be relevant.
- Timely: Useful measurements are timely. Information that is old is not as useful as fresh information.

A further outlined in this section of the PMBOK 7^{th} is the guidance on what should be measured in a project. This choice depends on the specific objectives of the project, its expected outcomes, and the operational environment in which it takes place. Specifically, the PMI identifies some common categories of metrics, including:

- Deliverable metrics: measures related to the quality and completeness of deliverables.

- Delivery: timing and mode of delivery.
- Baseline performance: comparison of planned and actual values.
- Resources: resource utilization and workloads.
- Business value: value generated in terms of benefits to the organization.
- Stakeholders: stakeholder satisfaction and involvement.
- Forecasts: forecasts of future project performance.

To measure these categories, the adoption of metrics that can also be traced back to more traditional methodologies, among which Earned Value Analysis (EVA), are proposed. The latter, in particular, is a established technique for measuring integrated performance in terms of time, cost, and progress, and will be the subject of a specific discussion in the next chapter.

3.3. Performance Measurement: Earned Value Analysis

Earned Value Analysis (EVA) integrates cost, schedule, and work performed by ascribing monetary values to each. EVA methodologies, which include Earned Value Management (EVM) and Earned Schedule Management (ESM), rely on three metrics: the budgeted cost of work performed (i.e., Earned Value), the actual cost of work performed (i.e., Actual Cost) and the budgeted cost of work scheduled (i.e., Planned Value).

- Planned Value (PV – BCWS)

  PV represents the value of work that, according to the project schedule, should be completed to a given point in time. This metric is calculated based on the total budget allocated for the project and the time distribution of planned activities.
- Earned Value (EV - BCWP)

  EV is a metric that indicates the value of the work actually completed up to a certain point in the project, expressed in terms of the planned budget. It measures the actual progress of the project, allowing you to assess whether the work has been performed in line with cost and schedule forecasts.
- Actual Cost (AC - ACWP)

  AC represents the total cost actually incurred for work completed up to a given point in time. It reflects the actual expenses incurred by the project.

Planned Value, Actual Cost and Earned Value S-curves can have six possible arrangements, as in the chart presented [Figure 3-9].

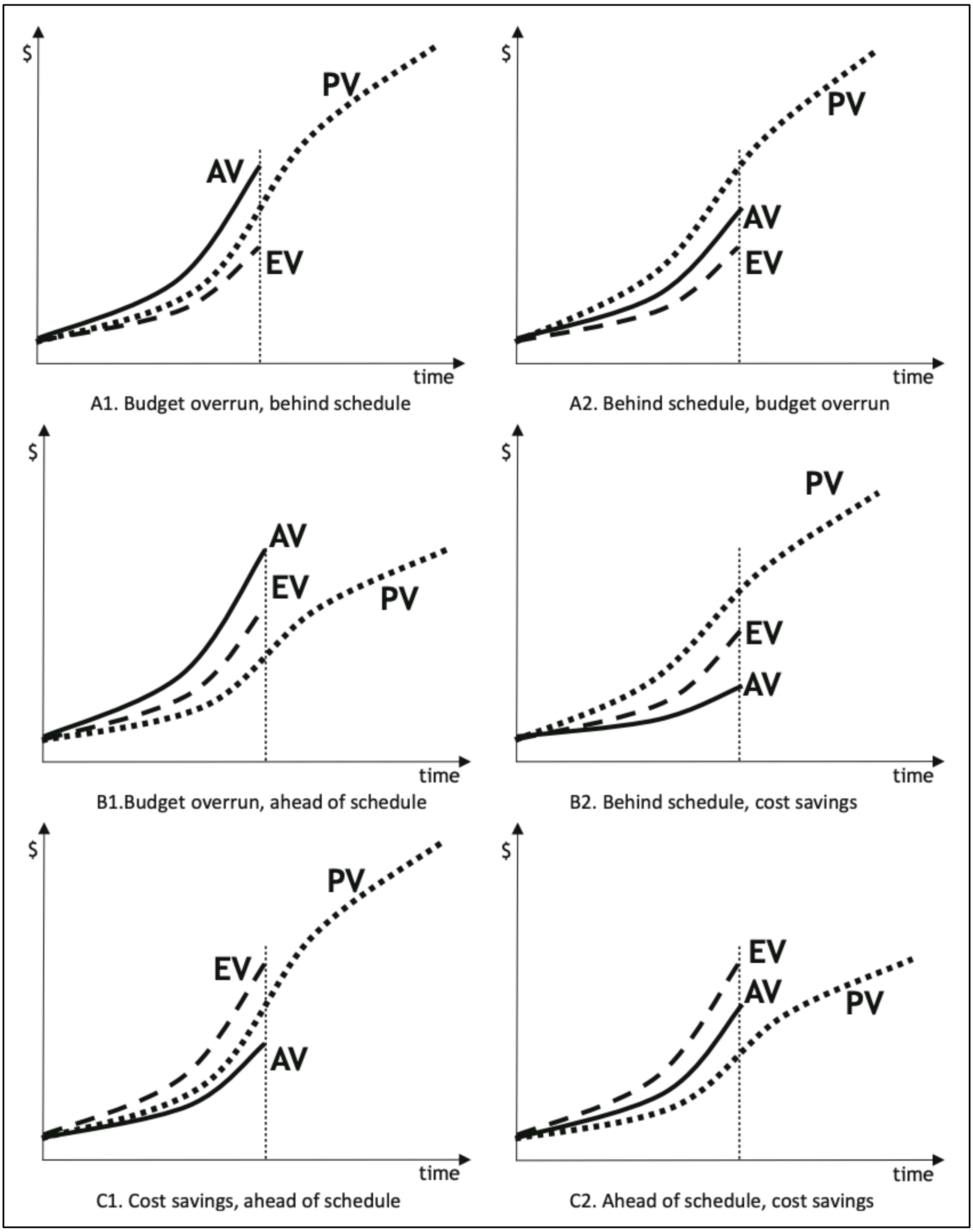

Figure 3-9, possible arrangements of S-curves indicating PV, AC, and EV, taken from [15]

3.3.1. Earned Value Management and Earned Schedule management

Earned Value Management (EVM) was first introduced by the e U.S. Department of Defense (DoD) in 1967, indeed, the need for a system such as EVM arose from the increasing complexity of projects, such as missile development programs, which often experienced significant cost overruns and delays. EVM established a structured framework that integrates essential elements of project scope, schedule, and cost performance, improving oversight and accountability in complex projects.

The Department of Defense sought to standardize project evaluation methods and improve transparency for stakeholders through consistent reporting practices by setting at $20 million the budget above which projects would have to be monitored through the use of EVM [5].

A key moment in the recognition of Earned Value Management as an international standard in project management was its official inclusion within the Project Management Institute standards. As early as 1996, with the publication of the first edition of the PMBOK, references to cost control techniques began to appear, although without a direct focus on EVM [16]. It is with the second edition in 2000, however, that we see a real turning point: EVM is formally recognized as a fundamental technique for monitoring and controlling project costs [17]. The final consolidation comes in 2005, with the publication of the Standard for Earned Value Management by PMI, which enshrines EVM as the benchmark methodology for objective measurement of project performance [18].

As a result of this development, EVM has become an important tool for project management, and communication with stakeholders has benefited from this standardization, but because it is based on predetermined methodologies it has encountered some difficulties in responding to changing project contexts.

Earned Schedule Management (ES) represents an enhancement to traditional Earned Value Management, and it was introduced by Lipke in 2003 [19]. It overcomes some limitations in forecasting project timelines and enables a clearer understanding of schedule performance by measuring the schedule progress in time units instead of dollar amounts.

ESM's focus on temporal variances not only simplifies interpretation but also equips project managers with practical tools to realign schedules proactively. This temporal perspective assesses the ambiguity present in traditional EVM metrics, particularly in projects with non-linear dynamics, providing a more reliable framework for evaluating and addressing delays [20].

3.3.2. Performance metrics

Earned Value Analysis uses a series of derived metrics to gain a better sense of project performance. These performance metrics are useful tools for the purpose of project monitoring and control, as they enable the evaluation of a project's performance against its objectives. Through the analysis of these metrics, it is possible to monitor resource utilization, adherence to budget and planned time-in fact, they aim to identify any deviations from initial planning, guiding any corrective measures. In this section, we will highlight a set of key metrics related to Earned Value Analysis, which are used to assess project performance in terms of cost and schedule efficiency.

Budget at Completion (BAC)

BAC indicates the overall amount of funds assigned to the project, reflecting the total of all planned expenses. It plays a key role in project management, as it outlines the full financial scope and acts as a benchmark the project manager strives to meet.

Cost Variance (CV)

The CV metric is an economic indicator of cost performance. It represents the difference between the Budgeted Cost of Work Performed and the Actual Cost of Work Performed[Figure 3-10]. The formula is:

$$CV = BCWP - ACWP = EV - AC$$

Where:

- $BCWP$ is Budgeted Cost of Work Performed or Earned Value
- $ACWP$ represents Actual Cost of Work Performed or Actual cost

Cost Performance Index (CPI)

The CPI is the corresponding relative measure of Cost Variance, representing the ratio between the earned value and the actual cost incurred. The formula is:

$$CPI = \frac{\mathrm{BCWP}}{\mathrm{ACWP}} = \frac{EV}{AC}$$

Where:

- $BCWP$ is Budgeted Cost of Work Performed or Earned Value
- $ACWP$ represents Actual Cost of Work Performed or Actual cost

Schedule Variance (SV)

The SV metric is a timeline indicator of time performance. It represents the difference between the Budgeted Cost of Work Performed and the Budgeted Cost of Work Scheduled [Figure 3-10]. The formula is:

$$SV^{EVM} = BCWP - BCWS = EV - PV$$

Where:

- $BCWP$ is Budgeted Cost of Work Performed or Earned Value
- $BCWS$ represents Budgeted Cost of Work Performed or Panned Value

For this metric, the Earned Schedule theory provides an extension to the traditional Earned Value Management approach. The formula is:

$$SV^{ES} = ES - AT$$

Where:

- ES is the point in time, according to the project baseline, when the current Earned Value (EV) should have been achieved
- AT represents the Actual Time, amount of time that has elapsed since the start of the project

Schedule Performance index (SPI)

The SPI is the corresponding relative measure of Schedule Variance, representing the ratio between the earned value and the planned value. The formula is:

$$SPI^{EVM} = \frac{\mathrm{BCWP}}{\mathrm{BCWS}} = \frac{EV}{PV}$$

Where:

- $BCWP$ is Budgeted Cost of Work Performed or Earned Value
- $BCWS$ represents Budgeted Cost of Work Performed or Panned Value

As for the SV also the SPI has his approach provided by Earnes Schedule theory

$$SPI^{ES} = \frac{\text{ES}}{\text{AT}}$$

Where:

- ES is the point in time, according to the project baseline, when the current Earned Value (EV) should have been achieved
- AT represents the Actual Time, amount of time that has elapsed since the start of the project

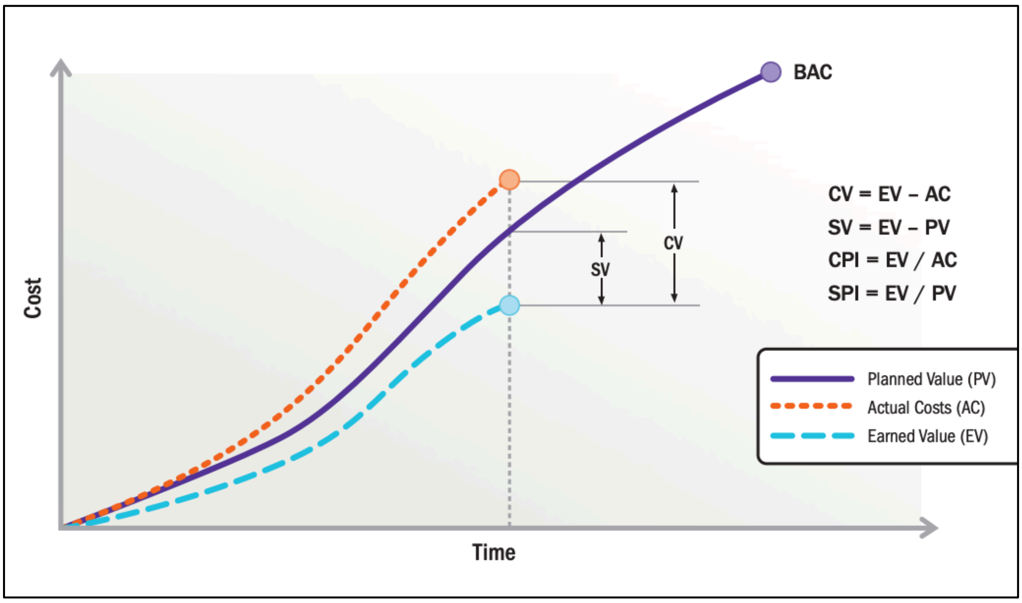

Figure 3-10, Earned Value Analysis Showing Schedule and Cost Variance, taken from [1]

Estimation At Compilation (EAC)

EAC is a critical metric that estimates the total cost or duration required to complete a project, based on current performance. This metric supports project managers in determining whether the project will meet the initial budget and schedule or whether changes will be required to align with objectives.

Cost Estimation At Compilation (CEAC)

CEAC represents a forecast of the total cost of the project, based on costs incurred to date (AC) plus an estimate of remaining costs (ETC) [Figure 3-11]. Although several CEAC formulas are available through the EVM, a common approach is a manual "bottom-up" estimation by the project manager, which combines actual costs with a new assessment of the costs required to complete the project:

$$CEAC = AC + ETC$$

Where:

- AC is Actual Cost
- ETC represents Estimate To Complete

In addition to manual estimation, there are three calculated methods for estimating EAC:

1. CEAC forecast for ETC work performed at the budgeted rate

This Method accepts the actual project performance to date (whether favorable or unfavorable) as represented by the actual costs and predicts that all future ETC work will be accomplished at the budgeted rate. The formula is:

$$CEAC = AC + (BAC - EV)$$

Where:

- AC is Actual Cost
- BAC represents Budget At Completion
- EV is Earned Value

2. CEAC forecast for ETC work performed at the present CPI

This method assumes that what the project has experienced to date can be expected to continue in the future. The formula is:

$$CEAC = \frac{BAC}{CPI}$$

Where:

- BAC represents Budget At Completion
- CPI is Cost Performance Index

3. CEAC forecast for ETC work considering both SPI and CPI factors

In this forecast, the ETC work will be performed at an efficiency rate that considers both the cost and schedule performance indices. This method is most useful when the project schedule is a factor impacting the ETC effort. The formula is:

$$CEAC = AC + \frac{BAC - EV}{CPI * SPI}$$

Where:

- AC is Actual Cost
- BAC represents Budget At Completion
- CPI is Cost Performance Index
- SPI is Schedule Performance Index

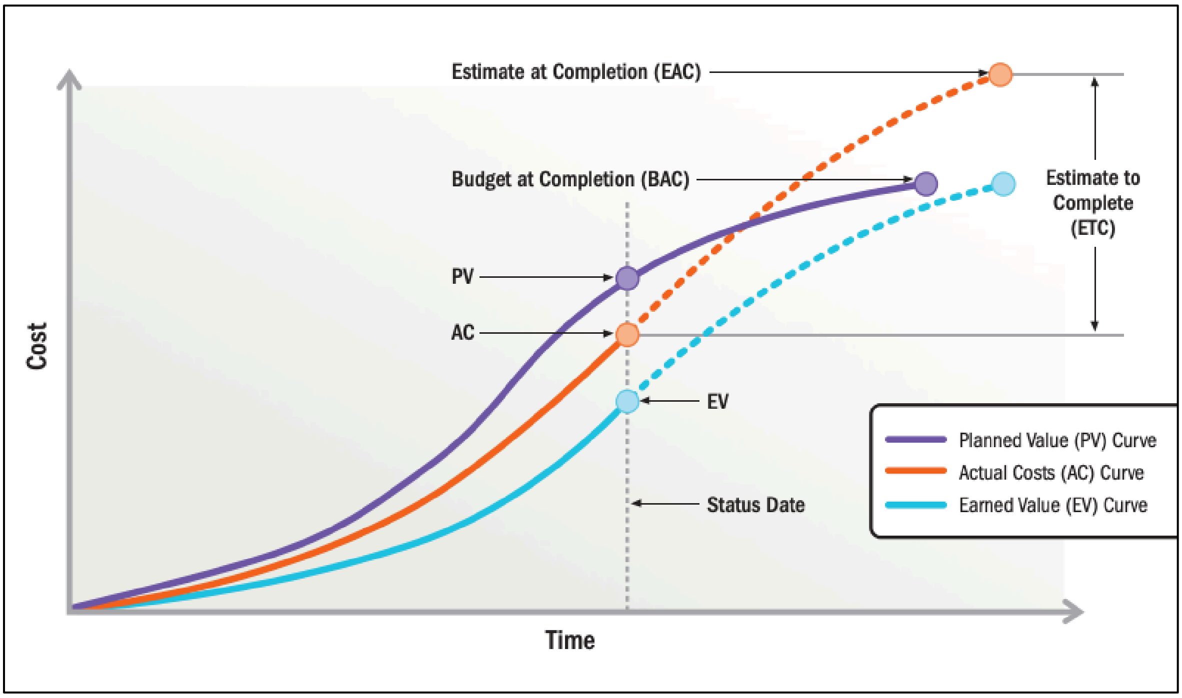

Figure 3-11, Forecast of CEAC and ETC, taken from [1]

Time Estimation At Compilation (TEAC)

TEAC represents a forecast of the total duration of the project to completion, based on the analysis of current time progress. This indicator is used to estimate how long it will take to complete the entire project, assuming that the scheduling performance observed to date will continue in the same manner. The formula is:

$$TEAC = \frac{Planned\ Duration}{SPI}$$

Where:

- PD is the Planned Duration
- SPI is the Schedule Performance Index

This formula is based on the assumption that the time performance observed up to the time of calculation (i.e., the SPI) will continue through the rest of the project. An SPI value < 1 indicates delays relative to the planned schedule, resulting in a TEAC greater than the planned duration; conversely, an SPI > 1 indicates faster-than-expected progress, resulting in a reduction in the estimated duration to completion.

3.4. Dynamic parameters

In the literature there are also other dynamic measures that assess schedule performance from different perspectives, considering various angles and datasets. When used individually, these metrics may lead to misleading conclusions or introduce bias into the model, to mitigate this risk, it is crucial to use a complementary set of metrics. For this reason, in this study, these additional measures have been combined with the more standard EVA metrics to ensure a more balanced and reliable analysis. A detailed explanation of these dynamic measures is provided below.

Total Float Consumption Index (TFCI)

The TFCI is a duration-based metric that measures the rate of total float consumption (total available time margin) relative to project progress. Unlike other metrics that capture only the current state of the project at a given point in time, the TFCI considers the average float consumption rate to estimate an expected completion date for the entire project [21]. The formula is the following:

$$TFCI = \frac{AD - CPTF\ (or\ BFV)}{AD}$$

Where:

- AD is the actual duration
- $CPTF$ represents the Critical Path Total Float
- BFV represents the Baseline Finish Variance which can be used in place of CPTF

A TFCI of less than 1.00 indicates that a project may not complete on-time and applying that indicator to the total program duration predicts where a project would complete if trends persisted, or corrective action is not taken.

Schedule Performance Index - Time vs. Earned Schedule (SPI vs. TSPI)

The To Complete Schedule Performance Index (TSPI) is a measure of the efficiency of the planned schedule for project completion. This index provides an estimate of the average rate of progress that must be maintained to complete the project on schedule. The TSPI can be compared with the SPI (Schedule Performance Index) to see if future schedule efficiency will be consistent with the efficiency demonstrated to date [21]. The formula of TSPI is defined as follows:

$$TSPI = \frac{PDWR}{RD}$$

Where:

- $PDWR$ is the time planned to go from current BCWP to BAC
- $CPTF$ represents the actual planned remaining duration based on current progress

The comparison of SPI and TSPI helps to asses whether the future performance of the project will be consistent with that observed so far. If the difference between the two indices falls within the range of -0.10 to 0.10, it means that the efficiency demonstrated so far is consistent with the forecast. Instead, when the difference exceeds 0.10 it could indicate an overly pessimistic forecast, suggesting a possible slowdown from the current trend and if the value falls below -0.10, the forecast is optimistic, implying an improvement in performance that may not be realistic.

Baseline Execution Index (BEI) and Baseline Execution Index at Start (BEI^{start})

Baseline Execution Index measures the number of tasks completed as a ratio to those tasks that should have completed to date according to the original (baseline) plan. This index provides a clear indication of the pace of project execution and helps to identify any risk of delayed completion in advance. Alongside the BEI, the Baseline Execution Index at Start is used, a variant that focuses on activities that should have been started versus those that were actually started, it useful for identifying problems with the activation of activities, highlighting whether the project is meeting the timeline of the initial phases of work [21]. BEI and BEI^{start} formulas are the following:

$$BEI = \frac{Task\ actually\ completed}{Task\ planned\ to\ be\ completed}$$

$$BEI^{start} = \frac{Task\ actually\ started}{Task\ planned\ to\ be\ started}$$

The ideal value is 1.00, which indicates that the project is proceeding in line with the schedule. If the BEI or BEI^{start} are greater than 1.00, it means that the number of completed or started activities exceeds expectations, while a value less than 1.00 signals that the project is lagging behind initial plans.

Earned Duration (ED(t))

Earned duration is a metrics introduced by Jacob and Kane is an extension of the EVM designed to improve the accuracy in estimating the project completion date [22]. Ed formula is the following:

$$ED(t) = t * SPI^{EVM}(t)$$

Where:

- t is the actual time of the project
- $SPI^{EVM}(t)$ represents Schedule Performance Index in EVM

Jacob and Kane's test results show that the ED provides reliable estimates in the early phase of the Project while toward the end it tends to be less accurate.

Time to Schedule Ratio (T/S) and Time to Cost Ratio (T/R)

The time-to-schedule ratio and time-to-cost ratio metrics proposed by Przywara and Rak are also an extension of the EVM introduced with the aim of improving the analysis of project financial and time deviations [23]. The formulas are as follows:

$$T/S = \frac{BCWS - ACWP}{BCWP - BCWS}$$

$$T/C = \frac{BCWS - ACWP}{BCWP - ACWP}$$

Where:

- BCWS is the Budgeted Cost of Work Scheduled (PV)
- ACWP is the Actual Cost of Work Performed (AV)
- BCWP is the Budgeted Cost of Work Performed (EV)

If T/S is positive, it means that the project is experiencing delays affecting liquidity and a negative T/C value indicates that the actual costs are exceeding the projected costs, negatively impacting liquidity.

Cost Schedule Index (CSI)

The Cost-Schedule Index (CSI) is a metric designed to provide a comprehensive measure of a project's overall efficiency in terms of both cost and schedule performance. By integrating EVA indicators, it offers a consolidated assessment of project progress.

$$CSI = CPI * SPI^{EVM}$$

Where:

- CPI is the Cost Performance Index
- SPI^{EVM} represents Schedule Performance Index in EVM

The value of CSI provides a direct indication of the overall condition of the project. When CSI is greater than 1, the project is in good condition because the cost and scheduling performance are both efficient. When CSI is equal to 1, the project is in line with the plan, with no significant deviations. Finally, when CSI is less than 1, the project is experiencing difficulties, suggesting delays and/or budget overruns.

3.5. Network Topology

Network topology indicators are tools that describe the structure of a project from the perspective of its network of activities and the dependencies between them. These indicators influence the ability to forecast during project control, especially when applying techniques such as Earned Value Management (EVM).

Project Seriality (SP)

Project Seriality describes how close the network structure of a project is to a serial or parallel network and can be represented by the serial/parallel project introduced in 1999 [24]. An SP value of 0 represents a fully parallel project, while an SP value of 1 represents a completely serial project. The SP formula is as follows:

$$SP = \frac{n_s - 1}{n_t - 1}$$

Where:

- n_s is the maximum number of subsequent activities in the network
- n_t is the total number of activities

For an evaluation of predictive accuracy in project control here there is a proposal classification into three categories [25]:

- 0% ≤ SP < 40% → parallel projects
- 40% ≤ SP ≤ 60% → serial-parallel projects
- 60% < SP ≤ 100% → serial projects

Regularity index indicator (RI)

Project regularity is an innovative concept introduced to measure how regular the distribution of Planned Value (PV) costs is over the entire time horizon of a project. Unlike other topological indicators (such as SP), which describe the logical structure of the project, RI describes the temporal shape of the PV curve and thus the trend of planned expenditure over time [26]. If the PV curve is perfectly linear, all time estimation formulas (such as Earned Duration Method, earned Schedule Method and Plannned Value Method) provide exact predictions. However, as soon as the PV curve deviates from linearity the risk of forecast errors increases. This motivated the introduction of the Regularity Index (RI) as a quantitative measure of this temporal regularity. The formula is the following:

$$RI = \frac{\sum_{i=1}^{r} m_i - \sum_{i=1}^{r} a_i}{\sum_{i=1}^{r} m_i}$$

Where:

- m_i is the maximal possible deviation of the project's PV-curve from a perfectly linear curve at time instance $i = 1, \dots, r$
- a_i is the actual deviation of the project's PV-curve from a perfectly linear curve at time instance $i = 1, \dots, r$
- r represents number of equidistant evaluation points

The RI works in a way that mirrors the logic behind the Serial/Parallel indicator; when a project's Planned Value follows a perfectly linear growth, its RI reaches the maximum value of 1, just like an SP of 1 represents a fully serial network. On the opposite end, a project where the Earned Value remains at zero for most of the timeline and then rises to match the Budget at Completion (BAC) near the conclusion would score an RI of 0. Most projects, however, exhibit regularity levels that lie between these two extremes.

Length of arcs indicator (LA)

The LA indicator describes the topological distance between two activities connected by an arc of precedence in the project network. It measures how "far" in time one activity is located relative to another to which it is connected [27]. It is defined as:

$$LA = PL_j - PL_i$$

Where:

- PL_j the progressive level of the end node j defined by [28]
- PL_i the progressive level of the end node i defined by [28]

The length of an arc thus represents the number of levels between two related activities, and reflects whether the dependence is immediate (short arc, LA=1) or more distant in time (long arc, LA>1).

Topological float indicator (TF)

The Topological Float indicator measures the structural flexibility of each activity within the network, that is, how many levels it can "move" in the design graph without violating logical dependencies [27]. It is given by:

$$TF = RL_i - PL_i$$

Where:

- RL_i the regressive level of the end node i defined by [28]
- PL_i the progressive level of the end node i defined by [28]

The aggregate Topological Float for a project is obtained as the average or sum of the floats of all activities. If F = 0 means activities on the critical path or strongly constrained (no freedom to move), if TF > 0 means flexible, non-critical activities with room to maneuver, at the and if TF is high means greater possibility to reoptimize the project in case of unforeseen events.

3.6. Machine Learning for Project Management Control

The growing interest in artificial intelligence has also brought a number of innovative tools to project management that can overcome the limitations of traditional techniques such as Earned Value Analysis. Among these, Machine Learning (ML) has proven particularly suited to support control activities due to its ability to learn from data and adapt to the complexity of real projects.

This chapter introduces the topic of ML applied to project management. After a brief overview of the fundamental concepts and differences from artificial intelligence and statistics, it goes on to describe the main existing models and the logic behind them. Finally, a review of the most recent literature is offered, with a focus on studies that apply ML to project cost and schedule forecasting, highlighting approaches, data used, and results obtained.

3.6.1. Fundamentals of Machine Learning

The innovation of AI originated with Alan Turing's publication titled "Computing Machinery and Intelligence", which proposed the Turing Test in 1950. For centuries philosophers and engineers have speculated over the amalgamation of mechanics and intelligence, and while we are still far from machines emulating humans, we have come a long way in terms of AI infrastructure. Today, diagnostic medicine, logistics, data mining, and many more fields are under the influence of AI.

Machine learning (ML) is one of the key tools on which modern AI is based, algorithms placed in ML empowers machines to perform specific tasks unaided. These algorithms are capable of interpreting data through classification, predictions, and recognizing patterns.

It is crucial not to mix up statistics, artificial intelligence, and machine learning, since they are all rooted in concepts and principles intertwine, albeit tangentially, yet they have distinct separate goals and methodologies. Artificial intelligence constitutes a broader framework which encompasses the attempt to reproduce certain characteristic features of human intelligence, including reasoning, problem solving, perception, and ultimately, self-awareness. Machine learning, which is a subfield of Artificial Intelligence that focuses on the capacity to learn from information, is also part of AI. Machine learning lacks “thinking ability” in the human sense, as it does not have consciousness or intent. Despite this, its immense capacity to process information allows it to conduct predictive analysis and enhance its performance with experience.

Another key distinction concerns the relationship between machine learning and statistics. Although ML shares many theoretical foundations with statistics-particularly with regard to data modeling and analysis-the two fields diverge in purpose and approach. Statistics is primarily geared toward describing and interpreting phenomena based on theoretical assumptions, such as a known distribution, and analysis is constructed to test a hypothesis or test a relationship. Machine learning, on the other hand, does not start from rigid assumptions, but learns directly from the data, even in the absence of a predefined structure. The output of ML is often aimed at predicting new outcomes, optimizing performance even on examples not in the original sample.

From a practical point of view, statistics works with structured and representative datasets, while machine learning fits unstructured data, such as images, text, or graphs. Moreover, ML models are designed to be generalizable: that is, to adapt not only to the current situation but also to new contexts, making them more flexible but also more demanding in terms of computational resources.

Large-scale adoption of machine learning poses significant hardware-related challenges. The datasets used are often enormous in size, resulting not only in the need for significant amounts of memory, but also for high-performance processors with many cores and high computational speed. One of the main problems is the inability, in certain contexts, to wait for days for the results of an analysis: analysts need quick responses, even at the expense of absolute accuracy.

Some key aspects to consider when working with machine learning include: obtaining a useful result before refining it, it is preferable to reach an initial functional solution, avoiding unnecessarily complicating the algorithm, which may become unsound or applicable only to a specific dataset. ask the right question, many failures result from an incorrect formulation of the initial problem. Endlessly optimizing an algorithm will never be effective if the starting question is wrong, do not blindly rely on intuition: the analyst's intuition is a starting point, but in ML, mistakes are more frequent than successes. It is therefore necessary to question one's assumptions, even when they seem reasonable.

3.6.2. Types of ML Models

As illustrated in the previous section, Machine Learning is a system capable of learning from data, reasoning, and experience. It is precisely the way in which this learning takes place that is the main distinction between the different types of Machine Learning, each characterized by a unique approach and specific problems it is able to address. ML models can be classified into four main categories, which will be presented in the following section of this chapter [29].

Supervised Learning

Supervised learning is one of the most common methods in machine learning. This method is based on the utilization of labeled data, in which the dataset example comprises an input and its corresponding output. The model aims to reduce the error between generated predictions and actual values by learning a function that can accurately map inputs to outputs. Regression and classification are two uses for this kind of learning. The most relevant ML methods in Project Management field belong to supervised regression, such as Linear Models, Bayesian Models, Gradient Descent-Based, Robust Regression, Nonlinear Models, Ensemble Methods, Neural Networks.

Unsupervised Learning

Unsupervised learning allows the algorithm to recognize patterns in the data on its own by learning from examples. As a result, the data are restructured into classes or new values that can be used in subsequent analyses. A practical example is automated recommendation systems, which estimate a user's preferences by comparing them with those of similar customers.

Semi-Supervised Learning

Semi-supervised learning combines the characteristics of supervised and unsupervised learning and it is particularly useful when labeling the data would complex. The model is trained on a small portion of labeled data and a large amount of unlabeled data, with the objective of learning features that can make accurate predictions about the output variable.

Reinforcement Learning

Reinforcement learning happens when an algorithm tosses out a guess and then hears, in plain yes or no terms, whether it nailed the shot or flubbed it. The approach fits any situation with real-time choices that matter, offering hard advice about what the system ought to do next. Its mechanics mirror the old trial-and-error habit of people, with each small price paid for a slip sharpening the next attempt. A classic example is the use of reinforcement learning in video games, where the algorithm explores different actions and learns which ones to avoid in order to survive or improve the outcome.

3.6.3. State of the Art: Machine Learning for Project Cost and Schedule Forecasting

In more recent years, the use of ML models in project time and cost forecasting has gained considerable attention. The focus of these works tends to address the restriction of traditional techniques like Earned Value Analysis which do not adequately account for the dynamic non-linear interactions present within projects. Instead, ML offers flexible and adaptive tools that can learn from historical data and progressively improve the accuracy of estimates.

This section presents and analyzes some of the most significant contributions in the recent literature, each proposing different approaches-from recurrent neural networks to hybrid models and regression techniques-applied to real or simulated datasets. [Table 3-1] summarizes their main features, while the following paragraphs provide a detailed description of each study.

Table 3-1, A summary of the reviewed studies on ML applications for project forecasting

Authors	Title	ML Method	Forecasted Variable
Santos et al. (2023)	Explainable Machine Learning for Project Management Control	Random Forest, Gradient Boosting, XGBoost + SHAP	Cost and time (EAC, TEAC)
İnan et al. (2022)	A Machine Learning Study to Enhance Project Cost Forecasting	LSTM (Recurrent Neural Networks)	Cost (EAC)
ForouzeshNejad et al. (2024)	Optimizing Project Time and Cost Prediction Using a Hybrid XGBoost and Simulated Annealing Algorithm	XGBoost + Simulated Annealing	Cost and time
Yalçın et al. (2024)	Evaluation of Earned Value Management-Based Cost Estimation via Machine Learning	ANN, M5Tree, GPR, ANFIS, LSTM, SVM	Cost (CEAC)
Ottaviani & De Marco (2022)	Multiple Linear Regression Model for Improved Project Cost Forecasting	Multiple Linear Regression (LASSO)	Cost (EAC)

“Explainable Machine Learning for Project Management Control”

Among the most significant and recent contributions in the field of predictive project management, the work [30] proposes an advanced extension of Earned Value Management analysis, integrating machine learning models and interpretability techniques to improve control capacity in project environments characterized by high uncertainty. This study is part of the strand of research that recognizes the limitations of traditional deterministic methods, such as EVM, which assume fixed durations and costs for activities and fail to dynamically model the variability observed in real projects.

The proposed methodology starts with stochastic project modeling, in which activities are represented as nodes in a Critical Path Method or PERT method and their durations follow probability distributions. From this framework, the authors generate several simulations of the project using Monte Carlo, obtaining a synthetic but representative dataset of the dynamic behavior of the project under uncertainty. On this dataset, several machine learning models-especially ensemble methods such as Random Forest, Gradient Boosting, and XGBoost are trained with the goal of predicting key variables such as the final duration of the project or the expected time at the time of control, depending on the analysis scenario.

In the modeling phase, the authors compared four ensemble algorithms (Random Forest, Gradient Boosting, XGBoost and AdaBoost), selecting the best one through 5-fold nested cross-validation. For the regression case, the prediction of the final project duration, the best performing model was Gradient Boosting Regressor, with an average Mean Squared Error (MSE) of 9.19. It was followed by Random Forest (MSE ≈ 9.20) and XGBoost (MSE ≈ 9.21), while AdaBoost reported lower performance (MSE ≈ 11.06) [Figure 3-12].

Model	MSE (mean)	MSE (stdv)
Gradient Boosting Regressor	9.1939	0.2734
Random Forest Regressor	9.2052	0.2514
XGBoost Regressor	9.2168	0.2687
AdaBoost Regressor	11.0620	0.1232

Figure 3-12, Results for regression model selection, taken from [30]

Also, in classification-that is, in predicting whether the project will finish late or not-Gradient Boosting achieved the best accuracy with an average of 85.7%, confirming the robustness of this approach in both analytical contexts [Figure 3-13].

Model	Accuracy (mean)	Accuracy (stdv)
Gradient Boosting Classifier	0.8570	0.0020
XGBoost Classifier	0.8564	0.0023
Random Forest Classifier	0.8544	0.0035
AdaBoost Classifier	0.8537	0.0028

Figure 3-13, Results for classification model selection, taken from [30]

The study also introduces a layer of explainability based on the SHAP (SHapley Additive exPlanations) technique, which allows ML model predictions to be interpreted in terms of the contribution of individual activities. SHAP analysis is in fact applied on the simulated activity durations: each activity is treated as an explanatory variable, and it is measured how much it contributes-positively or negatively-to the final model prediction, both globally and for individual simulations.

The analysis is conducted in two distinct but complementary modes: forward analysis and backward analysis. The first has a forward-looking orientation and aims to support execution and rescheduling decisions: through SHAP summary plots and dependence plots, the impact of future activities on the expected time or cost of the project is identified, highlighting any interactions between tasks. The second mode allows the current project situation to be explained by attributing root causes to specific activities, with the goal of providing causal analysis. At this stage, this study uses SHAP waterfall plots to visualize the cumulative contribution of activities to the prediction obtained at a given control point, strengthening the diagnostic capability of the model.

In addition to methodological aspects, the paper also carefully discusses managerial implications. It highlights how the combined use of simulations, ML and Explainable Artificial Intelligence (XAI) can provide project managers with a more transparent decision support tool than traditional methods, capable not only of predicting but also of explaining. This feature is particularly relevant in complex contexts, where it is critical to justify data-driven choices and effectively communicate results with stakeholders.

Overall, the proposed framework stands out for its generalizability (being model-agnostic), for its consistency with the stochastic nature of real projects, and for the analytical depth offered through visual and quantitative tools that make project control more interpretable, responsive, and accountable.

“A Machine Learning Study to Enhance Project Cost Forecasting”

This study [31] proposes a predictive model based on Machine Learning techniques to improve the accuracy of Cost Estimate at Completion forecasts. The Earned Value Management approach's structural limitations, specifically its forecasting component based on linear indices such as Cost Performance Index (CPI) and Schedule Performance Index (SPI), serve as the primary driving force behind the work. Despite their widespread use, these metrics are predicated on assumptions that frequently do not accurately represent the nonlinear patterns, often resembling "S" curves, that characterize the actual cost behavior in projects.

To overcome these limitations, the authors develop a supervised learning model based on recurrent neural networks, and in particular the Long Short-Term Memory (LSTM) architecture, which is particularly well suited to modeling temporal sequences. The model was trained from historical project data provided by Ghent University's OR&S database, same database used in this work, of which 41 were selected for the final analysis (all in the construction industry).

The input to the ML model consists of seven-dimensional feature vector: six of these are derived from the raw values and moving averages of the CPI and SPI, calculated with moving windows of two and three points; the seventh is the normalized time (given by the ratio of actual duration to planned duration). The target to be predicted is the CEAC, expressed as the final estimate of the cost of project completion.

The validation protocol involves a series of 300 experiments, in which for each iteration 12 projects are used for training and 3 for testing using a cross-validation approach.

The accuracy of the predictions was measured by Mean Absolute Percentage Error (MAPE). The results showed that in 75.33% of the cases, the ML model produced more accurate cost estimates than the linear EVM model used as a benchmark. About half of the projects showed an improvement in MAPE on the order of 1 percent, while in the remaining cases the improvement was even more pronounced [Figure 3-14]. The authors point out that while this is an apparently small deviation, the added value of the LSTM model lies in its ability to learn specific cost growth patterns from historical data, overcoming the rigidities of fixed formulas.

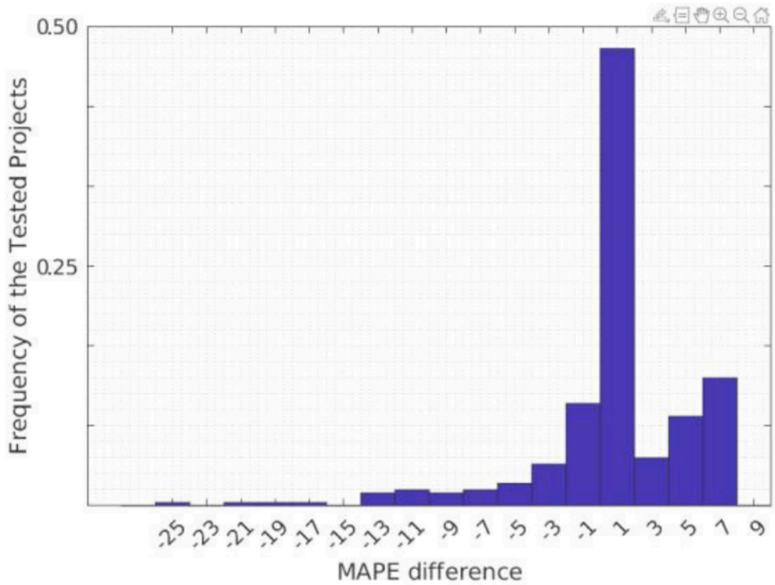

Figure 3-14, The MAPE difference between the proposed ML mode and the EVM model, taken from [31]

"Optimizing Project Time and Cost Prediction Using a Hybrid XGBoost and Simulated Annealing Algorithm"

In the context of construction projects, where the complexity of activity networks and environmental uncertainty make it difficult to obtain reliable predictions, the study [32] proposes an innovative model to simultaneously estimate project time and cost by integrating machine learning and metaheuristic techniques. The authors develop a hybrid approach based on the eXtreme Gradient Boosting (XGBoost) algorithm, enhanced with weight optimization via Simulated Annealing (SA), which allows for more precise calibration of model parameters.

The goal is to address the limitations of traditional methods which are based on static and linear formulas that are inadequate to represent the real dynamics of complex projects. In contrast, the XGBoost-SA model is data-driven, learning from historical data and dynamically adapting to each new project situation. The model was applied to a dataset consisting of 960 real construction project records, in which each record includes 15 features, including: number of activities, number of critical activities, minimum and maximum duration of activities, project progression, number of internal/external contracts, domestic/foreign production rates, human resources employed, as well as three uncertainty indices (economic, technical and environmental).

The optimization resulted in an improvement in MSE (Mean Squared Error) after about 380 iterations. The final metrics obtained (Accuracy = 0.923, Precision = 0.919, Recall = 0.921, F1-score = 0.919) show high performance in the test set.

The model was then compared with other ML algorithms, such as Artificial Neural Network (ANN), Support Vector Regression (SVR) and Decision Tree Regression (DTR). In all key metrics, the XGBoost-SA algorithm proved superior, confirming the validity of the hybrid approach, particularly in designs with high uncertainty and complex network structures [Figure 3-15].

Interpretability of the model was addressed through two approaches: Information Gain by XGBoost, which showed that the most influential features are the percentage of project progress, number of foreign contractors, and maximum durations of activities and SHAP analysis, which confirmed the importance of the same features and showed the directional effect (positive or negative) of each on the forecast.

The study also includes a direct comparison with EVM and ESM models. Out of 10 test designs the average error in time estimation was found to be reduced by 80% compared to EVM/ESM and the average error in cost estimation was reduced by about 50%.

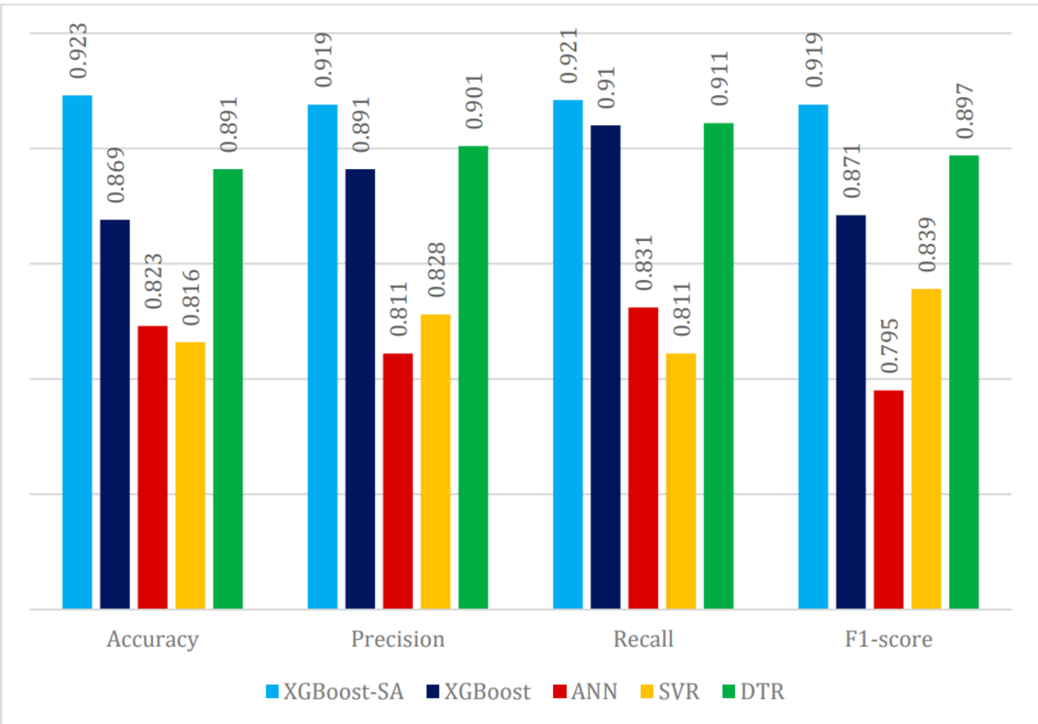

Figure 3-15,Comparison of algorithm evaluation indices, taken from [32]

“Evaluation of Earned Value Management-Based Cost Estimation via Machine Learning”

An interesting development in the application of machine learning in project control is the [33] which investigates how cost at completion estimation methods derived from the Earned Value Managemen approach can be enhanced through the integration of machine learning algorithms. The focus of the study is the comparative evaluation of 19 different EVM formulas for calculating CEAC, employing six different Machine Learning models, with the aim of improving predictive accuracy and adaptability to real-world and dynamic contexts.

The study is based on empirical data from a residential construction project in Turkey. Project activities were monitored daily for a period of 122 days, generating a dataset consisting of 2318 observations, each corresponding to the daily value of an CEAC method. The CEAC formulas employed are derived from both classic EVM metrics and more advanced metrics also incorporating combined approaches with α and β weights applied to cost and time [Figure 3-16].

ID	Method	EAC	Mathematical Equation	Explanation
1	EVM	PV1	$EAC_{PV1} = AC + (BAC - EV)$	Methods 1–5 [30] consider PF = 1, CPI, SPI, SCI, and weighted time/cost performance. For cost estimation, α = 0.8 and β = 0.2 as the cost performance index (CPI) should significantly outweigh the schedule performance index (SPI).
2	EVM	PV2	$EAC_{PV2} = AC + (BAC - EV)/CPI$	
3	EVM	PV3	$EAC_{PV3} = AC + (BAC - EV)/SPI$	
4	EVM	PV4	$EAC_{PV4} = AC + (BAC - EV)/SCI$	
5	EVM	PV5	$EAC_{PV5} = AC + (BAC - EV)/(\alpha\, CPI + \beta\, SPI)$	
6	EVM	ES1	$EAC_{ES1} = AC + (BAC - EV)/SPI(t)$	Methods 6–8 [12,13] consider PF = SPI(t), SCI(t), and weighted time/cost performance. For method 8, α = 0.8 and β = 0.2.
7	EVM	ES2	$EAC_{ES2} = AC + (BAC - EV)/SCI(t)$	
8	EVM	ES3	$EAC_{ES3} = AC + (BAC - EV)/\alpha\, CPI + \beta\, SPI(t)$	
9	EVM	SP1	$EAC_{SP1} = BAC/SCI$	Method 9 [12,31] considers the actual performance of the project in both terms of progress (duration) and costs.
10	ESM	ESM1	$EAC_{ESM1} = AC + (BAC - EV(e))$	Methods 10–19 are similar to Methods 1–8 [12,13].
11	ESM	ESM2	$EAC_{ESM2} = AC + (BAC - EV(e))/CPI(e)$	
12	ESM	ESM3	$EAC_{ESM3} = AC + (BAC - EV(e))/SPI(e)$	
13	ESM	ESM4	$EAC_{ESM4} = AC + (BAC - EV(e))/SCI(e)$	
14	ESM	ESM5	$EAC_{ESM5} = AC + (BAC - EV(e))/SPI(t)(e)$	
15	ESM	ESM6	$EAC_{ESM6} = AC + (BAC - EV(e))/SCI(t)(e)$	
16	ESM	ESM7	$EAC_{ESM7} = AC + (BAC - EV(e))/\alpha\, CPI + \beta\, SPI(e)$	
17	ESM	ESM8	$EAC_{ESM8} = AC + (BAC - EV(e))/\alpha\, CPI + \beta\, SPI(t)(e)$	
18	ESM	ESM9	$EAC_{ESM9} = AC + (BAC - EV(e))/\alpha\, CPI(e) + \beta\, SPI(e)$	
19	ESM	ESM10	$EAC_{ESM10} = AC + (BAC - EV(e))/\alpha\, CPI(e) + \beta\, SPI(t)(e)$	

Figure 3-16, CEAC method employed in [33]

To evaluate predictive performance, the study employs a full range of metrics: MAPE (Mean Absolute Percentage Error), RRMSE (Relative Root Mean Square Error), R^2, Nash-Sutcliffe Efficiency (NSE) and Overall Index of Performance (OI).

The results show that ML models outperform EVM formulas applied deterministically. In particular the ANN model stood out as the most accurate in terms of MAPE (0.2376%), RRMSE (0.3522) and R^2 (0.9878) in prediction by the EACPV1 method, the M5TREE model, although less effective in terms of R^2, performed best for the NSE (0.812) and OI (0.830) metrics, being the most effective in estimation by the EACPV5 method (weighted formula on CPI and SPI), the GPR and ANFIS algorithms showed good performance in specific cases, while LSTM and SVM performed the least well in almost all metrics.

Methodologically, the study has the merit of treating the integration of ML and EVM in a systemic way, offering a framework that can be replicated or adapted in other project contexts. In addition, it is highlighted how the careful selection of time variables (TED, ES(e), ED, DPI) can significantly improve the quality of estimates, making the model more sensitive to actual project performance.

"Multiple Linear Regression Model for Improved Project Cost"

[34] explores the improvement of cost forecasting accuracy using a Multiple Linear Regression (MLR) model within the framework of Earned Value Management (EVM). The goal is to develop a CEAC model that is both more accurate than classical EVM formulas and more stable, i.e., with less error variance. This study uses multiple linear regression, in contrast to many proposals based on intricate machine learning techniques that can occasionally be challenging to implement in industrial environments. The analysis is conducted on a dataset consisting of 805 observations from 29 real projects, extracted from the database developed by [25] and [24]. The variables considered are classic EVM variables, such as the Cost Performance Index (CPI), the percentage of work performed (WP), and an initial cost prediction (fEAC) calculated by indexed methods. These are combined in a multivariate regression model that aims to refine the estimate of EAC as the project progresses.

Three steps comprise the development of the model. First, a generalized linear model selection procedure (GLMSELECT) is used to identify the most significant regressors. Second, correlation analysis and multicollinearity tests are applied to verify whether there are any correlations between the variables. Lastly, to lower the chance of overfitting, a linear regression with LASSO and k-fold cross-validation is used.

The resulting model (rEAC) is based on three independent variables: the initial forecast of cost at completion (fEAC), the cost performance index (CPI), and the percentage of work completed (WP). The combination of these three elements makes it possible to construct a final cost estimate that incorporates both the historical economic performance of the project and its progress. Specifically, the initial forecast provides a solid base on which to anchor the estimate, while the CPI index adjusts its adjustment based on observed efficiency. The WP variable, on the other hand, has a corrective impact: its negative effect indicates that, as the project progresses, the influence of the first two components tends to diminish. This reflects greater data reliability in the advanced stages and a gradual reduction in uncertainty.

The model was compared with the traditional EVM methods. The results show a significant improvement in terms of prediction stability: the MAPE is reduced from 15.76% to 13.91%, while the standard deviation of the error drops from 0.199 to 0.110, showing greater predictive reliability [Figure 3-17]. The adjusted coefficient of determination R^2 is also very high (0.9842), indicating good explanatory power of the model.

Model	MAPE	Standard Deviation
fEAC	.1576	.1990
rEAC	.1391	.1102

Figure 3-17, Benchmarking of the fitted model performance, taken from [34]

Analysis of the residuals identified some patterns: the model tends to overestimate EAC in the early stages of projects, when the quality and availability of EVM data are lower. However, this behavior decreases as the WP approaches 100 percent, which confirms the underlying assumption that the reliability of the prediction improves as the project progresses.

4. Methodology

4.1. Dataset and data source

The data analyzed in this study comes from Ghent University, through the contribution of the Operations Research & Scheduling research group, which collected and structured a dataset consisting of 203 real projects from different companies [35]. This extensive database contains baseline scheduling data (network, resources, ...), risk analysis data (for Monte-Carlo simulations) and project control data (using EVM and ES metrics). These data are provided in the form of a list of projects, each accompanied by a summary sheet and an Excel file that includes visualizations and evaluations of the projects' temporal and economic performance.

The most significant data reported in this database and to be used in this study include tracking values, baseline schedule, and actual schedule.

- The tracking values provide a real snapshot of the project's progress, collecting Earned Value Analysis (EVA) metrics, with a focus on Earned Value Management (EVM) performance measures and forecasts derived from EVM Forecasting. Each project contains a different number of observations (snapshots) and also a different number of snapshots per progress stage. The next section will include a dedicated subsection [3.2.2.2] addressing this limitation.
- The baseline schedule represents the initial project plan, including information such as Budget at Completion (BAC) and planned duration.
- The actual schedule reflects actual progress, reporting real project duration and real cost at completion.

To ensure that the analyses in this study were based on reliable data, a stepwise selection process was undertaken on the 203 projects initially in the dataset. The main objective was to verify the authenticity of the data available for each project in order to obtain a robust and representative sample.

The first step involved assessing the presence of baseline schedule data. In particular, the Budget at Completion and the planned duration of the project were considered essential, as they are the pillars for any performance and forecast analysis. Only those projects that presented both of these pieces of information fully and consistently were retained, reducing the initial sample from 203 to 172 projects.

Next, the second step involved checking the availability of tracking values, which are those data that provide periodic updates on project performance through successive snapshots. These values are crucial for monitoring performance over time and applying Earned Value Management (EVM) methodologies. The application of this filter narrowed the sample to 118 projects, excluding those without such tracking.

Finally, a final quality check was conducted to ensure the reliability of the final dataset. 28 projects were removed that had tracking values but had inconsistent data, compromising their validity for the study. This process resulted in the final selection of 90 projects, which were considered complete and appropriate for analysis.

The result is thus a filtered dataset that can be considered robust and representative, suitable to support project performance analysis and provide reliable results for the objectives of this study.

4.2. Pipeline overview

The purpose of this section is to describe the proposed machine learning pipeline, going through the different stages and explaining their role in the process. The pipeline consists of four stages: data collection, data preprocessing, feature engineering, training and model evaluation. The following subsections will explain the purposes and activities of each step, providing a detailed overview of the workflow and techniques adopted.

4.2.1. Data collection

The data collection stage involves the gathering of data used in the pipeline from the dataset that has been previously filtered, as described in the selection steps given in the previous section of the thesis. Thus, this dataset includes EVA data including Budget at Completion (BAC), Planned Duration (PD), Actual Duration (AD), Actual Time (AT), Planned Value (PV), Earned Value (EV) and Actual Cost (AC). Moreover, it includes also number of tasks completed, number of tasks started, number of tasks planned to be completed or planned to be started, and data on the topology for each snapshot of the 90 selected projects.

To more clearly illustrate the data collected from the pipeline, a table showing the detail of a selected project will be presented below as an example. For visualization purposes, the example is provided split into two separate tables [Table 4-1, Table 4-2].

Table 4-1, project data collected by the pipeline

Code	Title	Type	Project	BAC	PD	AD	AC(AD)	AT	PV	EV	AC
C2011-05	Telecom System Agnes	Service	1	180485	5.0	5.0	180485	1	13527	13527	13527
C2011-05	Telecom System Agnes	Service	1	180485	5.0	5.0	180485	2	115358	115358	115358
C2011-05	Telecom System Agnes	Service	1	180485	5.0	5.0	180485	3	143433	129406	129406
C2011-05	Telecom System Agnes	Service	1	180485	5.0	5.0	180485	4	176069	166620	166620
C2011-05	Telecom System Agnes	Service	1	180485	5.0	5.0	180485	5	180485	180485	180485

Table 4-2, project data collected by the pipeline

Code	Title	Type	Project	Task completed	Task planned to be completed	Task planned to be started	Task started	SP_topology	AD_topology	LA_topology	TF_topology	RI_topology
C2011-05	Telecom System Agnes	Service	1	5	5	5	5	0.60	0.58	0.38	0.09	0.85
C2011-05	Telecom System Agnes	Service	1	11	11	13	13	0.60	0.58	0.38	0.09	0.85
C2011-05	Telecom System Agnes	Service	1	12	13	15	14	0.60	0.58	0.38	0.09	0.85
C2011-05	Telecom System Agnes	Service	1	15	18	18	16	0.60	0.58	0.38	0.09	0.85
C2011-05	Telecom System Agnes	Service	1	21	21	21	21	0.60	0.58	0.38	0.09	0.85

4.2.2. Data preprocessing

Data preprocessing includes the manipulations of available data necessary to ensure the accuracy and reliability of the analysis. In the pipeline, the process consists of two steps: data transformation and data balancing and augmentation. These steps are aimed at formatting and optimizing the data contained in the dataset presented in the previous section.

4.2.2.1. Data transformation

Data transformation, which consists of scaling EVA values, is meant to avoid underfitting and bias in the ML models used in this study. Various ML methods can be biased by different data scales, especially those that compute distances between points (e.g., K-Nearest Neighbors, Support Vector Machines) or use regularization techniques (such as L1 and L2 in Lasso and Ridge regression).

Scaling the data prevents quantities with higher values from dominating the model, thus improving predictions and making the learning process more stable and effective.

This stage of the pipeline scales EVA data by dividing cost metrics by BAC and time metrics by PD as shown in [Table 4-3]:

Table 4-3, Scaled metrics formula

Original metrics	Scaled metrics
BAC	BAC / BAC = 1
PD	PD / PD = 1
ADs	AD / PD
ATs	AT / PD
PS	PV / BAC
PC	EV / BAC
ACs	AC / BAC

The subscript "s" denotes the scaled metrics, except for PV and EV which scaled versions correspond to Percentage Scheduled (PS) and Percentage Completed (PC), respectively.

4.2.2.2. Data balancing and augmentation

Data balancing and augmentation are related steps which address the limitation of the collected data arising from the variability in the number of observation (snapshot) for each project and from the various number of snapshots per progress stage within each project.

Data balancing serves as a technique to equalize both the number of observations above-mentioned and data augmentation provides the needed methods to achieve this balance.

These steps help mitigate underfitting by ensuring that projects with a higher or lower number of observations do not influence the training process, thus reducing potential biases in the ML models. In addition, by balancing the number of observations for each progress stage it is possible to ensure that the models are trained on all projects stages.

The pipeline implements these methods by generating a fixed number of synthetic observations through linear interpolation of the project metrics, outlined in the previous subsections, at specific PC increments since the dataset's data point do not always align with these predefined increments. Given the chosen step size of 0.05, each project is represented by exactly 21 snapshots, ensuring a uniform distribution of data points from 0% to 100% completion.

The linear interpolation is defined by the following equation:

$$\tilde{x}(PC = z) = x_i + \frac{(z - PC_i)}{(PC_{i+1} - PC_I)} * (x_{i+1} - x_i)$$

Where:

- x is the metric to interpolate
- $\tilde{x}$ represents the interpolated value
- z denotes the specific value of PC at which interpolation is performed
- PC_i and PC_{i+1} are two known data point such as $PC_i \leq z \leq PC_{i+1}$

The following table illustrates the results of the data balancing and data augmentation step on the input dataset previously presented. For visualization purposes, the example is provided split into two separate tables [Table 4-4, Table 4-5].

Table 4-4, Project data following data balancing and data augmentation

Code	BAC	PD	AD	AC(AD)	AT	PV	EV	AC	Task completed	Task planned to be completed	Task planned to be started
C2011-05	180485	5	5	180485	0.00	0	0	0	0	0	0.00
C2011-05	180485	5	5	180485	0.67	9024	9024	9024	3.34	3.34	3.34
C2011-05	180485	5	5	180485	1.04	18049	18049	18049	5.27	5.27	5.36
C2011-05	180485	5	5	180485	1.13	27073	27073	27073	5.80	5.80	6.06
C2011-05	180485	5	5	180485	1.22	36097	36097	36097	6.33	6.33	6.77
C2011-05	180485	5	5	180485	1.31	45121	45121	45121	6.86	6.86	7.48
C2011-05	180485	5	5	180485	1.40	54146	54146	54146	7.39	7.39	8.19
C2011-05	180485	5	5	180485	1.49	63170	63170	63170	7.93	7.93	8.90
C2011-05	180485	5	5	180485	1.58	72194	72194	72194	8.46	8.46	9.61
C2011-05	180485	5	5	180485	1.66	81218	81218	81218	8.99	8.99	10.32
C2011-05	180485	5	5	180485	1.75	90243	90243	90243	9.52	9.52	11.03
C2011-05	180485	5	5	180485	1.84	99267	99267	99267	10.05	10.05	11.74
C2011-05	180485	5	5	180485	1.93	108291	108291	108291	10.58	10.58	12.44
C2011-05	180485	5	5	180485	2.14	119270	117315	117315	11.14	11.28	13.28
C2011-05	180485	5	5	180485	2.78	137305	126340	126340	11.78	12.56	14.56
C2011-05	180485	5	5	180485	3.16	148658	135364	135364	12.48	13.80	15.48
C2011-05	180485	5	5	180485	3.40	156572	144388	144388	13.21	15.01	16.21
C2011-05	180485	5	5	180485	3.65	164487	153412	153412	13.94	16.23	16.94
C2011-05	180485	5	5	180485	3.89	172401	162437	162437	14.66	17.44	17.66
C2011-05	180485	5	5	180485	4.35	177611	171461	171461	17.09	19.05	19.05
C2011-05	180485	5	5	180485	5	180485	180485	180485	21	21	21.00

Table 4-5, Project data following data balancing and data augmentation

Code	SP_topology	AD_topology	LA_topology	TF_topology	RI_topology	BAC_s	AT_s	WS	WP	AC_s	PD_s	AD_s	AC_s(AD_s)
C2011-05	0.60	0.58	0.38	0.09	0.85	1.00	0	0	0.00	0	1.00	1.00	1.00
C2011-05	0.60	0.58	0.38	0.09	0.85	1.00	0.13	0.05	0.05	0.05	1.00	1.00	1.00
C2011-05	0.60	0.58	0.38	0.09	0.85	1.00	0.21	0.10	0.10	0.10	1.00	1.00	1.00
C2011-05	0.60	0.58	0.38	0.09	0.85	1.00	0.23	0.15	0.15	0.15	1.00	1.00	1.00
C2011-05	0.60	0.58	0.38	0.09	0.85	1.00	0.24	0.20	0.20	0.20	1.00	1.00	1.00
C2011-05	0.60	0.58	0.38	0.09	0.85	1.00	0.26	0.25	0.25	0.25	1.00	1.00	1.00
C2011-05	0.60	0.58	0.38	0.09	0.85	1.00	0.28	0.30	0.30	0.30	1.00	1.00	1.00
C2011-05	0.60	0.58	0.38	0.09	0.85	1.00	0.30	0.35	0.35	0.35	1.00	1.00	1.00
C2011-05	0.60	0.58	0.38	0.09	0.85	1.00	0.32	0.40	0.40	0.40	1.00	1.00	1.00
C2011-05	0.60	0.58	0.38	0.09	0.85	1.00	0.33	0.45	0.45	0.45	1.00	1.00	1.00
C2011-05	0.60	0.58	0.38	0.09	0.85	1.00	0.35	0.50	0.50	0.50	1.00	1.00	1.00
C2011-05	0.60	0.58	0.38	0.09	0.85	1.00	0.37	0.55	0.55	0.55	1.00	1.00	1.00
C2011-05	0.60	0.58	0.38	0.09	0.85	1.00	0.39	0.60	0.60	0.60	1.00	1.00	1.00
C2011-05	0.60	0.58	0.38	0.09	0.85	1.00	0.43	0.66	0.65	0.65	1.00	1.00	1.00
C2011-05	0.60	0.58	0.38	0.09	0.85	1.00	0.56	0.76	0.70	0.70	1.00	1.00	1.00
C2011-05	0.60	0.58	0.38	0.09	0.85	1.00	0.63	0.82	0.75	0.75	1.00	1.00	1.00
C2011-05	0.60	0.58	0.38	0.09	0.85	1.00	0.68	0.87	0.80	0.80	1.00	1.00	1.00
C2011-05	0.60	0.58	0.38	0.09	0.85	1.00	0.73	0.91	0.85	0.85	1.00	1.00	1.00
C2011-05	0.60	0.58	0.38	0.09	0.85	1.00	0.78	0.96	0.90	0.90	1.00	1.00	1.00
C2011-05	0.60	0.58	0.38	0.09	0.85	1.00	0.87	0.98	0.95	0.95	1.00	1.00	1.00
C2011-05	0.60	0.58	0.38	0.09	0.85	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00

4.2.3. Feature engineering

Feature engineering is the process of transforming raw data into a more effective and machine-readable format input set, it involves creating, modifying, or selecting relevant features to enhance ML model performance.

In the pipeline, this stage involves combining EVA data to generate additional features, including both input features, which serve as independent regression variables, and target features, which represent the dependent variables to predict.

As part of this process, new metrics have been introduced using data from the input dataset, these features include:

- EVM and ESM metrics explained in the previous chapter, along with their respective scaled values: SV^{EVM} (Schedule Variance in EVM), SV^{ES} (Schedule Variance based on Earned Schedule), CPI (Cost Performance Index), SPI^{EVM} (Schedule Performance Index in EVM), SPI^{ES} (Schedule Performance Index based on Earned Schedule), and $TSPI^{ES}$ (Time-Schedule Performance Indicator)
- Other dynamic measures explained in the previous chapter: TFCI (Total Float Consumption Index), SPI vs. TSPI (Schedule Performance Index - Time vs. Earned Schedule), BEI (Baseline Execution Index), BEI^{start} (Baseline Execution Index at start), ED(t) (Earned Duration), T/S (Time to Schedule Ratio), T/R (Time to Cost Ratio), and CSI (Cost Schedule Index)

4.2.3.1. Input features

In this stage of the process, following the initial three steps, data collection, data preprocessing, and feature engineering, the fundamental task is to select the variables that will serve as inputs for the Machine Learning models. The feature selection process is driven by the objective of maximizing the model’s predictive performance while minimizing noise and redundancy in the data. The table below presents the selected input features. For visualization purposes, the example is provided split into two separate tables [Table 4-6, Table 4-7].

Table 4-6, Selected input features

Code	BAC_s	PD_s	AD_s	AC_s(AD_s)	AT_s	WS	WP	AC_s	ES_s	BEI	BEIstart	CV_s	SV^EVM_s	SV^ES_s	CPI
C2011-05	1.00	1.00	1.00	1.00	0.00	0.00	0.00	0.00	0.00	1.00	1.00	0.00	0.00	0.00	1.00
C2011-05	1.00	1.00	1.00	1.00	0.13	0.05	0.05	0.05	0.13	1.00	1.00	0.00	0.00	0.00	1.00
C2011-05	1.00	1.00	1.00	1.00	0.21	0.10	0.10	0.10	0.21	1.00	1.00	0.00	0.00	0.00	1.00
C2011-05	1.00	1.00	1.00	1.00	0.23	0.15	0.15	0.15	0.23	1.00	1.00	0.00	0.00	0.00	1.00
C2011-05	1.00	1.00	1.00	1.00	0.24	0.20	0.20	0.20	0.24	1.00	1.00	0.00	0.00	0.00	1.00
C2011-05	1.00	1.00	1.00	1.00	0.26	0.25	0.25	0.25	0.26	1.00	1.00	0.00	0.00	0.00	1.00
C2011-05	1.00	1.00	1.00	1.00	0.28	0.30	0.30	0.30	0.28	1.00	1.00	0.00	0.00	0.00	1.00
C2011-05	1.00	1.00	1.00	1.00	0.30	0.35	0.35	0.35	0.30	1.00	1.00	0.00	0.00	0.00	1.00
C2011-05	1.00	1.00	1.00	1.00	0.32	0.40	0.40	0.40	0.32	1.00	1.00	0.00	0.00	0.00	1.00
C2011-05	1.00	1.00	1.00	1.00	0.33	0.45	0.45	0.45	0.33	1.00	1.00	0.00	0.00	0.00	1.00
C2011-05	1.00	1.00	1.00	1.00	0.35	0.50	0.50	0.50	0.35	1.00	1.00	0.00	0.00	0.00	1.00
C2011-05	1.00	1.00	1.00	1.00	0.37	0.55	0.55	0.55	0.37	1.00	1.00	0.00	0.00	0.00	1.00
C2011-05	1.00	1.00	1.00	1.00	0.39	0.60	0.60	0.60	0.39	1.00	1.00	0.00	0.00	0.00	1.00
C2011-05	1.00	1.00	1.00	1.00	0.43	0.66	0.65	0.65	0.42	0.99	0.99	0.00	-0.01	-0.01	1.00
C2011-05	1.00	1.00	1.00	1.00	0.56	0.76	0.70	0.70	0.48	0.94	0.95	0.00	-0.06	-0.08	1.00
C2011-05	1.00	1.00	1.00	1.00	0.63	0.82	0.75	0.75	0.54	0.90	0.93	0.00	-0.07	-0.09	1.00
C2011-05	1.00	1.00	1.00	1.00	0.68	0.87	0.80	0.80	0.60	0.88	0.91	0.00	-0.07	-0.08	1.00
C2011-05	1.00	1.00	1.00	1.00	0.73	0.91	0.85	0.85	0.66	0.86	0.90	0.00	-0.06	-0.07	1.00
C2011-05	1.00	1.00	1.00	1.00	0.78	0.96	0.90	0.90	0.72	0.84	0.89	0.00	-0.06	-0.06	1.00
C2011-05	1.00	1.00	1.00	1.00	0.87	0.98	0.95	0.95	0.77	0.90	0.93	0.00	-0.03	-0.10	1.00

Table 4-7, Selected input features

Code	SPI^EVM	SPI^ES	ED(t)	(T/S)	(T/C)	SP_topology	AD_topology	LA_topology	TF_topology	RI_topology	CSI	SPIt - TSPIt	TFCI	cPF	sPF
C2011-05	1.00	1.00	0.00	1.00	1.00	0.60	0.58	0.38	0.09	0.85	1.00	0.00	1.00	1.00	1.00
C2011-05	1.00	1.00	0.13	0.00	0.00	0.60	0.58	0.38	0.09	0.85	1.00	0.00	1.00	1.00	1.00
C2011-05	1.00	1.00	0.21	0.00	0.00	0.60	0.58	0.38	0.09	0.85	1.00	0.00	1.00	1.00	1.00
C2011-05	1.00	1.00	0.23	1.00	1.00	0.60	0.58	0.38	0.09	0.85	1.00	0.00	1.00	1.00	1.00
C2011-05	1.00	1.00	0.24	0.00	0.00	0.60	0.58	0.38	0.09	0.85	1.00	0.00	1.00	1.00	1.00
C2011-05	1.00	1.00	0.26	1.00	1.00	0.60	0.58	0.38	0.09	0.85	1.00	0.00	1.00	1.00	1.00
C2011-05	1.00	1.00	0.28	1.00	1.00	0.60	0.58	0.38	0.09	0.85	1.00	0.00	1.00	1.00	1.00
C2011-05	1.00	1.00	0.30	1.00	1.00	0.60	0.58	0.38	0.09	0.85	1.00	0.00	1.00	1.00	1.00
C2011-05	1.00	1.00	0.32	0.00	0.00	0.60	0.58	0.38	0.09	0.85	1.00	0.00	1.00	1.00	1.00
C2011-05	1.00	1.00	0.33	1.00	1.00	0.60	0.58	0.38	0.09	0.85	1.00	0.00	1.00	1.00	1.00
C2011-05	1.00	1.00	0.35	1.00	1.00	0.60	0.58	0.38	0.09	0.85	1.00	0.00	1.00	1.00	1.00
C2011-05	1.00	1.00	0.37	1.00	1.00	0.60	0.58	0.38	0.09	0.85	1.00	0.00	1.00	1.00	1.00
C2011-05	1.00	1.00	0.39	0.00	0.00	0.60	0.58	0.38	0.09	0.85	1.00	0.00	1.00	1.00	1.00
C2011-05	0.98	0.98	0.42	-1.00	1.00	0.60	0.58	0.38	0.09	0.85	0.98	-0.03	1.00	1.00	1.01
C2011-05	0.92	0.86	0.51	-1.00	1.00	0.60	0.58	0.38	0.09	0.85	0.92	-0.32	1.00	1.00	1.18
C2011-05	0.91	0.86	0.58	-1.00	1.00	0.60	0.58	0.38	0.09	0.85	0.91	-0.38	1.00	1.00	1.24
C2011-05	0.92	0.89	0.63	-1.00	1.00	0.60	0.58	0.38	0.09	0.85	0.92	-0.35	1.00	1.00	1.24
C2011-05	0.93	0.91	0.68	-1.00	1.00	0.60	0.58	0.38	0.09	0.85	0.93	-0.34	1.00	1.00	1.25
C2011-05	0.94	0.92	0.73	-1.00	1.00	0.60	0.58	0.38	0.09	0.85	0.94	-0.35	1.00	1.00	1.27
C2011-05	0.97	0.89	0.84	-1.00	1.00	0.60	0.58	0.38	0.09	0.85	0.97	-0.87	1.00	1.00	1.75

4.2.3.2. Target features

The regression method alongside the forecasting targets of this study, which can be either cost forecasting or duration forecasting, set the target features of the pipeline. More specifically, direct regression (DR) employs as target variables directly $AC_S(AD_S)$ (cost forecasting) or AD_S (duration forecasting). While indirect regression (IR) assigns the target features to intermediate variables which is then used to compute the forecasting targets through specific formulas. The pipeline implements both regression methods for both forecasting targets.

The DR method evaluates forecasts as per

$$\hat{y} = f^{DR}(X)$$

Where:

- y is the real value of the target feature and $\hat{y}$ denotes the forecast
- X represents the set of input features
- f^{DR} denotes the regression models developed through DR using $AC_S(AD_S)$ or AD_S as target features.

the IR method evaluates cost forecasts as per

$$\hat{y} = AC_S + \frac{(1 - PC)}{\widehat{cPF}}$$

With

$$\widehat{cPF} = f^{IR}(X)$$

Where f^{IR} denotes the regression models developed through IR using cPF^* as target feature, which is defined as the value such that

$$AC_S(AD_S) = AC_S + \frac{(1 - PC)}{cPF^*}$$

Similarly, the IR method evaluates duration forecasts as per

$$\hat{y} = AT_S + \frac{(1 - ES_S)}{\widehat{sPF}}$$

With

$$\widehat{sPF} = g^{IR}(X)$$

Where g^{IR} denotes the regression models developed through IR using sPF^* as target feature, which is defined as the value such that:

$$AD_S = AT_S + \frac{(1 - ES_S)}{sPF^*}$$

4.2.4. Model training

The model training phase of the pipeline is a key step to ensure that the models not only learn patterns from the data but also generalize this learning to work effectively on unseen data. To achieve this goal, this phase combines cross-validation strategy, feature selection technique, and hyperparameter tuning. The following sections describe each of these components in detail, explaining how they are implemented and how they contribute to the construction of robust and reliable predictive models.

4.2.4.1. Cross validation

As mentioned above, a central question in supervised learning concerns the generalization ability of the resulting model. To assess this ability, data resampling methods such as cross-validation are used [36].

Since the input dataset contains multiple observations for each project, the goal is to test on entire groups. For this reason, the pipeline implements the Leave-One-Group-Out Cross-Validation (LOGO CV) procedure, both for model evaluation and within the training steps.

LOGO CV divides the dataset into P distinct folds, where P represents the total number of projects. In each fold, the observations of the p^{th}-project are used as the validation set, while other observations serve as the training set. This approach avoids group leakage, which would otherwise bias model evaluation, distorting model performance assessment.

Repeating the train-validate split for each project in the dataset ensures the consistency and reproducibility of results.

4.2.4.2. Feature selection

Having many features available may give the impression that the machine learning model possesses all the necessary information to solve the problem. However, this is not necessarily the case: the presence of numerous features does not automatically ensure good predictive performance. What is needed is the quality and relevance of the features in relation to the problem and the target. It is essential that the selected features are informative and capable of effectively guiding the model toward accurate and reliable predictions.

Feature selection represents an effective solution, as it involves identifying a subset of input features to develop the model. This technique reduces overfitting by narrowing the dimensionality of the set of input variables and limits the number of relationships that ML methods must analyze. It also speeds up the training process by eliminating unnecessary data and improves model clarity by focusing on the features that have the greatest impact on predictions.

The feature selection procedure adopted in the pipeline is the forward Sequential Feature Selection (fSFS) combined with LOGO CV. It starts with a model that includes no input features, and at each LOGO CV iteration, the procedure evaluates which feature would lead to the greatest improvement in predictive performance based on a cross-validation scoring metric. Only features that provide a performance gain above a certain threshold are added. The selection process continues iteratively until either all features have been evaluated, or the addition of new features no longer leads to improvements.

4.2.4.3. Hyperparameters tuning

The possible combinations of hyperparameter values for the ML models can be numerous, making the optimal selection of them complex in order to maximize model performance. To address this challenge, hyperparameter search and optimization techniques were developed to identify the most effective combination based on the score obtained. In the implemented pipeline, the approach used to test different combinations of hyperparameters, through LOGO CV, and select the best performing one is the Grid Search.

This procedure entails pinpointing a specific parameter grid and searching and evaluating all potential combinations of values utilizing LOGO CV on the provided dataset. The model is then iteratively trained and validated for various combinations, ultimately selecting the one with the optimal CV scorer as the final best metric.

4.2.5. Model evaluation

The Model evaluation phase focuses on validating the effectiveness of the ML methods in predicting project duration and cost. This phase involves using benchmarking the trained models against standard EVM and ESM models to assess their performance. Furthermore, to conduct a more in-depth analysis of the models and understand the influence of individual features on their predictions, a SHAP analysis is performed.

4.2.5.1. Benchmarking

To evaluate the performance of ML methods, this study benchmarks their results with those obtained through EVM and ESM models. The comparative analysis is based on two regression metrics used in the evaluation of predictive models: the Mean Absolute Error (MAE) and the Root Mean Square Error (RMSE).

Mean Absolute Error (MAE)

The MAE measures the mean accuracy of the model which is the average absolute difference between the predicted values and the actual target values, and it is determined as per

$$MAE = \frac{\sum_{i=1}^{n}|E_i|}{n}$$

Where:

- i denotes the ith record with $i = 1,2, \ldots, n$
- E_i represents the forecast residual of the ith observation, as per

$$E_i = y_i - \widehat{y_i}$$

Root Mean Square Error (RMSE)

The RMSE measures the mean precision of the model by penalizing larger errors, and it is determined as per

$$RMSE = \sqrt{\frac{\sum_{i=1}^{n} {E_i}^2}{n}}$$

Where:

- i denotes the ith record with $i = 1,2, \dots., n$
- E_i represents the forecast residual of the ith observation, as per

$$E_i = y_i - \widehat{y_i}$$

4.2.5.2. SHAP Analysis

This study aims to evaluate the effect of parameters, with a focus on dynamic parameters, on the performance of machine learning models and to understand how they affect predictions. To analyze this impact, the pipeline involves the use of SHAP (Shapley Additive Explanations) analysis. This analysis is a method based on the concept of Shapley value, introduced by Lloyd Shapley in 1953 as part of cooperative game theory, it makes it possible to quantify the contribution of each variable to the prediction of a machine learning model by assigning each feature a value that represents its impact on the predictions. [37]

Shapley value is a concept that distributes "merit" equally among participants in a collaborative system. Applied to machine learning, it allows the importance of a feature to be measured by evaluating its contribution to model prediction, considering all possible combinations of the other features.

The Shapley value for feature X_j in a model is given by:

$$X_j = \sum_{S \subseteq N \setminus \{j\}} \frac{k!\,(p-k-1)!}{p!} \left(f(S \cup \{j\}) - f(S)\right)$$

Where:

- p denotes the total number of features
- $N \setminus \{j\}$ is a set of all possible combinations of features excluding X_j
- S is a feature set in $N \setminus \{j\}$
- $f(S)$ is the model prediction with features in S
- $f(S \cup \{j\})$ is the model prediction with features in S plus feature X_j

SHAP developed by Lundberg and Lee (2017) represents an extension of Shapley values; the main innovation of SHAP is the generation of local additive attributions for features [38].

SHAP values can be approximated by several techniques, optimized for specific types of models and datasets, such as Tree SHAP, optimized for decision tree-based models, Kernel SHAP, a more general method, Deep SHAP, for neural networks, and LIME, which approximates SHAP for local interpretations.

In the pipeline developed in this study, SHAP analysis was implemented to automatically adapt to the type of model used, optimizing the calculation of SHAP values.

5. Results

This section presents a comparative analysis of the performance of 30 Machine Learning algorithms, implemented through the proposed pipeline, compared to traditional models based on EVM and ESM. The comparison was performed on a real dataset of projects, selected as a case study. The ML algorithms were chosen based on their widespread use in the literature, and cover a wide range of modeling categories, including Linear, Bayesian, Robust, Nonlinear, Ensemble, and Neural Network approaches. [Table 5-1] outlines the methods tested, specifying for each its category, subcategory, and abbreviation used for ease of reference.

The models were implemented in Python 3.13.1, using the Scikit-learn libraries for Sequential Feature Selection (SFS), validation via Leave-One-Group-Out (LOGO), and performance evaluation using Mean Absolute Error (MAE) and Root Mean Square Error (RMSE). The data manipulation and preparation steps were handled with Pandas and NumPy, while the visualization of the results was done through Matplotlib and Seaborn. Model interpretability and feature importance analysis were obtained through SHAP (SHapley Additive exPlanations).

In the following paragraphs, the results are presented on three levels of analysis: at the global dataset level, at the project progress stage level, and finally by interpretation of the features most influential on the predictive output.

Table 5-1, ML methods tested

Category/Subcategory	Method	Abbreviation
Linear	Ordinary Least Squares	OLS
Linear	Ridge	Ridge
Linear	Least Absolute Shrinkage and Selection Operator	Lasso
Linear	Elastic Net	EN
Linear	Least Angle Regression	Lars
Linear	Lasso Least Angle Regression	Lasso Lars
Linear	Orthogonal Matching Pursuit	OMP
Linear	Passive Aggressive	PA
Bayesian	Bayesian Ridge	BR
Bayesian	Automatic Relevance Determination	ARD
Generalized Linear Model	Tweedie	Tweedie
Stochastic Gradient Descent	Stochastic Gradient Descent	SGD
Stochastic Gradient Descent	One-Class Support Vector Machine using SGD	SGD1cSVM
Robust Regression	Random Sample Consensus	RANSAC
Robust Regression	Huber	Huber
Nonlinear	Kernel Ridge	KR
Nonlinear	Support Vector Regression	SVR
Nonlinear	Nu Support Vector Regression	NuSVR
Nonlinear	k-Nearest Neighbors	k-NN
Nonlinear	Gaussian Process	GP
Nonlinear	Decision Tree	DT
Nonlinear	Extremely Randomized Tree	ERT
Ensemble Methods	Random Forest	RF
Ensemble Methods	Extremely Randomized Trees	ERTs
Ensemble Methods	Adaptive Boosting	AdaBoost
Ensemble Methods	Gradient Boosting	GB
Ensemble Methods	Histogram-based GB	HGB
Ensemble Methods	XGBoost	XGB
Ensemble Methods	XGBoost RF	XGB RF
Neural Network	Multilayer Perceptron	MLP

5.1. Dataset level

[Table 5-2] reports the values of the regression metrics for the EVM, ESM models and the 30 Machine Learning algorithms tested, related to the prediction of both duration and cost at project completion, considering the entire dataset. MAE and RMSE values are reported for each model, calculated using both Direct Regression and Indirect Regression strategies. Values highlighted in bold represent the best performance obtained in each metric column.

The results confirm the superiority of ML models over traditional techniques, particularly in estimating duration. The EVM and ES models show significantly lower performance, the best result obtained by these methods for duration estimation is an MAE of 0.12803 and an RMSE of 0.18282, which, performs less well than the best machine learning algorithms analyzed: the lowest MAE is obtained by the SVR (DR) model with a value of 0.09680, while the lowest RMSE is recorded with AdaBoost (DR), which is 0.11396. In the comparison within IR models only, the Huber model stands out as the best performer, obtaining both the lowest MAE (0.11531) and the lowest RMSE (0.15014) within this category.

Regarding the cost prediction, on the other hand, the delta between traditional and ML methods is smaller but still significant, EVM models achieve their best results with RMSE = 0.06451 and MAE = 0.03829 but are outperformed by several ML algorithms.

The lowest MAE is recorded by Huber (DR) with 0.03549, while the best RMSE is obtained by Huber (DR) with a value of 0.06138, making it the most accurate model in absolute terms. Within the IR estimation the lowest MAE is recorded with SGD, which is 0.03564, and the lowest RMSE by HGB with a value of 0.0619.

Table 5-2, Dataset-level models regression metrics values.

Model	Duration estimation				Cost estimation			
	MAE		RMSE		MAE		RMSE	
	DR	IR	DR	IR	DR	IR	DR	IR
EVM(SPI^EVM)	0.23173	0.23173	0.49256	0.49256	-	-	-	-
ES(SPI^ES)	0.25153	0.25153	0.58072	0.58072	-	-	-	-
ES(1)	0.12803	0.12803	0.18282	0.18282	-	-	-	-
EVM(CPI)	-	-	-	-	0.04678	0.04678	0.09598	0.09598
EVM(1)	-	-	-	-	0.04129	0.04129	0.06751	0.06751
OLS	0.10855	0.12760	0.13199	0.17980	0.03758	0.03907	0.06646	0.06408
Ridge	0.10898	0.12774	0.13212	0.17880	0.04051	0.03906	0.06458	0.06639
Lasso	0.10854	0.12760	0.13199	0.17980	0.03757	0.03907	0.06324	0.06646
EN	0.10855	0.12760	0.13199	0.17980	0.03757	0.03907	0.06324	0.06646
Lars	0.10883	0.12760	0.13209	0.17980	0.03926	0.03907	0.06325	0.06646
Lasso Lars	0.10854	0.12760	0.13199	0.17980	0.03810	0.03907	0.06259	0.06646
OMP	0.12494	0.12815	0.16090	0.18094	0.03837	0.03907	0.06378	0.06646
PA	0.11253	0.12883	0.13597	0.17468	0.04309	0.03723	0.06257	0.06433
BR	0.10857	0.12619	0.13200	0.17800	0.03770	0.03898	0.06330	0.06608
ARD	0.10853	0.12619	0.13201	0.17800	0.03781	0.03898	0.06322	0.06608
Tweedie	0.14681	0.12468	0.18190	0.17421	0.05698	0.03863	0.08305	0.06530
SGD	0.13548	0.12770	0.17324	0.17336	0.05572	**0.03564**	0.08679	0.06558
SGD1CSVM	0.18757	0.12803	0.26877	0.18282	0.06355	0.03829	0.09256	0.06451
RANSAC	0.12936	0.13384	0.17550	0.18495	0.03852	0.03581	0.07163	0.06762
Huber	0.10735	**0.11531**	0.13265	**0.15014**	**0.03549**	0.03777	**0.06138**	0.06502
KR	0.10841	0.12863	0.13231	0.17931	0.04167	0.03948	0.06429	0.06747
SVR	**0.09680**	0.13349	0.11770	0.17925	0.04918	0.04157	0.07346	0.06758
NuSVR	0.10361	0.12547	0.12710	0.16973	0.03778	0.03644	0.06502	0.06316
k-NN	0.15435	0.12941	0.19589	0.18341	0.04563	0.03879	0.07471	0.06465
GP	0.11445	0.12739	0.14573	0.18003	0.04253	0.03895	0.07490	0.06596
DT	0.11428	0.14043	0.15604	0.18067	0.04058	0.04096	0.06677	0.07010
ERT	0.10422	0.13926	0.15948	0.18171	0.04275	0.03750	0.07130	0.06297
RF	0.10422	0.13830	0.15948	0.18044	0.04275	0.03768	0.07130	0.06355
ERTs	0.13300	0.12221	0.16931	0.16409	0.04775	0.03877	0.07381	0.06703
AdaBoost	0.09722	0.12780	**0.11396**	0.18057	0.04903	0.04109	0.07685	0.06940
GB	0.10538	0.12660	0.14623	0.16792	0.04405	0.03753	0.07162	0.06481
HGB	0.14184	0.12410	0.18881	0.16683	0.05412	0.03607	0.08199	**0.06191**
XGB	0.14500	0.12500	0.19500	0.17000	0.04520	0.03690	0.06980	0.06210
XGB RF	0.14750	0.12650	0.19850	0.17250	0.04680	0.03810	0.07320	0.06450
MLP	0.13735	0.12182	0.16996	0.16541	0.04751	0.03791	0.07073	0.06397

Analyzing the categories of models in duration prediction, Bayesian models perform best among those with DR approach. For the IR approach, the neural network (represents only by MLP) shows the most promising results. Models based on Stochastic Gradient Descent are found to be the least accurate in both approaches [Table 5-3].

Table 5-3, Dataset-level duration forecasting performance per model category.

Category	MAE DR	MAE IR	RMSE DR	RMSE IR
Bayesian	**0.108549**	0.126191	**0.132004**	0.178003
Ensemble Methods	0.124881	0.127215	0.167327	0.171765
Generalized Linear Model	0.146807	0.124678	0.181904	0.174205
Linear	0.111183	0.127842	0.136128	0.179174
Neural Network	0.137349	**0.121821**	0.169955	**0.165411**
Nonlinear	0.113730	0.132013	0.147750	0.179158
Robust Regression	0.118356	0.124578	0.154076	0.167545
Stochastic Gradient Descent	0.161522	0.127863	0.221003	0.178088

In cost prediction, Robust Regression models are the most accurate in terms of absolute error, achieving the lowest MAE in both the DR and IR approaches. In terms of RMSE, Bayesian models achieve the lowest value with the DR approach, while Neural Networks (represents only by MLP) achieve the best RMSE with IR [Table 5-4].

Table 5-4, Dataset-level cost forecasting performance per model category.

Category	MAE DR	MAE IR	RMSE DR	RMSE IR
Bayesian	0.037758	0.038977	**0.063260**	0.066080
Ensemble Methods	0.047099	0.038019	0.074081	0.064756
Generalized Linear Model	0.056978	0.038632	0.083048	0.065303
Linear	0.039005	0.038842	0.063714	0.065889
Neural Network	0.047513	0.037908	0.070735	**0.063974**
Nonlinear	0.042874	0.039097	0.070065	0.065983
Robust Regression	**0.037008**	**0.036789**	0.066507	0.066322
Stochastic Gradient Descent	0.059632	0.036965	0.089674	0.065043

5.2. Progress stage level

To assess the evolution of the predictive performance of Machine Learning models along the project life cycle, this section analyzes the results obtained at different stages of progress. For this purpose, reference is made to the Percentage Completed (PC) parameter, which represents the level of project completion in percentage terms, from 0% to 100%.

The analysis is carried out considering 19 intermediate snapshots (from 5% to 95% completion, at regular intervals of 5%), for each of which the error values (MAE and RMSE) are calculated for both types of regression: Direct Regression (DR) and Indirect Regression

(IR). The objective is to understand whether and how the accuracy of the models varies with the design stage considered, and which algorithms perform best at each stage.

From the analysis conducted on different stages of project progress (PC), important considerations emerge for both estimating duration and cost at completion. In general, Machine Learning models show significantly higher performance than traditional methodologies in the early and middle stages of the project.

However, the analysis also shows that at advanced stages (more than 85-90% PCs), traditional models revert to a central role, prove effective in prediction, and benefit from the increased accuracy of data collected at the end of the project. Under these conditions, the structural simplicity of traditional models is no longer a limitation, but rather ensures robust results that are comparable to, and in some cases superior to, those obtained with ML techniques.

	Duration estimation								Cost estimation							
	MAE				RMSE				MAE				RMSE			
	DR		IR		DR		IR		DR		IR		DR		IR	
PC	Model	Value	Model	Value	Model	Value	Model	Value	Model	Value	Model	Value	Model	Value	Model	Value
0.05	NuSVR	0.11375	Huber	0.12927	AdaBoost	0.13119	Huber	0.17160	NuSVR	0.05173	NuSVR	0.04716	NuSVR	0.07647	NuSVR	0.07454
0.1	SVR	0.10790	Huber	0.11696	AdaBoost	0.12754	Huber	0.15526	Huber	0.04691	NuSVR	0.04590	NuSVR	0.07329	NuSVR	0.07260
0.15	SVR	0.10352	Huber	0.11655	SVR	0.12367	Huber	0.15067	Huber	0.04661	RANSAC	0.04854	Passive Aggressive	0.07340	Huber	0.07403
0.2	SVR	0.10518	Huber	0.12560	AdaBoost	0.12401	Huber	0.16076	Huber	0.04645	RANSAC	0.04748	Passive Aggressive	0.07434	HGB	0.07717
0.25	AdaBoost	0.10322	Huber	0.12400	AdaBoost	0.12030	Huber	0.15663	GB	0.04700	RANSAC	0.04558	Passive Aggressive	0.07150	HGB	0.07619
0.3	SVR	0.09801	Huber	0.12012	AdaBoost	0.11537	Huber	0.15558	Ridge	0.04506	RANSAC	0.04253	Passive Aggressive	0.06833	HGB	0.07231
0.35	AdaBoost	0.09395	Huber	0.12291	AdaBoost	0.11049	Huber	0.15735	Lasso Lars	0.04433	RANSAC	0.04181	Lasso Lars	0.06687	HGB	0.07007
0.4	ERT	0.09168	Huber	0.12663	AdaBoost	0.11135	Huber	0.15925	Lasso Lars	0.04277	RANSAC	0.04101	Lasso Lars	0.06551	HGB	0.06864
0.45	AdaBoost	0.09140	Huber	0.12902	AdaBoost	0.10617	Huber	0.16127	Lasso Lars	0.04088	RANSAC	0.04016	Lasso Lars	0.06459	HGB	0.06791
0.5	SVR	0.09503	Huber	0.13017	AdaBoost	0.11280	Huber	0.16334	Lasso Lars	0.03879	SGD	0.03782	Huber	0.06433	HGB	0.06753
0.55	ERT	0.09420	Huber	0.13117	AdaBoost	0.11604	Huber	0.16307	Huber	0.03679	SGD	0.03624	Huber	0.06272	NuSVR	0.06555
0.6	ERT	0.09286	Huber	0.13143	SVR	0.11818	Huber	0.16351	Huber	0.03460	SGD	0.03413	Huber	0.06091	NuSVR	0.06082
0.65	AdaBoost	0.09458	Huber	0.12327	AdaBoost	0.11146	Huber	0.15338	RANSAC	0.03101	SGD	0.03035	Huber	0.05600	NuSVR	0.05273
0.7	ERT	0.08424	Huber	0.10843	AdaBoost	0.10180	Huber	0.14185	RANSAC	0.02816	SGD	0.02676	Huber	0.04778	NuSVR	0.04524
0.75	AdaBoost	0.08628	Huber	0.10698	AdaBoost	0.10325	Huber	0.14242	EVM(CPI)	0.02325	NuSVR	0.02284	Huber	0.03927	NuSVR	0.03797
0.8	SVR	0.09042	Bayesian Ridge	0.10149	SVR	0.10710	Huber	0.13597	EVM(CPI)	0.01857	EVM(CPI)	0.01857	Huber	0.03161	NuSVR	0.03250
0.85	SVR	0.08899	Bayesian Ridge	0.09306	SVR	0.10700	Huber	0.12979	EVM(CPI)	0.01448	EVM(CPI)	0.01448	Huber	0.02506	EVM(CPI)	0.02518
0.9	ES(1)	0.08013	k-NN	0.07913	AdaBoost	0.10225	Bayesian Ridge	0.11945	EVM(CPI)	0.01051	EVM(CPI)	0.01051	EVM(CPI)	0.01799	EVM(CPI)	0.01799
0.95	ES(1)	0.05877	ES(1)	0.05877	ES(1)	0.08269	Bayesian Ridge	0.08209	EVM(CPI)	0.00720	EVM(CPI)	0.00720	EVM(CPI)	0.01358	EVM(CPI)	0.01358

5.3. Features Analysis

To identify the features that most influenced the estimation of duration and costs, a SHAP analysis was conducted, aimed at identifying and understanding the contribution of the most relevant variables in the forecasting process.

For the purpose of accurate evaluation, the best-performing models were selected by subdividing them according to the type of prediction (Duration and Cost) and the regression method used (direct or indirect). This subdivision made it possible to compare the impact of features in different predictive approaches. Below, the results of the SHAP analysis are presented and analyzed.

For each selected model, the results of the SHAP analysis are represented through two separate but complementary views:

- SHAP Feature Importance:

This graph shows the average of absolute Shapley values calculated for each feature, the higher the average absolute value, the greater the importance of the feature in determining model predictions.

Features are sorted according to this average, allowing a global view of the variables that most influence the behavior of the model.

- SHAP Summary Plot:

This graph represents both the global importance of features and their local effect on individual predictions. Each point in the plot represents a SHAP value calculated for a feature on a specific instance of the dataset.

Y-axis shows the name of the feature, x-axis shows how much that feature contributed to increase (positive values) or decrease (negative values) the prediction, and the color represents the actual value taken by the feature (blue = low to red = high).

This combination of visualizations allows us to understand not only which variables are most important, but also how and to what extent they affect predictions, both on average and in specific cases.

Duration estimation through DR regression

SHAP analysis has been conducted on the SVR, NuSVR and AdaBoost models, which were selected as best performing for duration estimation by direct approach.

Analysis of the SHAP Feature Importance plots for the three selected models estimating duration by direct regression shows strong consistency among the models in selecting the most influential features. In all three cases, the variables with the highest average SHAP value, and therefore with the greatest impact on forecasts, turn out to be: the Baseline Execution Index (BEI), a dynamic metric introduced in this study; the Project Seriality index (SP), representative of the serial or parallel structure of the project; and the Schedule Variance according to the Earned Schedule (SV^{ES}), one of the established temporal indicators in traditional control techniques.

The SHAP Summary Plot allows us to delve into how these three metrics affect individual forecasts. It can be seen that a low BEI (blue values), indicative of delays relative to the planned budget, tends to increase the duration estimate, while a high BEI (red values), thus a signal of efficiency, leads the model to reduce the planned duration. Similarly, a low SP index, corresponding to structurally more parallel projects, tends to compress the forecast, while high (more serial) values extend it. Regarding Schedule Variance, we see that negative values (lagging behind schedule) push the model toward a longer forecast, while positive values shorten it. This consistent behavior between metric, value, and direction of the impact confirms the models' interpretive ability and alignment with the logic of the data used [Figure 5-1].

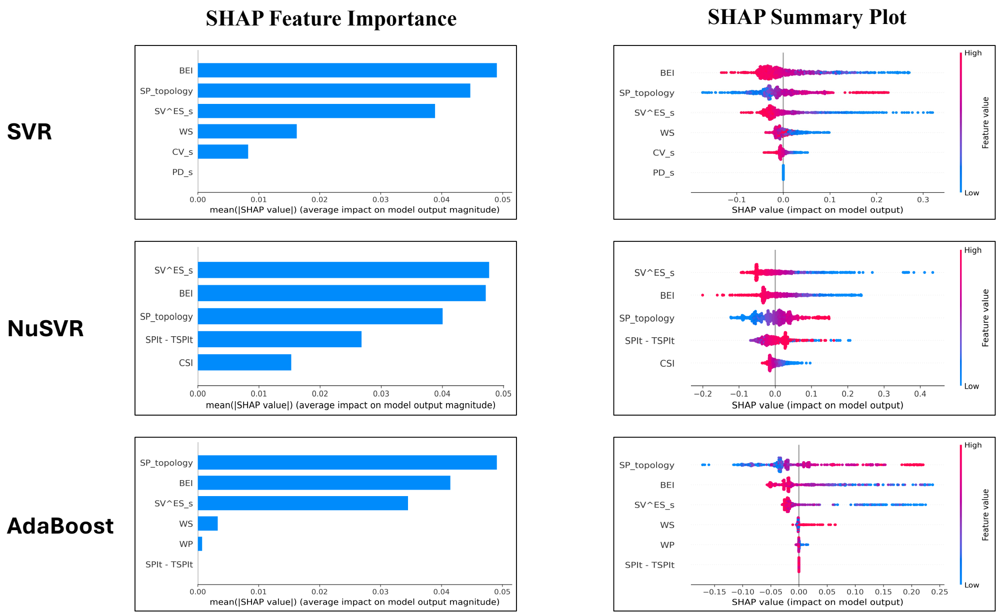

Figure 5-1, SHAP Feature Importance and Summary Plot for duration estimation (Direct Regression)

Duration estimation through IR regression

For the SHAP analysis of duration estimation by indirect regression, the Huber, HGB and ERTs models were selected and found to be among the best performing. As expected, the average SHAP values are generally lower than those of the direct regression models. This behavior is physiological, as indirect models do not directly predict final duration, but estimate an intermediate factor that is then transformed into output through external formulas. The result is less direct weight assignment to features, which results in smaller SHAP values overall.

Nevertheless, some features with significant impacts emerge in the Huber model, notably SP, Schedule Performance Index according to the Earned Schedule (SPI^{ES}) and BEI. Curiously, some of these variables show behavior in contrast to the expected: for example, a high value of SP (which normally indicates a more serial and thus longer structure) is associated with a shorter duration forecast. Similar behavior is also observed for BEI, which in some cases tends to increase the predicted duration when it takes high values, rather than reducing it.

In the HGB and ERTs models, although indices such as SP also appear, the relative SHAP values are very low. This confirms that indirect models, while making use of some relevant variables, tend to distribute their impact more flatly, or concentrate it in a few components that are not easily interpreted. These results, while less rich in terms of interpretation than DR, nevertheless offer interesting insights into the internal dynamics of IR models [Figure 5-2].

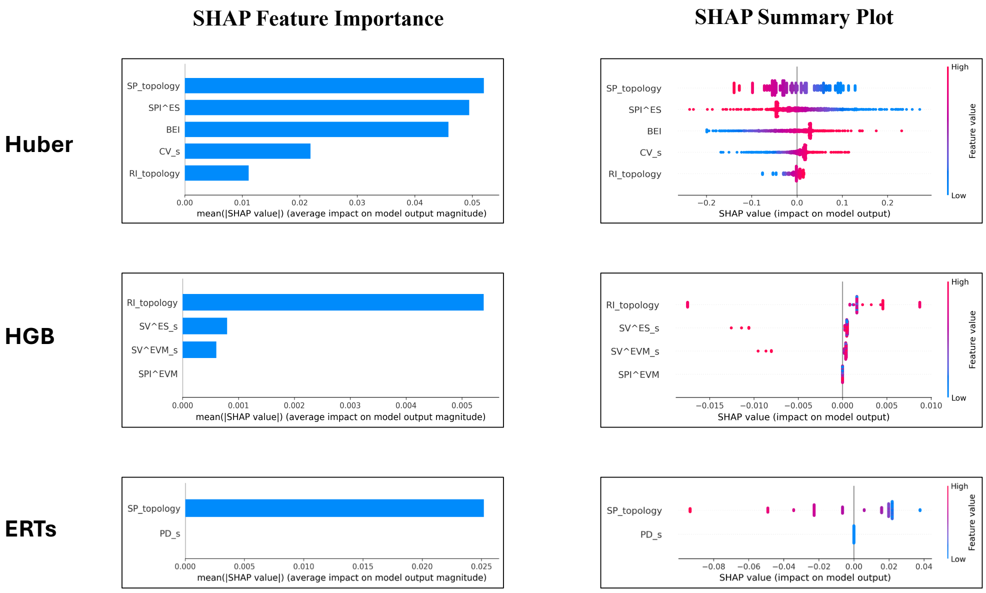

Figure 5-2, SHAP Feature Importance and Summary Plot for duration estimation (Indirect Regression)

Cost estimation through DR regression

For the SHAP analysis of features in cost estimation by direct regression, the Huber, Lasso and OLS models were selected and found to be among the best performing.. In all three, it becomes clear that Cost Variance is the feature with the highest average SHAP impact. This confirms the effectiveness of traditional Earned Value Analysis metrics in the cost estimation as well, consistent with the overall results of the thesis, where cost prediction showed similar performance between ML approaches and traditional techniques.

In the summary plots, Cost Variance shows an expected behavior: negative values (project over budget) push the model to increase the cost estimate at completion, while positive values (under budget) reduce it. Next to Cost Variance, scaled Actual Cost is also relevant for Lasso and OLS: when actual costs are low (so Actual Cost are high), the model tends to overestimate final costs, and vice versa.

Among the variables that emerged in cost estimation with direct regression, the TFCI (Total Float Consumption Index), which is a dynamic metric appeared as the third most influential feature in the Huber model, also deserves attention. This duration-based metric measures the rate of consumption of total available margin (total float) relative to project progress. A TFCI value below 1.00 indicates that the project consumes float faster than expected, signaling the risk of not completing on schedule. Its relevance in the model suggests that the trend in residual time margin also has a direct impact on cost estimates, reflecting the interconnection between time delays and potential extra costs.

Finally, albeit with minor impact, the presence of topological variables such as Project Seriality and Regularity index indicator (RI) is observed. The latter, which measures the temporal regularity of Planned Value: a high value indicates linear and steady growth in Planned Spending, while low values signal an uneven distribution, contributes non-negligibly to predictions in linear models [Figure 5-3].

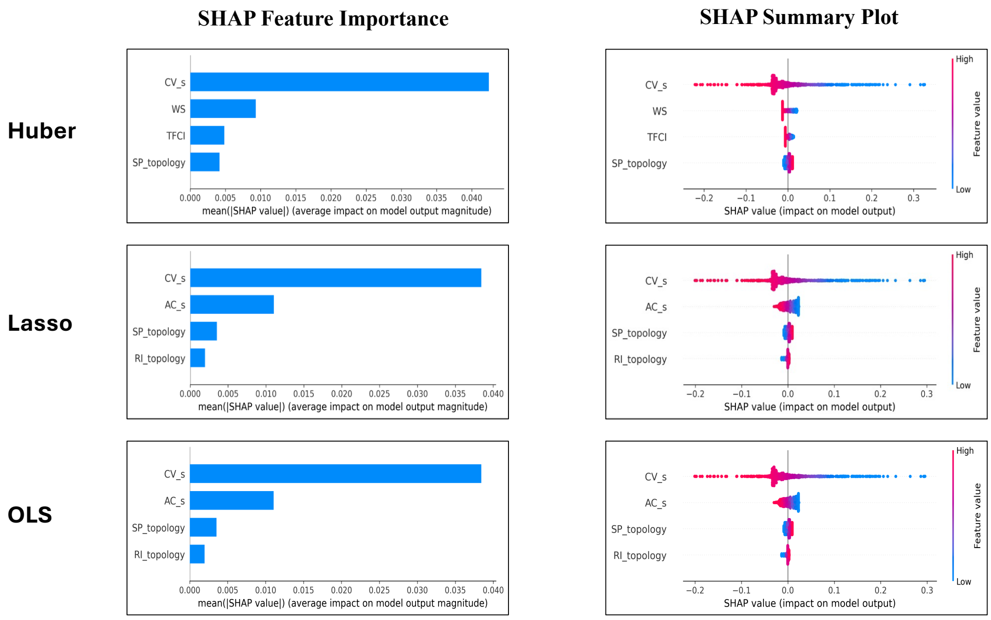

Figure 5-3, SHAP Feature Importance and Summary Plot for cost estimation (Direct Regression)

Cost estimation through IR regression

For cost estimation with indirect regression, SGD, HGB and NuSVR models were analyzed and selected based on their performance. As also observed in the duration, SHAP values are lower on average than in the direct regression. This is consistent with how the indirect approach works, where the model does not directly predict the final cost, but an intermediate parameter from which it is derived. This structure makes it more difficult to assign a direct net weight to individual features.

Nevertheless, two clearly relevant variables emerge in the SGD model: the Cost Performance Index (CPI) and SV^{ES}. The summary plot shows that a low CPI (indicative of cost inefficiency) tends to increase the final cost estimate, while a high value reduces it, consistent with the logic of the metric.

In the HGB model, although with lower SHAP values, dynamic feature is again observed: the comparison of SPI and TSPI [Figure 5-4].

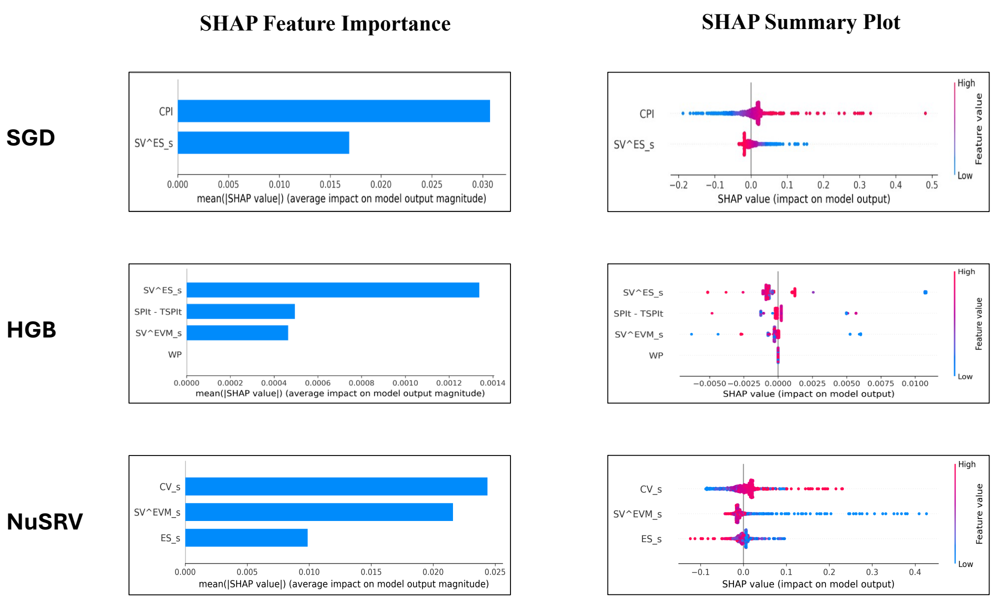

Figure 5-4, SHAP Feature Importance and Summary Plot for cost estimation (Indirect Regression)

6. Discussion

Analysis of the results obtained by applying the machine learning pipeline to a real-world dataset consisting of 90 projects highlighted important insights, both methodologically and interpretively. First, the developed pipeline demonstrated the ability to generate EAC and TEAC models with generally superior performance compared to those provided by traditional methods based on Earned Value Management and Earned Schedule. This advantage manifested itself in terms of average accuracy (MAE), precision (RMSE), and timeliness, especially in the early and middle stages of the project.

Within the context of decision-making in the real world, strategically anticipating time drifts and any out-of-control costs during the execution phase is a vital distinguishing strength. Due to the flexibility and non-linear learning capabilities of their structures, machine learning models are best suited for classifying incomplete, sparse, or noisy data, which is characteristic of early project snapshots. This characteristic makes them particularly useful tools for supporting proactive decisions where the scope for action is still significant.

One of the most distinctive elements of the present research was the integration within the models not only of the traditional EVA metrics or the network topological indicators but also of dynamic data. SHAP analyses showed that dynamic features were found to be decisive in almost all the best models, confirming the importance of considering the updated state of the project in forecasting. At the same time, network static indicators, although with smaller impacts, also showed a significant relationship with temporal estimates, particularly in linear models. This suggests that the structure and temporal regularity of the project network contribute to the accuracy of the estimates, an aspect rarely considered in traditional approaches.

At the methodological level, the use of direct regression provided more interpretable models, with more distributed and consistent SHAP values, while the indirect regression approach, while sometimes competitive in terms of performance, showed less clarity in assigning weight to features, as evidenced by summary plots that were often more “empty” or flat. However, IR remains useful in cases where modularization of prediction through intermediate parameters, such as Performance Factors, is preferred.

Finally, the application of techniques such as data interpolation and balancing, and validation through Leave-One-Group-Out Cross Validation (LOGO CV) prevented overfitting and ensured that each project was fairly represented. The pipeline's fully automatable approach also allows the method to be easily adapted to different datasets, promoting more extensive use even in operational contexts.

7. Conclusion

This study proposed a framework for project monitoring and control through the application of machine learning techniques, with the goal of improving predictive ability over traditional methods. Although approaches such as Earned Value Analysis (EVA) are widely used, they are often limited by the use of static, linear models, which are poorly suited to the complexity of real projects, especially in the more dynamic phases of execution.

In response to these limitations, this thesis proposes an ML pipeline that integrates 30 different learning algorithms, powered by a combination of static features, topological indicators of the project network, and, most importantly, dynamic metrics. It is precisely the systematic introduction of dynamic metrics that is one of the distinctive features of the research: unlike static variables, they evolve as the project progresses, allowing for more timely interception of deviations and critical issues. The SHAP analysis confirmed that these metrics play an important role in forecasting, particularly in the early and intermediate stages.

The results demonstrates that the ML models, through this combination of features, are able to produce more accurate forecasts than conventional models. At the same time, the analysis showed that some traditional metrics remain important, especially in forecasting final costs. In addition, the structure of the project network-described by topological indicators such as SP and RI also contributes to improving the quality of forecasts, allowing ML models to capture deviations from linear progress and the balance between serial and parallel execution.

Methodologically, balancing and interpolation techniques and the adoption of Leave-One-Group-Out cross-validation improved the robustness of the pipeline. Of the two types of regression, direct regression models proved to be more interpretable, while indirect regression models, while showing good performance, generated less readable SHAP analyses, suggesting potential room for improvement.

This research may open several perspectives for future developments. It will be interesting to test the pipeline on larger datasets, plus the use of artificial or simulated data could improve model training, especially in the early stages of projects where real data are often lacking. Another direction of development concerns the enrichment of topological features: in addition to SP and RI, it would be useful to explore variables related to network complexity, dependencies between activities, and resource allocation patterns. In parallel, the integration of indicators related to risk management could further increase the predictive ability of the models.

In conclusion, the developed pipeline proved to be an efficient tool for project control, capable of integrating static, dynamic and topological data. This information makes forecasts more reliable and aims to concretely support project managers in operational decisions and contributing to the evolution of project management practices toward an increasingly data-driven and proactive approach.

8. References

[1] Project Management Institute, A Guide to the Project Management Body of Knowledge (PMBOK Guide), vol. 7th, Newtown Square, PA, USA: PMI, 2021.

[2] E. Favari, Project management per comuni mortali, Esculapio, 2021.

[3] M. S. Dobson, The triple constraints in project management, Vienna: Management Concepts, 2004.

[4] P. O. Gaddis, "The Project Manager," *Harvard Business Review,* 1959.

[5] B. A. T. &. F. M. Wilson, Earned Value Management Systems: Challenges and Future Direction., Journal of Integrated Enterprise Systems, 2013.

[6] F. Webster, "The WBS," *PM Network,* vol. 8(12), pp. 40-46.

[7] D. F. Cioffi, "Work and Resource Breakdown Structures for Formalized Bottom-Up Estimating," *Cost Engineering,* 2004.

[8] K. E. I. S. A. Sakin, "A Literature-based Visualization Task Taxonomy for Gantt Charts," 2024.

[9] J. R. S. Cristòbal, "The S-curve envelope as a tool for monitoring and control of projects," in *CENTERIS / ProjMAN / HCist*, Barcelona, Spain, 2017.

[10] A. D. M. T. Narbaev, "A framework model for risk adjusted cost forecasting. Earned Value and Cost Contingency Management," *The journal of modern project management,* vol. https://doi.org/10.19255/JMPM01202, 2017.

[11] M. C. a. C. Chien, "Estimating project S-curves based on polynomial function and neural networks," in *Proceedings of the 2010 IEEE International Conference on Industrial Engineering and Engineering Management*, 2010.

[12] D. Cioffi, "Designing project management: A scientific notation and an improved formalism," *The Journal of the Operational Research Society,* vol. 58, 2005.

[13] Project Management Institute, A Guide to the Project Management Body of Knowledge (PMBOK® Guide), vol. 6th, Newtown Square, PA, USA: PMI, 2017.

[14] J. A. Figueiredo, "PMBOK 7th Edition, in the Good Direction, but How Effective?," Industrial Engineering, 2022.

[15] A. D. Marco, Project Management for Facility Constructions, Turin: Springer, 2018.

[16] Project Management Institute, A Guide to the Project Management Body of Knowledge (PMBOK® Guide), vol. 1st, Newtown Square, PA, USA: PMI, 1996.

[17] Project Management Institute, A Guide to the Project Management Body of Knowledge (PMBOK® Guide), vol. 2nd, Newtown Square, PA, USA: PMI, 2000.

[18] P. M. Institute, Practice Standard for Earned Value Management, Newtown Square, PA, USA: PMI, 2005.

[19] L. W., Schedule is different, Summer: The Measurable News, 2003.

[20] R. Corovic, "Why EVM is not good for schedule performance analyses," Project Management Institute, 2005.

[21] National Defense Industrial Association (NDIA), Integrated Program Management Division (IPMD), "Planning & Scheduling Excellence Guide (PASEG)," Arlington, VA, USA, 2022.

[22] S. Vandevoorde and M. Vanhoucke, "A comparison of different project duration forecasting methods using earned value metrics," Ghent university, 2005.

[23] D. P. a. A. Rak, "Monitoring of Time and Cost Variances of Schedule Using Simple Earned Value Method Indicators," *Applied Sciences,* vol. DOI: 10.3390/app11041357, 2021.

[24] M. C. J. D. D. M. B. &. T. L. Vanhoucke, "An evaluation of the adequacy of project network generators with systematically sampled networks," *European Journal of Operational Research,* 2008.

[25] M. V. J. Batselier, "Project regularity: development and evaluation of a new project characteristic," no. DOI: 10.1007/s11518-016-5312-6, 2017.

[26] M. K. D. Jacob, "Forecasting schedule completion using earned value metrics revisited," *The measurable News,* 2004.

[27] J. C. D. D. B. M. a. L. V. T. M. Vanhoucke, "An evaluation of the adequacy of project network generators with systematically sampled networks," *European Journal of Operational Research,* vol. 187, 2008.

[28] S. Elmaghraby, "Activity Networks: Project Planning and Control by Network Models," *John Wiley and Sons Inc,* 1977.

[29] I. H. Sarker, "Machine Learning: Algorithms, Real-World Applications and Research Directions," *SN Computer Science,* vol. 2, pp. doi: 10.1007/s42979-021-00592-x, 2021.

[30] M. P. V. A. a. J. M. G. J. I. Santos, "Explainable machine learning for project management control," *Computers & Industrial Engineering,* vol. 180, no. doi: 10.1016/j.cie.2023.109261, 2023.

[31] T. N. a. Ö. H. T. İnan, "A Machine Learning Study to Enhance Project Cost Forecasting," *IFAC-PapersOnLine,* vol. 55, no. doi: 10.1016/j.ifacol.2022.10.127, 2022.

[32] F. A. S. A. A. A. ForouzeshNejad, "Optimizing Project Time and Cost Prediction Using a Hybrid XGBoost and Simulated Annealing Algorithm," *Machines,* vol. 12, p. doi: 10.3390/machines12120867, 2024.

[33] S. B. H. Ç. G. Yalçın, "Evaluation of Earned Value Management-Based Cost Estimation via Machine Learning," *Buildings,* vol. 14, p. doi: 10.3390/buildings14123772, 2024.

[34] F. M. O. a. A. D. Marco, "Multiple Linear Regression Model for Improved Project Cost Forecasting," *Procedia Computer Science,* vol. 196, no. doi: 10.1016/j.procs.2021.12.079, 2022.

[35] Operations Research & Scheduling Research Group, "Real Project Data for Scheduling, Risk Analysis and Control," 2024. [Online]. Available: https://www.projectmanagement.ugent.be/research/data/realdata. [Accessed 2025].

[36] D. Berrar, "Cross-validation," 2019.

[37] Z. Li, "Extracting spatial effects from machine learning model using local interpretation method: An example of SHAP and XGBoost," *Computers, Environment and Urban Systems,* vol. doi: 10.1016/j.compenvurbsys.2022.101845, 2022.

[38] S. M. L. a. S.-I. Lee, "A Unified Approach to Interpreting Model Predictions," *Advances in Neural Information Processing Systems 30,* no. doi:10.48550/arXiv.1705.07874, 2017.

[39] M. Akhbari, "Project time and cost forecasting using Monte Carlo simulation and artificial neural networks.," International Journal of Industrial Engineering & Production Research, 2018.

[40] A. T. A. &. A. A. C. Cruz, " Traditional, agile and lean project management: A systematic literature review.," *Journal of Modern Project Management,* 2020.