**DataCronyx: AutoEDA & AutoTrainer**

**Project Report submitted in the partial fulfilment**

**Of**

Master of Technology

In

Artificial Intelligence

by

**Shardul Gore (R016)**

Under the supervision of

**Name of Faculty Mentor**

(Designation, Name of the department, MPSTME)

# SVKM’s NMIMS University

(Deemed-to-be University)



**MUKESH PATEL SCHOOL OF TECHNOLOGY MANAGEMENT & ENGINEERING (MPSTME)**

**Vile Parle (W), Mumbai-56**

**(2025-26)**

# CERTIFICATE

****

This is to certify that the project entitled DataCronyx: AutoEDA & AutoTrainer, has been done by **Mr. Shardul Gore** under my guidance and supervision & has been submitted in partial fulfilment of the degree of Master of Technology in Artificial Intelligence of MPSTME, SVKM’s NMIMS (Deemed-to-be University), Mumbai, India.

Project mentor (name and Signature) Examiner ( name and Signature) (Intenal Guide)

**Date**

**Place: Mumbai**

**(HoD)** (name and Signature)

# ACKNOWLEDGEMENT

Student Names

|  |  |  |
| --- | --- | --- |
| **NAME** | **ROLL NO.** | **SAP ID** |
| Shardul Gore | R016 | 70282400003 |
|  |  |  |
|  |  |  |
|  |  |  |

# ABSTRACT

This project presents DataCronyx, an intelligent, automated system designed to simplify and enhance the process of exploratory data analysis (EDA) and machine learning model development for diverse tabular datasets. The work addresses the common challenges faced by analysts and decision-makers in deriving insights and building predictive models efficiently, especially when technical expertise is limited.

DataCronyx offers a dual-mode functionality that significantly contributes to data analytics workflows. The first mode empowers users with a custom, user-friendly interface where they can manually guide the data cleaning, feature selection, and model training processes. This interactive environment enables fine control and customization suited to specific business needs. The second mode leverages AI agents to autonomously perform end-to-end data analysis and model training, reducing human intervention and accelerating the time from raw data to meaningful insights.

A key contribution of this work is the integration of automated EDA with high-quality visualizations and comprehensive data quality assessments, allowing for rapid and clear understanding of data patterns and anomalies. The system utilizes well-established machine learning libraries to support multiple modeling algorithms, providing robust model evaluation and comparison tools. Another novel aspect is the incorporation of large language model (LLM) technology—specifically Google Gemini API—to translate statistical findings and model outcomes into coherent, actionable narratives. This bridges the gap between complex analytics and accessible interpretation, enhancing decision-making for both technical and non-technical stakeholders.

Through the development and testing of DataCronyx, this project demonstrates how combining customizable workflows with intelligent automation can improve the efficiency, accuracy, and interpretability of data analysis pipelines. The system’s adaptability to various data sources, including file uploads and direct SQL database connections, further extends its practical applicability. Overall, DataCronyx advances the state of automated data science by delivering a scalable, explainable, and user-centric platform that empowers organizations to derive deeper insights and build predictive capabilities without extensive coding or domain expertise.

**Table of Contents**

**Topics Page**

**List of Figures** **i**

**List of Tables** **xvi**

**Abbreviations** **xix**

**Chapter 1 Introduction**

**1.1 Background of the project**

**1.2 Motivation and Scope**

**1.3 Problem Statement**

**1.4 Salient Contribution**

**Chapter 2 Literature survey 2.1 Introduction**

**2.2 Exhaustive Literature survey**

**Chapter 3 Methodology and Implementation**

**3.1 Methodology**

**3.2 Implementation**

**3.2.1 Project Structure & Module Functions**

**3.2.2 Workflow Example**

**3.2.3 Gantt Chart Alignment**

**3.2.4 Architecture and Extensibility**

**Chapter 4 Result and Analysis**

**4.1**

**4.2**

**Chapter 5 Advantages, Limitations and Applications**

* 1. **Advantages**
  2. **Limitations**
  3. **Applications**

**Conclusion and Future Scope**

**References**

**Appendix A: Soft Code Flowcharts Appendix B: Data Sheets Appendix C: List of Components**

**Appendix D: List of Paper Presented and Published.**

List of the figures

|  |  |  |
| --- | --- | --- |
| **Fig No** | **Name of the figure** | **Page No** |
| 1 | Architecture Diagram |  |
| 2 | Gantt Chart |  |

List of tables

|  |  |  |
| --- | --- | --- |
| **Table No** | **Name of the Table** | **Page No** |
|  |  |  |
|  |  |  |

Abbreviations

|  |  |  |  |
| --- | --- | --- | --- |
| **Table No** |  | **Name of the Table** | **Page No** |
|  |  |  |  |
|  |  |  |  |

**Note: Each table should be included on separate page.**

**Chapter 1 Introduction**

* 1. **Background of the project topic**

In the digital age, organizations and individuals generate and collect vast volumes of data from a variety of sources, including business transactions, sensors, social media, and online platforms. The effective use of this data through analytics and predictive modeling has become central to innovation, competitiveness, and informed decision-making across industries. Central to this process are two foundational practices: Exploratory Data Analysis (EDA) and machine learning (ML). EDA serves as the bridge between raw data and valuable insights, enabling users to investigate the structure, patterns, anomalies, and relationships within datasets. Once the quality and key features of the data are understood, machine learning techniques further unlock its predictive potential, informing forecasts, classifications, optimizations, and recommendations.

Despite remarkable advances in the field, the practical adoption of data analytics remains hindered for many users. Typical analytics workflows are fragmented, involving data cleaning, visualization, transformation, and modeling steps, each often requiring different tools and programming skills. Furthermore, there is a steep learning curve associated with understanding statistical outputs and the implications of model performance. This creates barriers for non-experts while leaving data scientists encumbered by repetitive, manual tasks that detract from higher-value creative work. The vision to lower these barriers and create a more seamless, interpretable, and efficient path from raw data to decision-ready insights is what motivates this project.

* 1. **Motivation and scope**

Motivated by the need to democratize access to advanced analytics, the present work focuses on developing DataCronyx, a comprehensive and intelligent data analytics platform. The motivation is twofold: First, to enable users—regardless of their technical background—to conduct thorough and insightful EDA coupled with robust machine learning, and second, to automate and accelerate these workflows using the latest advancements in artificial intelligence (AI) agents.

The scope of this report includes the design, implementation, and evaluation of a platform that can handle tabular data from a wide array of sources, including file uploads (CSV, Excel, JSON) and direct SQL databases. DataCronyx is envisioned as an easy-to-use web interface, incorporating both a traditional, interactive mode—where users retain full control over parameter selection—and an automated mode, where AI agents seamlessly process and analyze data with minimal input. This dual approach ensures that both novice and expert users can benefit from the platform, choosing either a guided experience or leveraging built-in automation for efficiency.

* 1. **Problem statement**

The field of data analytics continually faces several persistent challenges:

* Analysts often spend a disproportionate amount of time on routine data wrangling, quality checks, and plotting, rather than on generating actionable insights.
* Manual feature selection, preprocessing, and model evaluation are both error-prone and time-intensive, particularly for large or complex datasets.
* While automated solutions exist, many focus exclusively on either EDA or model automation, lacking an integrated, holistic approach or flexibility for user intervention.
* The presentation of findings is frequently in technical or statistical format, alienating business users or stakeholders who may lack domain expertise but are responsible for making key decisions.
* There is a growing demand for systems that combine robust automation with explainability, flexibility, and accessibility, narrowing the gap between data and impactful action.
  1. **Salient contribution**

This project brings forth several innovative contributions to the domain of automated analytics and AI-assisted decision support:

* A dual-mode data analysis workflow: DataCronyx allows users to select between a fully custom, user-driven EDA/ML flow, and an AI agent-driven automatic pipeline that manages data exploration, feature engineering, and model training end-to-end.
* Flexible data ingestion and preparation: The system facilitates both conventional file uploads and live connections to SQL databases, broadening its applicability and ensuring compatibility with real-world business environments.
* Comprehensive, automated EDA and reporting: With automated generation of descriptive statistics, high-resolution visualizations, and data quality diagnostics, users can rapidly assess and understand their datasets regardless of scale or complexity.
* Integrated machine learning with user choice: Multiple models are available, with side-by-side evaluation and leaderboard presentation; the best models can be exported for deployment or further analysis.
* Natural language, AI-powered explanation: By leveraging advanced large language model (LLM) technology (Google Gemini API), the platform summarizes results and model findings in clear, accessible language, closing the gap between technical analytics and business application.
* Accessible user interface: Intuitive design principles ensure DataCronyx is approachable for non-technical users, while also offering depth and customization for power users and data professionals.

**Chapter 2 Literature survey**

* 1. **Introduction to overall topic**

Exploratory Data Analysis (EDA) stands as a cornerstone of contemporary data science, acting as the critical bridge between raw datasets and actionable intelligence. Historically rooted in the work of John Tukey, EDA emphasizes understanding data distributions, patterns, relationships, and anomalies through both statistical summaries and visualizations, paving the way for appropriate modeling and robust, data-driven reasoning. As modern datasets have grown in complexity, size, and heterogeneity—spanning structured and unstructured sources across industries—the need for scalable, reproducible, and adaptable EDA methodologies has become acute. Manual, traditional approaches, though effective for preliminary investigation, often falter in the face of high-dimensional data, latent dependencies, and domain-specific nuances. This situation is further complicated by the pressure for rapid business intelligence, stringent regulatory requirements, and the need for transparent communication of findings to both technical and non-technical stakeholders. These factors have collectively inspired a wave of research into automating EDA, transforming it from reactive, manual examination to proactive, AI-driven pipelines that blend the best of human intuition and machine efficiency. The integration of AI, machine learning (ML), and large language models (LLMs) is revolutionizing how data is explored, interpreted, and acted upon—enabling deeper insights, greater accessibility, and robust handling of real-world data challenges [2][1][4].

* 1. **Exhaustive literature survey**

The literature exhibits rapid progress and diverse approaches in the area of automated EDA. Gaikwad et al. [2] provide a comprehensive overview of the evolution from traditional EDA to intelligent, automated frameworks. Their review underscores that while statistical analysis and human-driven visual exploration remain valuable, AI-based EDA delivers notable advantages: from automating tedious preprocessing, feature extraction, and pattern detection to supporting intuitive natural language interfaces. Their work highlights how AI-driven automation ensures scalability for large datasets, improves accessibility for users without deep technical expertise, and allows for more interactive, real-time discovery. Importantly, they also delineate the persistent challenges, such as managing data privacy, reducing algorithmic bias, and explaining complex model decisions to foster trust in the insights uncovered by AI agents.

Addressing the growing need for cross-domain adaptability and natural language understanding, Zhu et al. [1] introduce TiInsight, an advanced EDA system harnessing the power of LLMs. TiInsight’s architecture fuses hierarchical data context summarization, intent clarification, and dynamic visualization recommendation within a unified platform. Their system is distinguished by its ability to process complex, multi-table enterprise databases using text-to-SQL translation guided by user queries in everyday language. The authors' evaluations show that TiInsight outperforms existing tools by automatically clarifying vague or ambiguous user intent, generalizing analysis steps across domains without domain-specific fine-tuning, and effectively automating the creation of informative visualizations. Despite these strengths, Zhu et al. identify unresolved technical hurdles. These include handling especially ambiguous queries, dealing with the complexity of database schemas in real-world scenarios, and providing transparent model reasoning throughout the analytical pipeline, particularly in regulated or high-risk domains.

In pursuit of higher personalization and iterative insight discovery, Wu et al. [3] introduce AutoEDA-Segment, which applies Attribute Frequency Statistical Feature Ratio (AFSFR) for enhanced feature categorization and value assessment, supported by interactive clustering and visualization. Their iterative feedback approach allows analysts to refine focus areas, incorporate domain expertise, and iteratively improve the quality of features and clusters explored—demonstrating improved efficiency and user satisfaction in real-world case studies from meteorology and healthcare. Critically, their work elucidates the opportunities and practical challenges of bridging automated and user-guided workflows so that domain experts are able to inject contextual insights into the automated EDA process. Wu et al. also note limitations regarding the breadth of visualization options and the need for smarter, context-aware clustering—requirements that grow in importance as datasets scale and diversify.

Complementing these software-oriented advancements, Shi et al. [4] highlight the foundational importance of comprehensive benchmarking datasets by introducing ForgeEDA. This multimodal, open-source resource contains a diverse set of integrated circuit (IC) designs—including code, netlists, graph representations, and layouts—serving as a testbed for AI4EDA solutions in logic synthesis, placement, performance estimation, and more. Their experimental results unveil the performance disparities between open-source and commercial tools, advocating for open, large-scale datasets as necessary infrastructure for accelerating AI-driven EDA innovation. Beyond hardware, the philosophy of ForgeEDA points toward a broader gap: the need for domain-specific, yet flexible benchmarking datasets in business analytics, healthcare, and other data-rich fields; this would trustworthily evaluate both the generalizability and practical intelligence of automated EDA systems.

Pulling together the evidence from these works, clear patterns emerge about the current state and gaps in the field. Existing solutions either prioritize autonomous, fully automated analytics or user-driven customization, but rarely offer a cohesive, dual-mode experience that fluidly blends both to suit changing user needs or organization maturity [1][3]. Most frameworks are still limited by the format and source of the input data they can handle, often requiring manual adaptation or extensive retraining to generalize across domains—be it classic tabular business intelligence, unstructured text, or specialized formats like IC design layouts [4][2]. In practice, stakeholders demand not just fast and accurate results, but robust explanation, transparency, and adaptability in the face of ambiguous queries, changing objectives, or regulatory constraints [1][2]. Last, and importantly, the lack of comprehensive, cross-domain benchmarks curtails rigorous comparison, systematic improvement, and, ultimately, the real-world deployment of scalable, trustworthy automated EDA platforms [4].

Addressing these acute gaps, this project proposes to develop DataCronyx: a unified, extensible EDA platform for the modern era. It aims to combine interactive user-guided exploration with rich, AI-driven automation and advanced visualization, covering a spectrum of business and specialized (e.g., hardware) data. The ambition is to bridge technical and domain barriers, democratize data exploration with context-aware explainable analytics, and provide robust, scalable benchmarking to drive continuous improvement and innovation in EDA for all users.

**Chapter 3 Methodology and Implementation**

* 1. **Methodology**

The methodology behind DataCronyx centers on delivering a modular, scalable, and user-friendly platform for automated exploratory data analysis (EDA) and machine learning (ML) model training. The system is structured to solve real-world data analytics challenges by offering both automation for rapid insight discovery and interactive controls for expert fine-tuning. As illustrated in the architecture diagram, DataCronyx supports two main workflows—Agentic Workflow and Custom Workflow—which converge through a unified data ingestion process and culminate in a Streamlit-based web interface, ensuring flexibility and accessibility for users of any skill level.

A diagram of a process

AI-generated content may be incorrect.

Fig. 1 Architecture diagram

Data Ingestion serves as the foundation for both workflows. Users can upload CSV/XLS files or connect to SQL data sources; the system automatically parses schema information, validates formats, and prepares data for further steps. This universal entry point ensures that both custom and automated EDA can operate on diverse data sources.

In the Agentic Workflow, intelligent agents sequentially execute core stages of the data science pipeline. These agents are designed as modular components (as implemented in the agent\_module/ and agent.py files in the repo) that perform:

* EDA Agent: Automated statistical analysis, summary generation, and detection of missing values or anomalies.
* Preprocessing Agent: Selection and application of cleaning, imputation, encoding, and scaling techniques using heuristics or learned strategies.
* Feature Engineering Agent: Identification and transformation of optimal features (e.g., via PCA, selection algorithms) with minimal user intervention.
* Model Training Agent: Automated selection, training, and hyperparameter tuning of appropriate ML models for classification or regression.
* Model Evaluation Agent: Performance measurement using metrics such as accuracy, F1-score, RMSE, or R^2, and automatic leaderboard creation.

Each agent incorporates decision logic and, where appropriate, LLM or rule-based engines to tailor their actions to the dataset and desired output, reducing manual steps and speeding up end-to-end analysis.

In parallel, the Custom Workflow provides an interactive channel for users to manually guide each step:

* EDA & Visualization: Users select variables, filter data, and create custom charts using functions found in data\_analysis\_functions.py. Visualizations include summary tables, distributions, scatterplots, and correlation heatmaps.
* Preprocessing: Users decide on imputation strategies, encoding types, scaling, and outlier correction in data\_preprocessing\_function.py, with real-time feedback on changes.
* Feature Engineering: Manual execution of PCA, k-best selection, and other techniques through feature\_engineering.py, allowing selection based on domain knowledge or project goals.
* Model Training: Users pick from a suite of models (Logistic Regression, Random Forest, SVM, Decision Tree, Gradient Boosting, Ridge, Lasso, etc.), set parameters, and view model outputs. All logic is handled in model\_training.py and surfaced in the UI.
* Model Evaluation: Users analyze performance metrics, view confusion matrices or regression plots, and compare models head-to-head.

Both workflows are accessible from the Streamlit sidebar, allowing users to switch between full automation (agentic) and granular manual (custom) control. The outputs—charts, reports, model files—are downloadable, ensuring the system's utility for both quick experiments and deep analyses.

* 1. **Implementation**

The implementation of DataCronyx employs Python (>=3.8) and relies on well-known packages such as pandas, scikit-learn, seaborn, plotly, and Streamlit for UI development.

**3.2.1 Project Structure & Module Functions:**

* main.py: Orchestrates workflow logic, page routing, and sidebar navigation. Integrates both manual and autonomous workflows, and manages state across the session.
* home\_page.py: Supplies the landing dashboard and initial dataset selection interface. Handles onboarding for new users.
* data\_analysis\_functions.py: Implements summary statistics, advanced graphical EDA, and interactive charting.
* data\_preprocessing\_function.py: Contains all routines for missing value imputation, categorical encoding, feature scaling, and outlier checks. Supports both batch and interactive operations.
* feature\_engineering.py: Provides PCA, selection algorithms, and transformation utilities, supporting manual selection and agent-driven execution.
* model\_training.py: Encapsulates all model training logic (fit, predict, evaluate, compare) for both classification and regression tasks. Creates downloadable model files post-training.
* agent.py / agent\_module/: Implements rule-based agents, prototype workflows, and integration points for external agent frameworks (like CrewAI/LangChain), permitting scalable automation and experimentation.
* example\_dataset/: Sample CSV files (e.g., Titanic for classification, Insurance for regression).
* logs/: Application and experiment logs for debugging and reproducibility.

**Workflow Example:**

1. User uploads data or chooses a sample.
2. Data ingestion module parses and validates the dataset.
3. User selects workflow:
   * + In Agentic, agents autonomously handle EDA, preprocessing, feature engineering, model selection, and evaluation.
     + In Custom, the user interacts with each module, customizing choices and visualizations at every step.
4. Final results—model leaderboard, downloadable models, visualization charts, and EDA reports—are presented in the UI.

**Gantt Chart Alignment:**

The development followed the Gantt schedule referring to the figure:

* Setup: Initializing environment, preparing dataset and module layout.
* Sampling: Building ingestion and preprocessing modules, ensuring robust schema support.
* Core Development: Implementing visualization, feature engineering, training, and agentic logic modules week-wise.
* Integration: Unifying workflows in Streamlit UI, refining agentic logic.
* Testing: Using provided datasets, validating custom and automated paths.
* Documentation & Deployment: Finalizing notebooks, README, and help guides for user onboarding.

A graph with red and blue squares

AI-generated content may be incorrect.

Fig. 2 Gantt Chart

**Architecture and Extensibility:**

The architecture supports plug-and-play agent modules, making it easy to extend automation with newer frameworks or additional model types (see agent.py and agent\_module/ for example CrewAI and LangChain integrations). The visualization and preprocessing modules are generic, handling a wide range of real-world tabular data. Model training supports parameter tuning, leaderboard generation, and output export in standardized formats.

With both controlled and automated pathways, DataCronyx demonstrates how a modern EDA and AutoML system can leverage modular code, automation, and intuitive UI design to empower users at any expertise level—from rapid prototyping to thorough research—while remaining extensible for future improvements and integration into larger machine learning pipelines. This methodology positions DataCronyx as a practical, research-grade solution to the growing challenges and opportunities in data-driven decision-making.

**Chapter 4**

* 1. **Block diagram**
  2. **Software description, flowchart / algorithm**

This chapter can comprise of actual implementation photos and their description.

**Chapter 4 Results and Analysis**

This shall include a thorough evaluation and investigation carried out. It should also bring out your contributions from the study. The discussion shall logically lead to inferences and conclusions as well as scope for possible further future work.

**Note:**

**Include the IEEE or any other standards that you have adhered to test the validity of the results.**

**Link for IEEE standards**

https://[www.ieee.org/content/ieee-org/en/standards/index.html/](http://www.ieee.org/content/ieee-org/en/standards/index.html/)

**Chapter 5**

**Advantages, Limitations and Applications**

**Chapter 6 Conclusion and Future Scope**

* A brief report of the work carried out, conclusions derived from logical analysis presented in the Results and Discussions chapter.
* Scope for future work should be stated lucidly in this chapter.

**References**

**[1] Zhu, J. P., Niu, B., Cai, P., Ni, Z., Wan, J., Xu, K., ... & Liu, Q. (2024). Towards Automated Cross-domain Exploratory Data Analysis through Large Language Models. arXiv preprint arXiv:2412.07214.**

**[2] Prof. Jyoti Gaikwad, Aniket Manohare, Shweta Munde, Anwar Shaikh, and Diksha Subhedar, “AI-Based Exploratory Data Analysis”, Int. J. Sci. Res. Comput. Sci. Eng. Inf. Technol, vol. 11, no. 2, pp. 3876–3884, Apr. 2025, doi: 10.32628/CSEIT25112860.**

**[3] Wu, T., Wang, S., & Peng, X. (2024, October). AutoEDA: Iterative Data Focusing and Exploratory Analysis Based on Attribute Frequency. In 2024 IEEE International Conference on Systems, Man, and Cybernetics (SMC) (pp. 4113-4118). IEEE.**

**[4] Z. Shi et al., "ForgeEDA: A Comprehensive Multimodal Dataset for Advancing EDA," 2025 International Symposium of Electronics Design Automation (ISEDA), Hong Kong, China, 2025, pp. 778-783, doi: 10.1109/ISEDA65950.2025.11101194.**

**Appendix A: Soft Code Flowcharts**

**Appendix B: Data Sheets**

**Appendix C: List of Components**

**Appendix D: List of Paper Presented and Published**

* List of papers on the topic of the report published by the candidates.
* This may also be included in the contents.
* The candidates may also include reprints of his/her publications after the literature citation.