FASTMAN-JMP: All-in-one Tool for Data Mining and Model Building

Kyle Territo, \*Jose Romagnoli

Louisiana State University, Baton Rouge, United States

kterri3@lsu.edu, \*jose@lsu.edu

Abstract

In the contemporary realm of big data, the adept exploration of vast multivariate datasets is indispensable. Addressing this imperative, FASTMAN-JMP emerges as an innovative add-in for SAS's JMP software, seamlessly amalgamating a diverse range of machine learning algorithms into a unified and user-friendly tool. This integration serves as a catalyst for extracting profound insights from intricate data sets, insights that could easily elude discovery otherwise. Boasting an intuitively designed graphical user interface, FASTMAN-JMP leads users through a methodically structured process of knowledge discovery, finely tailored to their specific use cases.

**Keywords**: data-mining, dimensionality reduction, software, data exploration

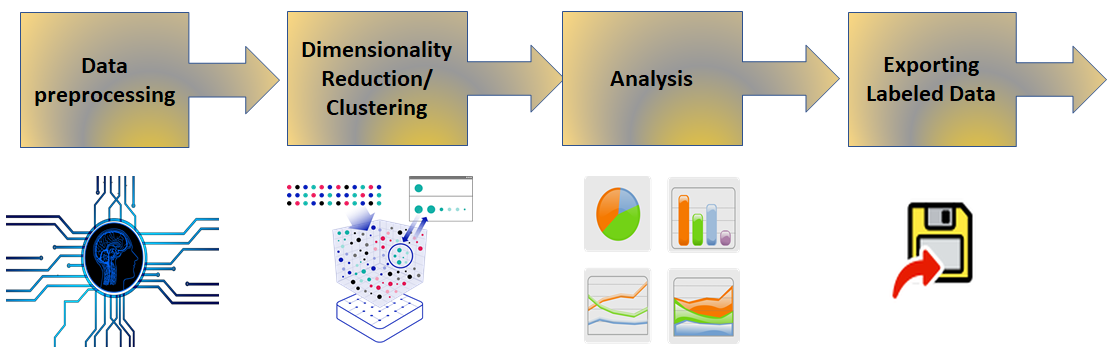
* 1. Introduction

Data is often regarded as the most important asset in the modern world because it holds the key to understanding complex patterns, behaviors, and trends, enabling informed decision-making in almost every aspect of business, science, technology, and even societal development. Due to the advancements brought about by Industry 4.0, such as the integration of the Internet of Things, artificial intelligence, and automation, data is now being generated at exponentially higher rates than ever before. The need for better ways to analyze this burgeoning big data is crucial for several reasons. Firstly, the sheer volume of data exceeds the capacity of conventional databases and analytical tools, necessitating more sophisticated and scalable solutions. Secondly, the complexity and diversity of this data require advanced analytics techniques, like machine learning and AI, to extract meaningful insights. These insights are vital for driving efficiencies, innovation, and competitive advantage in an increasingly data-driven world. Moreover, efficient analysis of big data enables predictive maintenance, enhanced customer experiences, and more informed decision-making, all of which are key components in realizing the full potential of Industry 4.0. This work introduces an environment designed to streamline the application of pattern recognition techniques to historical datasets. FASTMAN-JMP, an integrated tool for data exploration and fault detection, seamlessly merges the statistical data analysis software JMP [1] with the FASTMAN Python environment [2]. Packaged as an add-in utilizing JMP Scripting Language (JSL), FASTMAN-JMP eliminates the need for users to possess an in-depth understanding of the intricate implementation of each offered method. This includes tasks such as data cleaning, normalization, sampling, dimensionality reduction, and data clustering. Serving as the offline, model-building component in the process monitoring workflow, this tool conducts clustering on a historical database to train a model. This model can then be deployed for real-time process monitoring. All data used in this paper will be sourced from the Tennessee Eastman Process (TEP), a virtual plant that produces time series data often used for benchmark tests for process control and fault detection applications [3].

* 1. Data Mining & Knowledge Discovery

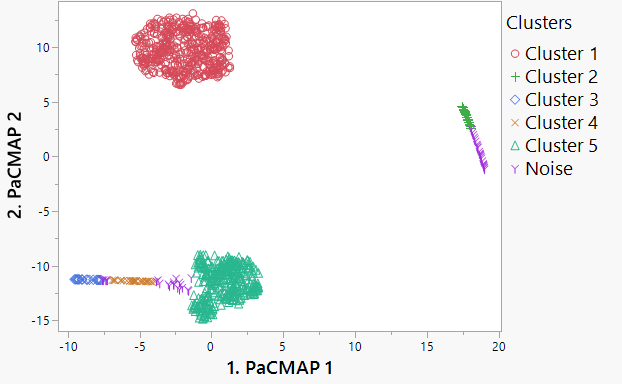
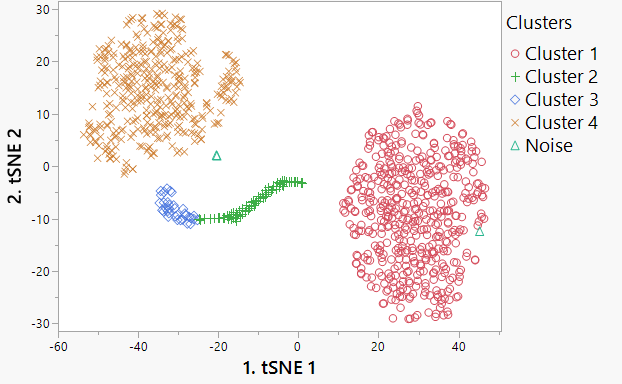
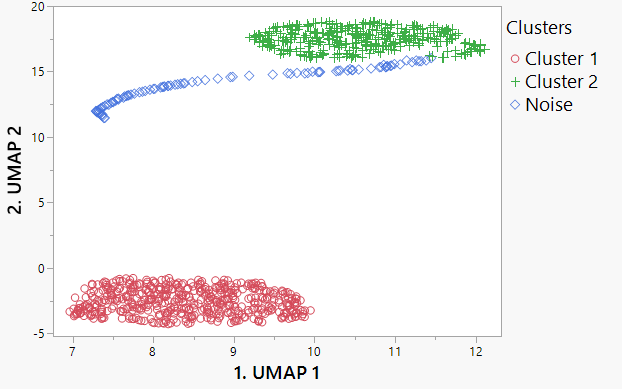
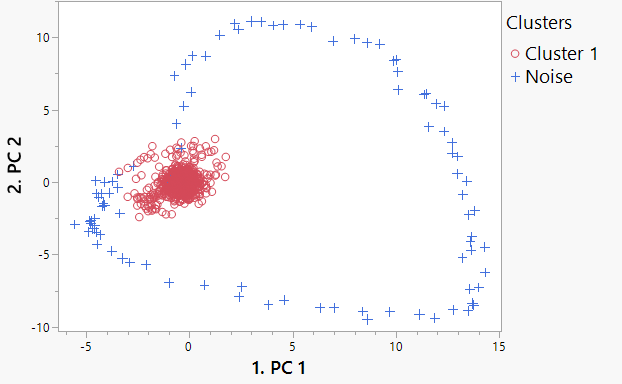
*Data Mining*

FASTMAN-JMP workflow is illustrated in ***Figure 1***. It begins with the data pre-processing phase, a frequently underestimated yet vital stage in the process. Often times, data is not clean. There may be missing values, outliers, or non-numeric values. In the context of chemical plants, malfunctioning sensors, process downtime, or human errors could cause missing data or outliers in the data. The next pre-processing step is data scaling. Scaling is a crucial step in the machine learning pipeline due to different ranges of the features. The environment has implemented several dimensionality reduction methods, including Principal Component Analysis (PCA), Independent Component Analysis (ICA), Spectral Embedding, t-distributed Stochastic Neighbour Embedding (t-SNE), Uniform Manifold Approximation and Projection (UMAP), Triplet Manifold Approximation (TriMAP), and most recently, Pairwise Controlled Manifold Approximation (PaCMAP). Users do not need to have a deep understanding of each method before using them, as FASTMAN-JMP will employ default parameters if none are specified. This feature allows users to become more familiar with the methods and later fine-tune parameters as needed. Furthermore, FASTMAN-JMP offers a range of clustering methods, including K-Means, MeanShift, Balanced Iterative Reducing and Clustering using Hierarchies (BIRCH), Density-Based Spatial Clustering of Applications with Noise (DBSCAN), and Hierarchical Density-Based Spatial Clustering of Applications with Noise (HDBSCAN). Additionally, users have the option to upload custom cluster labels to overlay on the preprocessed and reduced data projection. Just like with the dimensionality reduction methods, FASTMAN-JMP does not demand that users fully comprehend each clustering method upfront, and default parameters are used in the absence of user specifications.



***Figure 1:*** FASTMAN-JMP workflow for Data Mining

As an example, consider the data from the Tennessee Eastman Process (TEP). Specifically, data for TEP-1, where the A/C feed ratio is varied, and the B composition remains constant (step change). ***Figure 2*** displays 2D projections of the data using four combinations of DR (PCA, UMAP, t-SNE, and PaCMAP) and DBSCAN clustering techniques. It's worth noting that all the methods—namely t-SNE, UMAP, and PaCMAP- show excellent separability among classes (clusters). The only difference between them is that UMAP and PaCMAP subdivide the transient region between clusters (due to the action of the controller) into an additional region, showing a better characterization of the local process behavior. On the other hand, when using PCA for dimensionality reduction, the DBSCAN algorithm falls short in proper classification, illustrating the necessity of selecting an appropriate combination of techniques to fully and accurately classify the data.

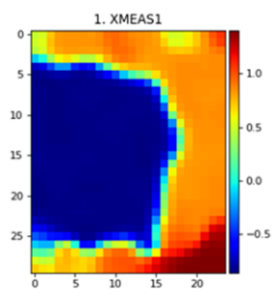
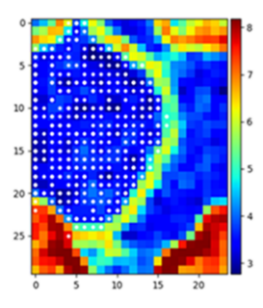
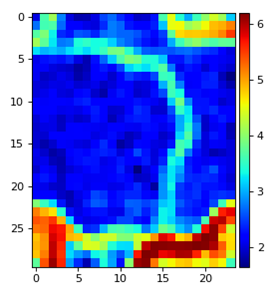


***Figure 2*:** DR and Clustering results for TEP-1

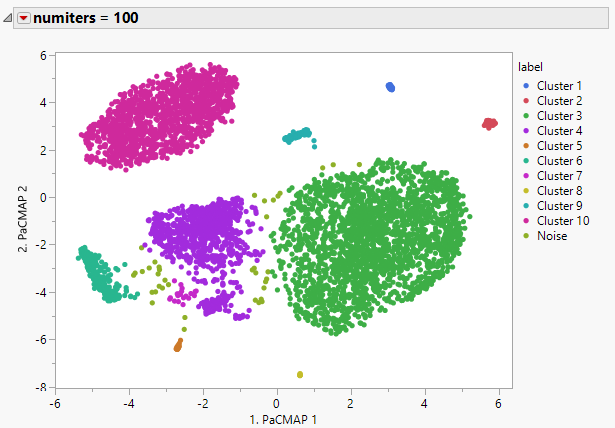
*Analysis Features*

After gaining insight from the dimensionally reduced, clustered data, there are more features that can be used to extract more information and further reassure the users initial conclusions. First, contribution plots can be created. These serve a crucial role in identifying the features that most significantly influence the difference between two specific clusters. This is super effective—with just a few clicks of a button the user can go from knowing little about their data to quickly pinpointing the exact variables responsible for a fault in the process. Another notable feature is the use of Self-Organizing Maps (SOMs), which are unsupervised, neural network-based models that arrange data points onto a lower-dimensional grid. These maps serve as a valuable complement to traditional 2-D dimension reduction plots, offering comparable results and insights. *Figure 3* shows the raw SOM, an original cluster projected to the raw SOM, and a feature projected to the raw SOM.

After verifying the dimensionality reduction and clustering with SOM, the user can conduct sensitivity analysis with FASTMAN-JMP. This analysis, applicable to tasks like data cleaning and normalization, involves selecting parameters (e.g., outlier count for cleaning, cluster numbers for k-means) and setting their range and interval. As the parameter varies within this range, clustering metrics (silhouette, Davies-Bouldin scores) and visual plots of the reduced and clustered data are generated, showcasing the analysis progression. For instance, the impact of varying iterations in PaCMAP dimensionality reduction can be examined. *Figure 4* illustrates the data at 50 and 100 PaCMAP iterations, allowing users to observe the evolution of the dimensionally reduced and clustered data with increasing iterations. This method is valuable for understanding how changes in parameter values influence the final results. On the other hand, FASTMAN-JMP offers an optimizer that optimizes all hyperparameters for the chosen dimensionality reduction and clustering algorithms. After the optimizer is run, it returns a data table with the optimum hyperparameters.



***Figure 3*:** SOM results for TEP-1



***Figure 4:*** FASTMAN-JMP sensitivity analysis for PaCMAP number of iterations

* 1. Predictive Modelling

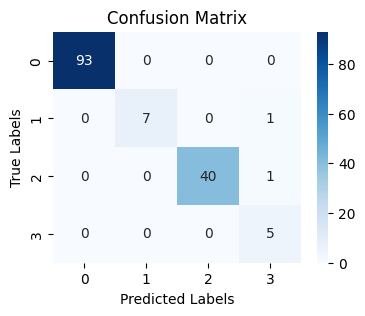
By this stage, the user is expected to have derived significant insights from their data via unsupervised techniques such as dimensionality reduction, clustering, and Self-Organizing Maps (SOMs). Understanding the impact of each feature on the clustering process is further enhanced by examining contribution plots. The next step involves developing predictive models. The analysed data will be employed to train a selected model. It's crucial to generate visualizations to validate the model's effectiveness, ensuring it is capable of making accurate predictions and that the model can be exported for external applications. This process is illustrated in *Figure 5*. FASTMAN-JMP provides the capability to produce both classification and regression models, tailored to the user’s specific application. For instance, classification models can be developed for chemical plant data, labelling data points into predefined classes that represent various plant operating conditions. Conversely, regression models can be constructed to function as 'soft sensors', predicting the value of one feature based on the values of other features. This is particularly useful for features that are challenging or costly to measure directly with physical sensors.



***Figure 5*:** FASTMAN-JMP workflow for predictive modeling

*Classification Model Building*

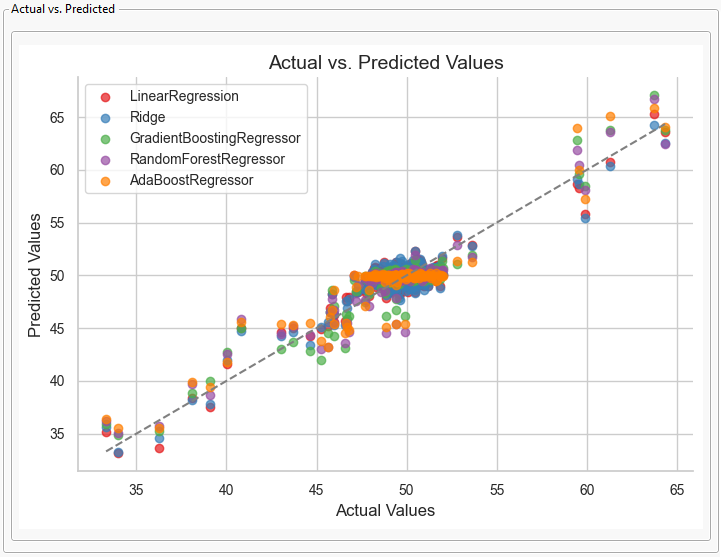
In FASTMAN-JMP, a classification predictive model is created using labelled data from prior clustering. This process involves choosing from available model types like Random Forest, Boosted Tree, Decision Tree, Neural Network, SVM, Linear Classifier, and Naïve Bayes classifier. The procedure includes dividing the data into training and test sets, adjusting hyperparameters, and training the model. After training, the model's accuracy is assessed using the test data, with the option to fine-tune hyperparameters through grid-search optimization if necessary. Depending on the chosen model, specific visualizations are generated, such as a confusion matrix and feature importance plot for a Random Forest model. As shown in *Figure 6*, the confusion matrix plots the true labels vs. the predicted labels and looks for mismatches that can be seen by a heatmap. A good confusion matrix results in a diagonal with darker colors—showing that the predicted labels match with the true labels. These visual aids are crucial for evaluating prediction accuracy and the impact of different features. Finally, the model can be either exported for further use or employed to predict new clustering labels on novel data.



***Figure 6*:**FASTMAN-JMP confusion matrix

*Regression Model Building*

On the other hand, FASTMAN-JMP also offers regression model building. These models do not require labelled data as they are just predicting continuous values. The model types for use are Random Forest, Gradient Boosted Tree, Decision Tree, Linear, Ridge, SVR, ElasticNet, Lasso, and Ada Boost regressors. The process begins with splitting the data into training and testing sets and choosing a suitable model. Key steps involve defining the target variable and identifying predictors. Once the model is trained, its performance is evaluated by plotting actual values against predicted ones, with an ideal model showing data along a diagonal line. Metrics like R-squared value and mean-squared-error are calculated to assess the model. An additional feature in FASTMAN-JMP is model screening. Here, all of the model types are tested and the top five models based off of MSE values are plotted. This model screening process is illustrated in *Figure 7*. Finally, one can optimize hyperparameters using grid-search, export the model, or use it to make predictions on new data.



***Figure 7*:**FASTMAN-JMP plot for different regression models

* 1. Conclusions

By addressing the challenges posed by the volume, velocity, and variety of data generated in modern times, Fastman-JMP offers a comprehensive solution for data exploration and fault detection. Its capabilities in data pre-processing, dimensionality reduction, clustering, and predictive modelling make it an invaluable tool for extracting meaningful insights from complex datasets. The use of the Tennessee Eastman Process data in this study underscores the tool's applicability in real-world scenarios, particularly in process control and fault detection. As industries continue to evolve in a data-driven landscape, tools like FASTMAN-JMP will play a pivotal role in harnessing the power of big data to drive innovation, efficiency, and competitive advantage. This paper not only highlights the features and functionalities of FASTMAN-JMP but also sets the stage for future research and development in the realm of advanced data analytics.

References

*[1] Statistical Software*. JMP. (n.d.). https://www.jmp.com/en\_us/home.html

*[2]* Romagnoli, J. (n.d.). Fault Detection and Knowledge Discovery. PSE LSU. <https://pse.che.lsu.edu/DA.php>

[3] Shen Yin, Steven X. Ding, Adel Haghani, Haiyang Hao, Ping Zhang, A comparison study of basic data-driven fault diagnosis and process monitoring methods on the benchmark Tennessee Eastman process, Journal of Process Control,Volume 22, Issue 9, 2012, Pages 1567-1581, ISSN 0959-1524, <https://doi.org/10.1016/j.jprocont.2012.06.009>. (https://www.sciencedirect.com/science/article/pii/S0959152412001503)