1. Introduction to LabTech Assistant Application

LabTech Assistant is an application created to support scientists and engineers in the statistical analysis of experimental data, design of experiments, and optimization of models using the Response Surface Methodology (RSM). The application has been designed using Python and the Streamlit library, ensuring ease of use and interactivity, enabling users to perform complex data analyses quickly and efficiently without the need for deep programming knowledge.

LabTech Assistant offers a suite of tools for experiment design (DOE - Design of Experiments), including full factorial and fractional factorial designs, Box-Behnken designs, and central composite designs, which are key for effectively studying the impact of multiple independent variables on one or more response variables. With integration of libraries such as Pandas, Numpy, Statsmodels, Matplotlib, Plotly, PyDOE2, and SciPy, the application provides extensive computational and visualization capabilities, essential for statistical analysis and data exploration.

The goal of LabTech Assistant is to enhance research productivity by automating routine data analysis tasks, allowing users to focus on interpreting results and making data-based decisions. With its user-friendly interface and rich functionality, this application is an invaluable tool for scientists in the fields of biological sciences, chemistry, physics, engineering, and many others, who are looking for efficient solutions for analysis and experiment design.

2. Configuration and Requirements

LabTech Assistant is a web application based on the Streamlit platform, providing an intuitive and flexible environment for users without deep programming knowledge. The application has been designed to be compatible with most modern web browsers, offering accessibility and ease of use.

System Requirements:

- Web browser: The latest version of Google Chrome, Mozilla Firefox, Safari, or another modern browser.
- Internet connection: A stable internet connection is required to use the application and access its features.

2.1 Environment Setup:

The application does not require local installation or configuration of a complex environment. Users only need internet access and a web browser to utilize the full functionality of LabTech Assistant. There is no need for additional software or tools configuration.

2.2 Installation and Launch:

As a web application, LabTech Assistant does not require traditional installation. Users can access the application through the dedicated URL labtech-assistant-fvjtcpcqlnkhg9goqc5acs.streamlit.app in their web browser. This process eliminates the need for local setup and makes the application easily accessible from any location and device connected to the internet.

2.3 Supported Libraries:

LabTech Assistant utilizes a range of advanced Python libraries for data analysis and visualization, including:

- Pandas for data manipulation and analysis,
- Numpy for numerical computations,
- Statsmodels for advanced statistical techniques,
- Matplotlib and Plotly for data visualization,
- PyDOE2 for experiment design,
- SciPy for optimization and other mathematical applications.

3. Application Functionality

3.1 Experimental Design

LabTech Assistant offers advanced features for Design of Experiments (DOE), enabling users to select and configure various types of experimental designs according to their specific research needs. Below is a detailed description of the available methods:

- General Full-Factorial Design: This design allows for the full exploration of all possible combinations of factor levels. It enables the identification of the main effects of each factor as well as their interactions. This method is highly flexible, allowing for any number of levels for each of the factors.
- 2-level Full-Factorial Design: This design is limited to analysing two levels (e.g., high and low) for each of the factors. It is an efficient method for identifying main effects and interactions between factors with a limited number of experiments.
- 2-level Fractional Factorial Design: The fractional factorial design allows for the reduction in the number of experiments by selecting a representative fraction of all possible factor combinations. This method is useful in cases of a large number of factors where a full factorial design would be impractical.
- Plackett-Burman Design: Plackett-Burman designs are used for the rapid screening of many factors to identify those that have the most significant impact on the response variable. They are particularly useful in the initial stages of research when a large number of potential factors are analyzed.
- Box-Behnken Design: Box-Behnken is a type of Response Surface Methodology (RSM) design that is effective in fitting quadratic models using fewer experiments than full factorial designs. It does not include all combinations at extreme levels, making it more cost-effective.
- Central-Composite Design (CCD): Central-Composite Design is another RSM method, providing excellent capabilities for precise modeling of the relationship between factors and the response variable. It allows for the estimation of linear, quadratic effects, and interactions between factors. CCD is particularly useful in the optimization of processes and products

3.2 Response Surface Analysis

3.2.1 Create Model

LabTech Assistant facilitates advanced statistical analysis and modeling, utilizing various statistical techniques and tools for analyzing experimental data. Below are the key functionalities related to statistical analysis and modeling:

- Data Loading: Users can easily load data from CSV or Excel files containing the results of their experiments. This feature supports the detection of separators and decimal formats, ensuring correct data loading into the application.
- Data Transformation: Through integration with the PolynomialFeatures library from sklearn.preprocessing, LabTech Assistant enables the automatic transformation of experimental data into a form suitable for statistical analysis and modeling. This function allows for the easy generation of polynomial variables, which are crucial in the modeling process.
- Regression Model Creation: Users can create linear regression models using OLS (Ordinary Least Squares) from the statsmodels.api library. This method allows for the estimation of model coefficients that best describe the relationships between independent variables and the dependent variable.
- Variable Selection: LabTech Assistant allows users to select variables to be included in the model. This enables users to experiment with different combinations of variables to find the model that best describes their data.

3.2.2 Model Visualisation

LabTech Assistant provides advanced data visualization capabilities, enabling users to perform deep visual analysis of their experimental data and modeling results. Here's a detailed description of the available visualization features:

- 3D Response Surface Plots: These allow users to visualize complex relationships between two independent variables and the response variable. The style of the plots includes color gradients and three-dimensional geometry to show how independent variables affect the dependent variable. The aim of this type of visualization is to identify and understand the nature of the interactions between variables.

- Contour Plots: Present the relationships between two independent variables and the response variable in the form of a contour map. These plots are particularly useful when identifying areas of similar response values or locating the optimum. Similar to 3D plots, they help understand how changing one variable affects the response variable while keeping the other variable constant, but in a 2D plane.
- Residual Plots: Are used to assess the quality of the statistical model fit, displaying the differences (residuals) between observed and predicted response variable values. They function to identify possible patterns in data that may have been missed by the model and are a key diagnostic tool in assessing whether the model adequately describes the data.

Interactive capabilities mean that users can interactively change the ranges of variables on the plots, allowing for detailed examination of areas of interest within the data. Features such as zooming, panning, or rotating 3D plots, facilitated by the use of the Plotly library, enhance detailed data analysis. Users have the ability to customize the plots, adjusting elements such as axis titles or the plot title, allowing for visualization to be tailored to individual presentation needs.

Thanks to these advanced visualization functions, LabTech Assistant greatly simplifies the interpretation of complex datasets and modeling results, enabling users to quickly and intuitively analyze and present their findings.

3.2.3 Optimization

LabTech Assistant also provides an optimization tool, essential for finding optimal variable values that maximize the model's responses. The key optimization features in the application include:

- Setting Optimization Ranges: Users can define lower and upper limits for each of the variables included in the model, allowing for optimization under realistic experimental conditions. This functionality is crucial for maintaining the practical utility of optimization results.

Model Parameter Optimization: Optimization is performed using the minimize method from the SciPy library, which adjusts model variables to maximize the predicted response value. This is a key tool enabling users to find the best experimental conditions.

- Support for Coded and Natural Variables: LabTech Assistant offers support for optimization in both the space of coded variables (e.g., -1, 0, 1) and natural variables. This allows for direct conversion of optimization results to real variable values, facilitating interpretation and application of the results.

3.2.4. Validation

The model validation process in LabTech Assistant is a crucial step ensuring that the created statistical models are not only fitted to training data but also effectively predict outcomes on new, unseen data. Here are the key functionalities related to model validation:

- Loading Validation Data: Users can upload independent validation data sets using CSV or Excel files. This feature allows for a quick comparison of model predictions with actual experimental outcomes.
- Calculating Prediction Errors: LabTech Assistant automatically calculates key prediction error indicators, such as Mean Squared Error (MSE), Root Mean Squared Error (RMSE), and Mean Absolute Error (MAE), providing users with quantitative measures of model accuracy.
- Model Validation: The validation process involves comparing model-predicted response variable values with actual values contained in the validation data. This comparative analysis assesses how well the model generalizes to new data.
- Optimization Based on Validation Results: Based on the validation outcomes, users can decide on further model optimization, adjusting parameters or selecting different modeling methods to increase prediction accuracy.

4. Methodology

LabTech Assistant utilizes advanced algorithms and statistical methods for the analysis of experimental data and experimental design. Key methods employed include:

4.1. Experimental Design (DoE)

Uses factorial designs methods, including general full-factorial, 2-level full-factorial, fractional factorial, and Plackett-Burman, to study the effects of multiple factors simultaneously. It also uses response surface methodology (RSM) with Box-Behnken and central-composite designs (CCD) for optimizing experimental conditions and modeling response surfaces.

4.2. Statistical Analysis and Modeling

OLS (Ordinary Least Squares) modeling with polynomial transformation in LabTech Assistant is a statistical modeling process that allows for the analysis and interpretation of relationships between independent variables and the dependent variable. Within the application context, polynomial transformation is utilized to generate new features (independent variables) by raising existing variables to various powers and creating interactions among them, allowing for the modeling of more complex nonlinear relationships. The OLS algorithm applies the least squares method to estimate model parameters, minimizing the sum of the squares of the differences between observed and predicted values of the dependent variables, leading to the best model fit to the data.

4.3. Optimization Algorithm

The optimization algorithm in LabTech Assistant identifies optimal experimental conditions by maximizing the predicted response value of the model. Initially, the user defines lower and upper bounds for each of the model variables, which is crucial for conducting optimization under realistic conditions, considering experimental constraints. The algorithm uses the defined variable bounds to prepare input data for the optimization process, setting starting conditions and constraints. This process relies on the minimize method from the SciPy library. The Sequential Least Squares Programming (SLSQP) method is used for optimization, efficiently finding the maximum of the objective function while meeting specified constraints. Upon optimization completion, the algorithm returns a set of optimal variable values along with the objective function value, enabling the identification of the most favorable experimental conditions.

4.4. Model validation

LabTech Assistant allows users to upload independent validation data sets, enabling the comparison of model predictions with actual experimental outcomes. This functionality supports the calculation of key prediction error indicators such as Mean Squared Error (MSE), Root Mean Squared Error (RMSE), and Mean Absolute Error (MAE) using ready-made scikit-learn functions, providing users with quantitative measures of model accuracy.

4.5. Model Visualization

LabTech Assistant employs the Plotly library to implement algorithms for creating interactive and high-quality data visualizations. Key concepts and methods used in plotting include:

4.5.1. 3D Response Surface and Contour Plots

The 3D plotting algorithm uses model prediction data in three dimensions: two independent variables and one dependent variable (model response). Initially, based on the range of values for two independent variables, the algorithm creates a coordinate grid (meshgrid), serving as a base for calculating predicted response values. For each grid point, the predicted model response value is calculated using previously estimated model parameters. The obtained data are then used to create contour or 3D plots.

4.5.2 Residual Plots

Residual plots are used to assess model fit by presenting differences between observed and predicted response values. For each data point, the difference (residual) between the observed value and the model-predicted value is calculated. Residuals are then displayed on a scatter plot (go.Scatter), where the X-axis represents predicted values, and the Y-axis represents residual values.

5. Example Use Case

Generating and analyzing a Box-Behnken Experiment for Yeast Growth Study

Step 1: Experimental Design Selection

The user begins by selecting the Box-Behnken design in the experimental design (DoE) section. The objective is to analyze the impact of three factors (e.g., acetate, propionate, and butyrate concentrations) on a selected response (e.g., dry yeast mass). The user specifies the number of factors and selects the number of central points, which is crucial for evaluating experimental conditions and the capability of response surface modeling.

Concentration Levels:

- Level -1: 0 g/L
- Level 0: 10 g/L
- Level +1: 20 g/L

Response: Dry yeast mass.

Step 2: Generating Results

After conducting experiments under conditions defined by the Box-Behnken design, the user inputs the obtained results into the application. These data are then used for further analysis and modeling.

Step 3: Model Creation and Parameter Analysis

Using the OLS method with polynomial transformation, the user creates a statistical model describing the relationships between the factors and the experimental response. Analyzing the model parameters, such as regression coefficients and fit statistics (e.g., R-squared), allows for evaluating how well the model represents the experimental data.

Step 4: Visualization

With visualization functions, the user can generate 3D response surface plots or contour plots to visually represent the relationships between the factors and the response. Residual plots may be used to assess the model fit quality.

Step 5: Optimization

Using the optimization algorithm, the user defines the optimization goal (e.g., maximizing yeast dry mass) and operational boundaries for each of the factors. The algorithm searches for a set of experimental conditions that optimize the predicted response value. The result is a set of optimal experimental conditions.

Step 6: Validation

In the final stage, the user can validate the model using additional experimental data. By comparing the model-predicted and actually observed response values, it is possible to assess how well the model performs in predicting outcomes under new, untested experimental conditions. Metrics such as Mean Squared Error (MSE) and Mean Absolute Error (MAE) can be used to evaluate model accuracy.

This detailed example demonstrates how LabTech Assistant can be utilized for a specific study on yeast growth influenced by acetate, propionate, and butyrate concentrations, showcasing the application's capabilities from design and modeling to optimization and validation.

6. Summary

The LabTech Assistant documentation provides a comprehensive overview of the application designed to aid in the statistical analysis of experimental data, experiment design, and optimization using Response Surface Methodology. It covers system requirements, supported libraries, and detailed functionalities for experimental design, response surface analysis, optimization, and model validation. The methodology section discusses the application's use of statistical methods and algorithms for analysis, modeling, and visualization, using advanced Python libraries. Example use cases demonstrate the application's practical application, such as a Box-Behnken experiment for studying yeast growth. This documentation is essential for users looking to maximize the utility of LabTech Assistant in their research endeavors.