

Software implementation of probabilistic-deterministic design of a chemical experiment on R

Vitaliy Fomin

Buketov Karaganda University; vityfomin@mail.ru

Abstract: This paper presents a software implementation of an algorithm for processing experimental data obtained using the probabilistic-deterministic design of experiments (PDDoE) method in the R environment. The tool enables the construction of partial and generalized dependencies between results and varying factors, supports templates of different dimensionalities, automates the selection of approximation models, and evaluates their quality using R^2 , the nonlinear multiple correlation coefficient (R_M), and its significance (tR_M). Implementation is carried out entirely in R using packages such as openxlsx, dplyr, and tcltk. The system supports arithmetic, geometric, and harmonic averaging methods, automatic orthogonality checks, and prioritized model selection based on tR_M , with emphasis on physically meaningful functions. Generalized equations are constructed by enumerating combinations of significant partial dependencies until the maximum tR_M is achieved. The tool provides result visualization, table export, and a modular structure allowing easy extension. Users can load custom plan templates, add new metrics and approximating functions, and manually define the form of both partial and generalized models. Designed for researchers, educators, and engineers working with multifactorial systems, the tool is applicable in scientific research, chemical technology, and education. The ChatGPT language model was used in R code generation. Potential applications of AI tools in experimental data processing are briefly discussed.

Citation: Fomin, V. (2025). Software implementation of probabilistic-deterministic design of a chemical experiment on R. Bulletin of the L.N. Gumilyov ENU. Chemistry, Geography. Ecology Series, 151(2), 130-142. <https://doi.org/10.32523/2616-6771-2025-151-2-130-142>

Keywords: probabilistic-deterministic design of experiment; R; approximation; modeling of chemical-technological processes.

Academic Editor:
E.Ye. Kopishev

Received: 14.05.2025
Revised: 16.05.2025
Accepted: 19.06.2025
Published: 30.06.2025



Copyright: © 2025 by the authors. Submitted for possible open access publication under the terms and conditions of the Creative Commons Attribution (CC BY NC) license (<https://creativecommons.org/licenses/by-nc/4.0/>).

1. Introduction

Methods of describing complex processes through one-factor dependencies, traditionally used in scientific research, became resource-intensive, economically inexpedient and inefficient in modern conditions. There is a growing interest in mathematical methods of experimental planning (Design of Experiments, DoE), which allow to significantly optimize time and resource consumption while maintaining sufficient accuracy of results.

Historically, the foundation of DoE was laid by Ronald Aylmer Fisher (Fisher, 1926, 1935), who developed the principles of Latin squares, randomization, and analysis of variance. Subsequently, George Edward Box, who worked with G. Jenkins, proposed steepest ascent and

sequential experimentation methods, and justified the approach to time series modeling and experimental data processing (Box et al., 2015). These methods gave impetus to the application of DoE in chemical engineering and process engineering problems. Current reviews (Jankovic et al., 2021; Lee et al., 2022; Eriksson et al., 2014; Barad, 2014) confirm that DoE remains important, but classical full-factor plans become unwieldy with the increase of the number of factors. Significantly, any of the classical approaches allows only one response function to be optimized at a time, and linear functions are used for approximation.

More flexible methods are required to describe processes with pronounced nonlinearity and interaction of factors. One of them is probabilistic-deterministic design of experiment (PDDoE), developed based on the ideas of M.M. Protodyakonov and R.I. Tedder (Protodyakonov & Tedder, 1970), who proposed to use multiplication of partial dependencies instead of polynomial regressions. The main contribution to the formalization and development of PDDoE was made by V.P. Malyshev, a scientist from Kazakhstan, who demonstrated its applicability in metallurgy and chemistry, and pointed out a number of advantages over classical multifactorial design (Malyshev, 1981, 1994 Belyaev & Malyshev, 2008).

Simultaneously, the method of G. Taguchi was developing in the world, oriented at increasing the stability of processes. His approach differed in criteria and structure, not being an alternative to the PDDoE. As in most classical DoE approaches, Taguchi's method optimizes one aggregate result - the so-called loss function, which includes dispersion and deviation from the target, but still represents the only output indicator. G. Taguchi's approach is well explained in his classic work (Taguchi, 1986).

PDDoE is actively used in the Republic of Kazakhstan and CIS countries in chemical-technological experiments (Akberdin et al., 2018; Akhmetkarimova et al., 2017; Gogol et al., 2023; Ibishev et al., 2017; Troeglasova, 2020), as well as for the tasks of optimization of settings and calibration of physicochemical analysis instruments (Fomin et al., 2021a, 2021b, 2022, 2024; Turovets et al., 2024). In general, the PDDoE method yields stable mathematical models with predictive power in most of the mentioned works and many other works.

However, with the growing interest to the method, examples of methodological errors have also appeared: not well-founded attempts to describe generalized dependencies through polynomial regressions, as well as the use of six-level plans, which is mathematically impossible, since it is strictly proved that there are no orthogonal Latin squares of order 6 (Bose et al., 1960).

Existing software for PDDoE is extremely limited and is often represented by Excel macros. Only two programs implementing the core functionality of the PDDoE are available (Author's Certificate RK No. 26, 2018; No. 5515, 2019). The first one implements almost all the functionality required in the "classical" PDDoE, from plotting to quality assessment of the generalized equation, but its further development is difficult for a number of reasons, and the reports generated by it need manual refinement to use the data in publications. The second one is intended for work with atomic emission spectra and is not very convenient for general tasks. For some time, the program analiz3 was available for purchase, distributed through the analiz3.com website (No authors, 2013). The program used a manually set "reference point" and grid search for approximation. Currently, the website is not functioning, and the program is not officially distributed. This raises the need to develop flexible, reproducible, and accessible software tools oriented to the current scientific practice of using and further developing PDDoEs.

Modern approaches to processing chemical information increasingly rely on machine learning, text mining, and scientific programming. In particular, tools are being developed for the automated extraction and interpretation of knowledge from unstructured sources. Large language models (LLMs) have been shown to be highly efficient in recognising chemical entities (Kumari et al., 2025), extracting synthesis conditions (Shi et al., 2025), and automating the structuring of synthetic protocols (Ai et al., 2024).

Rampal et al.'s (2024) work represents a step towards building comprehensive knowledge bases for training AI systems to synthesise substances. These approaches are enhanced by

specialised chemical language models, such as ChemLM, which have demonstrated a high level of accuracy in predicting molecular properties (Kallergis et al., 2025).

Particular attention has been paid to constructing mathematical models and optimising experimental conditions. For instance, Mahmood et al. (2025) demonstrate how the kernel ridge regression method can be improved when applied to chemical datasets. Song and Sun (2025) analyse the potential for local optimisation of reaction conditions using active learning and Bayesian approaches. These efforts generally aim to reduce the amount of experimentation required and increase its informativeness.

Integrating machine learning into the study of natural compounds (Shi et al., 2025) and the structural analysis of texts (Schilling-Wilhelmi et al., 2025) highlights the transdisciplinary potential of these methods. However, the interpretability, verifiability, and universalisation of algorithms remain matters of debate, as discussed in Cooper's (2025) review of the Faraday Discussion on data-driven chemistry.

Together, these studies emphasise the necessity of tools that combine the rigour of mathematical modelling, the flexibility of the R statistical language, and the adaptability of modern AI tools, which is the essence of the proposed PDDoE approach.

The present work aims at automating the construction and correctness checking of experiment plans by the PDDoE method using the R programming language. It is based on both classical approaches and the author's own developments, including publications and registered certificates of authorship.

2. Materials and methods

The mathematical basis of the data processing procedure implemented in this work is based on the classical publications of V.P. Malyshev (1981) and subsequent joint work with S.V. Belyaev. These sources describe an approach based on the use of Latin squares for constructing orthogonal plans of experiment, sampling values by factor levels, and obtaining partial dependencies with their subsequent combination into a generalized multifactorial equation.

At the current stage of implementation of the methodology, the R environment of versions 4.4 and 4.5 was used. The main packages involved in the scripts are:

- openxlsx - for working with Excel files, including experiment plan files;
- dplyr and tidyr - for data transformation;
- teal and shiny - for interactive interaction with the user;
- base and stats - for basic calculations and approximation.

The experiment plans constructed from the literature data are stored in the Excel file Plans.xlsx. However, the presence of Microsoft Excel on the user's computer is not required - interaction with the files is performed through the openxlsx package. Adding new plans is possible by simply extending the structure of the file, without changing the code of the scripts.

The development scripts were created in the RStudio environment version 2024.12.1. During the development OpenAI ChatGPT v4o system is used, which provides automatic generation of code blocks in R language and logic of their interaction by text description of necessary algorithms. All final scripts were manually checked, finalized and debugged on real data.

The detailed processing algorithms, including the implementation of averaging functions, construction of partial dependencies and selection of approximation models, are presented in the next section.

3. Results

3.1. Implemented experiment plan templates

As a part of the development of the software tool, templates of experiment plans corresponding to orthogonal Latin squares of various orders were implemented and tested. The raw data on such plans were obtained from previously published sources (including Protodyakonov,

Malyshev, and Belyaev) and adapted to an electronic format in the form of Excel spreadsheets. All plan templates are stored in the user-accessible file Plans.xlsx.

At the time of this article, the following plans are implemented in the file:

3×2 plan, 3 factors, 2 levels, convenient for combining PDDoE with Box-Wilson type methods;

4×3 plan;

5×4 plan;

the popular 6×5 plan, for six factors with five levels;

8×7 plan;

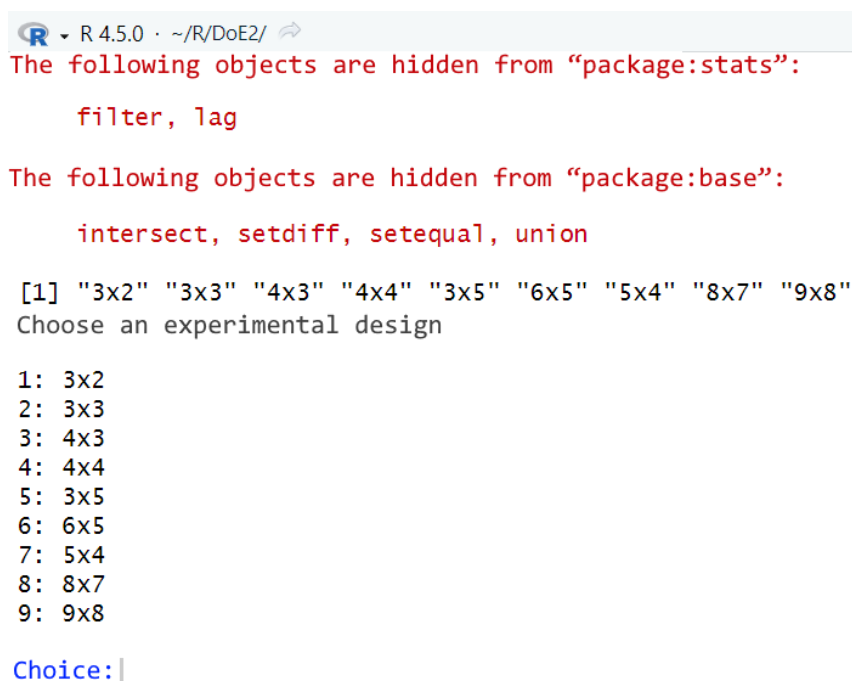
9×8 plan;

several test and auxiliary variants for debugging and experimentation.

Each of the plans is constructed with consideration of orthogonality conditions: each level of one factor meets each level of the other factors once and only once. Validity of the plans is checked automatically during loading and transformation. A validity checking mechanism is added in case the user changes or adds plans.

The user can select any of the available plans or add a new one by saving it as a separate sheet in the Plans.xlsx file. No changes to the script code are required in this case.

When starting the script, the user is prompted to select a plan by entering its number (Fig. 1).



```
R 4.5.0 · ~/R/DoE2/
The following objects are hidden from "package:stats":
  filter, lag

The following objects are hidden from "package:base":
  intersect, setdiff, setequal, union

[1] "3x2" "3x3" "4x3" "4x4" "3x5" "6x5" "5x4" "8x7" "9x8"
Choose an experimental design

1: 3x2
2: 3x3
3: 4x3
4: 4x4
5: 3x5
6: 6x5
7: 5x4
8: 8x7
9: 9x8

Choice: |
```

Figure 1. Interface for selecting one of the available experiment plans when running the script in the R console

After selecting a plan, the script automatically receives information about the number of factors conceived by the experimenter and the number of levels of their variation.

3.2 Dialog for describing the results

The next step is to describe the results to be measured. The user is asked to specify how many different results will be recorded in the experiment (e.g. element content, mass of residue, recovery factor, etc.). For each result it is necessary to enter its designation, which will be used in tables and formulas (Fig.2).

After that, for each result the number of repetitions is specified - how many times the measurement will be performed for each combination of factors. Thus, the system forms a table

structure, in which each result is represented as a set of columns by the number of repetitions (e.g. Y1_Rep_1, Y1_Rep_2).

All dialogs are implemented in the interactive console of the R environment and do not require a separate interface. If necessary, the script can be supplemented with a shiny-based user interface, but at the current stage the focus is on providing the necessary functionality.

The requested information is used further to form the final experiment plan and further processing of the results.

```
Selection: 4
How many different measurement results will there be? 2
Enter a designation for the result 1: As
Enter a designation for result 2: Ds
How many repetitions for result As? 3
How many repetitions for result Ds? 3
Fill in the Factors.xlsx table and save it. Press Enter to continue...
|
```

Figure 2. Dialog for describing the structure of measurement results in the R console interface

3.3. Filling the table with factors

At the next stage, the user needs to fill in the table Factors.xlsx, automatically generated by the script. This table contains the names of factors, their symbols, types (N - normal, V - vacant), and specific values for each level of variation (Fig.3). The "Label" field contains the desired designation of the factor in the formulas, and the "Dependence" field contains the numerical designation of the type of approximating function ("0" - automatic selection).

The file is opened and edited in an external table editor. Currently, this requires Microsoft Excel or compatible software (LibreOffice, OnlyOffice, WPS Office, etc.). The possibility of interacting with this table through the interface provided by R extensions has not been implemented yet.

	A	B	C	D	E	F	G	H
1	Factor	Label	Type	Dependence	Level1	Level2	Level3	Level4
2	Energy, mJ	E	N	0	3.1	5.6	7	10
3	Temper., C	T	N	0	100	150	200	240
4	Time, min	t	N	0	60	120	180	220
5	Vacant	V	V	0	1	2	3	4
6								

Figure 3. Example of completing the Factors.xlsx table with description of factors, their labels, types and levels

3.4 Forming the working plan of the experiment

After loading information about factors and results, the script automatically generates the final matrix of the experiment. During the generation process, the validity check is performed:

- correspondence of levels to the number of factors;
- automatic replacement of level numbers in the plan template with their values;
- consistency of names and designations;
- orthogonality of the plan.

If all conditions are met, the script generates a table that includes:

- the ordinal number of the experiment;
- combinations of factor levels;

empty columns for entering experimental results for each repetition (Fig. 4).

The file is saved in Excel format and is ready to be filled in. The user can enter values at any time and then proceed to data processing using the second script.

After completing the table, the user returns to the console and confirms the completion of editing by pressing Enter. Then the data is loaded back into R and used to form the final experiment plan with substitution of actual values of factors.

Experiment	Energy, mJ	Temper., C	Time, min	Vacant	As_Rep_1	As_Rep_2	As_Rep_3	Ds_Rep_1	Ds_Rep_2	Ds_Rep_3
1 Exp. # 1	3.1	100	60	1	NA	NA	NA	NA	NA	NA
2 Exp. # 2	3.1	150	120	2	NA	NA	NA	NA	NA	NA
3 Exp. # 3	3.1	200	180	3	NA	NA	NA	NA	NA	NA
4 Exp. # 4	3.1	240	220	4	NA	NA	NA	NA	NA	NA
5 Exp. # 5	5.6	100	120	3	NA	NA	NA	NA	NA	NA
6 Exp. # 6	5.6	150	60	4	NA	NA	NA	NA	NA	NA
7 Exp. # 7	5.6	200	220	1	NA	NA	NA	NA	NA	NA
8 Exp. # 8	5.6	240	180	2	NA	NA	NA	NA	NA	NA
9 Exp. # 9	7.0	100	180	4	NA	NA	NA	NA	NA	NA
10 Exp. # 10	7.0	150	220	3	NA	NA	NA	NA	NA	NA
11 Exp. # 11	7.0	200	60	2	NA	NA	NA	NA	NA	NA
12 Exp. # 12	7.0	240	120	1	NA	NA	NA	NA	NA	NA

Figure 4. Final view of the experiment plan generated automatically by the selected scheme

3.5 Processing a filled plan

After the user has entered the measured values into the corresponding columns of the plan, the main data processing script (Partial2.R) can be run. This script automatically loads the table with the results, checks the presence of all necessary data and immediately proceeds to the generation of generalized models by searching possible combinations of partial dependencies. The user is not offered to view the partial models separately - they are used in the background only for quality assessment and selection.

Partial dependencies are formed by averaging values over the levels of the corresponding factor. Three types of averaging are available so far:

arithmetic: $\bar{x} = (1/n) * \sum x_i$;

geometric: $\bar{x}(g) = (\prod x_i)^{(1/n)}$, applies only for all positive values;

harmonic: $\bar{x}(h) = n / \sum (1/x_i)$, also not applicable in the presence of zero and negative values.

The choice of averaging type can be made by the user or determined automatically by the presence of corresponding columns in the table. If all three types of averages are present, the comparison of models based on them is allowed.

Before averaging, the user can choose the way of working with repetitions:

simple averaging;

Cochrane homogeneity test with rejection of suspicious values and averaging.

For each partial dependence the selection of models from the set is performed:

linear function;

step dependence;

exponential;

logarithmic;

inverse;

extended forms of exponential and logistic approximation.

Dependences with integer degrees 2 and 3, as well as 1/2 and 1/3, are considered separately from the stepped dependence with automatically fitted degree. Each model is characterized by coefficients A, B (C if necessary), and evaluated by the metrics R^2 , RM, and tRM. In the first step, the three best models per factor for each outcome are selected, prioritizing the physical (basic, intrinsically linear) functions (linear, stepwise, exponential, logarithmic, hyperbolic, inverse). In the

case of a two-level plan, only the linear approximation is used. If there are not three functions with $tRM \geq 2$ among the basic ones, then interpolating functions are added to the set (polynomial of the second (number of levels $n > 3$) and third ($n > 4$) orders, exponential-degree function). If $n \geq 5$, logistic function can be used. A total of 15 approximating functions are implemented.

At the second stage, all possible combinations of the selected models are enumerated in order to construct generalized dependencies. For each such combination the values are recalculated, compared with the initial values, and the final values of R^2 , R_M , tR_M for the generalized model are calculated. Among them, those that satisfy the significance criterion (usually $tR_M \geq 2$) are selected.

The results are stored in R environment variables and can be additionally exported, visualized or analyzed by the user manually or by R tools.

3.6 Visualization and output of results

The script automatically generates a console report containing brief information on all results, models built, values of quality metrics (R^2 , R_M , tR_M) and selected functions. The report allows the user to immediately evaluate the success of the approximation and identify cases in which none of the models passed the quality criteria.

A graphical comparison of experimental and calculated values (e.g., via ggplot2 or plot) can be plotted for clarity. The user can also visualize distributions of residuals, comparison of different models or the result of generalized approximation (Fig.5).

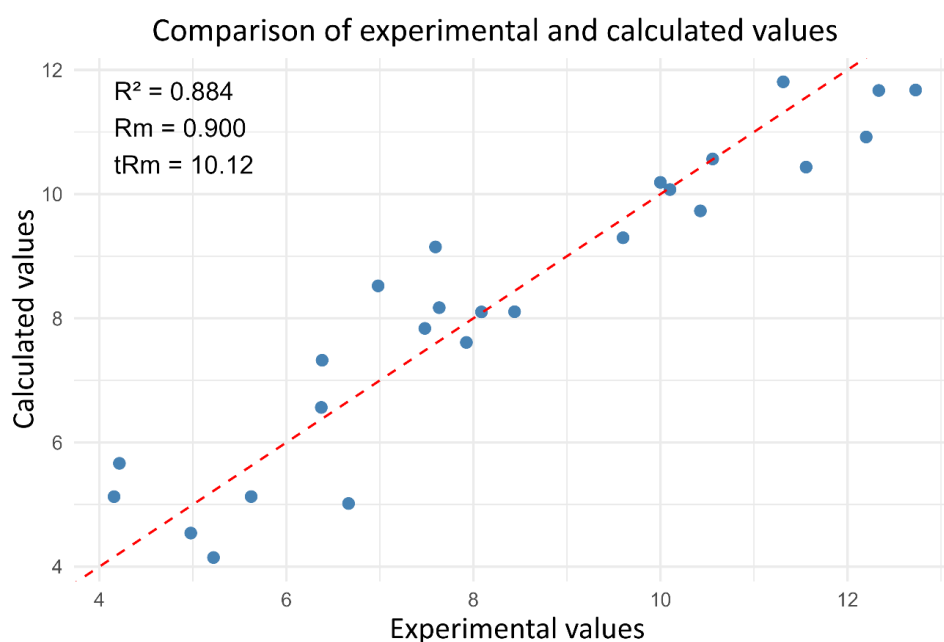


Figure 5. Visualization of comparison of experimental and computational models by means of R

The output can be saved in HTML, PNG or PDF format. Tables with model coefficients, metrics and selected feature numbers are also saved, suitable for further analysis or inclusion in reporting documentation.

3.7 Co-working with ChatGPT

The OpenAI ChatGPT artificial intelligence system was used extensively during the script development process. Initially, the model had no built-in information about the methodology of probabilistic-deterministic design of experiment, which required step-by-step additional training based on the literature provided by the author (including the works of Protodyakonov, Malyshev, and modern applied research).

Special attention was paid to correctly distinguishing the PDDoE from more popular methods such as the classical DoE and Taguchi approaches. The model tended by default to substitute

specific elements of the method for more widely known analogs. Gradual refinement and refinement training allowed the model to stabilize its behavior and achieve meaningful dialogue on a highly specialized topic.

Due to the limitations of the system architecture, there is currently no way to directly execute the R code inside the model. This required external debugging of all proposed fragments manually in the RStudio environment. Despite this, interaction with the language model significantly accelerated the formation of the script structure, automation of template sections and development of solution algorithms for all stages of data acquisition and processing by the PDDoE method.

4. Discussion

The choice of R language and environment for the implementation of the PDDoE method, including the selection and filling of the plan, statistical processing of parallel measurement results and automated selection of the generalized function by the maximum tRM, is explained by the initial focus of the language on statistical data processing, the free distribution and cross-platform nature of the environment, its popularity among scientists in the world. R, unlike “classical” PLs, natively supports vectorized computations, which is very convenient when working with objects like PDDoE plans.

The implemented algorithm allows the user without deep knowledge of mathematical statistics or programming to obtain adequate models of dependence of experimental results on factors. At the same time, the flexibility of the method characteristic for PDDoE is preserved: it is possible to use plans of different dimensionality, different functions for approximation of partial dependencies and averaging methods.

The presence of the mechanism of automatic model search and tRM filtering provides high sensitivity of the method to the adequacy of approximation, while the use of physically interpretable functions at the first stage makes the models not only statistically but also physically meaningful.

Compared to existing software solutions, the list of approximating functions has been significantly expanded. If necessary, any known methods of approximation can be implemented in the script.

One of the significant limitations of the current implementation is the lack of a full-fledged user interface: interaction with the program is performed through the R console and external table editors. This may be difficult for users without experience in the corresponding environment. However, the structure of the code and the packages used make it possible to implement a graphical interface in the future using shiny, golem or flexdashboard.

It also seems promising to add the calculation of dCor, Chatterjee's ξ (see, e.g., Székely et al., 2007 for dCor; Chatterjee, 2021 for ξ), and other dependency metrics that may be useful in problems with a large number of factors and weak linearity of the relationship.

The possibility of adapting the script for problems with specific constraints deserves special attention: for example, in the presence of fixed groups of factors, composite variables (Fomin et al., 2017; Fomin & Dik, 2015), or the need to interpret the result as a mixture of contributions. These directions are under active development.

The author invites colleagues to discuss, criticize and jointly develop the tool, including in terms of expanding the model library, adaptation to specific subject areas and integration with other software tools for analyzing experimental data. The ultimate goal is to publish the scripts as an R package in the official CRAN repository to promote the PDDoE method.

5. Conclusion

This paper describes a software tool for processing experimental data obtained using the technique of probabilistic deterministic design of experiments (PDDoE). The implemented algorithm allows to automate key stages: from the selection and generation of the experimental plan to the construction of generalized models with the assessment of their quality.

The principal features of the approach are the flexibility of the system, the preferred use of physically interpretable functions in the approximation of partial dependencies, as well as the automatic search of models according to the tR_M metric to identify the best combination of partial dependencies. The introduction of automatic processing and the possibility of adding new patterns makes the tool suitable for a wide range of tasks in chemical-technological, physical-chemical and other applied research.

The possibility of productive interaction with the ChatGPT artificial intelligence system is shown, allowing to speed up the development without reducing the scientific validity of solutions. The work is performed in the console environment R with preservation of maximum openness and possibility of code modification.

The proposed tool is already used in laboratory practice, and its further development involves expanding the model library, integrating additional metrics, and creating a user interface. The results obtained can be useful both for researchers mastering or using the PDDoE method and for teachers interested in its implementation in the educational process.

6. Supplementary Materials: No supplementary material.

7. Author Contributions

The author carried out all work on script development and preparation of the article.

8. Author Information

Fomin Vitaly Nikolaevich - Head of LEP “PhChMI”, Buketov Karaganda University, 28, Universitetskaya str., Karaganda, Kazakhstan, 100028; e-mail: vitfomin@mail.ru, <https://orcid.org/0000-0002-2182-2885>

9. Funding: This research was funded by the Science Committee of the Ministry of Science and Higher Education of the Republic of Kazakhstan (Grant No. AP19677716 «Development of methods for qualitative and quantitative analysis of lead and copper alloys using laser induced breakdown spectroscopy and chemometrics»).

10. Acknowledgements: The author expresses his gratitude to the staff of LEP “PhChMI” of Buketov Karaganda University for support in script development and writing the article.

11. Conflicts of Interest: The authors declare no conflicts of interest.

12. References

1. Modifitsirovannyi metod veroyatnostno-determinirovannogo planirovaniya eksperimenta (MVE) [Modified method of probabilistic deterministic experiment planning (MPE)]. (2013). Internet Archive: <https://web.archive.org/web/20130401063933/http://analiz3.com/>
2. Ai, Q.X., Meng, F.W., Shi, J.L., et al. (2024). Extracting structured data from organic synthesis procedures using a fine-tuned large language model. *Digital Discovery* 3(9), 1822–1831. <https://doi.org/10.1039/d4dd00091a>
3. Akberdin, A.A., Kim, A.S., Sultangaziev, R.B. (2018). Experiment Planning in the Simulation of Industrial Processes. *Steel Transl* 48, 573–577. <https://doi.org/10.3103/S0967091218090024>
4. Akhmetkarimova, Z.S., Baikenov, M.I., Dyusekenov, A.M. (2017). Mathematical simulation of the hydrogenation of borodino coal. *Solid Fuel Chemistry* 51(2), 111–114. <https://doi.org/10.3103/S0361521917020021>
5. Barad, M. (2014). Design of experiments (DOE) - A valuable multi-purpose methodology. *Applied Mathematics* 5(14), 2120–2129. <https://doi.org/10.4236/am.2014.514206>
6. Belyaev, S.V., Malyshev, V.P. (2008). Puti razvitiya veroyatnostno-determinirovannogo planirovaniya eksperimenta [Development paths of probabilistic-deterministic experimental design]. In *Kompleksnaya pererabotka mineral'nogo syr'ya Kazakhstana. Sostoyanie*,

- problem, resheniya [Complex processing of Kazakhstan's mineral raw materials. State, problems, solutions]* 9(8), 599–633.
7. Bose, R.C., Shrikhande, S.S., Parker, E.T. (1960). Further results on the construction of mutually orthogonal Latin squares and the falsity of Euler's conjecture. *Canadian Journal of Mathematics* 12, 189–203
 8. Box, G.E.P., Jenkins, G.M., Reinsel, G.C., Ljung, G.M. (2015). Time series analysis: Forecasting and control (5th ed.). John Wiley & Sons.
 9. Chatterjee, S. (2021). A new coefficient of correlation. *Journal of the American Statistical Association* 116(536), 2009–2022. <https://doi.org/10.1080/01621459.2020.1773984>
 10. Cooper, A.I. (2025). Concluding remarks: Faraday Discussion on data-driven discovery in the chemical sciences. *Faraday Discussions* 256, 664–690. <https://doi.org/10.1039/d4fd00174e>
 11. Eriksson, L., Johansson, E., Wold, S. (2014). Design of experiments (DoE) and process optimization: A review of recent publications. *Organic Process Research & Development* 19(11), 1605–1633. <https://doi.org/10.1021/op500169m>
 12. Fisher, R.A. (1926). The arrangement of field experiments. *Journal of the Ministry of Agriculture of Great Britain* 33, 503–513.
 13. Fisher, R.A. (1935). The design of experiments. Oliver and Boyd.
 14. Fomin, V.N., Dik, A.V. (2015). Using one-way analysis of variance in the stochastic determined design of experiment. *Bulletin of Karaganda University. Ser. Chemistry* 1(58), 17–20.
 15. Fomin, V.N., et al. (2021a). Optimization of the parameters of a laser induced breakdown spectrometer (LIBS) using probabilistic-deterministic design of experiment. *Industrial Laboratory Diagnostics of Materials* 87(5), 14–19. <https://doi.org/10.26896/1028-6861-2021-87-5-14-19>
 16. Fomin, V.N., et al. (2021b). Optimization of coal tar gas chromatography conditions using probabilistic-deterministic design of experiment. *Bulletin of the University of Karaganda-Chemistry* 104, 39–46. <https://doi.org/10.31489/2021Ch4/39-46>
 17. Fomin, V.N., et al. (2022). Method for qualitative and quantitative analysis of ancient lead enamel using laser inducted breakdown spectroscopy. *Bulletin of the University of Karaganda-Chemistry* 108, 107–117. <https://doi.org/10.31489/2022Ch4/4-22-16>
 18. Fomin, V.N., et al. (2024). Method of classification and quantitative analysis of vein quartz using LIBS and chemometric techniques. *Bulletin of the L. N. Gumilyov Eurasian National University. Chemistry Geography Ecology Series* 147(2), 48–60. <https://doi.org/10.32523/2616-6771-2024-147-2-48-60>
 19. Fomin, V.N., Kovaleva, A.A., Aldabergenova, S.K. (2017). Ispolzovanie mnogofaktornykh peremennykh v veroyatnostno-determinirovannom planirovanii eksperimenta [Use of multifactor variables in probabilistic-deterministic design of experiment]. *Vestnik Karagandinskogo Universiteta. Seriya Khimiya [Bulletin of Karaganda University. Chemistry Series]* 3(87), 91–100.
 20. Gogol, D.B., Rozhkovoy, I.E., Sadyrbekov, D.T., Makasheva, A.M. (2023). Deposition of transition metal onto carbonate materials surface: Theoretical evaluation of optimal parameters. *Eurasian Journal of Chemistry* 28, 4(112). <https://doi.org/10.31489/2959-0663/4-23-13>
 21. Ibishev, K.S., Malyshev, V.P., Kim, S.V., Sarsembaev, B.S., Egorov, N.B. (2017). Preparation of nanosized nickel powder by direct-current electrolysis combined with high-voltage spark discharge. *High Energy Chemistry* 51(3), 219–223. <https://doi.org/10.1134/S0018143917030055>
 22. Jankovic, A., Chaudhary, G., Goia, F. (2021). Designing the design of experiments (DOE) – An investigation on the influence of different factorial designs on the characterization of complex systems. *Energy and Buildings* 250, 111298. <https://doi.org/10.1016/j.enbuild.2021.111298>

23. Kallergis, G., Asgari, E., Empting, M., et al. (2025). Domain adaptable language modeling of chemical compounds identifies potent pathoblockers for *Pseudomonas aeruginosa*. *Communications Chemistry* 8(1), 114. <https://doi.org/10.1038/s42004-025-01484-4>
24. Kumari, M., Chauhan, R., Garg, P. (2025). Can LLMs revolutionize text mining in chemistry? A comparative study. *Computer Standards & Interfaces* 94, 103997. <https://doi.org/10.1016/j.csi.2025.103997>
25. Lee, B.C.Y., Mahtab, M.S., Neo, T.H., Farooqi, I.H., Khursheed, A. (2022). A comprehensive review of Design of Experiment (DOE) for water and wastewater treatment application - Key concepts, methodology and contextualized application. *Journal of Water Process Engineering* 47, 102673. <https://doi.org/10.1016/j.jwpe.2022.102673>
26. Mahmood, S.W., Basheer, G.T., Algamal, Z.Y. (2025). Quantitative structure-activity relationship modeling based on improving kernel ridge regression. *Journal of Chemometrics* 39(5), e70027. <https://doi.org/10.1002/cem.70027>
27. Malyshev, V.P. (1981). Probabilistic-deterministic design of experiment [Veroyatnostno-determinirovannoe planirovanie eksperimenta in Russian]. Nauka, Almaty.
28. Malyshev, V.P. (1994). Probabilistic-deterministic mapping [Veroyatnostno-determinirovannoe otobrazhenie in Russian]. Gylm, Karaganda.
29. Protodyakonov, M.M., Tedder, R.I. (1970). Methodology of rational experimental design [Metodika ratsional'nogo planirovaniya eksperimentov in Russian]. Nauka, Moscow.
30. R Core Team. (2024). R: A language and environment for statistical computing (Version 4.4.1). R Foundation for Statistical Computing. <https://www.R-project.org/>
31. Rampal, N., Wang, K.Y., Burigana, M., et al. (2024). Single and multi-hop question-answering datasets for reticular chemistry with GPT-4-Turbo. *Journal of Chemical Theory and Computation* 20(20), 9128–9137. <https://doi.org/10.1021/acs.jctc.4c00805>
32. Schilling-Wilhelmi, M., Ríos-García, M., Shabih, S., et al. (2025). From text to insight: Large language models for chemical data extraction. *Chemical Society Reviews* 54(3), 1125–1150. <https://doi.org/10.1039/d4cs00913d>
33. Shi, S.W., Huang, Z.W., Gu, X.X., et al. (2025). From 2015 to 2023: How machine learning aids natural product analysis. *Chemistry Africa* 8(2), 505–522. <https://doi.org/10.1007/s42250-024-01154-3>
34. Shi, Y., Rampal, N., Zhao, C.B., et al. (2025). Comparison of LLMs in extracting synthesis conditions and generating Q&A datasets for metal-organic frameworks. *Digital Discovery*. <https://doi.org/10.1039/d5dd00081e>
35. Song, W.H., Sun, H.G. (2025). Local reaction condition optimization via machine learning. *Journal of Molecular Modeling* 31(5), 143. <https://doi.org/10.1007/s00894-025-06365-0>
36. Székely, G.J., Rizzo, M.L., Bakirov, N.K. (2007). Measuring and testing dependence by correlation of distances. *Annals of Statistics* 35(6), 2769–2794. <https://doi.org/10.1214/009053607000000505>
37. Taguchi, G. (1986) Introduction to Quality Engineering: Designing Quality into Products and Processes. Asian Productivity Organization, Tokyo.
38. Troeglasova, A.V. (2020). Veroyatnostno-determinirovannoe planirovanie eksperimenta po razlozheniyu kremniysoderzhashchikh obraztsov [Probabilistic-deterministic design of experiment for decomposition of silicon-containing samples]. *Fiziko-khimicheskie issledovaniya [Physicochemical Research]* 8(2), 49–55. <https://doi.org/10.33764/2618-981X-2020-8-2-49-55>
39. Turovets, M.A., et al. (2024). Chemometric approach for the determination of vanadium by the LIBS method. *Eurasian Journal of Chemistry*. <https://doi.org/10.31489/2959-0663/4-24-10>

R программалау тілінде химиялық эксперименттің ықтималдық-детерминирленген жоспарлауын бағдарламалық қамтамасыз ету

Виталий Фомин

Андатпа. Мақалада R ортасында ықтималдық-детерминирленген эксперименттік жобалау (ЫДЭЖ) әдісімен алынған эксперименттік деректерді өңдеу алгоритмінің бағдарламалық іске асырылуы ұсынылған. Өзірленген құрал нәтижелер мен айнымалы факторлар арасындағы жеке және жалпылама тәуелділіктерді қалыптастыруға, әртүрлі өлшемді жоспарлардың шаблондарын пайдалануға, жуықтау модельдерін таңдауды автоматтандыруға және олардың сапасын R^2 метрикасына, R_M сызықты емес еселік корреляция коэффициентіне және оның tR_M маңыздылығын бағалауға мүмкіндік береді. Іске асыру алғаш рет R ортасында openxlsx, dplyr, tcltk және басқа пакеттер арқылы орындалды. Орташа алудың әртүрлі әдістеріне (арифметикалық, геометриялық, гармониялық), жоспарлардың ортогональдылығын автоматты түрде тексеруге, физикалық негізделген жуықтау функцияларын таңдау және басымдық беру арқылы tR_M бойынша модельдердің басымдық рейтингісіне қолдау көрсетіледі. Жалпыланған теңдеу ең жоғары tR_M мәніне жеткенше маңызды ерекше функциялардың комбинацияларын санау арқылы құрастырылады. Жүйе нәтижелерді визуализациялауды, кестелерді экспорттауды қамтамасыз етеді және функционалдылықты кеңейтуге мүмкіндік беретін модульдік архитектураға ие. Теңшелетін жоспар үлгілерін жүктеуге, жаңа көрсеткіштерді және теңшелетін жуықтау функцияларын қосуға, ішінара жуықтау функцияның түрін және жалпыланған теңдеуді қолмен тағайындауға және нақты пән аймақтарының тапсырмаларына бейімдеуге болады. Құрал көп факторлы эксперименттермен жұмыс істейтін зерттеушілерге, оқытушыларға және инженерлерге арналған, сонымен қатар ғылыми қызметте, химия инженериясында және оқу тәжірибесінде қолданылуы мүмкін. ChatGPT үлкен тіл үлгісі кодты жасау және жөндеу кезінде пайдаланылды. Химия және химиялық технология мәселелерін шешу үшін қолданбалы бағдарламалауда тілдік модельді пайдалану мүмкіндіктері қысқаша қарастырылады.

Түйін сөздер: эксперименттің ықтималды-детерминирленген жоспарлауы; R; жуықтау; химиялық-технологиялық процестерді модельдеу.

Программная реализация вероятностно-детерминированного планирования химического эксперимента на R

Виталий Фомин

Аннотация. Представлена программная реализация алгоритма обработки экспериментальных данных, полученных по методике вероятностно-детерминированного планирования эксперимента (ВДПЭ), в среде R. Разработанный инструмент позволяет формировать частные и обобщённые зависимости между результатами и варьируемыми факторами, использовать шаблоны планов различной размерности, автоматизировать выбор моделей аппроксимации и оценивать их качество по метрикам R^2 , коэффициенту нелинейной множественной корреляции R_M и его значимости tR_M . Реализация впервые выполнена в среде R с использованием пакетов openxlsx, dplyr, tcltk и других. Поддерживаются различные методы усреднения (арифметическое, геометрическое, гармоническое), автоматическая проверка ортогональности планов, приоритетное ранжирование моделей по tR_M с выделением и приоритезацией физически обоснованных аппроксимирующих функций.

Обобщенное уравнение строится методом перебора сочетаний значимых частных функций до достижения максимума tR_M . Система обеспечивает визуализацию результатов, экспорт таблиц и имеет модульную архитектуру, допускающую расширение функционала. Возможна загрузка пользовательских шаблонов планов, добавление новых метрик и пользовательских аппроксимирующих функций, ручное назначение вида аппроксимирующей частной функции и обобщенного уравнения, адаптация под задачи конкретных предметных областей. Инструмент предназначен для исследователей, преподавателей и инженеров, работающих с многофакторными экспериментами, и может быть использован в научной деятельности, химической технологии и в образовательной практике. Большая языковая модель ChatGPT использовалась на этапе генерации и отладки кода. Кратко обсуждаются возможности использования языковой модели в прикладном программировании для решения задач химии и химической технологии.

Ключевые слова: вероятностно-детерминированное планирование эксперимента; R; аппроксимация; моделирование химико-технологических процессов.