Arch. Min. Sci. **70** (2025), 4, 561-574

Electronic version (in color) of this paper is available: http://mining.archives.pl

DOI: https://doi.org/10.24425/ams.2025.157447

MATEUSZ ZARĘBA[1], MARTA SKIBA[2*]

MACHINE LEARNING METHODS FOR TIME SERIES PREDICTION OF THE DIFFUSION CURVE

This study employed machine learning techniques to predict time series of diffusion curves, generated in Python with NumPy library. The data was structured as a time series to enable efficient model training and evaluation. Various approaches – statistical, neural, and regression-based – were tested to model the diffusion dynamics. Results revealed notable performance differences: regression models achieved the highest accuracy with the lowest error rates, while time series models like ARIMA and TCN performed worse, likely due to difficulties in capturing the process's complexity.

Keywords: Effective diffusion coefficient (D_e); machine learning; time series; regression model

1. Introduction

Methane hazard is directly related to the presence of methane in the rock mass and its emission from mining activities. Despite significant progress in identifying and counteracting this hazard, there is an increasing trend in its intensity in many mining regions. This phenomenon is mainly due to mining at ever-increasing depths, the increased methane sorption capacity of coal seams, and increasing methane pressure in the seam [1,2].

The release of methane from coal is a result of the coal-methane system striving to reach a state of thermodynamic equilibrium. In the context of methane accumulation in the porous coal structure and its emission, the term sorption is often used in a broader sense than the mechanism of the phenomenon itself would suggest. Many researchers treat sorption (and desorption) not

[1] AGH UNIVERSITY OF KRAKOW, FACULTY OF GEOLOGY, GEOPHYSICS AND ENVIRONMENTAL PROTECTION, AL. MICKIEWICZA 30, 30-059 KRAKOW, POLAND

[2] STRATA MECHANICS RESEARCH INSTITUTE OF THE POLISH ACADEMY OF SCIENCES, REYMONTA 27, 30-059 KRAKOW, POLAND

* Corresponding author: skiba@imgpan.pl

only as a phenomenon occurring at the surface, but as a whole set of processes culminating in sorption [3].

Under laboratory conditions, two key parameters are most often used to characterize the properties of the coal-gas system: sorption capacity (*a*) and effective diffusion coefficient (*De*) [4-6]. These parameters are an important complement to *in situ* studies, in the context of assessing methane and ejection hazards in mines. The sorption capacity reflects the ability of the seams to store gas, while the effective diffusion coefficient determines the rate of gas emission.

Among the available test methods, gravimetric techniques play an important role [7], which allow direct determination of the amount of sorbed gas based on the measurement of the increase in sorbent mass after sorbate introduction, under constant pressure and temperature conditions. These methods, despite their numerous advantages [8], are characterized by the high cost of the apparatus and significant time-consumption associated with the need to achieve sorption equilibrium, which limits their practical application in the ongoing prediction of gas and rock outburst hazards.

Determining the diffusion coefficient of a gas involves taking into account many variables and making numerous assumptions, both in experimental and computational approaches. Due to the complexity of the process, the determination of the diffusion coefficient requires simplifications that allow the differential equations describing diffusive transport to be linearized with sorption [9,10]. For this purpose, the following assumptions are made:

- sorption follows Henry's linear isotherm,
- before the initial moment ($t < 0$), the grain is uniformly saturated and is in equilibrium with the surrounding gas concentration,
- at time $t = 0$, there is a step change in the gas concentration in the vicinity of the grain, which initiates the processes of desorption and sorbate transport,
- the coal material is treated as homogeneous, which means ignoring the variation in maceral composition and ash content of grains of different sizes,
- the grain fraction within the isolated grain size class is homogeneous,
- gas is released from a particle with a regular shape (e.g., spherical),
- the process takes place under isothermal conditions,
- the kinetics of sorption and desorption are so fast that their duration can be neglected, considering that the gas release is determined solely by the kinetics of its diffusive transport inside the grain.

2. Machine learning

Machine Learning (ML) is a subfield of Artificial Intelligence (AI) focused on developing algorithms that can learn automatically from data and improve their performance over time without the need for explicit programming [11]. In recent years, ML has been increasingly applied in industries such as mining and energy, where the complexity of physical processes and the high dimensionality of data require more sophisticated analytical approaches [2,5,12]. Traditional methods often fall short in accurately modeling such systems, particularly when non-linear interactions and heterogeneous material structures are involved. In the context of coal mining and methane-related hazards, ML offers promising tools for modeling key processes such as gas sorption and diffusion in coal seams. In particular, ML can be used to predict time-dependent diffusion curves, which are typically derived from laboratory experiments measuring methane

release. These curves provide essential information about the rate and dynamics of gas emissions from the coal matrix – critical for assessing explosion risks and ensuring safe mining operations. Conventional modeling of diffusion behavior usually relies on fitting experimental data using empirical or semi-empirical functions. However, these approaches may struggle to fully represent the complex, multi-factorial nature of gas transport in coal, especially in samples affected by tectonic deformation or structural anomalies. ML overcomes some of these limitations by learning patterns directly from data and uncovering hidden relationships between variables such as porosity, coal rank, pressure, and diffusion dynamics. This research does not address all of these challenges; however, it represents an initial step toward accelerating traditional gravimetric measurements by enabling partial prediction of the diffusion curve. Such an approach may reduce the time required for laboratory testing and open the door to more comprehensive and large-scale analysis of coal seam behavior in future studies. A more advanced subfield of ML, known as Deep Learning (DL), uses multi-layered neural networks capable of learning high-level representations from complex datasets. In the context of methane sorption and diffusion, deep learning models can be trained to predict the full diffusion curve. This allows for more accurate modeling of gas behavior.

In this study, ML techniques were applied to forecast diffusion curves based on sorption data collected under laboratory conditions. The aim is to develop a data-driven method for estimating gas emission potential in various coal seams, thereby complementing traditional methane hazard assessment techniques used in mining. The integration of ML into this field offers opportunities to improve the prediction of methane-related risks, enhance underground safety, and support more informed decision-making in coal exploitation.

In general we can distinguish two types of approaches: one is unsupervised and other is supervised learning.

2.1. Supervised learning

Supervised learning is a fundamental paradigm in machine learning that involves training predictive models using datasets where each observation is associated with both an input and an output [13]. In other words, for every sample in the training data, the corresponding desired outcome – often referred to as a "label" – is known in advance. This allows the algorithm to learn a mapping from input features to output values.

Let us formally define the data structure involved in supervised learning:

- Let $X = \{x_1, x_2, ..., x_n\}$ represent the set of input features, where each $x_i \in \mathbb{R}^d$ is a d-dimensional feature vector corresponding to the i-th observation.
- Let $Y = \{y_1, y_2, ..., y_n\}$ denote the set of target values, where each y_i is either a continuous value (in the case of regression tasks, $y_i \in \mathbb{R}$) or a categorical class label (for classification task) $y_i \in \{c_1, c_2, ..., c_k\}$.
- The full dataset is expressed as a set of paired observations $M = (X, Y)$ comprising n instances in total.

Typically, the dataset M is split into two disjoint subsets:

- A training set $M_{TR} \subset M$, used to fit the model
- A test set $M_{TE} \subset M$ used to evaluate model performance, such that $M_{TR} \cup M_{TE} = M$ and $M_{TR} \cap M_{TE} = \varnothing$.

The main goal of supervised learning is to identify a function $f: X \rightarrow Y$ that best approximates the relationship between inputs and outputs. This function should generalize well to new, unseen data. Predictive performance is evaluated using a loss function $L(y,\hat{y})$, where $\hat{y} = f(x)$ denotes the model's prediction and y represents the true target value. The loss function measures the discrepancy between predicted and true values.

Common loss functions include [14]:

1. For regression problems, a typical choice is the Mean Squared Error (MSE):

$$MSE = L\left(y,\hat{y}\right) = \frac{1}{n}\sum_{i=1}^{n}\left(y_i - \widehat{y_i}\right)^2 \tag{1}$$

2. For classification tasks, a widely used loss is the cross-entropy loss, defined as:

$$L\left(y,\hat{y}\right) = -\sum_{i=1}^{n} y_i \log \widehat{y_i} \tag{2}$$

The learning process involves optimizing the model parameters θ in such a way that the loss function is minimized. This is commonly achieved using optimization algorithms like gradient descent, which updates the parameters iteratively:

$$\theta^{(t+1)} = \theta^{(t)} - \eta \nabla_{\theta} L\left(y,\hat{y}\right) \tag{3}$$

where η is the learning rate, a hyperparameter that controls the step size in each iteration.

Supervised learning techniques are widely applied across various domains, offering powerful tools for predictive modeling. These techniques can be divided into two main categories: classification and regression, each serving distinct purposes [11]:

- In classification tasks, the target variable y is categorical, and the model learns to assign inputs to one of several predefined categories. Applications include image recognition, such as identifying objects or faces in images, fraud detection where financial transactions are classified as fraudulent or legitimate based on historical patterns, spam filtering to categorize emails as spam or non-spam based on content and sender, and medical diagnosis where patient data is classified to detect diseases, aiding doctors in decision-making.
- In regression tasks, y is a continuous numerical value, and the model predicts a numeric outcome based on input features. Applications of regression include predicting housing prices by estimating home values based on features like size, location, and amenities, energy consumption forecasting for predicting future energy usage of buildings or industrial systems, and financial risk modeling to estimate risk or returns in investment portfolios. By learning from labeled datasets, supervised learning models provide a robust method for making predictions about unknown data, making them essential for data-driven decision-making in fields ranging from finance to healthcare.

2.2. Unsupervised learning

Unsupervised learning is a category of machine learning algorithms where the model analyzes data without predefined labels and independently uncovers hidden patterns, structures, and relationships within the dataset [15]. Unlike supervised learning, where labeled examples

are provided (input-output pairs), unsupervised learning algorithms do not rely on labeled data and instead explore the data for inherent groupings, correlations, or structures. In this type of learning, the model works with raw, untagged data and attempts to learn the underlying structure or distribution of the data by detecting similarities or differences between the data points. Let $X = \{x_1, x_2, ..., x_n\}$, where $x_i \in \mathbb{R}^d$, represent a dataset consisting of n objects, each having d features. These objects can be anything from customer records to sensor readings, and the features represent the attributes of each object, such as age, income, or time of occurrence. In unsupervised learning, the aim is to learn a function $f : X \rightarrow Z$ where Z is a new representation of the data that reflects the hidden structure, such as clusters or reduced dimensions. This transformation allows the model to discover previously unknown relationships and patterns that are not explicitly visible in the original dataset. The goal of unsupervised learning is to find this structure within the data, which may include clustering similar data points into groups (e.g., customer segmentation) or dimensionality reduction (e.g., reducing the number of features while retaining the most important information). Such transformations enable more efficient data analysis, helping identify trends, anomalies, or features that could be used for further analysis or decision-making [16]. For example, unsupervised learning is commonly used in data exploration, anomaly detection, and feature extraction, and is widely applied in areas like market analysis, recommendation systems, and natural language processing [17].

3. Material and methods

3.1. Theoretical model

To test the machine learning models on a controlled dataset, the diffusion process was modeled using an analytical solution to the diffusion equation for ideal boundary conditions. The equation describing the change in mass over time has the form [18]:

$$\frac{M(t)}{M_\infty} = 1 - \frac{6}{\pi^2}\sum_{1}^{\infty}\exp\left(-\frac{n^2\pi^2 D_e t}{R^2}\right) \tag{4}$$

where: M_∞ [g] is the total mass of gas accumulated in the grains, $M(t)$ [g] is the mass of gas accumulated at time t, R [cm] is the equivalent grain radius.

3.2. Simulation of theoretical diffusion curves

To simulate synthetic diffusion curves, various parameters were randomly selected within defined realistic ranges. Below is a summary of the parameters and their corresponding values used for generating the diffusion curves:

The general process for diffusion curve generation is as follows:

1. Time Grid Creation: A time grid was created to simulate the diffusion process. The time range spans from 0 to n time units, with the grid's time steps defined by the user's input. This time grid was generated using linear steps, which was done using the NumPy library. The flexibility of NumPy allowed for efficient handling of large datasets and precise time-step calculations, making it suitable for creating the time intervals necessary for simulating the diffusion curves.

TABLE 1

Summary of the parameters and their corresponding values used for generating the diffusion curves

Parameter	Description	Value/Range
$M(\infty)$	Asymptotic methane sorption capacity in coal	Normal distribution: Mean = 2.1 cm^3/g, Standard Deviation = 0.6 cm^3/g
D_e (Diffusion Coefficient)	Diffusion rate of methane through the coal matrix	Uniform distribution: From 2×10^{-10} to 2×10^{-8} cm^2/s
R (Particle Radius)	Radius of the coal particles	Fixed value: 0.011157 cm

2. Diffusion Curve Calculation: For each randomly selected set of parameters, the diffusion curve was computed using the analytical solution, with the first 150 terms of the series considered. This step strikes a balance between calculation accuracy and computational efficiency. The use of NumPy allowed for optimized numerical operations, particularly when summing the terms of the series efficiently. The series calculation benefited from the array-based operations provided by NumPy, which is crucial for managing large numerical computations like these.
3. Repeating the Process: This process was repeated for 5000 random parameter sets, which generated a wide variety of diffusion curves. The computed diffusion curves were stored as arrays, with time and corresponding methane release values, using Pandas for tabular storage. Pandas enabled easy manipulation and structuring of the generated data, facilitating further analysis and ensuring that the data was organized in a way that was both efficient and easily accessible.

The core calculations and operations were executed using Python. Libraries like NumPy provided the computational power necessary to handle the mathematical operations, while Pandas was used to manage and store the results in an organized format for subsequent analysis.

4. Machine learning for time series prediction of the diffusion curve

In the context of this study, we compared two distinct forecasting approaches: a simpler, traditional linear regression method and a more complex deep learning model using a Temporal Convolutional Network (TCN) available in Darts library [19]. The primary goal is to evaluate how these methods perform on the same diffusion curve data and to analyze their respective strengths and weaknesses. The regression model represents a straightforward method, using historical time-series data to make predictions based on linear relationships. In contrast, the TCN, a more advanced method, captures complex, non-linear dependencies in the data through multiple layers of convolution, enabling it to model intricate temporal patterns.

The data science process involved following steps

- Data preprocessing once the diffusion curves are generated, we preprocess the dataset by performing the following steps:
- Data independent variable conversion – the data index (here seconds) is converted into a time delta format and set as the index of the DataFrame.

- Resampling – the time series data is resampled at 0.5-second intervals, and any missing values are interpolated to ensure a consistent time series. This step ensures that the time resolution is uniform across all curves.
- Index re-conversion – the time index is then converted into a total seconds format, which is used as the new time index. The time index is then set as the main index for the dataset. It allowed for better results visualization.

The resulting dataset is prepared for model training, where individual curves are treated as time series components.

Time Series Preparation is another major step in the process. The data is converted into time series format using the Darts library, which facilitates time series forecasting. For each of the 1000 curves, the data is split into training and validation sets. Data for seconds 1000-2000 were used for validation, while the data for 0-1000 seconds was used for training. The training data is further organized into a list of time series components. Each time series corresponds to one curve from the original dataset, making it easier to handle each curve's forecast using the global model.

In this study we used two forecasting models to predict future values of the diffusion curves:

1. Linear Regression Model: A simple regression model with a lag of 12 time steps is used to forecast future values of the diffusion curves. This model is trained on the past values of each time series component.
2. Temporal Convolutional Network (TCN): A more advanced model, the TCN, is used to capture the temporal dependencies in the data. The TCN is configured with an input chunk length of 16 time steps, an output chunk length of 8 time steps, and is trained for 40 epochs. The model uses dropout and weight normalization to improve generalization.

Both models are trained with the global assumptions, and predictions are generated for each curve. The forecasts are then stored for later evaluation.

To better understand the diffusion curves half-asymptote time was calculated. The point at which the curve reaches half of its final asymptotic value. This is computed by identifying the time at which the curve's value is closest to half of the final value.

To enhance the interpretability of the TCN model's predictions, a smoothing step is applied to the raw predicted values. This smoothing helps mitigate high-frequency noise and provides a clearer visualization of the long-term trends in the data. A uniform filter with a window size of 30 was used to perform the smoothing. The smoothed predictions are then stored as a new TimeSeries object and plotted alongside the original TCN predictions. The smoothed predictions are displayed in red for easy differentiation.

To evaluate the accuracy of the predictions, several performance metrics are computed on the validation set for each model [20]:

- Mean Absolute Error (MAE): Measures the average magnitude of the errors in a set of predictions, without considering their direction.
- Root Mean Square Error (RMSE): Measures the square root of the average squared differences between predicted and actual values, giving higher weight to larger errors.
- Mean Absolute Percentage Error (MAPE): Measures the accuracy of predictions as a percentage of the actual values.
- Symmetric Mean Absolute Percentage Error (SMAPE): A variation of MAPE that accounts for the magnitude of both predicted and actual values symmetrically.

These metrics allow for an objective comparison of the performance of the regression and TCN models.

After the predictions are made, the results are saved to an Excel file for further analysis. In addition to the predictions, metadata for each selected curve is saved, including the parameters used to generate the curve, as well as the extracted features.

5. Results and discussion

Fig. 1 provides a combined visualization of the diffusion curves for six synthetic simulations of representative coal samples, modeled using both traditional regression-based ML methods and a Temporal Convolutional Network (TCN) available in Darts library. M_TXXX refers to the synthetic diffusion curve derived from parameter set no. XXX, as presented in Fig. 1. Each subplot displays the diffusion curve $M(t)$, divided into training (gray) and testing (green) time series. The predicted curve is presented in two approaches: a linear regression model (black dashed line) and the TCN model (light red line), with an additional smoothed TCN output (solid red line). The vertical blue dashed line indicates the half-asymptotic time. From a descriptive standpoint, several observations can be made:

1. All samples follow the general characteristic sigmoidal trend of diffusion-controlled sorption, with an initially rapid uptake that slows asymptotically as equilibrium is approached.
2. The regression model generally provides a good, smooth, and stable prediction curve that aligns well with the testing data. Only one not significant part with overprediction for M_T242 can be seen on the final edge in the testing.

The complex architecture of the Temporal Convolutional Network (TCN) enables it to capture intricate, nonlinear patterns in the data; however, this flexibility sometimes leads to overfitting or misrepresentation of the actual sorption behavior in the testing region. Despite extensive tuning of the model parameters – including input/output chunk size, number of filters, and kernel size – to achieve the smoothest possible output, the TCN still tends to predict a residual cyclical component. Interestingly, the magnitude of this oscillatory behavior appears to increase across successive samples, which may be partially attributed to the fixed chunk size used for training, as well as inherent characteristics of the TCN architecture that favor the learning of local temporal patterns. It is worth noting that not all prediction curves are equally affected by this phenomenon. The degradation of prediction quality is most pronounced in samples M_T242, M_T241, and M_T68, where the TCN significantly overestimates the last part of the curve. A slight underestimation is visible in M_T371 and M_T337 in the central part of the testing curve. Notably, smoothing helps to mitigate these issues to some extent, leading to improved prediction curves in all cases.

Fig. 2 presents a comparative analysis of the Regression and Smoothed Temporal Convolutional Network (TCN) models using four standard error metrics: MAE, RMSE, MAPE, and SMAPE. The evaluation is conducted across six chosen curves types: M_T371, M_T242, M_T68, M_T337, M_T241, and M_T10. In all four subplots, the Regression model consistently achieves significantly lower error values than the Smoothed TCN model, indicating superior predictive performance and generalization capability. In the MAE plot (top-left), the Regression model maintains error values predominantly below 10^{-2} across all curves, whereas the Smoothed TCN model records substantially higher errors, typically ranging from 10^{-2} to nearly 10^{-1}. A similar

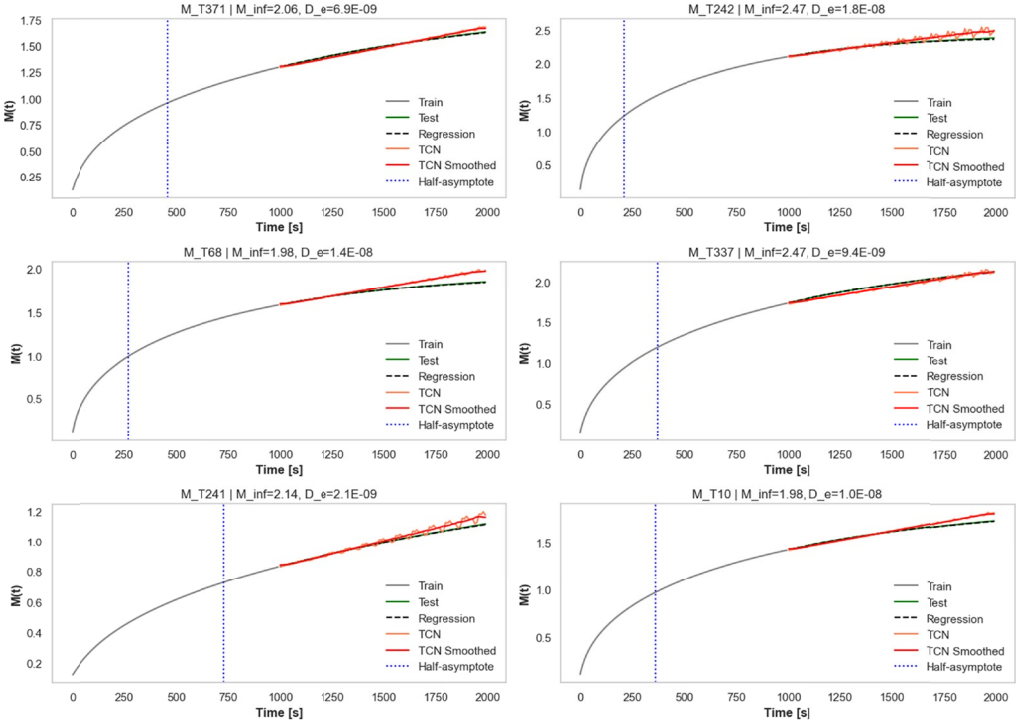

Fig. 1. Diffusion curves predictions. Each subplot shows the training (gray) and testing (green) time series, with predictions from the regression model (black dashed line) and TCN model (light red line), along with a smoothed TCN output (solid red line). The vertical blue dashed line marks the half-asymptotic time

pattern is observed in the RMSE plot (top-right), where the Regression model again shows much lower root mean squared errors, with particularly notable improvements for curves M_T242, M_T68, and M_T241, where the Smoothed TCN model's RMSE approaches or exceeds 10^{-1}. The performance gap is even more pronounced in the percentage-based metrics. In the MAPE plot (bottom-left) and SMAPE plot (bottom-right), the Regression model yields values well below 10^{-1}, often near or under 10^{-2}, while the Smoothed TCN consistently exhibits much higher percentage errors, indicating less reliable performance in proportional terms. These results clearly illustrate that the Regression model outperforms the Smoothed TCN model by a substantial margin across all tested metrics and curve scenarios. Compared to the baseline TCN model without smoothing, the application of the smoothing filter yielded, on average, a 30% reduction in MAE. Nevertheless, the error remains relatively high, suggesting that while smoothing alleviates the problem to some extent, it does not fully overcome the model's predictive limitations. The significant discrepancy in error magnitudes, especially under the logarithmic scale used in the plots, underscores the robustness and precision of the linear regression approach, making it a more reliable choice for the given prediction task.

Fig. 3 illustrates the relationship between the variable D_e and MAE for two models: the Regression model (left) and the Smoothed TCN model (right). Both scatter plots display D_e on the x-axis and MAE on the y-axis, with each point representing a specific case from 6 curves

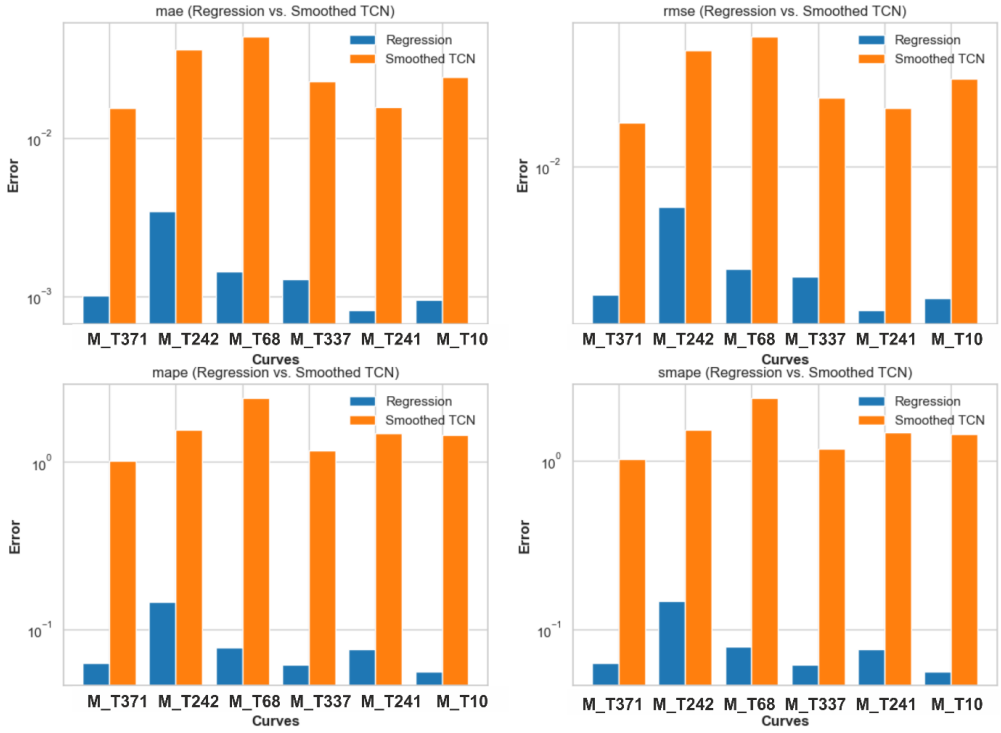

Fig. 2. Regression vs. Smoothed TCN: Evaluation Using MAE, RMSE, MAPE, and SMAPE

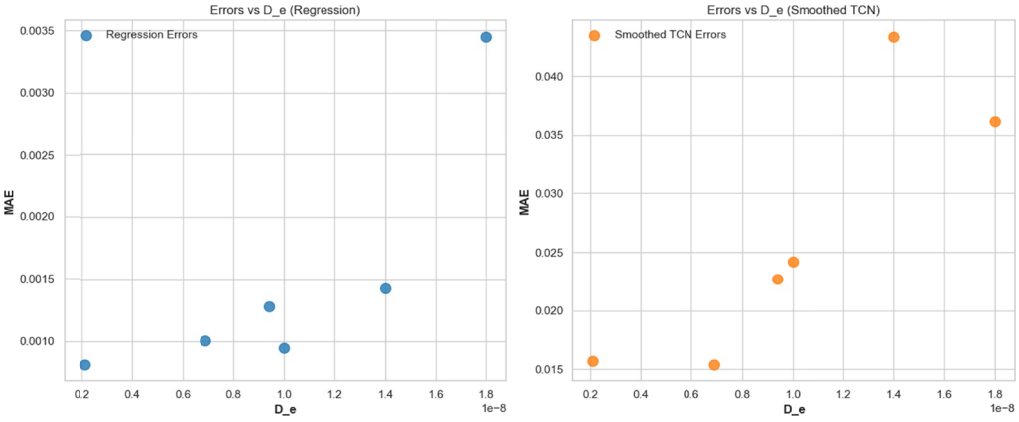

Fig. 3. MAE errors for Regression (blue) and smoothed TCN (orange) as function of D_e

presented in Fig. 1. In the Regression model plot, MAE values remain consistently low, mostly below 0.0015, with no clear trend relative to D_e. This indicates that the model maintains stable and accurate performance across different D_e values, demonstrating robustness and low sensitivity to variations in the input. On the other hand, the Smoothed TCN model exhibits significantly higher

MAE values, ranging from approximately 0.015 to over 0.04. A positive correlation between D_e and MAE is observed, suggesting that the model's performance deteriorates as D_e increases. This indicates that the Smoothed TCN is more sensitive to variations in D_e, leading to less consistent and less reliable predictions. The Regression model outperforms the Smoothed TCN model in terms of MAE, showing higher stability and accuracy across varying conditions.

Fig. 4 shows a similar set of graphs for $M(\infty)$. In the left panel, the regression model exhibits generally low MAE values, all below 0.0035. The minimum error is observed near $M(\infty) \approx 2.15$, with a slight increase in error observed at both ends of the $M(\infty)$ spectrum. Notably, the highest error appears around the edge $M(\infty)$ values suggesting potential model instability at the edges of parameter space (similar for D_e). In the right panel, the smoothed TCN model presents significantly higher MAE values in comparison to the regression model, ranging approximately from 0.015 to 0.042. Although a minimum in MAE is also observed around $M(\infty) \approx 2.15$, the variation is more pronounced, and the error increases sharply at both the lower and upper bounds of $M(\infty)$. This could indicate a higher sensitivity of the TCN model to the value of $M(\infty)$, or an artifact introduced by the smoothing process. The comparison suggests that while both models exhibit a performance optimum around $M(\infty) = 2.15$, the regression model demonstrates higher robustness and lower absolute error across the tested $M(\infty)$ range, making it potentially more reliable in applications requiring high predictive accuracy.

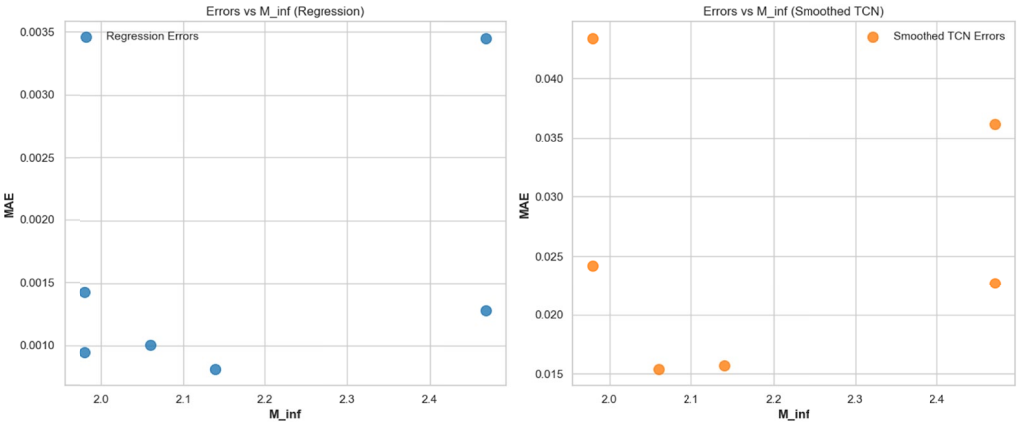

Fig. 4. MAE errors for Regression (blue) and smoothed TCN (orange) as function of $M(\infty)$

6. Conclusions

This study provides evidence supporting the use of ML methodologies in significantly accelerating the prediction of diffusion-controlled sorption curves. Specifically, the application of regression-based models has proven to be not only computationally efficient but also remarkably accurate across a range of representative synthetic datasets. When compared to more complex architectures, such as the Temporal Convolutional Network implemented via the Darts library, the linear regression model consistently outperforms in terms of all considered error metrics, including MAE, RMSE, MAPE, and SMAPE.

The comparative analyses indicate that while the TCN architecture is theoretically well-suited for capturing nonlinear and temporally-dependent features, in practice it tends to introduce oscillatory artifacts in the predicted signal. These are particularly evident in the later stages of the sorption curves, where the network frequently overestimates values, even after extensive hyperparameter tuning and smoothing procedures. Furthermore, the TCN's sensitivity to parameters such as the D_e and the $M(\infty)$ suggests that its generalization capability is inferior to that of the linear model within the examined parameter space.

Both models exhibit reduced predictive performance near the boundaries of the training distribution, particularly for extreme values of D_e and $M(\infty)$, which are indicative of the outer envelope of physically relevant sorption behaviors. This pattern underscores the necessity for caution when applying data-driven models in extrapolative regimes, and emphasizes the importance of confining predictions to regions of the input space well represented in the training set.

From a computational standpoint, the efficiency gains afforded by the regression approach are substantial. For the selected experimental setting (35 training epochs), the linear regression model completes training approximately 300 times faster than the TCN counterpart. This efficiency, combined with superior accuracy, renders simple models highly suitable for deployment in time-sensitive or resource-constrained applications.

7. Recommendations and Directions for Future Work

a) Restrict the application of ML-based prediction models to the central region of the training data distribution, thereby avoiding the elevated uncertainty associated with boundary values of D_e and $M(\infty)$.
b) Adopt simple regression models (e.g., linear regression) as a default approach for modeling diffusion curves from, particularly in scenarios where computational resources are limited and rapid inference is required.
c) Undertake future studies using real-world experimental datasets to verify the transferability of the observed performance trends from synthetic to empirical settings.
d) Investigate domain-specific neural architectures, tailored to the physical characteristics of sorption kinetics, which may outperform general-purpose TCN models in both accuracy and robustness. Preferably with a broader context for a multi-model network.
e) Explore adaptive training strategies and uncertainty quantification frameworks, particularly for neural models, to enhance their interpretability and reliability in edge-case predictions.

8. Limitations of using synthetic data

The results presented in this study are based exclusively on synthetic diffusion curves generated from theoretical boundary conditions. While this approach offers the advantage of controlled parameter variation and facilitates proof-of-concept testing, it also introduces limitations regarding the transferability of the models to empirical settings. In particular, synthetic data do not fully capture the heterogeneity of coal structure, the influence of impurities, or experimental uncertainties that arise under laboratory and in situ conditions. As such, the predictive perfor-

mance reported here should be interpreted as an upper bound rather than a direct reflection of real-world accuracy. Future work will address this limitation by validating the proposed models on laboratory measurements of methane desorption and diffusion curves. Our plan is to reproduce experimental runs corresponding to selected synthetic parameter sets, enabling a one-to-one comparison between modeled and measured outcomes. This will provide a critical test of the models' robustness and allow for the refinement of both regression and deep learning approaches in the context of realistic coal-gas systems.

Acknowledgements

The work was financed within the framework of the statutory research of the Strata Mechanics Research Institute of the Polish Academy of Sciences and the AGH University of Krakow, Faculty of Geology, Geophysics and Environmental Protection (Program "Excellence initiative – research university" for the AGH University of Krakow).

References

[1] B. Dutka, K. Godyń, Predicting variability of methane pressure with depth of coal seam. Przem. Chem. **97** (8), 1344-1348 (2018). DOI: https://doi.org/10.15199/62.2018.8.20

[2] M. Skiba, B. Dutka, M. Młynarczuk, MLP-Based Model for Estimation of Methane Seam Pressure. Energies **14**, 7661 (2021). DOI: https://doi.org/10.3390/en14227661

[3] M. Vorokhta, M. Svabova, Z. Weishauptova, Co2 Kinetic Sorption Rate Study Depending On Pressure And Concentration Step 16th International Multidisciplinary Scientific Geoconference (SGEM 2016) Energy And Clean Technologies Conference Proceedings, SGEM 2016, Vol. III, pp. 107-114 (2016).

[4] S. Chattaraj, D. Mohanty, T. Kumar, G. Halder, Thermodynamics, kinetics and modeling of sorption behaviour of coalbed methane – A review. Journal of Unconventional Oil and Gas Resources **16**, 14-33 (2016). DOI: https://doi.org/10.1016/j.juogr.2016.09.001

[5] M. Skiba, M. Młynarczuk, Estimation of Coal's Sorption Parameters Using Artificial Neural Networks. Materials **13**, 5422 (2020). DOI: https://doi.org/10.3390/ma13235422

[6] N. Skoczylas, A. Pajdak, K. Kozieł, L. Braga, Methane emission during gas and rock outburst on the basis of the unipore model. Energies **12** (10), 1999 (2019). DOI: https://doi.org/10.3390/en12101999

[7] L. Zhang, T.X. Ren, N. Aziz, A stydy of laboratory testing and calculation methods for coal sorption isotherms. J. Coal Sci. Eng. **19** (2), 193-202 (2013). DOI: https://doi.org/10.1007/s12404-013-0214-4

[8] A. Saghafi, M. Faiz, D. Roberts, CO2 storage and gas diffusivity properties of coals from Sydney Basin, Australia. Int. J. Coal Geol. **70**, 240-254 (2007). DOI: https://doi.org/10.1016/j.coal.2006.03.006

[9] M. Pillalamarry, S. Harpalani, S. Liu., Gas diffusion behavior of coal and its impact on production from coalbed methane reservoirs. Int. J. Coal Geol. **86**, 342-348 (2011). DOI: https://doi.org/10.1016/j.coal.2011.03.007

[10] K. Kozieł, A. Gajda, M. Skiba, N. Skoczylas, A. Pajdak, Influence of Grain Size and Gas Pressure on Diffusion Kinetics and CH4 Sorption Isotherm on Coal. Arch. Min. Sci. **69**, 1, 3-23 (2024). DOI: https://doi.org/10.24425/ams.2024.149824

[11] C.M. Bishop,. Pattern recognition and machine learning. Springer (2006). DOI: https://doi.org/10.1007/978-0-387-31073-2

[12] S.K. Singh, D. Chakravarty, Efficient and Reliable Prediction of Dump Slope Stability in Mines using Machine Learning: An in-depth Feature Importance Analysis. Arch. Min. Sci. **68** (4), 685-706 (2023). DOI: https://doi.org/10.24425/ams.2023.148157

[13] J.E. van Engelen, H.H Hoos, A survey on semi-supervised learning. Mach Learn **109**, 373-440 (2020). DOI: https://doi.org/10.1007/s10994-019-05855-6

[14] I. Goodfellow, Y. Bengio, A. Courville., Deep learning. MIT Press (2016).

[15] M. Zareba, T. Danek, M. Stefaniuk, Unsupervised Machine Learning Techniques for Improving Reservoir Interpretation Using Walkaway VSP and Sonic Log Data. Energies **16** (1), 493 (2023). DOI: https://doi.org/10.3390/en16010493

[16] I.H. Sarker, Machine Learning: Algorithms, Real-World Applications and Research Directions. SN Comput. Sci. **2**, 160 (2021). DOI: https://doi.org/10.1007/s42979-021-00592-x

[17] E. Alpaydin, Introduction to machine learning (3rd ed.). MIT Press (2014).

[18] J. Crank, The mathematics of diffusion. Clarendon Press, England (1975).

[19] J. Herzen, F. Lässig, S.G. Piazzetta, T. Neuer, L. Tafti, G. Raille, T. Van Pottelbergh, M. Pasieka, A. Skrodzki, N. Huguenin, M. Dumonal, J. Kościsz, D. Bader, F. Gusset, M. Benheddi, C. Williamson, M. Kosinski, M. Petrik, G. Grosch, Darts: User-friendly modern machine learning for time series (2022) (arXiv:2110.03224v3 [cs.LG]). DOI: https://doi.org/10.48550/arXiv.2110.03224

[20] R.J. Hyndman, A.B. Koehler, Another look at measures of forecast accuracy. Int. J. Forecast. **22** (4), 679-688 (2006). DOI: https://doi.org/10.1016/j.ijforecast.2006.03.001