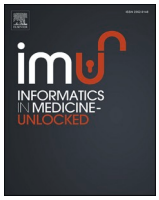
Informatics in Medicine Unlocked 47 (2024) 101471

Contents lists available at ScienceDirect

Informatics in Medicine Unlocked

journal homepage: www.elsevier.com/locate/imu



Advancing pneumonia virus drug discovery with virtual screening: A cutting-edge fast and resource efficient machine learning framework for predictive analysis

Ochin Sharma [a], G.S. Pradeep Ghantasala [b], Iacovos Ioannou [c,*], Vasos Vassiliou [c]

[a] Chitkara University Institute of Engineering & Technology, Chitkara University, Punjab, India
[b] Department of Computer Science and Engineering, Alliance University, Bengaluru, India
[c] University of Cyprus and CYENS Centre of Excellence, Nicosia, Cyprus

ARTICLE INFO

Index terms:
Medicine
Industry
Drugs
Employability
Machine learning

ABSTRACT

Pneumonia, a severe respiratory infection characterized by a significant morbidity and fatality rate, afflicts many individuals globally. The demand for highly effective antiviral medications has experienced a surge because of the emergence of novel pneumonia viruses, such as the COVID-19 coronavirus. Due to their inherent time and cost constraints, conventional drug development strategies sometimes need to be more manageable. Exploring alternative approaches is crucial to identifying and establishing effective therapy choices. This work introduces a computational methodology for analyzing the chemical space of medications targeting the pneumonia virus, employing Python-based data mining tools. Using computer-aided analysis in drug molecules aims to enhance the efficiency of identifying and evaluating potential new therapeutic candidates using Machine Learning (ML). The research successfully discovered two therapeutic compounds by utilizing the Bayesian Ridge approach, which is the most accurate with the least mean squared error, is less computationally expensive in terms of power, memory, and CPU, and is the fastest of the investigated approaches. It discovered the CHEMBL433378 and CHEMBL93653, with promising docking scores of −4.3 and −4.2, respectively. Additionally, both molecules demonstrated significant inhibitory activity against their respective targets, as seen by their IC50 values of 0.0018 and 0.001. Both compounds meet the criteria for the B. Mann Whitney U Test and Lipinski test.

1. Introduction

Exploring alternative approaches is crucial to identifying and establishing effective therapy choices. This work introduces a computational methodology for analyzing the chemical space of medications targeting the pneumonia virus, employing Python-based data mining tools. Initially, it is crucial to amass an extensive assortment of chemical compounds derived from several origins, including databases and chemical repositories. The exploration of chemical space is predicated upon the utilization of these compounds. To enhance the quality and relevance of the dataset, it is necessary to do data pre-processing tasks such as filtering, normalization, and feature extraction. These operations can be accomplished using Python tools and methods [1,2].

The requirement to find effective remedies for diverse medical conditions propels the continuous advancement of the pharmaceutical industry. Pneumonia, a severe respiratory illness, is a significant worldwide health concern due to its substantial morbidity and fatality rates. The significance of potent antiviral medications has become increasingly apparent considering the emergence of novel pneumonia viruses such as the COVID-19 coronavirus. Exploring alternative approaches to identify and evaluate new drug candidates is imperative, as conventional drug development techniques involve laborious and costly experimental procedures. In recent years, silico approaches, which rely on computer-based procedures and data analysis, have gained significant prominence as valuable tools for accelerating the drug discovery process [3–6].

A key component of drug discovery is analyzing the chemical space, which includes many chemical substances and their attributes. By searching this chemical space, researchers can find compounds with the appropriate properties, such as antiviral activity against pneumonia viruses. Data analysis is essential in this endeavor because it allows for extracting valuable insights from vast and complex databases. Python, a

* Corresponding author.
E-mail addresses: ochin.sharma@chitkara.edu.in (O. Sharma), ggspradeep@gmail.com (G.S.P. Ghantasala), ioannou.iakovos@ucy.ac.cy (I. Ioannou).

https://doi.org/10.1016/j.imu.2024.101471
Received 23 December 2023; Received in revised form 6 March 2024; Accepted 7 March 2024
Available online 22 March 2024

flexible programming language, has become more well-liked in the scientific community due to its extensive library of resources and tools for machine learning and data analysis. In this paper, we suggest an in-silico method for examining the chemical space for pneumonia viral medications by using Python-based data analysis approaches. We aim to speed up prospective drug candidates' identification and evaluation using computational tools and algorithms. We can prioritize compounds that are most likely to display antiviral activity by using the power of data analysis to find hidden patterns and correlations within the chemical universe [7,8].

Data cleansing and data preparation techniques such as filtering, normalization, and feature extraction have improved the chemical dataset's quality and relevance in this examination. Further, we use various data analysis techniques to explore the chemical space and find compounds with the desired features, such as clustering, similarity analysis, and machine learning algorithms [9,10]. These methods allow us to categorize molecules with comparable properties, pinpoint molecules with different structural traits, and forecast the probable effectiveness of new drugs against pneumonia viruses. Robust tools for data processing, statistical analysis, machine learning, and molecular visualization are available thanks to Python's diverse ecosystem of libraries, which includes NumPy, Pandas, Scikit-learn, and RDKit. These tools make it easier to quickly collect and analyze chemical data, enabling researchers to make defensible choices about prospective medication candidates [9,11].

This can speed up the drug development process by using data analysis in a silico environment, leading to the discovery of novel pneumonia virus drugs. Unlike conventional experimental procedures, this strategy has the advantage of being less expensive and time-consuming. The goal of our research is to aid in creating potent antiviral medications to prevent pneumonia infections with the use of figures to identify the most appropriate medicine targeting the enhancement of overall health outcomes [11,12]. The research innovates with a reliable framework for drug discovery for Pneumonia disease. The entirety of the work has been partitioned into multiple portions.

This research introduces a groundbreaking framework that seamlessly integrates machine learning and virtual screening for predictive computational strategies in pneumonia virus drug discovery. The novelty lies in the unique amalgamation of advanced machine learning algorithms with virtual screening techniques, paving the way for more efficient, accurate, and rapid identification of potential drug candidates. This innovative energy and resource-efficient approach marks a significant leap in computational drug discovery, especially for combating pneumonia viruses, by harnessing the power of data-driven insights and computational efficiency. The contribution of the paper is the following.

- Innovative Integration: Our work uniquely blends machine learning and virtual screening, significantly enhancing the efficacy of drug discovery processes.
- Advanced Predictive Modeling: We utilize sophisticated machine learning models, enabling precise predictions of drug effectiveness against pneumonia viruses, thereby marking a leap forward in predictive analytics.
- Efficiency in Drug Screening: Our approach markedly reduces the time and resources traditionally required for drug candidate identification, streamlining the screening process through computational methods.
- Enhanced Accuracy: By integrating machine learning into virtual screening, we significantly improve the process's accuracy, yielding more reliable and promising drug discovery outcomes.
- Pioneering Approach in Pneumonia Virus Drug Discovery: The methodology we propose is specifically tailored to address the unique challenges inherent in pneumonia virus drug discovery, positioning our work at the forefront of this field.
- Data-Driven Insights: Our framework heavily relies on extensive data analysis, leveraging these insights to refine and optimize the entire drug discovery process.
- Scalable Framework: We have developed a versatile and scalable solution designed to be adaptable across various drug discovery endeavors, not limited to just pneumonia viruses.
- Emphasis on Green Energy and Rapid Decision-Making: Our research also introduces a novel focus on green energy and resource optimization, alongside the capability for rapid decision-making in time-sensitive scenarios. This dual emphasis is particularly relevant in high-stakes environments such as emergency medicine and public health crises, where therapeutic agents' efficient and timely development can be crucial. Our framework's consideration of environmental sustainability in computational processes, alongside its focus on speed and accuracy, addresses an emerging and critical gap in the research landscape.

Following the Introduction section, the subsequent section includes a Literature Review outlining, identifying, and synthesizing relevant research and prior work related to our investigation. The third section articulates the Problem Statement, defining the specific challenges, questions, or issues to be addressed. The fourth section explains the Methodology employed in developing the framework, outlining the research approach, data collection methods, tools, and techniques used. Next, the fifth section presents an in-depth analysis of the conducted experiments, including the experimental setup, data analysis procedures, and obtained results. Finally, the sixth section draws the research to a Conclusion, summarizing key findings, implications, and contributions.

2. Background work

This section provides the background work regarding our examination (e.g., what a lazy regression is). It also provides the techniques for preparing the data to become a dataset for our approach.

In machine learning, an essential task is the examination of the data. Thus, data must be examined, cleaned, transformed, and interpreted to find significant patterns, insights, and information. It is essential in several disciplines, including business, science, healthcare, finance, and social sciences. In this examination, we used the Principal Component Analysis (PCA) for dimension reduction and "Abide pIC50" for molecule characterization, which we will explain in the following paragraphs.

Principal Component Analysis (PCA) [13]: is a widely used dimensionality reduction technique and data analysis method in statistics and machine learning. It aims to transform high-dimensional data into a lower-dimensional representation while preserving as much of the original variance as possible. PCA achieves this by identifying a set of new orthogonal axes, called principal components, along which the data varies the most. These principal components are linear combinations of the original features, and they are ranked in order of the amount of variance they explain. In the following list, we provide the steps of PCA.

1. Data Standardization: To perform PCA, it's essential to standardize the data to have zero mean and unit variance. This step ensures that all features have the same scale, preventing some variables from dominating the analysis due to their larger magnitude with the use of Eq. (1).

$$X_{\text{standardized}} = \frac{X - \text{mean}(X)}{\text{std}(X)} \tag{1}$$

2. Covariance Matrix Calculation: Calculate the covariance matrix of the standardized data. The covariance matrix shows how feature pairs vary and provides information about the relationships between features using Eq. (2).

$$Cov(X,Y) = \frac{\sum_{i=1}^{N}(X_i - \overline{X})(Y_i - \overline{Y})}{(N-1)} \quad (2)$$

3. Eigen decomposition: Calculate the eigenvectors and eigenvalues of the covariance matrix (as shown in Eq. (3)). Eigenvectors represent the directions in the original feature space along which the data varies the most. In contrast, eigenvalues indicate the amount of variance explained by each eigenvector.

$$C \cdot v = \lambda \cdot v \quad (3)$$

4. Selecting Principal Components: Decide how many principal components to retain based on the variance you want to preserve in the reduced-dimensional data.
5. Projection: Project the original data onto the selected principal components to obtain the reduced-dimensional data.

If **V** represents the matrix of selected eigenvectors, and **X** represents the standardized data matrix, the projection **Z** is given by Eq. (4):

$$Z = X \cdot V \quad (4)$$

Applications of PCA: PCA is widely used for various purposes, such as data compression, noise reduction, visualization, and feature selection. It helps simplify complex datasets while preserving essential patterns and reducing computational complexity in subsequent analyses. The principal components obtained through PCA can also provide insights into the underlying structure of the data and reveal relationships between features.

The term "Abide pIC50" pertains to the pIC50 value linked to bioactivity data in drug discovery and molecular pharmacology. Let us engage in a comprehensive analysis of the concept to enhance our comprehension [14]. The pIC50 value is determined by taking the negative logarithm (base 10) of the IC50 value. The acronym IC50 represents the term "half maximal inhibitory concentration." The term "half-maximal inhibitory concentration" (IC50) refers to a drug or inhibitor concentration necessary to inhibit a biological activity or response by 50%. In pharmaceutical research and development, the phenomenon under consideration may manifest as suppressing enzymatic activity or blocking receptor function. The IC50 value is a widely employed quantitative measure for assessing the effectiveness of a molecule in suppressing a particular biological or metabolic activity.

The conversion of IC50 to pIC50 involves the application of the negative logarithm to IC50 values. This transformation enables researchers to manipulate and analyze IC50 results more comprehensively, particularly when confronted with a broad spectrum of IC50 values that may differ significantly in magnitude. A drug's potency can be inferred from its pIC50 value, as a higher pIC50 value corresponds to a lower IC50 value. This indicates that the compound is more potent, as a smaller quantity of it is required to induce an equivalent amount of inhibition. The term "Abide pIC50" is commonly used to denote the pIC50 measurement corresponding to a particular chemical or dataset, often in the context of research conducted by Abide Therapeutics.

So, let's continue to the technique used in this investigation, which is called Lazy Regressor. This technique is used as it is simple and cost-effective. Data analytics tools (python) provide the built-in package to use many machine learning algorithms abiding by Lazy Regressor. A Lazy Regressor is a specialized machine learning approach primarily employed for regression tasks. Regression, within the realm of machine learning, involves predicting a continuous output value (typically numeric) based on input features. Lazy regressors, also known as "lazy learners," stand apart from conventional models in terms of how they make predictions. Here's an in-depth explanation of Lazy Regressors in machine learning.

2.1. Lazy learning vs. eager learning

Lazy Regressors fall under the broader category of "lazy learning" algorithms. Unlike "eager learning" algorithms, which construct an explicit model during training, lazy learning algorithms don't build a model. Instead, they store the entire training dataset and predict new data by comparing it to the stored training data [15].

2.2. How Lazy Regressors work

Lazy Regressors operate by memorizing the training data and using this stored information to make predictions when provided with new input. The central idea is to retrieve the most similar training data points to the input and then apply some form of aggregation or interpolation to predict the output based on the values of these retrieved training points [16].

2.3. Common lazy regressor algorithms

Several widely used Lazy Regressor algorithms include.

- **K-Nearest Neighbors (K-NN)** [17]**:** K-NN is a lazy regressor that identifies the K training data points closest to the new input point and calculates their target values' average (or weighted average) as the prediction.
- **Local Weighted Regression (LWR)** [18]**:** LWR assigns weights to nearby training data points based on their proximity to the input point. It then performs a weighted linear regression to predict the output.
- **Gaussian Process Regression (GPR)** [19]**:** GPR models the relationships between input and output as a probabilistic distribution. Given a new input, it computes the distribution over outputs, which can provide uncertainty estimates alongside point predictions.
- **Decision Trees (with Lazy Learning)** [20]**:** Decision trees can be employed lazily by navigating the tree to find the leaf node corresponding to the input data point. The prediction is then based on the average target values of training samples in that leaf node.

In the current research investigation, these approaches have been experimented:

Bayesian Ridge, Poisson Regressor, Ridge CV, SGD Regressor, Lasso, CVElastic Net CV, Hist Gradient Boosting Regressor, Tweedie Regressor, Support Vector Regression, LGBM Regressor, Nu SVR, Huber Regressor, Ridge, K Neighbors Regressor, Gamma Regressor, Orthogonal Matching Pursuit, MLP (Multi-Layer Perceptron) Regressor, Lasso Lars, CV Passive Aggressive Regressor, Ada Boost Regressor, Orthogonal Matching Pursuit CV, Random Forest Regressor, Gradient Boosting Regressor, Linear SVR, Bagging Regressor, XGB Regressor, Lars CV, Decision Tree Regressor, Extra Trees Regressor, Extra Tree Regressor, Elastic Net, Transformed Target Regressor, Linear Regression, Lasso Lars, Lasso, Dummy Regressor, Quantile Regressor, Gaussian Process Regressor, Kernel Ridge, Least Angle Regression [16,21–23].

2.4. Advantages of Lazy Regressors

- Simplicity: Lazy Regressors are easy to understand and implement since they do not involve complex model training procedures.
- Adaptability: Lazy Regressors can quickly adapt to changes in the data since they do not require retraining the entire model.
- Interpretability: Predictions made by Lazy Regressors can be more interpretable since they are often based on nearby training data points.

limitations of Lazy Regressors

- Computational Cost: Lazy Regressors can be computationally expensive, mainly when dealing with large datasets, as they involve searching for similar training data points. As for a drug molecule based upon the descriptors values, a limited number of molecules are to be searched, so it is not computationally expensive for the current study.
- Sensitivity to Noise: They can be sensitive to noisy data points or outliers since they consider all training data equally. In the current study, no noise in the data set is observed as the data is extracted through the well-known libraries in the structured form.

A. Not Suitable for High-Dimensional Data: Lazy Regressors may struggle with high-dimensional data as proximity becomes less meaningful in high-dimensional spaces. In the present study, the data dimension has been reduced by applying a variance threshold of 1/3, as mentioned in section V, Subsection: Predictive Modeling and Machine Learning.

2.6. Use cases

Lazy Regressors are employed in scenarios where interpretability and adaptability are more important than model complexity. Some everyday use cases include.

- Predicting housing prices based on property features.
- Estimating the price of a used car based on its characteristics.
- Forecasting stock prices based on historical market data.

2.7. Formulation

The general formulation of a Lazy Regressor, such as K-Nearest Neighbors (K-NN), can be expressed as follows.

- Given a training dataset with features represented as X_train and target values as y_train, and a new input data point X_new:
- Find the K training data points in X_train that are closest to X_new based on some distance metric (e.g., Euclidean distance).
- Retrieve their corresponding target values from y_train.
- Calculate the predicted output for X_new by aggregating (e.g., averaging) the target values of the K nearest neighbors.

The study utilizes data visualization tools and techniques, such as charts, graphs, and box plots, to visually present the findings comprehensibly, facilitating interpretation, insights, and reporting. Upon completion of the study, the findings are examined to draw crucial conclusions and offer essential recommendations, employing the visualization technique. The stage above holds significant importance in the process of data analysis. Subsequently, the findings of the research and the main conclusions are typically disseminated through a formal report or a presentation. Practical and concise reporting is vital in effectively conveying research findings to decision-makers and stakeholders. It should be noted that the Lazy Predict package in Python can be utilized to assess the efficacy of various machine-learning models on a given dataset. However, it is essential to mention that the machine learning models being compared must belong to the lazy regression type and be supported by the library. When addressing a machine learning problem centered around classification or regression, a researcher will seek to evaluate and contrast the performance of several models on a specific dataset. In this study, multiple machine learning models must undergo distinct training and testing processes using the dataset under investigation. The Python Lazy Predict package is utilized in this context. This feature facilitates the assessment of the performance of each classification or regression model. Hence, the evaluation of the performance of several machine learning models can be facilitated by employing Python's lazy prediction package, thereby enabling the identification of the most suitable model for a given task.

3. Literature review

This section provides a literature review regarding the methods that investigate medicine chemical space for correct medicines in the open literature. As shown in the following papers, the significance of employing data analytical tools in examining prospective chemical compounds for treating the pneumonia virus is underscored in the literature study. Researchers have expedited the drug development process by employing Python-based tools and algorithms to choose promising therapeutic options, rank molecules for further investigation, and identify potential areas of study. In silico technologies offer valuable insights in developing strong antiviral medications, thereby contributing to the monitoring and treatment of pneumonia infections.

In the study [24], potential COVID-19 antiviral medicines were discovered using a drug repurposing technique called in silico drug discovery. By analyzing the chemical space and using Python-based data analysis methods, including molecular docking, molecular dynamics simulations, and virtual screening, the researchers were able to identify potential inhibitors of pneumonia viruses.

In a different study [25], they employed machine learning techniques in Python to forecast the antiviral effectiveness of chemical compounds against the influenza 'A' virus. They employed clustering and similarity analysis to uncover compounds with potential antiviral activity despite their structural variety. The study demonstrated how new pneumonia virus treatments can be developed using data analysis techniques.

Researchers in Ref. [26] used Python-based chemoinformatics tools and machine-learning techniques to investigate the chemical space for potential antiviral medicines against the respiratory syncytial virus (RSV). The researchers employed clustering and similarity analysis to uncover compounds that structurally resembled known RSV inhibitors, which contributed to creating new treatment options.

Similarly, in Ref. [27], authors observed that modeling infection requires in vitro human lung epithelium, which includes a variety of proximo-distal axis cell types. The statistical analysis was done with the aid of GraphPad Prism version 8. Drug validation experiments were analyzed using one-way ANOVA. Methods details: drug validation tests, bulk RNA sequencing, proximal airway epithelial cell culture, and differentiation at the air-liquid interface.

In [7] study, the significance of finding new antiviral medications is emphasized. The requirement for biosafety confinement facilities (e.g., Category 3 and 4 containment levels are required, respectively, for SARS-CoV-2 and EBOV) is one of the significant technological constraints of antiviral testing on emerging RNA viruses. These facilities demand high supervision because they are expensive to construct, install, and maintain.

4. Problem statement

This Section provides the problem statement that our approach must tackle: the fast and accurate identification of the correct medicine for pneumonia. Pneumonia is one of the paramount global health challenges, characterized as an acute respiratory disease that has significantly impacted morbidity and mortality rates worldwide. Recent years have witnessed the emergence of novel pneumonia-causing viruses, most notably the COVID-19 coronavirus. This rise emphasizes the dire need to expedite the development and availability of potent antiviral medications to counteract these infections effectively. However, the contemporary landscape of drug discovery presents several formidable barriers. Prolonged durations and exorbitant costs often mar conventional methodologies in identifying and developing therapeutic agents. This extended timeline, coupled with substantial financial outlays, impedes the swift introduction and assessment of novel drug candidates,

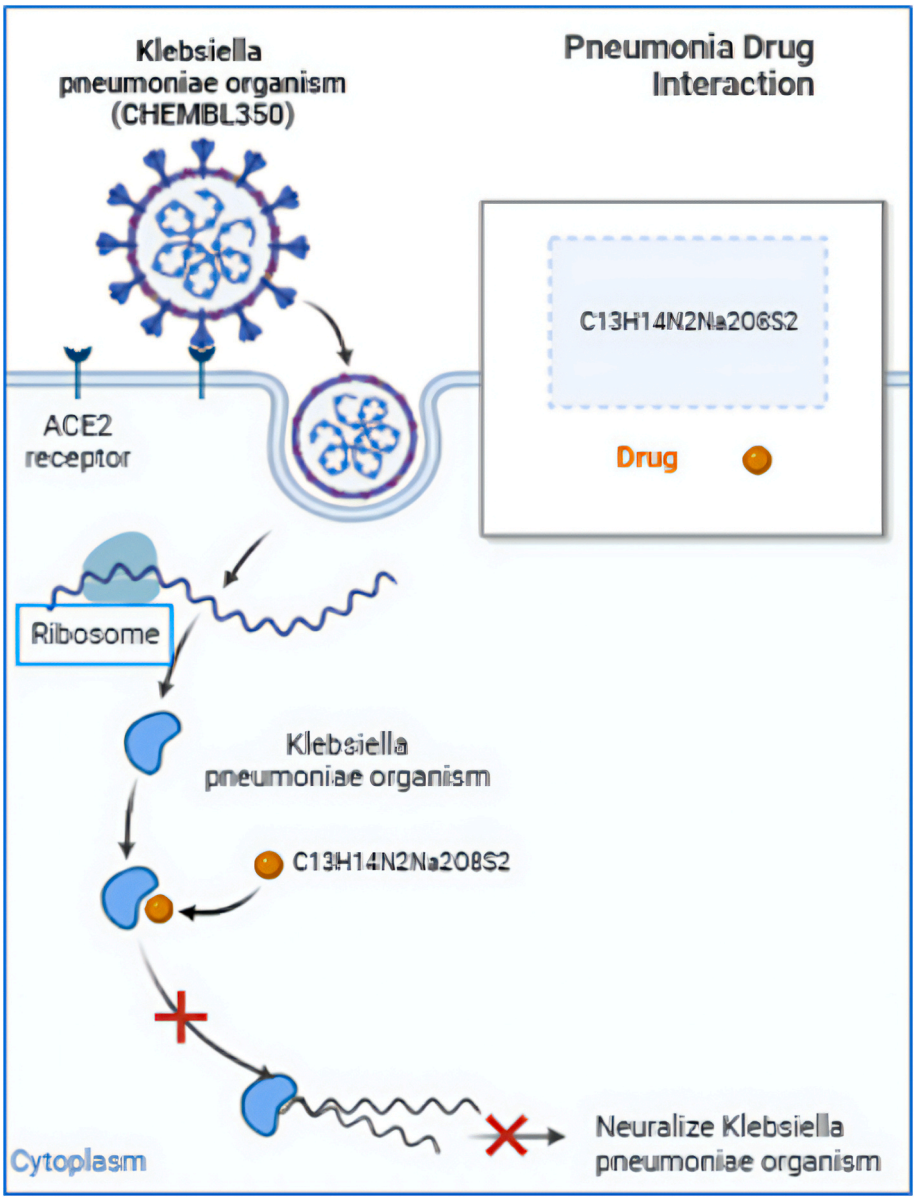


Fig. 1. Methodological drug interaction layout.

which is crucial in the face of rapidly evolving viral threats. Furthermore, the actual formulation of drugs is another pivotal aspect. There is a growing demand for optimized drug formulations to facilitate smaller pill sizes, thereby reducing the associated toxicity and enhancing patient compliance. Additionally, as the viruses evolve, yielding new variants, the need for innovative molecules that can effectively target these novel strains becomes increasingly pressing. In light of these challenges, this research seeks to delineate the problems that our proposed approach aims to address. The primary objective is to innovate and refine the current drug discovery and formulation paradigms, ensuring a timely and effective response to the ever-evolving threat of pneumonia and its associated pathogens. There is an urgent need to develop effective antiviral medications to treat these infections due to the advent of novel pneumonia viruses, such as the COVID-19 coronavirus. On the other hand, traditional drug discovery techniques are frequently time- and money-consuming, making it challenging to identify and assess new therapeutic candidates quickly. New drug formulation methods may lead to smaller pills with less toxicity. Moreover, new molecules could be helpful in new variants of the diseases.

By suggesting an in-silico method to analyze the chemical space for pneumonia viral medications utilizing data analytic techniques in a Python-based silico environment, this study seeks to address this issue. The study will use computational tools and algorithms to quickly identify and evaluate compounds with potential antiviral efficacy against pneumonia viruses.

5. Methodology

This section provides the methodology used in our approach to tackle the problem above. The experimental methods employed in our inquiry focus on creating a framework for predicting computational strategies in the field of Pneumonia virus medication discovery. This framework utilizes Machine Learning and Virtual Screening techniques, as depicted in Fig. 1. The steps of our framework are shown in the following subsections.

5.1. Data gathering

Various chemical compounds have been explored from multiple resources [24–26,28] related to Streptococcus pneumoniae,

Chlamydophila pneumoniae, *Klebsiella pneumoniae*, and Mycoplasma pneumoniae are essential organisms contributing to pneumonia. Thus, our approach gathers data associated with the compounds mentioned above.

5.2. Data preparation

Data preparation is performed for efficient data analysis, as explained in the background work section. For this, data is being prepared using the Chembl dataset for.

- The *Klebsiella pneumoniae* organisms play a vital role in the disease phenomena.
- Next, using the RDKIT library, prepared data for the molecules that can reduce the *Klebsiella pneumoniae* organism (phenomena) growth to 50% (IC50)
- For better visualization converted IC50 values to pIC50 values.
- Prepared data for the chemical descriptors for chemical property calculations and applied machine-learning algorithms

Specifically, molecular descriptors are quantitative descriptions of the compounds in the dataset. Finally, we will prepare this into a dataset for subsequent model building.

Here are some crucial components and procedures for data analysis that this investigation is following.

- Data Collection: Molecule Descriptor data is collected with the help of the Padel library of Python based upon the molecules that have a potency of desired IC50 values. Collecting Molecule Descriptor data relies on the Python-based Padel library to extract molecular information for compounds exhibiting desired IC50 values, indicating their potency in specific biological activities. Compounds with IC50 values meeting predefined criteria are initially chosen from a compound database. Subsequently, the Padel library computes a diverse set of molecular descriptors for each selected compound, encompassing structural, electronic, and property-related characteristics. These descriptors serve as numerical representations, encapsulating comprehensive information about the molecular makeup of the compounds. The resulting dataset combines compound-specific details with the corresponding computed descriptors, providing a valuable resource for in-depth analysis, modeling, and research in fields such as drug discovery and cheminformatics.
- Data pre-processing: Dimensions of descriptors have been reduced by applying 80 percent variance. The following technique is employed to reduce the dimensionality of descriptors while preserving essential information: A dataset containing descriptors representing various data features is initially collected and pre-processed to handle data quality issues. Subsequently, a dimensionality reduction technique, such as Principal Component Analysis (PCA), is applied to extract a set of principal components that capture the maximum variance in the data. The total variance explained by these components is calculated, and a threshold is set to retain at least 80% of the total variance. The selected principal components are then subjected to a formulation algorithm, resulting in a reduced set of descriptors that succinctly represent the original data while maintaining the desired level of variance. This technique enables more efficient data representation for subsequent analysis and modeling tasks.
- Data Preparation: Abide pIC50 as a class variable in the descriptors dataset. “Abide pIC50” is a pivotal class variable within the descriptors dataset, carrying significant information regarding the potency of molecules based on their IC50 values. In pharmacology and cheminformatics, IC50 values represent the inhibitory concentration at which a compound achieves a particular activity level in a biological assay. Utilizing the negative logarithm (base 10) of these IC50 values, denoted as “pIC50,” a more linear and interpretable measure of potency is obtained. In the dataset, “Abide pIC50” takes on the role of a class variable, implying its central importance in analyses and predictive modeling tasks. Researchers and data analysts often employ this variable as the target for regression models, enabling the prediction of inhibitory potency based on associated molecular descriptors. Consequently, “Abide pIC50” serves as a cornerstone for drug discovery efforts and the development of molecules with desired bioactivities, facilitating valuable insights and advancements in pharmaceutical research.
- Using Machine Learning: Using the Lazy regressor machine learning technique, predict the pIC50 value of a molecule based on its descriptors to decide whether this predicted molecule could be an excellent choice for the drug molecule using the pneumonia dataset [29,30]. In this context, the Lazy Regressor machine learning technique is employed to predict the pIC50 values of molecules based on their respective descriptors, which are numerical representations of various molecular properties. These pIC50 values serve as critical indicators of a molecule’s inhibitory potency in a biological assay, a crucial aspect of drug discovery. The Lazy Regressor technique is favored for its simplicity and efficiency in regression tasks, making it suitable for initial predictive modeling. The prediction task involves determining whether a molecule, based on its descriptors, is a promising candidate for drug development. A dataset associated with pneumonia denoted as the pneumonia dataset [29,30], is utilized to achieve this. Researchers use the Lazy Regressor to estimate the pIC50 values for molecules within the dataset. Suppose the predicted pIC50 values are sufficiently high. In that case, it suggests that these molecules possess the potential to effectively inhibit the target of interest, thus qualifying them as candidates for further experimental evaluation and optimization in the pursuit of discovering novel and efficacious drugs for the treatment of pneumonia or related conditions.

5.3. Space chemical exploration

As shown above, this research utilizes data analysis methods to investigate the chemical space and discover trends and connections between the substances. To classify molecules with related features, use K-means or hierarchical clustering methods. This aids in locating groups of substances that may have antiviral properties against pneumonia viruses. Compare novel compounds with well-known antiviral drugs utilizing similarity analysis methods like molecular fingerprinting or molecular descriptors. This analysis aids in determining how well novel chemicals combat pneumonia viruses.

More specifically, in this research, data analysis techniques are employed to delve into chemical space, which is a vast multidimensional space representing various molecular properties and characteristics. The primary objective is uncovering patterns, trends, and relationships among chemical substances. Clustering methods such as K-means or hierarchical clustering are applied to achieve this. These techniques allow for grouping molecules with similar features or properties. By doing so, researchers can identify clusters of substances that exhibit potential antiviral properties against pneumonia viruses.

Furthermore, the research involves a comparative analysis where novel chemical compounds are assessed in relation to well-established antiviral drugs. Similarity analysis methods, such as molecular fingerprinting or molecular descriptors, facilitate this comparison. These techniques provide a means to quantify the similarity between different chemical structures and their antiviral efficacy. By conducting such analyses, researchers aim to determine how effectively novel chemical compounds combat pneumonia viruses, offering insights into their potential as antiviral agents. This comprehensive approach leverages data analysis to inform the discovery and evaluation of promising compounds for combating pneumonia viruses, contributing to the field of drug discovery and antiviral research.

```
! cat molecule.smi | head -5

CC(=O)N/C=C\SC1=C(C(=O)[O-])N2C(=O)[C@@H](C(C)(C)O)[C@H]2C1.[Na+]       CHEMBL327797
CC(=O)N/C=C\[S+]([O-])C1=C(C(=O)[O-])N2C(=O)[C@@H](C(C)(C)O)[C@H]2C1.[Na+]      CHEMBL327917
CC(=O)N/C=C/[S+]([O-])C1=C(C(=O)[O-])N2C(=O)[C@@H](C(C)(C)O)[C@H]2C1.[Na+]      CHEMBL328990
CC(=O)N/C=C\SC1=C(C(=O)[O-])N2C(=O)[C@@H](C(C)OS(=O)(=O)[O-])[C@H]2C1.[Na+].[Na+]       CHEMBL433378
CC(=O)N/C=C\[S+]([O-])C1=C(C(=O)[O-])N2C(=O)[C@@H](C(C)(C)OS(=O)(=O)[O-])[C@H]2C1.[Na+].[Na+]   CHEMBL93653
```

Fig. 2. Molecule smiles.

5.4. Predictive Modeling and Machine Learning

As explained in the background work section and above, building predictive models using machine learning techniques like random forests, support vector machines, or KNN is especially important. To train these models, the usage of chemical properties, bioactivity profiles, and other pertinent traits are well-known in antiviral drugs. Also, the trained models are utilized to forecast the effectiveness or potential of new compounds as antibiotics for the pneumonia virus.

After creating the dataset with molecule simplified molecular-input line-entry system (SMILES) and Chambl IDs, as shown in Fig. 2, molecular descriptors have been generated from the molecule smiles using

Table 1
Review Summary of some Important studies.

S. No	Title	Author	Work Done	Gap/Limitation	Reference
1	In silico drug repurposing: Screening potential therapeutic compounds against COVID-19 using molecular docking and molecular dynamics simulation.	Rahman, M. M., Hosain, M. Z., Rahman, M. S., & Moni, M. A. (2020).	Potential COVID-19 antiviral medicines were discovered using a drug repurposing technique called in silico drug discovery. By analyzing the chemical space and using Python-based data analysis methods, including molecular docking, molecular dynamics simulations, and virtual screening	The researchers were only able to identify potential inhibitors of pneumonic viruses. Machine Learning Techniques can be used to justify the proper use of a silico environment.	[24]
2	Identification of potential antiviral compounds against influenza A virus subtype H7N9 via computational screening.	Li, Y., Zhang, X., Ji, D., & Yang, R. (2019).	Researchers employed machine learning techniques in Python to forecast the antiviral effectiveness of chemical compounds against the influenza 'A' virus.	Experiments were employed using only the clustering technique. Due to this, data points with outliers are not being addressed and with increased overall complexity.	[25]
3	In silico discovery of candidate drugs against Covid-19 viruses	Cava, C., Bertoli, G., & Castiglioni, I. (2020).	The authors used a silico environment to see the protein interactions and work upon already existing drugs for the potential to act against the COVID-19 drug. They concluded that diagnosing is a potential drug.	The authors used already existing drugs as agents and didn't explore the new database and new drugs to neutralize the disease.	[26]
4	SARS-CoV-2 infection of primary human lung epithelium for COVID-19 modeling and drug discovery	Mulay, A., Konda, B., Garcia, G., Yao, C., Beil, S., Villalba, J. M., … & Stripp, B. R. (2021)	Researchers observed that modeling infection requires in vitro human lung epithelium, which includes a variety of proximo-distal axis cell types. The statistical analysis was done with the aid of GraphPad Prism version 8. Using One-way ANOVA,	Limited statistical tools have been used and do not utilize the capacity of machine learning and other modern tools.	[27]
5	Antiviral drug discovery: preparing for the next pandemic	Adamson, C. S., Chibale, K., Goss, R. J., Jaspars, M., Newman, D. J., & Dorrington, R. A. (2021).	The significance of finding new antiviral medications is emphasized, and also discussed the requirement for biosafety confinement facilities to carry out safe experiments in vitro environment is.	It is a potential paper emphasis to prepare for the next pandemic, but no framework was suggested as such to overcome the situation in an organized way. Challenges.	[7]
6	Drug candidates and model systems in respiratory syncytial virus antiviral drug discovery	Heylen, E., Neyts, J., & Jochmans, D. (2017)	Substantial efforts have been made to explore the potential of the latter as a target for inhibition of RSV replication. It is an essential protein found in all cells. Potential host factor targets for treatment or prophylaxis include some of the tentative RSV receptors, co-receptors or co-factors, such as the fractalkine receptor.	Vivo and Vitro experiments, which are time- and cost-related, are conducted. Existing drugs are being focused on finding the solution to pneumonia that might occur due to a different virus. In this situation, machine learning techniques are most helpful.	[21]
7	Potential drugs for the treatment of the novel coronavirus pneumonia	Pan, X., Dong, L., Yang, L., Chen, D., & Peng, C. (2020).	Chemical drugs have the advantages of clear composition, rapid onset of action, and strong antiviral ability. Due to the urgency of COVID-19 treatment, kaletra, ribavirin, chloroquine, remdesivir, arbidol, and favipiravir were clinically used to antagonize SARS-COV-2 in China.	Pneumonia led by the COVID-19 virus is discussed, and no efforts are made to suggest new molecules that might occur with the *Klebsiella pneumoniae* organism.	[22]
8	Current strategies of antiviral drug discovery for COVID-19	Mei, M., & Tan, X. (2021).	Authors suggest that repurposing existing drugs has demonstrated power by bringing several drugs to approval for treating COVID-19 patients, such as remdesivir and dexamethasone. However, these drugs still suffer from suboptimal therapeutic effects or known strong side effects.	The authors suggested working on new potential drugs to solve pneumonia-related problems rather than suggesting old drugs.	[31]

The current study uses a framework with proper methodology, including feature reduction, data analysis, and machine learning techniques. The work focuses on the discovery of molecules in curing the phenomena symptoms.

Padel, an open-source library. Molecular fingerprints are a way of encoding the structure of a molecule in a series of binary digits (bits). Molecular fingerprints encode molecular structure in a series of binary digits (bits) representing the presence or absence of substructures in the molecule.

Continuing, comparing fingerprints will allow determining the similarity between two molecules. So, a descriptor dataset is being created based on the molecules that can interact with the target *Klebsiella pneumoniae* organism. X represents the descriptors (columns, and Y represents the value of the independent variable pIC50).

The X. shape function given the data (193, 881) means 193 rows and 881 columns. It is essential to mention that for feature reduction, the removal of low-variance features is done [29,30].

```
selection = VarianceThreshold(threshold =
(.8 * (1 - .8)))
X = selection.fit_transform(X)
X.shape
```

After applying for zero variance features that are 80% similar, the system reduced features to (193, 215). Train_test_split model selection is used in the split of 80:20, where 80 represents training data instances, and 20 represents test data instances.

Various machine learning techniques are available in Python packages like Scikit-learn for creating and analyzing predictive models (as shown in Refs. [5,6]). Our investigation examined the vast amount of 40 ML approaches and ended up with the most accurate, Bayesian Ridge (as shown in Table 5).

5.5. Validation

The K-fold cross-validation method is employed to validate and assess the models under investigation. Specifically, K-fold cross-validation with k set to 5 is utilized to ensure the model is well-generalized and capable of achieving consistent metrics across different subsets of data points. This process divides the dataset into k uniformly sized folds. Subsequently, the first (k-1) folds are used to train the model, and the average accuracy is calculated. The k-th fold serves as the test set for evaluating the model obtained.

5.6. Visualization

Visualization is essential as it is to make the doctors easily understand the results of the machine learning process. Also, use Python modules like RDKit and molecular visualization programs to see molecules' characteristics and chemical structures.

By employing this methodology, this study seeks to analyze the chemical space for pneumonia virus drugs effectively by leveraging data analytic tools in a silico setting with Python. Identifying promising medication candidates and prioritizing substances for additional experimental research can speed up drug discovery and eventually create vital antiviral treatments for pneumonia infections.

6. Experimentation

This section provides the experimentation steps executed by the proposed approach. In this study, Google Collab Notebook has been used as an overall environment to avoid various package compatibility issues. Experiments were performed using Windows 10, processor –i5, RAM-8GB. Python Packages, such as PaDEL-Descriptor, RDKIT, Pandas, NumPy, and chembl_webresource_client, have been utilized.

In this examination, the experiment explored various organisms that may cause pneumonia. The *Klebsiella pneumoniae* organism (CHEMBL350) was selected for the purpose of the study. The bacteria known as *Klebsiella pneumoniae* typically reside in intestines and feces. They are referred to as Gram-negative, enclosed, and nonmobile bacteria by experts. They are also highly susceptible to developing antibiotic resistance. For experimentation purposes, target molecules are enlisted (Table 2) with the help of the CHEMBL database and Python using code snippet.

Table 2
IC50 values of molecules under study [30].

Molecule_chembl_id	canonical_smiles	Standard value (IC50)	Bioactivity _class
CHEMBL327797	CC(=O)N/C=C\SC1=C(C(=O)[O-])N2C(=O)[C@@H](C(C)(C)O)[C@H]2C1.[Na+]	0.22	active
CHEMBL327917	CC(=O)N/C=C\[S+]([O-])C1=C(C(=O)[O-])N2C(=O)[C@@H](C(C)(C)O)[C@H]2C1.[Na+]	0.0038	active
CHEMBL328990	CC(=O)N/C=C/[S+]([O-])C1=C(C(=O)[O-])N2C(=O)[C@@H](C(C)(C)O)[C@H]2C1.[Na+]	0.08	active
CHEMBL433378	CC(=O)N/C=C\SC1=C(C(=O)[O-])N2C(=O)[C@@H](C(C)OS(=O)(=O)[O-])[C@H]2C1.[Na+].[Na+]	0.0018	active
CHEMBL93653	CC(=O)N/C=C\[S+]([O-])C1=C(C(=O)[O-])N2C(=O)[C@@H](C(C)(C)OS(=O)(=O)[O-])[C@H]2C1.[Na+].[Na+]	0.001	active

```
activity = new_client.activity
res =
activity.filter(target_chembl_id=selected_target).
filter(standard_type=" IC50")
```

Based upon the IC50 molecular weight list, the author has chosen two molecules (highlighted in Table 1) that can target *Klebsiella pneumoniae* organism. These molecules are selected based on lower values of IC50. IC50 represents the amount of molecule needed to block the organism up to 50%. The chemical named "chembl" database ID of these molecules are CHEMBL433378, CHEMBL93653 and their molecular formula is C13H14N2Na2O8S2, C14H16N2Na2O9S2 respectively (as shown in Fig. 3A and B). The docking with these molecules is shown in Figs. 4 and 5 which is −4.3 for CHEMBL433378 and -4.2 for CHEMBL93653.

Notice in Table 2 that both molecules C13H14N2Na2O8S2 and C14H16N2Na2O9S2 are in a preclinical state. Preclinical substances containing bioactivity information, such as CHEMBL6300, are substances with bioactivity information gleaned through academic research. The max_phase is set to null because the primary sources for drugs and clinical candidate drug data in ChEMBL do not indicate that this molecule has entered clinical trials.

In the following subsection, we show the examinations executed using our approach to show the quality of our investigations.

6.1. Lipinski

According to Lipinski's rule, an orally active medication has no more than one violation of these standards. The rule has specific directions, which are the following.

- The total amount of nitrogen-hydrogen and oxygen-hydrogen bonds cannot exceed five hydrogen bond donors.
- Ten or fewer hydrogen bond acceptors, all of which must be oxygen or nitrogen atoms.
- A molecular weight of under 500 Da.
- An octanol-water partitioning coefficient (log P) is calculated to be no greater than 5.

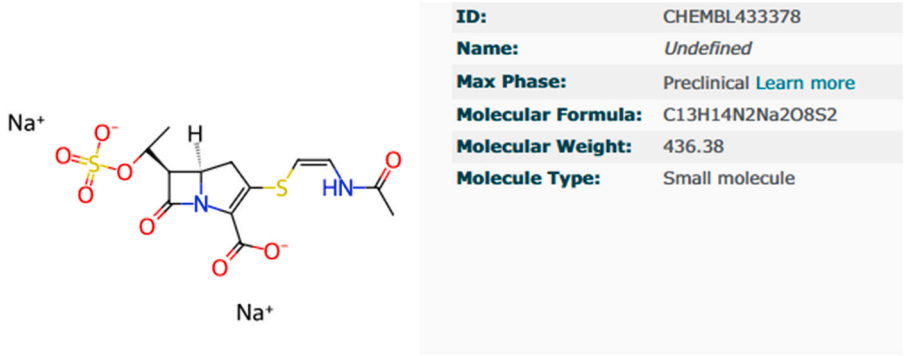


Fig. 3A. Molecular structure of CHEMBL433378.

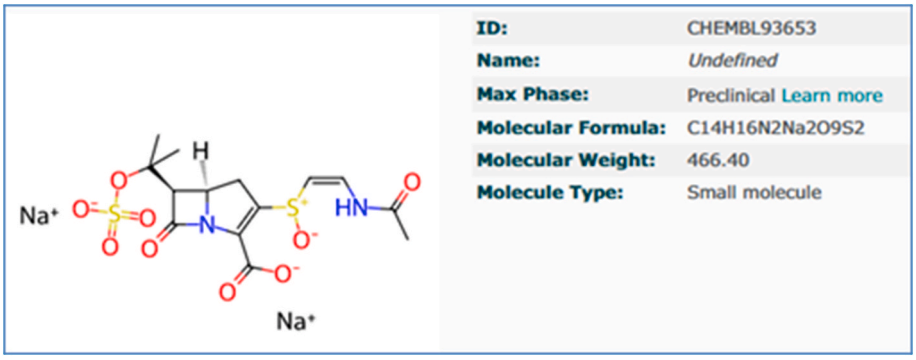


Fig. 3B. Molecular structure of CHEMBL93653.

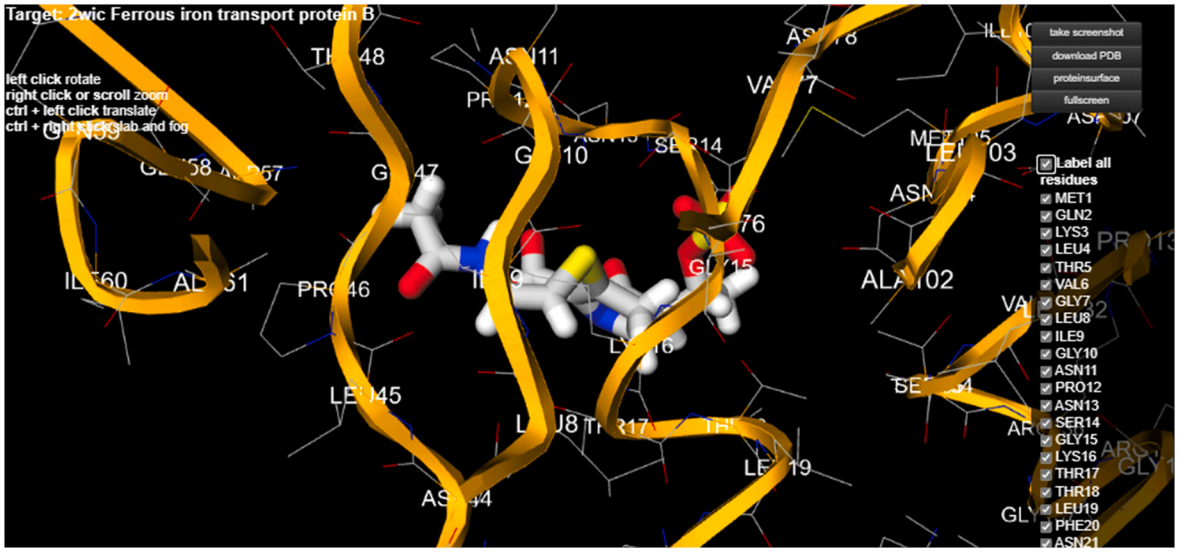


Fig. 4. Docking of CHEMBL433378 with *Klebsiella pneumoniae* (Score −4.3).

It can be observed in Table 3 that both C13H14N2Na2O8S2 and C14H16N2Na2O9S2 pass the Lipinski test.

6.2. Mann Whitney U test

The Mann-Whitney *U* Test measures whether the active and inactive groups differ regarding LogP, molecular weight, number of H acceptors, number of H donors, and pIC50. It can be observed in Table 4 that both classes have significant differences, and hence, evaluating the active drugs component may lead to some practical results. It can be observed in Table 3 that the Whitney *U* test shows that the analysis of active drug molecules and inactive drug molecules are different in terms of LogP, Number of Hydrogen molecule acceptors, donor, and pIC50.

The test statistic is denoted as U and is the smaller of U1 and U2, as defined below:

$$U1 = n1n2 + n1(n1+1)/2 - R1 \quad (1)$$

$$U2 = n1n2 + n2(n2+1)/2 - R2 \quad (2)$$

Where n1 and n2 are the sample sizes for samples 1 and 2, respectively, and R1 and R2 are the sum of the ranks for samples 1 and 2, respectively.

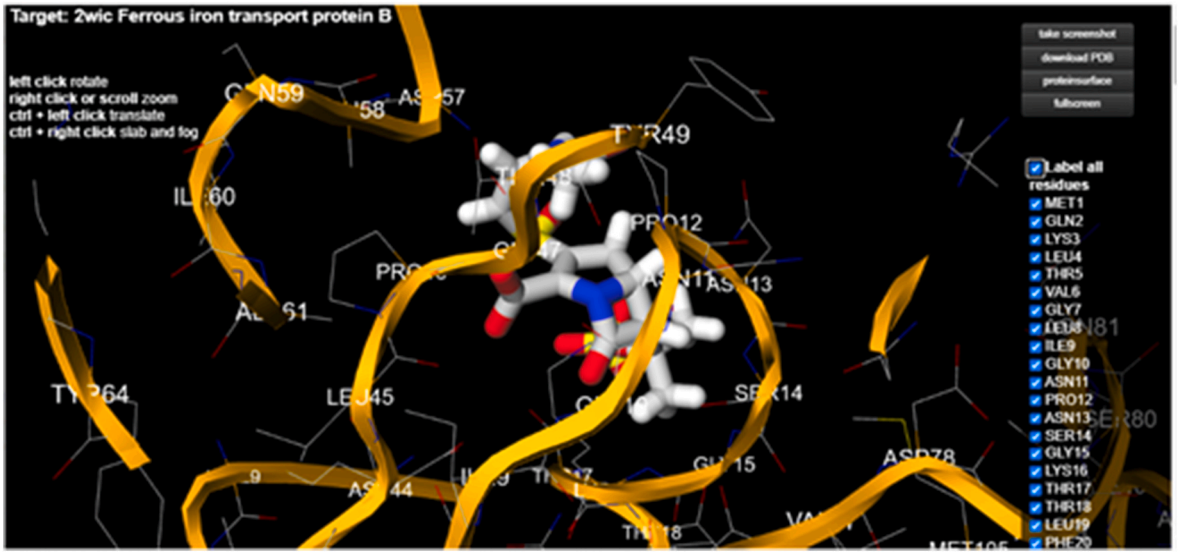


Fig. 5. Docking of CHEMBL433378 with *Klebsiella pneumoniae* CHEMBL93653 (Score −4.2).

Table 3
Lipinski values of CHEMBL433378, CHEMBL93653 molecules [29].

S No	molecule_chembl_id	bioactivity_class	MW	LogP	Num HDonors	Num HAcceptors	pIC50
1	CHEMBL327917	Active	341.36	−1.69	2	6	11.42
2	CHEMBL328990	Active	341.36	−1.69	2	6	10.09
3	CHEMBL433378	Active	390.39	−1.61	1	9	11.74
4	CHEMBL93653	Active	420.42	−2.21	1	9	12
5	CHEMBL92019	Active	406.3	−2.600	1	9	13.096

Table 4
Mann Whitney *U* Test analysis for null hypothesis evaluation [30].

S No	Descriptor	Statistics	p	alpha	Interpretation
1	LogP	1662	6.44E-12	0.05	Different distribution (reject H0)
2	MW	3951.5	0.516139	0.05	Same distribution (fail to reject H0)
3	NumHAcceptors	6087	9.00E-08	0.05	Different distribution (reject H0)
4	NumHDonors	5076	0.014089	0.05	Different distribution (reject H0)
5	pIC50	8134.5	5.99E-27	0.05	Different distribution (reject H0)

6.3. Results visualization for examination and comparison of the investigated approaches

A box plot is used to visualize the results. It is a way of graphically displaying the localization, spread, and skew of groups of numerical information through their quartiles in descriptive statistics. The lines from the box indicate variation between the upper and lower quartiles. The box plot is a valuable representation of a dataset that gives information sabot datasets. For example, Fig. 6 shows that the number of hydrogen donors for active molecules is average between 1 and 3; however, in rare cases, it may go up to 12 hydrogen donors. Fig. 7 shows that most molecules' molecular weight is less than 500, and the small dots represent these active molecules. Fig. 8 shows the frequency of active molecules is greater than that of inactive molecules. Fig. 9 shows

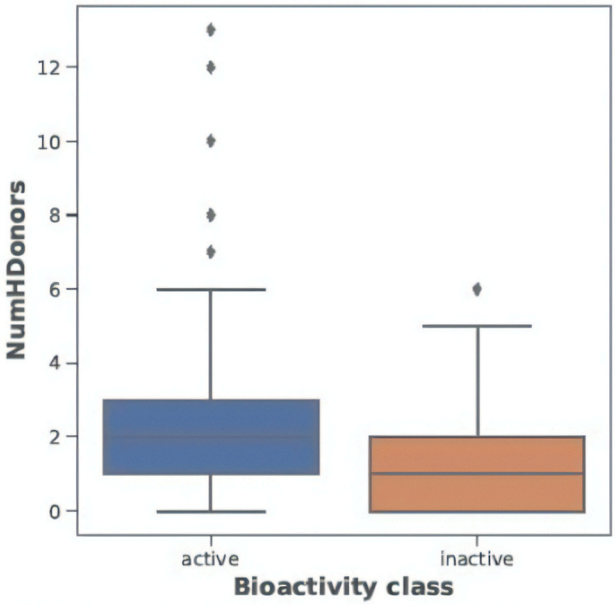


Fig. 6. Box Plot for Number of H Donors vs. Bioactivity Class.

That pIC50 average values are higher in active molecules than in non-active molecules. Fig. 10 shows active molecules have acceptance of hydrogen molecule capacity between 5 and 7.

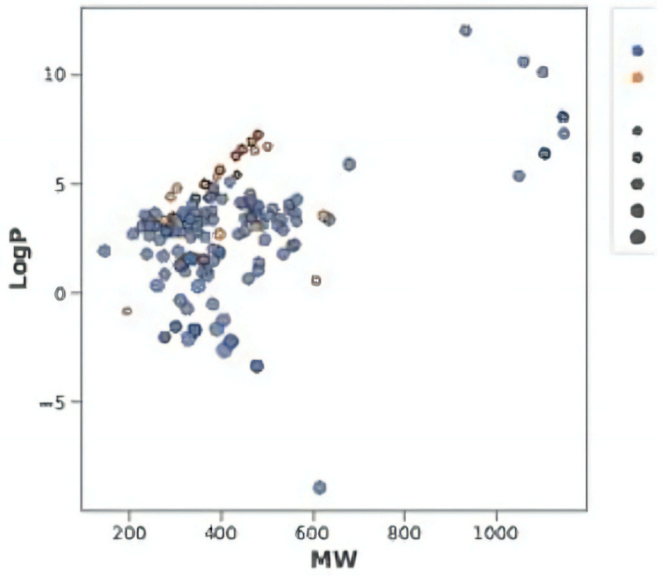


Fig. 7. Scatter Plot for LogP vs. Molecular Weight.

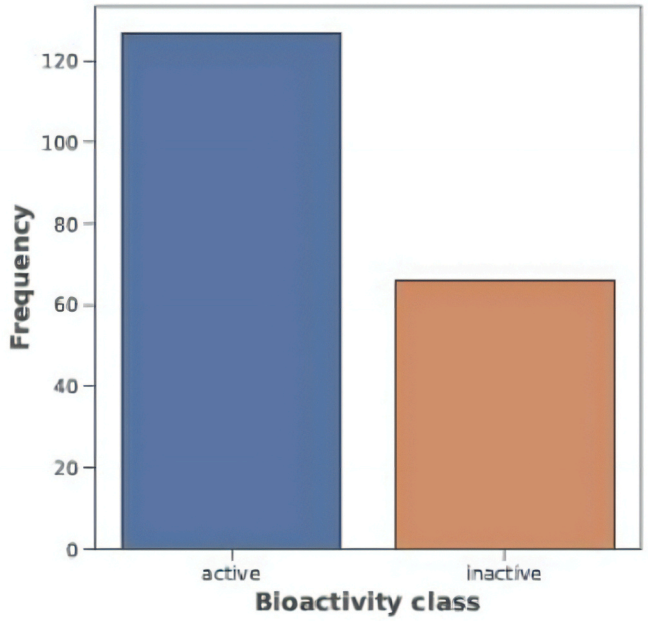


Fig. 8. Box Plot for Frequency vs. Bioactivity Class.

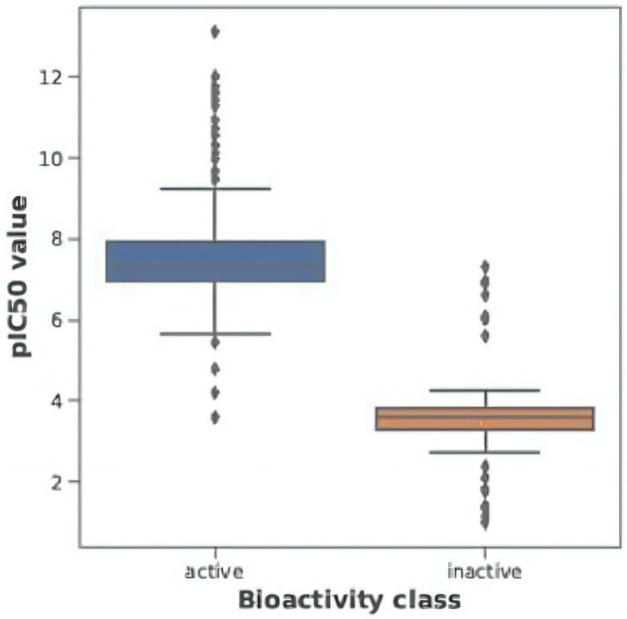


Fig. 9. Box Plot for pIC50 vs. Bioactivity Class.

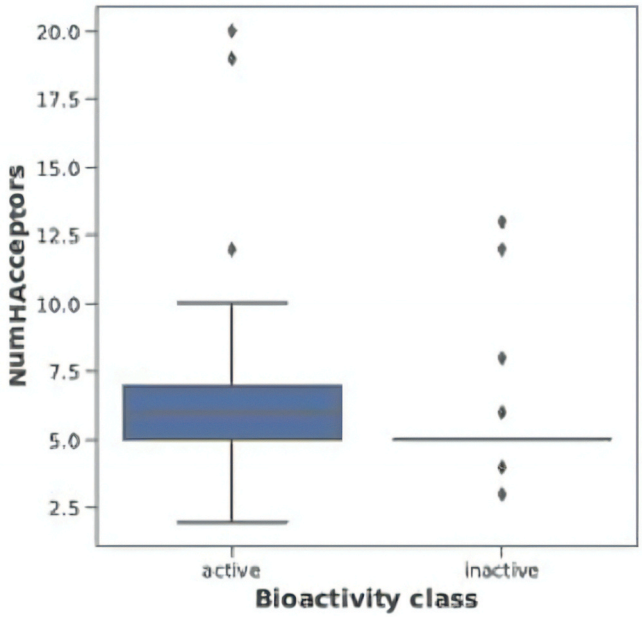


Fig. 10. Box Plot for Number of H Acceptors vs. Bioactivity Class.

6.4. *Prediction using machine learning*

As the background work section mentioned, Lazy Regressor has been used as a machine learning approach in many medical applications. In our research, machine learning prediction was applied based on molecule descriptors as an independent variable and pIC50 as a class variable. In this study, various machine learning regression models have been used. A regression model can predict values that are lower or higher than the actual value. As a result, the only way to determine the model's accuracy is through residuals. The R^2 score (pronounced R-squared score) is a statistical measure that tells us how well our model makes all its predictions from zero to one.

$$R^2 = 1 - \frac{RSS}{TSS} \tag{3}$$

$R^2 =$ *coefficient of determination*

RSS = Sum of squares of residuals

TSS = Total sum of squares

Python contains many powerful libraries. Lazy Predict is one of those libraries. It's an excellent tool for machine learning and data science. LazyPredict is an open-source Python library that helps you semi-automate your machine-learning task. It can build multiple models effortlessly and is being used to build several types of models to check prediction accuracy quickly. Table 5 shows the best-suited model near the value ONE for the adjusted R squared. In this way, Bayesian Ridge and Elastic Net CV are the best-suited models. Similarly, in Tables 5 and

Table 5
Evaluation of various Regression Models Based upon R Squared, RMSE values, Execution Time, Energy consumption, memory, and CPU Usage.

Model	Adjusted R-Squared	R-Squared	RMSE	Execution Time	Energy (mW)	Memory (kBytes)	CPU (%)
Bayesian Ridge	1.02	0.90	0.76	0.05	74	1080	15
Poisson Regressor	1.02	0.90	0.77	0.22	73	1050	14
Ridge CV	1.02	0.90	0.78	0.02	100	1400	15
SGD Regressor	1.02	0.89	0.82	0.03	110	1450	20
Lasso CV	1.02	0.89	0.82	9.67	250	2000	60
Elastic Net CV	1.02	0.88	0.82	4.96	230	1900	55
Hist Gradient Boosting Regressor	1.03	0.88	0.84	0.24	180	1700	35
Tweedie Regressor	1.03	0.87	0.88	0.35	160	1650	40
SVR	1.03	0.86	0.89	0.02	90	1300	18
LGBM Regressor	1.03	0.86	0.89	0.10	140	1550	28
Nu SVR	1.03	0.86	0.91	0.02	95	1350	19
Huber Regressor	1.03	0.85	0.92	0.06	125	1500	22
Ridge	1.03	0.85	0.93	0.02	105	1400	16
K Neighbors Regressor	1.03	0.85	0.94	0.02	100	1375	17
Gamma Regressor	1.03	0.85	0.94	0.52	170	1600	45
Orthogonal Matching Pursuit	1.04	0.82	1.04	0.02	85	1280	12
MLP (Multi-Layer Perceptron) Regressor	1.04	0.79	1.10	0.35	190	2100	65
Lasso Lars CV	1.05	0.76	1.19	0.13	130	1800	32
Passive Aggressive Regressor	1.05	0.76	1.20	0.02	80	1250	14
Ada Boost Regressor	1.05	0.75	1.20	0.14	135	1750	33
Orthogonal Matching Pursuit CV	1.05	0.75	1.21	0.03	88	1300	20
Random Forest Regressor	1.06	0.74	1.23	0.38	200	2200	70
Gradient Boosting Regressor	1.06	0.73	1.25	0.20	160	1850	50
Linear SVR	1.06	0.71	1.30	0.15	140	1700	38
Bagging Regressor	1.06	0.70	1.32	0.05	110	1450	25
XGB Regressor	1.07	0.69	1.35	0.21	165	1900	55
Lars CV	1.09	0.60	1.53	0.55	220	2300	75
Decision Tree Regressor	1.10	0.55	1.63	0.05	115	1475	26
Extra Trees Regressor	1.10	0.55	1.63	0.40	195	2150	68
Extra Tree Regressor	1.10	0.53	1.66	0.03	90	1325	15
Elastic Net	1.11	0.47	1.76	0.03	85	1300	16
Transformed Target Regressor	1.13	0.38	1.90	0.03	80	1275	15
Linear Regression	1.13	0.38	1.90	0.04	82	1290	11
Lasso Lars	1.18	0.15	2.23	0.03	78	1240	15
Lasso	1.18	0.15	2.23	0.03	78	1240	16
Dummy Regressor	1.23	−0.05	2.49	0.04	83	1295	20
Quantile Regressor	1.27	−0.24	2.70	0.49	210	2400	80
Gaussian Process Regressor	1.89	−3.15	4.93	0.03	75	1200	15
Kernel Ridge	2.46	−5.81	6.32	0.02	70	1150	24
Lars	1.14	−5.31	5.58	0.16	145	1750	45

it can be observed that the time taken for the highest-performing model is 0.02 units of time (ms).

From Table 5, the Root Mean Square Error (RMSE) is less for the Bayesian Ridge and Poisson Regressor. So, these two regression models are comparatively better than other enlisted models. Similarly, the R square value tells if a model's value is near to 1, the model is good in prediction, so Bayesian Ridge and Poisson Regressor are better models as per the conducted experiments.

From Table 5, the Energy consumption, memory, and CPU.

Usage is less for the Bayesian Ridge and Poisson Regressor. So, these two regression models are comparatively better in terms of resource usage than other enlisted models. Moreover, both methods are the quickest methods to finish the examination according to execution time. To measure the power usage in computing environments, we employ a tool known as "powertop". Powertop is a powerful utility designed for Unix-like operating systems that allows users to monitor the power consumption of their system and provide insights into the power usage of individual applications. This tool is particularly useful for identifying which processes and devices consume the most energy, thereby enabling users to make informed decisions about how to optimize their system's power consumption for improved energy efficiency. For assessing memory percentage and CPU percentage, we use a tool called "top." The top command is a task manager program found in many Unix-like operating systems that provides a dynamic real-time view of a running system. It displays a detailed overview of system processes, including the percentage of CPU and memory each process is using. This allows users to monitor their system's resource utilization in real-time, identify processes that are consuming too many resources, and take necessary actions such as terminating or optimizing those processes. The top tool is invaluable for system administrators and users alike for managing system resources effectively, ensuring that the system runs smoothly without unnecessary load on the CPU or excessive memory usage.

Bar charts are plotted based on the best prediction algorithms by considering R Squared and Root Mean Square Error values, as shown in Table 5, Figs. 11 and 12. From the performance perspective, time is taken, and various machine learning algorithms simulations have been recorded using Python tools, as shown in Table 5. In Fig. 11, Poisson Regression helps analyze both count and rate data. Hence, it is performing well. Bayesian regression models are that if you use the proper measures, automatic variable selection in your model. Root Mean Square Error (RMSE) pre-measured the average difference between the predicted and observed values. When this error is low for a model, the model's prediction performance is increased. Fig. 11 is more user-friendly and shows the bar plot of the accuracy of various models (Table 5). Simultaneously, Fig. 12 shows the root mean square error of multiple models. Hence, Fig. 12 provides good evidence for Fig. 11 and shows why some models perform better than others. For example, Bayesian Ridge and Poisson Regressor are near to the value ONE (1) and hence best fitted and have fewer error rates than the other displayed machine learning models in Fig. 12.

7. Conclusion and future works

The present study presents a pragmatic paradigm to expedite the process of developing efficacious medicinal therapies for pneumonia viruses. This research introduces a viable methodology for improving

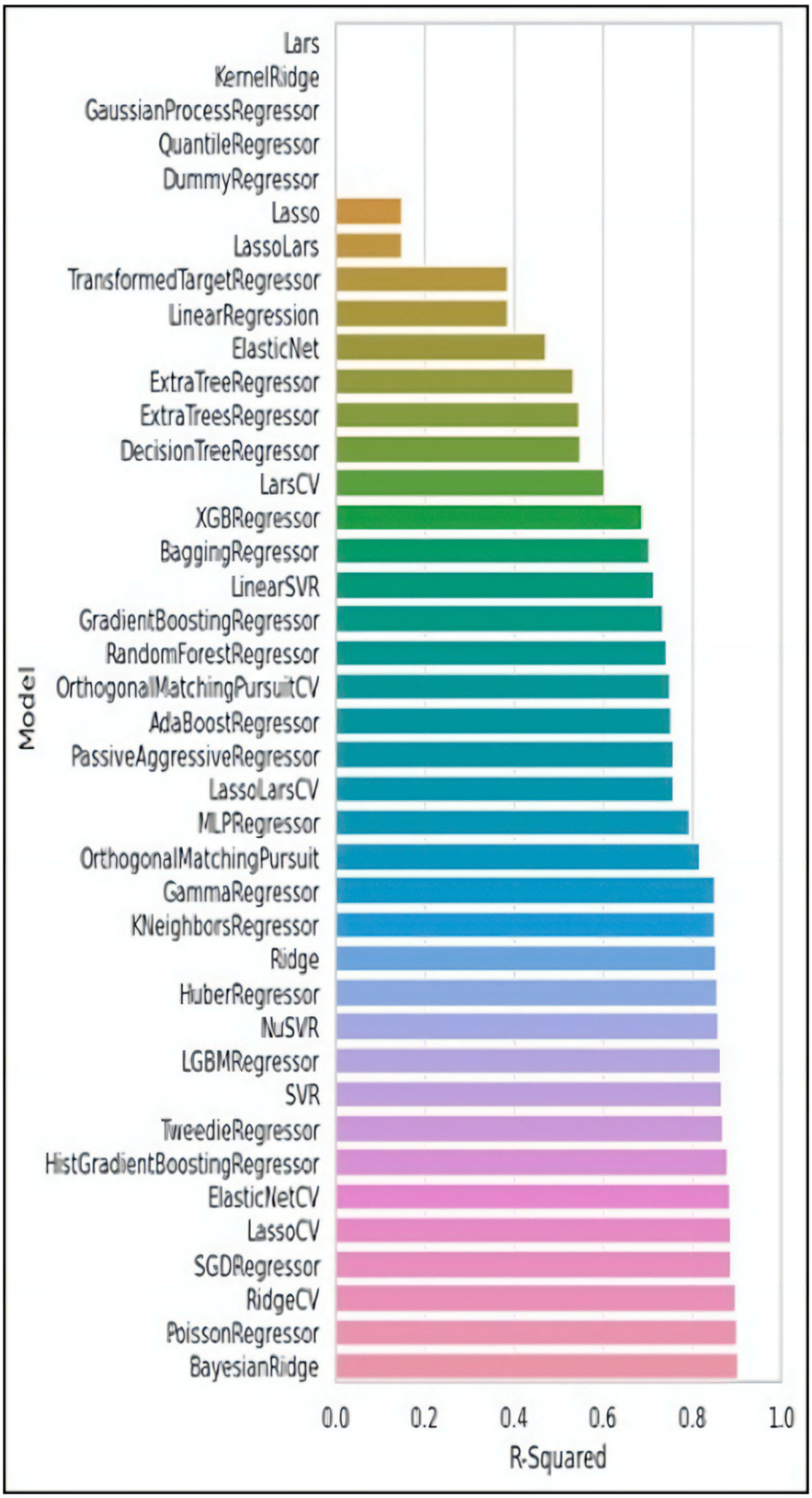


Fig. 11. Bar plot of R-squared for Various Machine Learning Algorithms.

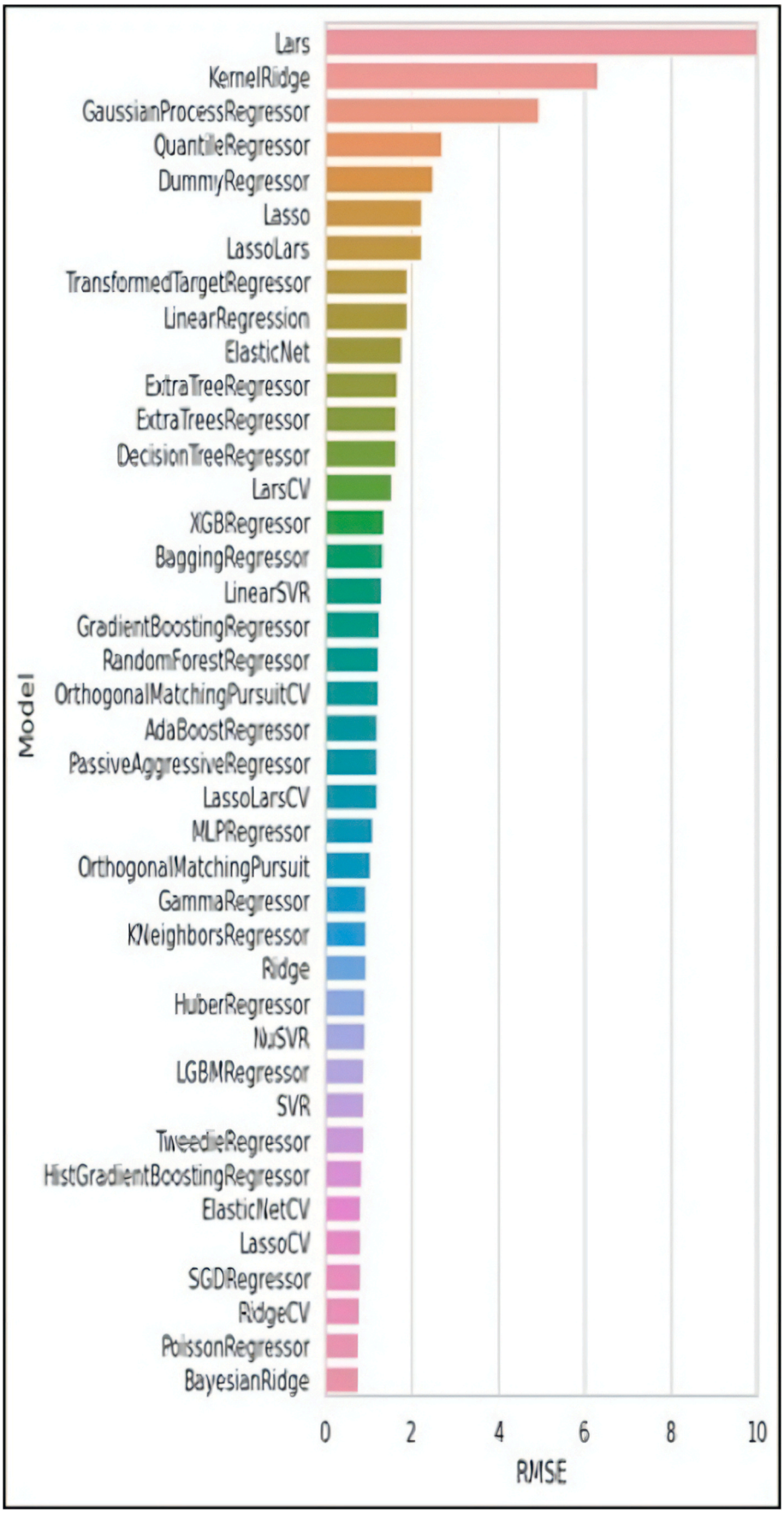


Fig. 12. Bar plot for RMSE vs. Various Machine Learning Algorithms.

the efficacy of antiviral drug identification and evaluation through Python data analytics techniques. This study emphasizes the crucial role of data preparation techniques, such as filtering, normalization, and feature extraction, in ensuring the data's quality and relevance.

These methods are implemented to ensure the robustness of future data analysis and yield meaningful insights. Using machine learning algorithms was crucial in this procedure, as they facilitated the creation of predictive models to evaluate possible therapeutic candidates. These models considered various criteria, such as the compounds' chemical properties, bioactivity profiles, and other relevant attributes. These models enhance the process of experiment selection for testing and enable virtual screening, leading to significant reductions in time and cost expenditures. The examination of graphical representations and the subsequent discoveries yield substantiating evidence for the effectiveness of this framework and its capacity to augment the drug development process.

The effectiveness of the proposed methodology was enhanced by the capabilities of the Python programming language, together with its affiliated libraries, including NumPy, Pandas, Scikit-learn, and RDKit. Python is a widely accepted and versatile programming language extensively utilized by the scientific community for undertaking silico research of chemical space. Using data analysis, machine learning, and Python-based tools in tandem with this methodology holds considerable potential in accelerating the identification of medicinal therapies for the pneumonia virus. Ultimately, it played a role in the ongoing efforts to advance effective antiviral treatments for pneumonia infections. This study can potentially enhance global health outcomes by accelerating drug discovery and addressing the pressing demand for novel therapeutics targeting pneumonia viruses.

Additionally, as we analyze our selected approach, it becomes evident that the Bayesian Ridge and Poisson Regressor models stand out in terms of Root Mean Square Error (RMSE) and R-squared values, showcasing their superior prediction accuracy over other models. These models not only excel in predictive performance but also demonstrate lower energy consumption, memory usage, and CPU utilization, highlighting their efficiency in resource usage. Their rapid execution times further underscore their effectiveness. The analysis, conducted using Python tools, visually represented through bar charts, confirms the predictive reliability and resource efficiency of these models. Specifically, the Poisson Regression's adept handling of count and rate data and the Bayesian Ridge's capability for automatic variable selection emphasize their utility. This comprehensive evaluation establishes the Bayesian Ridge and Poisson Regressor as the preferred models for machine learning applications, balancing prediction accuracy with resource efficiency.

In light of the achievements of current research, future endeavors can explore integrating more advanced machine learning and deep learning models to improve drug predictions. Adapting the established framework to address a broader spectrum of infectious diseases, incorporating real-time data on emerging viral strains, and fostering interdisciplinary collaborations with pharmaceutical entities will be pivotal. Enhancing model transparency and establishing a feedback loop with laboratory validations can further refine predictions, paving the way for a comprehensive, efficient response to global health threats.

Ethical Statement for Solid State Ionics

Hereby, I Iacovos Ioannou consciously assure that for the manuscript "Advancing Pneumonia Virus Drug Discovery with Virtual Screening: A Cutting-edge Fast and Resource Efficient Machine Learning Framework for Predictive Analysis" the following is fulfilled.

1) This material is the authors' own original work, which has not been previously published elsewhere.
2) The paper is not currently being considered for publication elsewhere.
3) The paper reflects the authors' own research and analysis in a truthful and complete manner.
4) The paper properly credits the meaningful contributions of co-authors and co-researchers.
5) The results are appropriately placed in the context of prior and existing research.
6) All sources used are properly disclosed (correct citation). Literally copying of text must be indicated as such by using quotation marks and giving proper reference.
7) All authors have been personally and actively involved in substantial work leading to the paper, and will take public responsibility for its content.

The violation of the Ethical Statement rules may result in severe consequences.

To verify originality, your article may be checked by the originality detection software iThenticate. See also http://www.elsevier.com/editors/plagdetect.

I agree with the above statements and declare that this submission follows the policies of Solid State Ionics as outlined in the Guide for Authors and in the Ethical Statement.

CRediT authorship contribution statement

Ochin Sharma: Writing – review & editing, Writing – original draft, Visualization, Validation, Supervision, Software, Resources, Project administration, Methodology, Investigation, Funding acquisition, Formal analysis, Data curation, Conceptualization. **G.S. Pradeep Ghantasala:** Writing – review & editing, Writing – original draft, Visualization, Validation, Supervision, Software, Resources, Project administration, Methodology, Investigation, Funding acquisition, Formal analysis, Data curation, Conceptualization. **Iacovos Ioannou:** Writing – review & editing, Writing – original draft, Visualization, Validation, Supervision, Software, Resources, Project administration, Methodology, Investigation, Funding acquisition, Formal analysis, Data curation, Conceptualization. **Vasos Vassiliou:** Writing – review & editing, Writing – original draft, Visualization, Validation, Supervision, Software, Resources, Project administration, Methodology, Investigation, Funding acquisition, Formal analysis, Data curation, Conceptualization.

Declaration of competing interest

The authors declare the following financial interests/personal relationships which may be considered as potential competing interests:

Dr Iacovos Ioannou reports financial support was provided by CYENS. Dr Iacovos Ioannou reports a

Relationship with CYENS Centre of Excellence Ltd that includes: employment.

No other Author has financial help from any other institution.

IX Acknowledgement

This research is part of a project that has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement N°739578 and the government of the Republic of Cyprus through the Directorate General for European Programmes, Coordination and Development.

References

[1] Sliwoski G, Kothiwale S, Meiler J, Lowe EW. Computational methods in drug discovery. Pharmacol Rev 2014;66(1):334–95.
[2] Leach AR, Gillet VJ. An introduction to chemoinformatics. Springer Science & Business Media; 2007.
[3] Dhiman P, et al. A novel deep learning model for detection of the severity level of the disease in citrus fruits. Electronics 2022;11(3):495.

[4] Fourches D, Muratov E. Tackling COVID-19 drug discovery with explainable artificial intelligence. Expet Opin Drug Discov 2019;14(9):961–4.
[5] Chen H, Engkvist O, Wang Y, Olivecrona M. The rise of deep learning in drug discovery. Drug Discov Today 2018;23(6):1241–50.
[6] Singh G, Mantri A, Sharma O, Kaur R. Virtual reality learning environment for enhancing electronics engineering laboratory experience. Comput Appl Eng Educ 2021;29(1):229–43.
[7] Adamson CS, Chibale K, Goss RJ, Jaspars M, Newman DJ, Dorrington RA. Antiviral drug discovery: preparing for the next pandemic. Chem Soc Rev 2021;50(6):3647–55.
[8] Puhl AC, Bernardes GJ. New trends in peptide-based anticancer drug design. Pharmaceuticals 2017;10(4):34.
[9] Shukla PK, Sandhu JK, Ahirwar A, Ghai D, Maheshwary P, Shukla PK. Multiobjective genetic algorithm and convolutional neural network based COVID-19 identification in chest X-ray images. Math Probl Eng 2021:1–9.
[10] Singh G, Mantri A, Sharma O, Dutta R, Kaur R. Evaluating the impact of the augmented reality learning environment on electronics laboratory skills of engineering students. Comput Appl Eng Educ 2019;27(6):1361–75.
[11] Noor S, Ismail M, Ali Z. Potential drug-drug interactions among pneumonia patients: do these matter in clinical perspectives? BMC Pharmacol. Toxicol. 2019;20(1):1–16.
[12] Sharma R, Mehta K, Sharma O. Exploring deep learning to determine the optimal environment for stock prediction analysis. In: 2021 international conference on computational performance ecvaluation (ComPE); 2021. p. 148–52. https://doi.org/10.1109/ComPE53109.2021.9752138.
[13] Maćkiewicz A, Ratajczak W. Principal components analysis (PCA). Comput \& Geosci 1993;19(3):303–42.
[14] Deng J, Yang Z, Wang H, Ojima I, Samaras D, Wang F. A systematic study of key elements underlying molecular property prediction. Nat Commun 2023;14(1):6395.
[15] Wei C-C. Comparing lazy and eager learning models for water level forecasting in river-reservoir basins of inundation regions. Environ Model \& Softw 2015;63:137–55.
[16] Kulkarni JA, Jayaraman KV, Kulkarni DB. Review on lazy learning regressors and their applications in QSAR. Comb Chem High Throughput Screen 2009;12(4). https://doi.org/10.2174/138620709788167908.
[17] Peterson LE. K-nearest neighbor. Scholarpedia 2009;4(2):1883.
[18] Cleveland WS, Devlin SJ. Locally weighted regression: an approach to regression analysis by local fitting. J Am Stat Assoc 1988;83(403):596–610.
[19] Schulz E, Speekenbrink M, Krause A. A tutorial on Gaussian process regression: Modelling, exploring, and exploiting functions. J Math Psychol 2018;85:1–16.
[20] De Ville B. Decision trees. Wiley Interdiscip Rev Comput Stat 2013;5(6):448–55.
[21] Heylen E, Neyts J, Jochmans D. Drug candidates and model systems in respiratory syncytial virus antiviral drug discovery. Biochem Pharmacol 2017;127:1–12.
[22] Pan X, Dong L, Yang L, Chen D, Peng C. Potential drugs for the treatment of the novel coronavirus pneumonia (COVID-19) in China. Virus Res 2020;286:198057.
[23] Wettschereck D, Aha DW, Mohri T. A review and empirical evaluation of feature weighting methods for a class of lazy learning algorithms. Artif Intell Rev 1997;11:273–314.
[24] Kumar Y, Singh H, Patel CN. In silico prediction of potential inhibitors for the main protease of SARS-CoV-2 using molecular docking and dynamics simulation based drug-repurposing. J Infect Public Health 2020;13(9):1210–23.
[25] Li Y, Zhang X, Ji D, Yang R. Identification of potential antiviral compounds against influenza A virus subtype H7N9 via computational screening. Front Microbiol 2019;10.
[26] Cava C, Bertoli G, Castiglioni I. In silico discovery of candidate drugs against Covid-19. Viruses 2020;12(4):404.
[27] Mulay A, et al. SARS-CoV-2 infection of primary human lung epithelium for COVID-19 modeling and drug discovery. Cell Rep 2021;35(5).
[28] Mittal L, Kumari A, Srivastava M, Singh M, Asthana S. Identification of potential molecules against COVID-19 main protease through structure-guided virtual screening approach. J Biomol Struct Dyn 2021;39(10):3662–80.
[29] O. Sharma, "Pneumonia drug exp data 1. Retrieved on 06/03/2024." Elsevier Datasets, 2023. doi: 10.17632/jt48kvz4fb.1. .
[30] O. Sharma, "Pneumonia drug exp data. Retrieved on 06/03/2024." Elsevier Datasets, 2023. doi: 10.17632/8bmpx4zvs8.2. .
[31] Mei M, Tan X. Current strategies of antiviral drug discovery for COVID-19. Front Mol Biosci 2021;8:671263.