Recent Advances in Pharmaceutical & Medical Sciences

EDITION 5

Dr. S. Padmavathi

Edited Book

Recent Advances In Pharmaceutical & Medical Sciences

Edition- 5

Dr. S. Padmavathi

Professor & HOD, Department of Pharmaceutical Chemistry and Phytochemistry, Nirmala College of Pharmacy, Atmakur, Mangalagiri, Guntur, AP, India

Publisher

Prime Publication

54/5 Ashok Nagar, New Delhi, India

Email: editor@primepublication.info

Edited Book:

Recent Advances In Pharmaceutical & Medical Sciences

(Edition- 5)

Publication Year: October, 2024

MRP: INR 499

© Prime Publication

ISBN: **978-81-978950-2-9**

Copyright © Authors and Publisher

Publisher Address

54/5 Ashok Nagar, New Delhi, India

Email: editor@primepublication.info

Contact: +91, 9630232035

Printer: Prime Publisher

Chapter- Six

KNIME workflow for molecular modeling and docking studies of drugs

Tanishqua Sharma, K. Venkateswara Swamy

MIT School of Bioengineering Sciences & Research, MIT Art, Design and Technology University, Pune, India

Abstract

The molecular docking workflow designed for drug discovery research on the KNIME Analytics Platform, tailored to combine public/commercial tools and softwares. The designed workflow integrates RDKit and OpenBabel cheminformatics for molecular modeling and docking with AutoDock Vina for ligand-receptor docking, extends the capabilities of KNIME using Python scripts. To describe the step-by-step installation process and the configuration of different plugins in the KNIME environment, had to frequently deal with issues of data integration errors, licensing issues, and troubleshooting. This study illustrates how KNIME makes streamlined, modular, and cost-effective drug discovery research crucial for academic settings.

Keywords: *KNIME Analytics Platform, Molecular modeling and Docking, Drug Discovery, RDKit, AutoDock Vina, Workflow Automation, Python Integration, Cheminformatics.*

Introduction

1.1 Importance of Molecular Modeling and Docking:

The quest for new medicines is a daunting endeavor. Pharmaceutical and biotechnology companies must invest heavily in the discovery of a single breakthrough drug, one that can potentially vanquish a disease or provide relief from debilitating symptoms. This complex process demands substantial resources, underscoring the challenges inherent in modern drug development.

However, a revolutionary approach is transforming the landscape*: In-silico* drug discovery, also known as chemobiological modeling. By harnessing the power of computers, researchers can streamline the discovery process, saving both time and money. Computational techniques are increasingly becoming integral to drug development, gaining widespread acceptance for their efficiency and effectiveness.
This innovative methodology leverages computer simulations to identify potential therapeutic compounds, expediting the journey from laboratory to marketplace. As the pharmaceutical industry continues to evolve, *In-silico* drug discovery is poised to play a pivotal role in unlocking new treatments and improving patient outcomes.

Molecular Modeling
Molecular modeling refers to all theoretical methodologies and computational techniques used to model or simulate molecular behavior. The techniques are applied in computational chemistry, drug design, computational biology, and materials science to investigate molecular systems ranging from small chemical systems to giant biological molecules and material assemblies [1]. The most basic calculations can be done by hand, but computers are required to perform molecular modeling on any substantially sized system. The atomic-level representation of molecular systems is common to all molecular modeling methodologies based on various algorithms.

Molecular Docking
Although molecular modeling is a large field, the three most extensively utilized components of computer modeling are molecular docking, MD simulation, and ADMET modeling, which have proved critical in the quick identification of leads for experimental in vitro and in vivo testing [2]. Molecular docking is an in-silico method for identifying the right binding posture of a protein-ligand complex and evaluating its effectiveness using multiple scoring functions. The best pose created by each molecule is then scored. Docking strategies aim to fit a ligand

into a target protein's binding site by combining and optimizing factors such as hydrophobic, steric, and electrostatic complementarity, consequently measuring their binding free energy [2]. The primary goal of molecular docking is to achieve an ideal docked conformer of both interacting molecules in order to reduce the free energy of the entire system [3].

1.2 KNIME as a tool for drug discovery:

The Konstanz Information Miner (KNIME, version 2.12.0) is an open-source tool that allows you to create programs called workflows that mine data. KNIME contains several nodes that can be utilized for cheminformatics, such as learning and prediction models. (KNIME, 2016) RDKit Collection. This collection comprises various nodes used in cheminformatics, such as substructure filters, molecular fragments, and more. (RD Kit, 2016) [4]. KNIME offers a diverse set of tools, supported by a large community of contributors, to enable ligand and structure-based drug creation [5]. It also provides a combined approach for data mining requirements throughout the drug discovery process by visualizing data workflows and drawing on a large tool repository [6].

1.3 Objective:

Most of these are licensed, and the open version will not be available to researchers who haven't paid licenses. This can undermine the usage of such a platform, especially among smaller research settings and academics.

To overcome this limitation, we developed a KNIME-based workflow that avoids using those expensive license nodes by incorporating open-source alternatives and custom scripting. The method eliminates the cost barriers related to licensed software but still allows the researchers to conveniently and easily access molecular docking post-docking analysis. This workflow, combining open source with a user-friendly interface, can effectively dock ligand-receptor interactions while offering a scalable basis for lead optimization in a one-pot synthesis procedure.

2) KNIME Setup and Workflow Design

2.1 Introduction of KNIME Interface and Nodes:

KNIME is an open-source data analytics tool that offers an advanced and user-friendly interface for processing, analyzing, and illustrating data [7]. The platform provides a graphical environment in which workflows comprised of interconnected nodes demonstrate various data processing procedures, which are simple to develop and run [8].

KNIME User Interface

The interface of user is intuitive to use and not difficult at all to navigate. The central area of the main workflow is located on the screen where users can easily drag and drop nodes into their analysis pipeline to build them. On the right side of the interface, users have access to the node and workflow repositories, where a wide range of pre-built nodes and workflows is available as contributed by the KNIME [9][10][11][12].

KNIME has become popular because it is heterogeneous and quite inclusive to support widely, coming from the community side. The central element of this software, which is the workflow, is shown within the nodes as offered by KNIME.

2.2 Obtaining KNIME Analytics Platform:

2.2.1 Downloading the KNIME Analytics Platform:

Getting started with KNIME involves a preliminary step of downloading the KNIME Analytics Platform software. Below are the steps to follow:

1] Visit the KNIME website: https://www.knime.com/knime-analytics-platform

2] Click the "Download" button.

3] Fill out the opened form and check the checkboxes to optionally receive emails from the mlr teacher bot or get info about new releases or events in your area [15].

4] Select the operating system to use: Windows, Linux, or Mac, and the flavor of the installer you want to download.

5] Accept the terms and conditions and click "Download."

6] Once your download is complete, double-click on the installer file to get the installation running.

7] You can now follow the prompts given to you to install.

8] Once you're done installing, you start the KNIME Analytics Platform by double-clicking the icon on your desktop.
Note: You need to have Java installed on your computer so you can run them; you can download Java from https://www.java.com/en/download/.

2.2.2 Installing the necessary plugins for the workflow:
To use various nodes of KNIME like RDKIT, Python, Marvin, etc., you will have to download the necessary plugins by clicking on the preference icon on the KNIME platform. Once clicked, it will open a dialogue box that will show multiple options, and then select the desired plugin and click install.
For example: If the user wants to install Python for scripting, then they should follow the below protocol:
1] Open KNIME Analytics Software

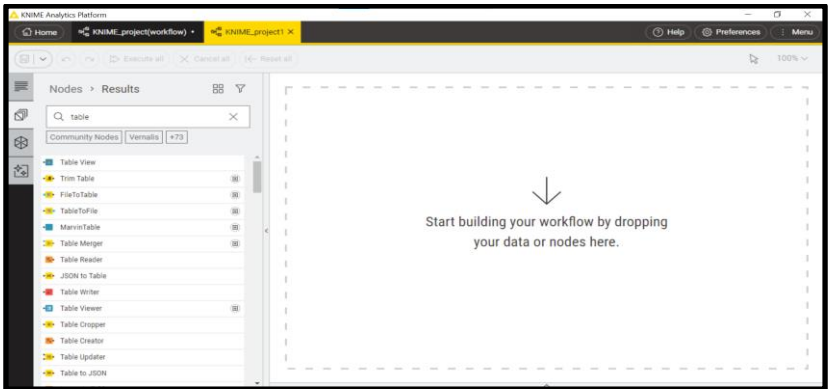


Fig 1: KNIME Analytics Software

2] Click on the preference icon which is present on the top right side corner

Fig 2: Click on the red highlighted icon

3] Choose Python script from the given options

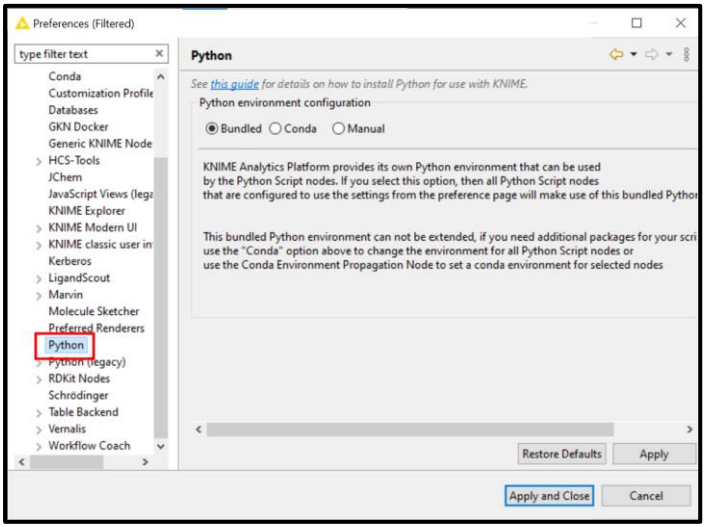
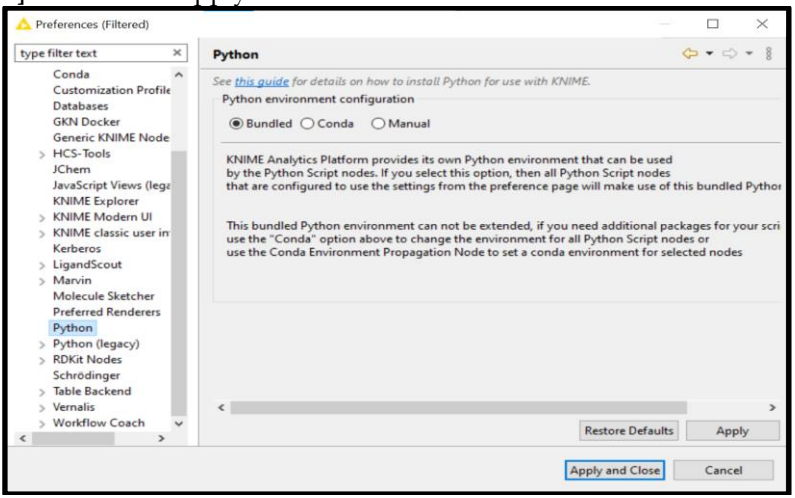


Fig 3: Click on the highlighted preference

4] Click on the apply and close button



Similarly for all the other necessary plugins, the same method has to be followed

2.2.3 Other software dependency:

Installing KNIME and other plugins is not enough to create a molecular docking workflow; various third-party software must

be loaded on the computer to ensure that the workflow runs well.

1] Python Script - this is one of the most important applications that must be downloaded.

Python can be downloaded from https://www.python.org/downloads/.

2] Auto dockVina - this program is required for the generation of a configuration file, which is a must-have file for conducting the docking method.

The program can be downloaded from https://vina.scripps.edu/downloads/.

3]Open Babel is necessary for converting .pdb files to .pdbqt and vice versa, allowing for output visualization of the .pdbqt file, once converted to pdb and viewed using pymol

4] Pymol - This software plays a very important role in the visualization of the output file generated after the docking process is completed, and in KNIME the pymol node will only work if pymol is installed in the computer because the node has a requirement of adding the path of Pymol installed in computer.

The program must be downloaded from https://pymol.org/dokuwiki/doku.php?id=installation.

3] Molecular Descriptor Calculation for Ligands

3.1 Importance of Molecular Descriptors in Drug Discovery

Many modern applications in computer-aided drug development and cheminformatics rely on representing molecules with descriptors that reflect their structural traits and behaviors. Variance analysis, library construction, as well as virtual screening are a few examples of such uses. Hundreds of molecular descriptors have been included in the literature, ranging from simple bulk properties to difficult three-dimensional forms and complex molecular fingerprints that can include thousands of bit locations. Knowledge-based choices for descriptors appropriate for certain applications are a critical problem in cheminformatics research. If descriptors are to be chosen on rational basis rather than guesswork or chemical intuition, a thorough examination of their performance is necessary [13].

The QSAR/QSPR strategy includes the determination of relevant molecular descriptors, molecular structure representation, and mathematical tools for developing and analyzing models. The remarkable progress in chemometrics and cheminformatics over the last few years has resulted in new strategies for discovering mathematically meaningful relationships between molecular structure and biological activities, the physicochemical, toxicological, and environmental properties of chemicals [14].

3.2 KNIME Workflow for Descriptor Calculation

We used a set of compounds in the SMILES format for preparing the molecular descriptors required for docking. We used the following KNIME nodes for conversion and calculating the descriptors:

1] Excel Reader: Loaded a data set from an Excel file having a compound and its SMILES format in each row. The data was read and made available within the KNIME workflow for further processing.

2]Molecular Type Cast: After importing SMILES data, this node has converted the SMILES strings to a SMILE form that is interpreted by the cheminformatics tools of KNIME, which is indispensable for further molecular manipulations as well as descriptor calculations.

3] RDKit From Molecule: This node converts the molecular representations into RDKit molecules. The RDKit nodes fit perfectly for use with KNIME. An open-source cheminformatics library, in which lots of functionality may be found and directly implemented for molecular manipulation and analysis.

Before the computation of descriptors, the molecular structures were visualized in three dimensions so that we can have the successful conversion from SMILES to the molecular structure, as we like to visualize the molecular structure before calculating the descriptors to identify any error or anomaly in molecular structure.

4] RDKit Descriptor Calculator: The node was applied to compute a range of molecular descriptors like molecular weight, topological polar surface area, the number of rotatable bonds, and other physicochemical properties. The different computed descriptors here are of extreme importance for the understanding of molecular characteristics as well as in subsequent docking studies.

5] Table View: Calculated descriptors were viewed from the Table View node. This means we can check and evaluate the result in a tabular format. This is an important step to confirm whether the descriptor is correctly calculated before undergoing the workflow of docking.

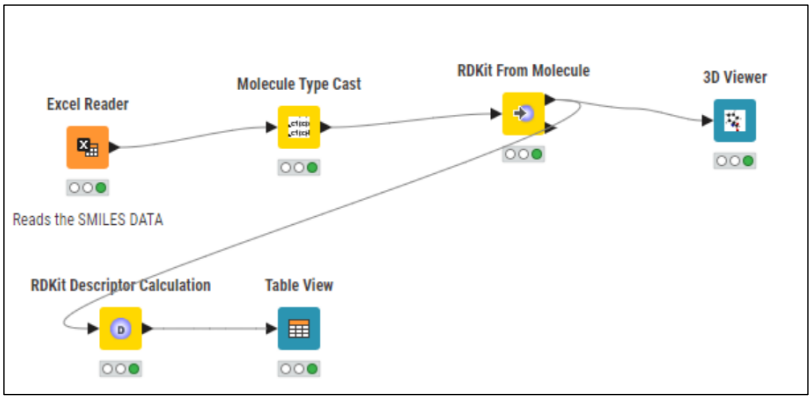


Fig 5: KNIME Workflow depicting Molecular descriptors calculation and visualization

4] Docking Workflow Using AutoDock Vina in KNIME

4.1 Ligand Preparation:

The next step in the molecular docking pipeline is the preparation of ligands, where SMILES representation of the ligands was used to transform them into appropriate formats for docking simulations, namely PDB and PDBQT.

1] *Molecule Type Cast*: With the same set of ligands in SMILES format, the Molecule Type Cast node converted SMILES strings into a PDB format. PDB format contains the 3D molecular structures that are necessary for visualization and further analysis in the simulation of docking.

2] *The OpenBabel node* converted the PDB files to the final PDBQT format to prepare the ligands for docking using AutoDock Vina. PDBQT is a specific PDB format containing atomic charges and torsion tree information required for molecular docking simulations. OpenBabel efficiently performed this conversion, as it is an open-source chemical toolbox that can be easily integrated into the workflow.

This step in preparing the ligand ensured that the ligands were subsequently prepared for docking simulations in the right format, contained correct 3D structural data, and included all the atomic and charge information.

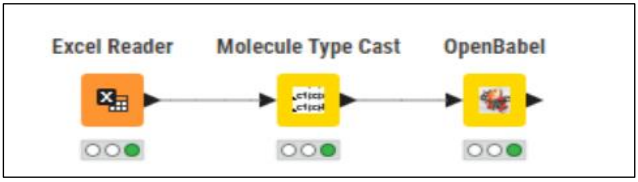


Fig 6: KNIME Workflow depicting the conversion of SMILES to a pdbqt file

4.2 Molecular Docking

For molecular docking, we employed Auto Dock Vina, which can directly be accessed within KNIME by a custom Python script. This means that users do not depend on license-dependent molecular docking nodes in their workflow, so it will be more freely accessible, flexible, and adaptable.

File Reader: First, we read the prepared ligands in PDBQT format into the KNIME workflow using the File Reader node. Each PDBQT the actual PDBQT corresponds to one ligand- was loaded into KNIME as separate entries.

Then, individual PDBQT files were concatenated into a table with the help of a Concatenate node. It assisted in processing and managing all ligand files together as a batch for molecular docking in a batch manner thereby making the workflow even simpler and consistent throughout the docking simulations.

Configuration file creation: The file included some of the critical docking parameters. Among them were those described below:

Center X, Y, Z - coordinates of the center of the binding site in a protein target where docking simulation would be focused.

Size X, Y, Z: The Grid box is composed of the three dimensions that determine the search space for ligand binding.

Energy Range: This parameter controlled the energy range over which ligand conformations would be scored.

Exhaustiveness: This parameter determines to which extent the search was explored. The more exhaustive the docking algorithm was, the higher is the value of this parameter.

The configuration file ensured that all docking simulations followed consistent and optimized parameters for accurate results.

We built a node of Python script in KNIME to do docking simulations. Once we set up the Python script it became the interface for the user to pass information to AutoDock Vina's executable, so to make a more direct communication between a workflow and docking software; it handled submitting ligands, targets from proteins, and docking parameters to AutoDock Vina and received results from each run afterward.

This custom Python script removed the docking nodes requirement from the licensed KNIME setup so that any user can simply perform docking simulation without any licensure constraint, thereby increasing the workflow's accessibility and flexibility toward different docking protocols.

This semi-automated docking setup allowed for the efficient treatment of several ligands and the calculation of docking scores and binding affinities that further simplified the post-simulation analysis.

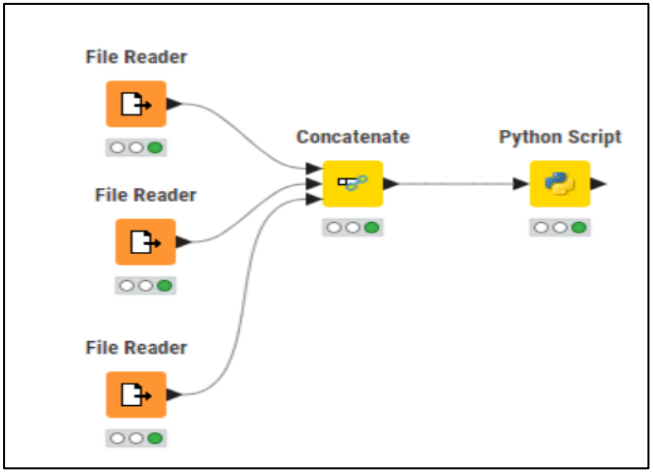


Fig 7: KNIME Workflow executing molecular docking process

The script creates a new output directory which consists of the output.pdbqt file and the log file for storing the result of the docking.

4.3 Advantages of Using KNIME for Molecular Docking and Modeling

1) Visual Workflow Design:

Because KNIME uses drag-and-drop, it makes the workflow design visually intuitive rather than requiring lines of code upon lines. It is easier in the hands of those who are not too technical but still command complex molecular docking and modeling tasks.

Another feature is that the workflows are far more visual than in traditional scripting-based approaches; you can see the workflow at every step of the process.

2) Reproducibility:

In the case of traditional approaches, command line tools and stand-alone software often used within the approaches pose a kind of challenge in reproducing identical workflows. Workflows in KNIME can be saved, shared, and reused, ensuring full reproducibility of the whole process.

KNIME enables easy workflow annotation and versioning, and thus makes collaboration and documentation rather simple.

3) Integration with Various Tools:

KNIME can be integrated with cheminformatics tool packages like RDKit and Open Babel, with easy access to many external software applications, such as Auto Dock Vina through Python scripting. Combination through integration brings together the best tools for a seamless workflow. There are quite a number of conventional methods that essentially necessitate several installations of software, scripting, and manual file manipulation. All this is taken care of by KNIME in one unified environment.

4) Automated Data Manipulation:

The workflow can be initiated automatically during registration. It can also be used retrospectively throughout the corporate collection. Automation alleviated the chemist's workload while submitting the compound to the registration system. Automation at scale also saved many hours of time spent back-checking compounds in the corporate chemical library.[14].

5) Open-source and Extensible

The workflow system KNIME is an open source software that tries to overcome these challenges by providing a platform that is easily extensible with additional tool integrations, has a highly

typed data system, and allows workflow developers to record the process's steps in detail [8].

5] Case study: Docking Betulin Ligands against Estrogen Receptor-alpha in Breast Cancer.

5.1 Background of Betulin and Target Receptor:

Estrogen Receptor alpha

ERα is a key target for treating breast cancer due to its overexpression in most cases and its activation by oestrogen, which stimulates tumour growth in hormone receptor-positive cells. Tamoxifen, a popular selective oestrogen receptor modulator for ERα therapy, has certain limitations.

SERMs effectively suppress ERα activation in breast tissue. However, it is likely that these chemicals will function as partial agonists in other tissues, which may be undesirable. Finally, there is fear that tamoxifen will lose efficacy over time as resistance develops [16].

Betulin

Betulin has gained attention as a promising medicinal chemical for treating cancer, particularly breast cancer. Betulin, a natural chemical obtained from birch bark, has anti-inflammatory, antioxidant, and anticancer effects. Betulin has selective cytotoxic effects on cancer cells, sparing normal cells [17]. It has been shown to induce apoptosis (programmed cell death), decrease tumor growth, and prevent cancer cell migration and invasion. Furthermore, betulin has demonstrated the capacity to overcome medication resistance and improve the efficacy of chemotherapeutic drugs [18].

5.2 Applying Descriptor and Docking Workflows

5.2.1 Applying Molecular Descriptor workflow-

An excel was created which consisted of 12 betulin and their derivatives namely: Betulin, Betulin-28-oxime, Betulin 28-acetate, 30-Hydroxy Betulin, Betulin-28-yl beta-glucopyranoside, Acyclovir & Betulin, Betulin diphosphate, Betulin-3-yl beta-glucopyranoside, Betulin-3,28-Dioxime, Betulin 3-Acetate, 4-Hydroxytamoxifen.These compounds were chosen based on their binding affinity to 40 betulin compounds, which were manually determined using Autodock Vina.

After processing the SMILES sequence and the compounds, the workflow generated an output in the form of a table view with 12 molecular descriptors: binding affinity, SlogP, TPSA, Exact Molecular weight, Number of Lipinski Hydrogen Bond Acceptors and Donors, Number of Hetero atoms, HBD, and HBA, as shown in figure 8, as well as a 3D viewer node that provided a 3D view of the compound according to their SMILE sequence. All of these features are highly significant when screening lipids for docking against various chemical molecules.

5.2.2 Applying Molecular Docking Workflow-

From the list of ligands, the compound that showed the best molecular property was chosen for further analysis which was **Betulin-28-yl.** First, the SMILE sequence was converted to .pdb format file using the Open Babel node and then later converted to a pdbqt file using the same node since the .pdbqt file is required for ligand preparation. Once done it was processed into the above-mentioned docking workflow.

Once done, the output.log file which was generated was viewed using Table view node for further analysis.

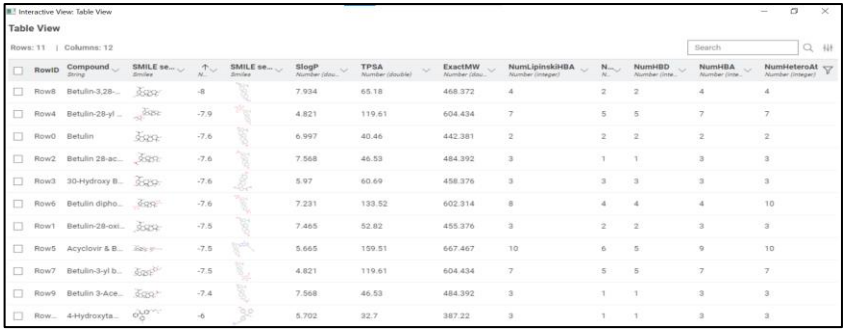
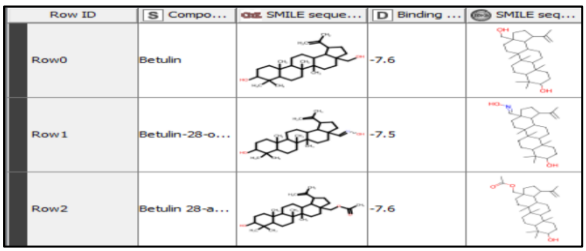


Table View

Rows: 11 | Columns: 12

RowID	Compound	SMILE se...	SMILE se...	SlogP	TPSA	ExactMW	NumLipinskiHBA	N...	NumHBD	NumHBA	NumHeteroAt
Row8	Betulin-3,28-...		-8	7.934	65.18	468.372	4	2	2	4	4
Row4	Betulin-28-yl ...		-7.9	4.821	119.61	604.434	7	5	5	7	7
Row0	Betulin		-7.6	6.997	40.46	442.381	2	2	2	2	2
Row2	Betulin 28-ac...		-7.6	7.568	46.53	484.392	3	1	1	3	3
Row3	30-Hydroxy B...		-7.6	5.97	60.69	458.376	3	3	3	3	3
Row6	Betulin dipho...		-7.6	7.231	133.52	602.314	8	4	4	4	10
Row1	Betulin-28-oxi...		-7.5	7.465	52.82	455.376	3	2	2	3	3
Row5	Acyclovir & B...		-7.5	5.665	159.51	667.467	10	6	5	9	10
Row7	Betulin-3-yl b...		-7.5	4.821	119.61	604.434	7	5	5	7	7
Row9	Betulin 3-Ace...		-7.4	7.568	46.53	484.392	3	1	1	3	3
Row...	4-Hydroxyta...		-6	5.702	32.7	387.22	3	1	1	3	3

Fig 8: Result for molecular descriptors



Row ID	Compo...	SMILE seque...	Binding ...	SMILE seq...
Row0	Betulin		-7.6	
Row1	Betulin-28-o...		-7.5	
Row2	Betulin 28-a...		-7.6	

Fig 9: A representation of the output of the 3D view node

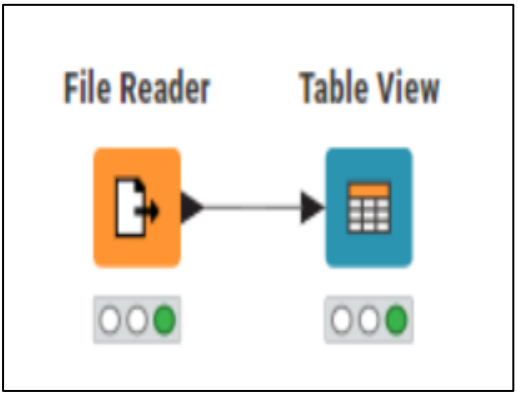


Fig 10: Viewing of log file using Table view

The table view provided the results in the exact same format as the one that was generated using the traditional Auto Dock Vina tool . The Figure 11 depicts the result.

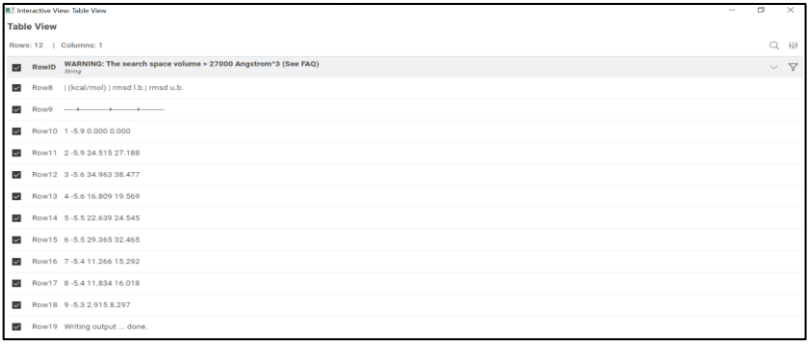

Interactive View: Table View

Table View

Rows: 12 | Columns: 1

RowID	WARNING: The search space volume > 27000 Angstrom^3 (See FAQ) String
Row8	| (kcal/mol) | rmsd l.b.| rmsd u.b.
Row9	—+———-+———-+———-
Row10	1 -5.9 0.000 0.000
Row11	2 -5.9 24.515 27.188
Row12	3 -5.6 34.963 38.477
Row13	4 -5.6 16.809 19.569
Row14	5 -5.5 22.639 24.545
Row15	6 -5.5 29.365 32.465
Row16	7 -5.4 11.266 15.292
Row17	8 -5.4 11.834 16.018
Row18	9 -5.3 2.915 8.297
Row19	Writing output ... done.

Fig 11: Results for molecular docking

Once the output.pdbqt file was generated it was viewed using the pymol node but since it only accepts input in .pdb format the .pdbqt file was converted to .pdb file using Open Babel via python script which stored this output in the local computer which was then viewed using a PDB reader node connected to the pymol node and in the PDB reader the location of the .pdb file was given.

The workflow is depicted in the Fig 12, the results in figure number 13

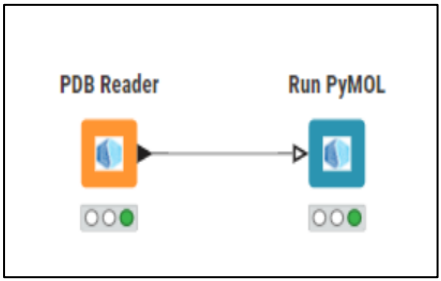


Fig 12: Workflow for viewing the output.pdb file

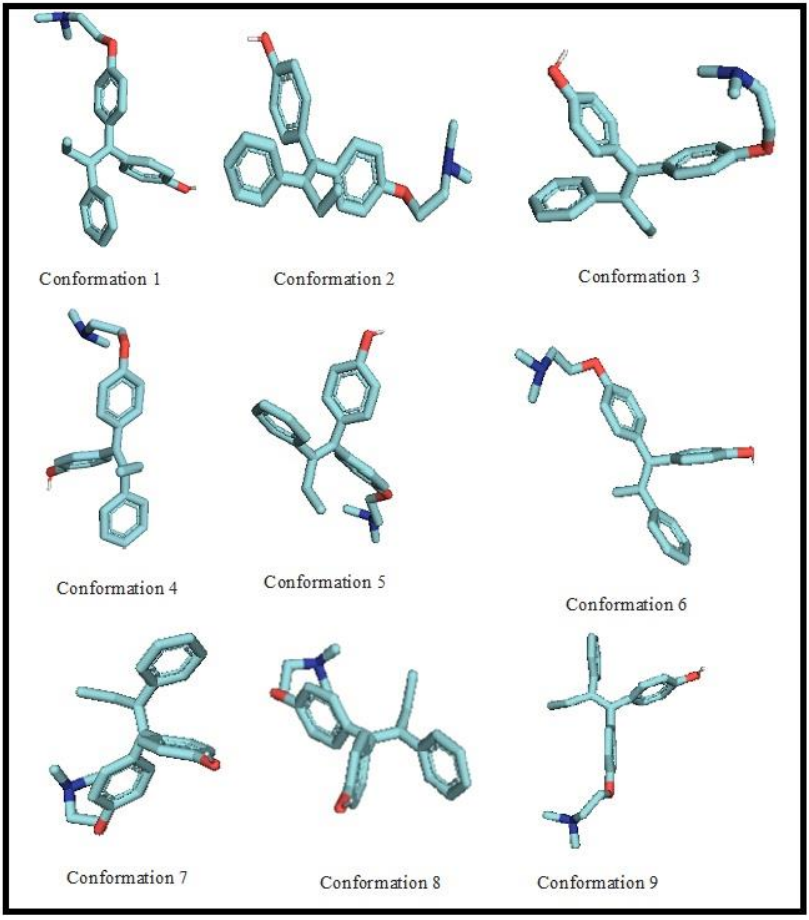


Fig 13: Table showing the 9 conformations of the output file

6] Problems and Solution

6.1 Resolve licensing issues on molecular docking nodes

One of the difficulties one encounters in carrying out molecular docking using KNIME is that by default, the docking nodes are proprietary and thus licensed. They are critical for the process of the automatic interaction of ligands with receptors but sometimes are not freely available. In the interest of research, researchers have modified open-source workflows in KNIME that can make use of open-source tools in automating the process like AutoDock Vina. Through using external scripts through Python nodes, docking operations can be executed without necessarily depending on the proprietary nodes.

This flexibility in user utilization also harnesses accessible tools so that the accessibility and collaboration of academic research continue to be enhanced. Moreover, by using file reading nodes, data of ligands and receptors may be imported into the format PDBQT to allow for a smooth automation process for docking and without having to depend on proprietary docking tools.

6.2 Tailoring KNIME for Open-Source Solutions

The open-source nature makes it an excellent tool for the integration of third-party tools in the form of RDKit, OpenBabel, and AutoDock Vina. Customization with Python scripts or shell commands allows the execution of molecular simulations and docking that would otherwise be paid licenses.

This flexibility in customization enables researchers to build workflows that are easily reproducible and can be shared quickly and then adapted to a multitude of research settings, making KNIME the best choice for budgeted large-scale drug discovery projects.

Errors and Troubleshooting in KNIME Workflows

Like most modular workflow environments, errors often originate from the lack of nodes, incorrect file formats, or improper versions of plugins. The most common errors include

6.3 Common Errors

1] *Data Input Errors*: Sometimes, workflow fails due to the reason of incorrect input data files (for example, ligand /receptor data). Thus, using nodes such as Molecule Type Cast helps maintain consistency in the data.
2] *Plugin Version Conflict*: Sometimes it may create conflicts if the versions of plugins differ in the workflow: for example, using RDKit or Python nodes with different versions. KNIME has the Preferences menu to make it easier to manage plugins and update them to ensure compatibility.
3] *Script errors in the Python nodes*: Syntax or runtime errors on custom scripts in the Python nodes can cause the workflow to crash. Console provides verbose error messages that will help you trace where exactly the issue is, so avoid such issues by using best practices in your workflow, where: before running scripts which need to mash data together, ensure your data is compatible, and all your plugins are up to date.

7] Conclusion and Future Prospects

7.1 Workflow Accomplishments:

This chapter demonstrated how a workflow for molecular docking based on KNIME could be developed-from importing data, through calculation of molecular descriptors and docking simulations. It bypasses the constraint from such licensed docking nodes by using open-source tools like RDKit and AutoDock Vina, which allow interaction studies of ligands and receptors at a low cost.
The inclusion of Python scripts in the workflow actually permitted automatic running of molecular docking with all flexibility and adaptability in the drug discovery pipeline.

7.2 Implications for Research in Drug Discovery

The flexibility and accessibility of KNIME workflows play a vital role in drug discovery. Complicated computational tasks can be streamlined into processes such as high-throughput virtual screening and docking studies through the recognition by researchers, hence hastening the identification process of potential drug candidates. Open-source tools further this

research at the global level by allowing reproducibility and by making collaboration easier.

Additional Requirements

A git hub link is provided from which the users can access data along with a python script for performing the auto docking in python node of KNME: https://github.com/tanishqua27/KNIME-Data-

References

1. Eweas, A. F., Maghrabi, I. A., & Namarneh, A. I. (2014). Advances in molecular modeling and docking as a tool for modern drug discovery. Der Pharma Chemica, 6(6), 211-228.
2. Adelusi, T. I., Oyedele, A. K., Boyenle, I. D., Ogunlana, A. T., Adeyemi, R. O., Ukachi, C. D., Idris, M. O., Olaoba, O. T., Adedotun, I. O., Kolawole, O. E., Xiaoxing, Y., & Abdul-Hammed, M. (2022). Molecular modeling in drug discovery. Informatics in Medicine Unlocked, 29, 100880. https://doi.org/10.1016/j.imu.2022.100880
3. Agarwal, S., & Mehrotra, R. J. J. C. (2016). An overview of molecular docking. JSM Chem, 4(2), 1024-1028.
4. DiTommaso, J. A. (2017). Implementation of virtual workflows in KNIME for medicinal chemistry. Journal of Student Science and Technology, 10(1). https://doi.org/10.13034/jsst.v10i1.123
5. Mazanetz, M. P., Goode, C. H., & Chudyk, E. I. (2020). Ligand- and structure-based drug design and optimization using KNIME. Current Medicinal Chemistry, 27(38), 6458-6479. https://doi.org/10.2174/0929867326666190409141016
6. Mazanetz, M. P., Marmon, R. J., Reisser, C. B. T., & Morao, I. (2012). Drug discovery applications for KNIME: An open-source data mining platform. Current Topics in Medicinal Chemistry, 12(18), 1965-1979.
7. Meinl, T., Jagla, B., & Berthold, M. R. (2012). Integrated data analysis with KNIME. In Elsevier eBooks (pp. 151-171).

8. Fillbrunn, A., Dietz, C., Pfeuffer, J., Rahn, R., Landrum, G. A., Berthold, M. R., & Cichonska, A. (2017). KNIME for reproducible cross-domain analysis of life science data. Journal of Biotechnology, 261, 149-156. https://doi.org/10.1016/j.jbiotec.2017.07.028
9. Das, S., Boucher, X. L., Rogers, C., Makowski, C., Chouinard-Decorte, F., Klein, K. O., Beck, N., Rioux, P., Brown, S. T., Mohaddes, Z., Zweber, C., Foing, V., Forest, M., O'Donnell, K. J., Clark, J., Meaney, M. J., Greenwood, C. M. T., & Evans, A. C. (2018). Integration of "omics" data and phenotypic data within a unified extensible multimodal framework. Frontiers in Neuroinformatics, 12. https://doi.org/10.3389/fninf.2018.00056
10. Hastreiter, M., Jeske, T., Hoser, J., Kluge, M., Ahomaa, K., Friedl, M., Kopetzky, S. J., Quell, J., Mewes, H., & Küffner, R. (2017). KNIME4NGS: A comprehensive toolbox for next generation sequencing analysis. Bioinformatics, 33(10), 1565-1567. https://doi.org/10.1093/bioinformatics/btx003
11. KNIME Analytics Platform. (2019).(https://www.knime.com/)
12. Xue, L., & Bajorath, J. (2000). Molecular descriptors in chemoinformatics, computational combinatorial chemistry, and virtual screening. Combinatorial Chemistry & High Throughput Screening, 3(5), 363-372.
13. Consonni, V., & Todeschini, R. (2009). Molecular descriptors. In Challenges and advances in computational chemistry and physics (pp. 29-102).
14. Lumley, J. A., Sharman, G., Wilkin, T., Hirst, M., Cobas, C., & Goebel, M. (2020). A KNIME workflow for automated structure verification. SLAS Discovery, 25(8), 950-956. https://doi.org/10.1177/2472555220946867
15. https://www.youtube.com/watch?v=8ISIeFKkoOE
16. Clarke, R., et al. (2001). The role of estrogen receptor-alpha and -beta in mediating chemoresistance in breast cancer. Cancer Research, 61(20), 7315-7320.

17. Yang, Y., et al. (2019). Betulin induces mitochondrial cytochrome c release associatedvapoptosis in human cancer cells. Molecular Carcinogenesis, 58(2), 203-216.
18. Tao, X., et al. (2018). Betulin inhibits triple-negative breast cancer metastasis and growth by blocking the epidermal growth factor receptor signaling pathway. PLoS One, 13(5), e0197565.