RDKit + Vernalis extensions workshop

Greg Landrum (KNIME)
Steve Roughley (Vernalis)

Workshop materials:
http://tinyurl.com/y4bdtab3
What we will do

• Exercise 1: MMPs
  – Generate MMPs for a dataset with the Vernalis nodes
  – Do some filtering
  – Visualize results

• Exercise 2: Patent analysis
  – Filter compounds by properties
  – Find “key” structure
  – Find “scaffold”
  – Perform R-Group decomposition
  – Enumerate missing compounds

Workshop materials: http://tinyurl.com/y4bdtab3
Brief intro to the RDKit
The RDKit: An open-source toolkit for cheminformatics

- Business-friendly BSD license
- Runs on Linux/Mac/Windows
- Commercial support available
- Releases every six months
- Active and engaged community
- Core data structures and algorithms in C++
- Usable from Python (2 or 3), C#, or Java
- Strong integration with other tools like KNIME, Jupyter, Pandas, and PostgreSQL
- Pretty good documentation
- Basic functionality highlights:
  - Chemical reactions
  - 2D depiction
  - Substructure searching
  - Canonical SMILES
  - Gasteiger-Marsili charges
  - Molecular standardization
- 2D Functionality highlights:
  - RECAP and BRICS support
  - Multi-molecule MCS
  - Similarity maps
  - Functional group filters
  - Diversity picking
- Supported fingerprint highlights:
  - Morgan/Feature Morgan (ECFP/FCFP-like)
  - RDKit (Daylight-like)
  - Atom-pairs and topological torsions
  - MACCS keys
  - Avalon
- Descriptor highlights:
  - Hall-Kier $\chi$ and $\kappa$ descriptors
  - SLogP, SMR, TPSA
  - MQN
  - “MOE-like” VSA
  - Compositional (number of donors, number of rings, number of heterocycles, etc.)
- 3D Functionality highlights:
  - 2D->3D conversion/conformational analysis via distance geometry
  - UFF and MMFF94/MMFF94S implementations for cleaning up structures
  - Feature maps and feature-map vectors
  - Shape-based similarity
  - RMSD-based molecule-molecule alignment
  - Open3DAlign implementation
  - Integration with PyMOL
  - Torsion Fingerprint Differences

www.rdkit.org
The RDKit code ecosystem

C++:
- Core data structures and algorithms

Python:
- Jupyter
- pandas
- Boost.Python
- PostgreSQL

Java
- KNIME

C#
- SWIG

The exact same implementation is available in all endpoints
The RDKit and KNIME

- Open-source wrappers for KNIME maintained by NIBR and the open-source community
- Useful for:
  - Descriptor calculation
  - Cleaning structures
  - Canonical SMILES and InChi conversion
  - Fingerprints
  - Scaffolds/substructures
  - Reaction simulation
  - Conformation generation
  - and more...

https://github.com/rdkit/knime-rdkit
Download data/exercises

http://tinyurl.com/y4bdtab3
Brief intro to MMPs
MMPs workflow
Background and Idea

• Construct and explore MMPs for a set of compounds measured on a collection of CYP assays
• Find a set of MMPs that show up multiple times for a given assay and that always increase/decrease activity
Tip: enabling “master-detail” selection in views
Patent workflow
Background and Idea

Here we’re working with compounds extracted from a pharmaceutical patent (data are from SureChEMBL)
- Find the “key” compound in a pharmaceutical patent
- Construct an estimate of the chemical scaffold, or “Markush” structure for the patent
- Do R-group decomposition to see what’s in the patent
- Enumerate some compounds that aren’t in the patent
Tip: Removing duplicates

• Use the GroupBy node
Tip: interactive filtering
Tip: Interactive selection of a subset of rows