Challenge One

Reconstruction of Chemical Structures from Hashed Fingerprints

>background

As of today, chemistry-centric data (e.g. chemical structures and experimental results on their properties) is typically "buried" in a few large pharmaceutical research organizations. Due to restrictions from patent / IP law, sharing of such data with the broader community is challenging – which limits the ability of academic research groups and smaller organizations to make significant contributions to the field. Approaches supporting the IP-preserving sharing of data would thus be highly interesting. The goal of this topic would be to "pressure test" various approaches and to assess the "reverse-engineering risk" of various descriptors.

>setup

Prepare a standard **Anaconda / Minconda** installation with typical packages such as numpy, pandas, scipy, scikit-learn, jupyter, etc.
Install domain-specific packages such as rdkit and mordred.

>data

We will share with the participants three datasets containing pre-calculated descriptors / identifiers for a set of approximately 1k chemical structures each. The chemical structures themselves will be kept confidential for the participants. The task for the participants would be to recover / reverse engineer as many chemical structures as possible from the descriptors.

Boehringer Ingelheim

>descriptors

Used descriptors are:

1. **RDKit Topological Fingerprints**. A string of 2048 bits encoding the presence / absence of certain features in a chemical structure. Fingerprints of this type can be used to measure the similarity of chemical structures (similarity searching, clustering), but also for machine learning. (See rdkit.org fingerpreinting and molecular similarity and a presentation on fingerprints).

2. **Mordred Descriptors**. A set of about 1800 numbers describing the properties of a chemical structure. Descriptors of this type are frequently used for machine learning applications (See github page on mordred).

3. **InChiKeys**. A unique hash of fixed length used to identify a chemical structure. These identifiers can be used to identify duplicates or to compare large compound collections in an efficient way. The hashing is based on a cryptographic hash function (SHA-256), so it is widely believed that brute-force enumeration of chemical structures is the only way to reverse-engineer an InChiKey (See about the InChi standard and international chemical identifier).

>approaches

A small subset of the shared datasets consists of chemical structures known in the public domain. Recovery of these structures is possible with a simple lookup in public databases (e.g. using resolver services like Cactus or by downloading databases like PubChem and implementing one's own lookup mechanism).

The remaining structures are proprietary, which makes their recovery significantly more challenging. One possible approach would be to use a generic optimization algorithm (see the Jensen Group's github or BenevolentAI's for an example) and to optimize a distance / similarity measure defined on the respective descriptors. Such approaches could be seeded with similar structures from a public database (see above links).

>bottom line

The challenge can be seen in **two different ways**:

1. Create an algorithm able to use as few molecular descriptors as possible to efficiently reverse engineer molecules

2. Create a molecular descriptor that totally prevents the reverse engineering process.