Prediction of Cytotoxicity Related PubChem Assays Using High-Content-Imaging Descriptors derived from Cell-Painting

Vorhersage zytotoxizitätsbezogener PubChem-Assays unter Verwendung von High-Content-Imaging-Deskriptoren aus Cell-Painting Assays

Master thesis by Luis Vollmers

Date of submission: March 15, 2021

1. Review: Prof. Dr. Katja Schmitz, TU Darmstadt

2. Review: Dr. Andreas Bender, University of Cambridge

Darmstadt

TECHNISCHE UNIVERSITÄT DARMSTADT

Erklärung zur Abschlussarbeit gemäß §22 Abs. 7 und §23 Abs. 7 APB der TU Darmstadt

Hiermit versichere ich, Luis Vollmers, die vorliegende Masterarbeit ohne Hilfe Dritter und nur mit den angegebenen Quellen und Hilfsmitteln angefertigt zu haben. Alle Stellen, die Quellen entnommen wurden, sind als solche kenntlich gemacht worden. Diese Arbeit hat in gleicher oder ähnlicher Form noch keiner Prüfungsbehörde vorgelegen.

Mir ist bekannt, dass im Fall eines Plagiats (§38 Abs. 2 APB) ein Täuschungsversuch vorliegt, der dazu führt, dass die Arbeit mit 5,0 bewertet und damit ein Prüfungsversuch verbraucht wird. Abschlussarbeiten dürfen nur einmal wiederholt werden.

Bei der abgegebenen Thesis stimmen die schriftliche und die zur Archivierung eingereichte elektronische Fassung gemäß §23 Abs. 7 APB überein.

Bei einer Thesis des Fachbereichs Architektur entspricht die eingereichte elektronische Fassung dem vorgestellten Modell und den vorgelegten Plänen.

Darmstadt, March 15, 2021

L. Vollmers

Contents

Prediction of Cytotoxicity Related PubChem Assays Using High-Content-Imaging Descriptors derived from Cell-Painting

Das Vorhersagepotenzial neuartiger High-Content-Imaging Datensätze, aus Cell-Painting-Assays, soll in dieser Masterthesis anhand von statistischen und praktischen Methoden überprüft werden. Dabei werden die prozessierten Rohdaten mit Datenbanken verglichen, die Informationen über toxikologische Endpunkte enthalten. Weiterhin werden verschiedene Machine Learning Algorithmen anhand der Datensätze trainiert und die Ergebnisse extensiv analysiert. Dabei wird besonderes Augenmerk auf die Unterschiede im Vorhersagepotential zwischen den einzelnen Endpunkten gelegt, um daraus Informationen über die Anwendbarkeit der Cell-Painting Datensätze zu gewinnen.

•**Applied computing**~**Physical sciences and engineering**~**Chemistry**
•Applied computing~Life and medical sciences~Computational biology ~Recognition of genes and regulatory elements
•Applied computing~Life and medical sciences~Systems biology
•Computing methodologies~Machine learning~Machine learning algorithms~Feature selection
•Computing methodologies~Machine learning~Machine learning approaches~Classification and regression trees
•Computing methodologies~Machine learning~Cross-validation

1 Summary

The pharmaceutical industry is centred around small molecules and their effects. Apart from the curative effect, the absence of adverse or toxicological effects is cardinal. However, toxicity is as elusive as it is important. A simple definition is: 'toxicology is the science of adverse effects of chemicals on living organisms'.[1] However, this definition comprises several caveats. What is the organism? Where do therapeutic and adverse effects start and end? Even for the comparably simple cytotoxicity the mechanisms are manifold and difficult to unravel. Hence, it remains obscure which characteristics a compound has to combine to be labelled as toxic.
One attempt to illuminate these characteristics are novel cell-painting (CP) assays. For a CP assay, cells are perturbed by libraries of small compounds, which might affect the cellular morphology before images are taken via automated fluorescence microscopy. Five fluorescent channels are used for imaging, and these channels correspond to certain cell organelles.[2] Therefore CP data contains information about cell structure variations caused by each compound. Which sub-information is actually valuable within these morphological fingerprints remains elusive. Therefore a significant part of the project presented here is dedicated to comparatively exploring the CP data and their predictive capabilities. They will be compared against different descriptors for a variety of bioassays. The CP data used in this project contains roughly $30\,000$ compounds and 1800 features.[3]
In chemistry, the structure determines the properties of a compound or substance. Therefore, apart from CP, structural fingerprints are used as a benchmark descriptor set for comparison. In this project extended-connectivity fingerprints (ECFPs) were used to encode the compounds' structures as numerical features.
This work is concerned with morphological changes that correspond to toxicity. Thus, the CP data were combined with toxicological endpoints from specific assays selected from the PubChem database. The selection process implemented a minimum number of active compounds, a size criterion and the occurrence of toxicologically relevant targets.[4]
After the selected assays were combined with each of their descriptors, machine learning models were trained, and their predictive power was evaluated against specific metrics. The predictions can be divided into four cycles. In the first cycle, the CP data are used as descriptors, the second cycle used the structural fingerprints, and the third cycle used a subset of both. A rigorous feature engineering process selected the subsets. The last cycle skipped the feature engineering and combined all CP and ECFP descriptors into one large set of inputs.
The evaluation of the prediction metrics illuminates which strengths and shortcomings the morphological fingerprints entail compared to the structural fingerprints. It turned out that

there are two groups of assays: those PubChem assays that are generally better predicted with CP features and those that have higher predictive potential when using ECFP. Additionally, it was revealed that ECFP comprise higher specificity compared to CP data which in turn show higher sensitivity. A high sensitivity means the prediction rarely mislabels a sample as negative (e.g. non-toxic) compared to the number of correctly labelled positive samples (e.g. toxic compounds.). Based on these results, CP is better suited for toxicity prediction and drug safety evaluations since any mislabelled, toxic compound can lead to expenses or even damage to health.
Furthermore, based on the fluorescent channels from which the data stems, an enrichment measure was introduced and calculated for the aforementioned two groups of PubChem assays. This enrichment connects predictive performance with cell organelle activity. The hypothesis was that PubChem assays, reliably predictable from CP data, should exhibit increased enrichment, which was the case for four out of five fluorescence microscopy channels.
As a next step, phenotypic terms were manually generated to categorize the different PubChem assays. These terms corresponded to cellular mechanisms or morphological processes and were generated unbiasedly. Nevertheless, they are not extensive and subject to human error. The phenotypic annotations that are found to be enriched for successful modelling approaches might guide the pre-selection of bioassays in future projects. The enrichment analysis of phenotypic annotations detected that PubChem assays that could be well predicted via CP data are related to immune response, genotoxicity and genome regulation and cell death.
Finally, the assays are assigned gene ontology (GO) terms obtained from the GO database.[49,50] These terms comprise a controlled, structured vocabulary that explicitly describes the molecular function and biological processes of a given gene product. For PubChem assays associated with a protein target, the GO terms are collected. If an assay is particularly well predicted via CP descriptors, the associated GO terms can relate this finding to cellular function. Even though the analysis with go terms suffers from a minimal sample size, it was found that CP related assays usually correspond to processes concerning deoxyribonucleic acid (DNA) and other macromolecules. This finding is in good agreement with the analysis of the channel enrichment as well as the phenotypic enrichment.

1.1 Zusammenfassung

Diese Arbeit befasst sich mit zellulären, morphologischen Veränderungen in Zusammenhang mit Toxizität. CP-Daten wurden hierbei mit toxikologischen Endpunkten aus spezifischen Assays kombiniert, die aus der PubChem-Datenbank ausgewählt wurden. Das Auswahlverfahren imple-mentierte eine Mindestanzahl von Wirkstoffen, ein Größenkriterium und das Auftreten

toxiko-logisch relevanter Endpunkte.[4]
Nachdem die ausgewählten Assays mit ihren Deskriptoren kombiniert worden waren, wurden Modelle für machine learning (ML) trainiert und ihre Vorhersagekraft anhand spezifischer Kenngrößen bewertet. Die Vorhersagen können in vier Zyklen unterteilt werden. Im ersten Zyklus wurden die CP-Daten als Deskriptoren verwendet, im zweiten Zyklus wurden strukturelle Merkmale verwendet, und im dritten Zyklus wurde eine Teilmenge beider verwendet. Ein ausgiebiger Feature-Engineering-Prozess wählte die Teilmengen aus. Im letzten Zyklus wurde das Feature-Engineering übersprungen und alle CP- und ECFP-Deskriptoren zu einem großen Datensatz zusammengefasst.
Die Auswertung der Vorhersagemetriken zeigt, welche Stärken und Mängel die morphologischen Fingerabdrücke im Vergleich zu den strukturellen Merkmalen aufweisen. Es stellte sich heraus, dass es zwei Gruppen von Assays gibt: jene PubChem-Assays, die mit CP-Daten im Allgemeinen besser vorhergesagt werden können, und jene, die bei Verwendung von ECFP ein höheres Vorhersagepotential haben. Zusätzlich wurde gezeigt, dass ECFPs eine höhere Spezifität aufweisen als CP-Daten, die andererseits eine höhere Empfindlichkeit zeigen. Eine hohe Empfindlichkeit bedeutet für eine Vorhersage, dass eine Probe im Vergleich zur Anzahl korrekt markierter positiver Proben (z. B. toxische Verbindungen) selten falsch als negativ (z. B. nicht toxisch) vorausgesagt wird. Basierend auf diesen Ergebnissen sind CP-Daten besser für die Vorhersage der Toxizität und die Bewertung der Arzneimittelsicherheit geeignet, da eine falsch ausgewiesene, toxische Verbindung zu Kosten oder sogar zu Gesundheitsschäden führen kann.
Darüber hinaus wurde basierend auf den Daten der Fluoreszenzmikroskopiekanäle eine Enrichment-Größe eingeführt und für die oben genannten zwei Gruppen von PubChem-Assays berechnet. Diese Enrichment-Größe verbindet die Vorhersageleistung mit der Aktivität der Zellorganellen. Die Hypothese war, dass PubChem-Assays, die zuverlässig aus CP-Daten vorhersagbar sind, eine erhöhte Enrichment-Größe aufweisen sollten, was bei vier von fünf Fluoreszenzmikroskopiekanälen der Fall war.
Als nächster Schritt wurden phänotypische Kennwörter manuell generiert, um die verschiedenen PubChem-Assays zu kategorisieren. Diese Begriffe entsprachen zellulären Mechanismen oder morphologischen Prozessen und wurden unvoreingenommen generiert. Trotzdem unterliegen sie menschlichen Fehlern und stellen keine vollständige Liste dar. Die phänotypischen Annotationen, die für erfolgreiche ML Modelle angereichert sind, könnten die Vorauswahl von Bioassays in zukünftigen Projekten vereinfachen. Die Enrichment-Analyse phänotypischer Annotationen ergab, dass PubChem-Assays, die über CP-Daten gut vorhergesagt werden konnten, mit Immunantworten, Genotoxizität und Genom-regulation sowie Zelltod zusammenhängen.
Schließlich werden den Assays GO-Begriffe zugewiesen, die aus der GO-Datenbank stammen.[49,50] Diese Begriffe umfassen ein kontrolliertes, strukturiertes Vokabular, das die molekulare Funktion und die biologischen Prozesse eines bestimmten Genprodukts explizit beschreibt. Für PubChem-

Assays, sofern sie einem Protein Target zugeordnet sind, wurden die GO-Begriffe gesammelt. Wenn ein Assay über CP-Deskriptoren besonders gut vorhergesagt wird, können die zugehörigen GO-Terme diesen Befund mit der Zellfunktion in Beziehung setzen. Obwohl die Analyse mit GO-Begriffen durch eine kleine Stichprobengröße eingeschränkt sind, wurde festgestellt, dass CP-bezogene Assays normalerweise Prozessen entsprechen, die DNA und andere Makromoleküle betreffen. Dieser Befund stimmt gut mit dem Enrichment der Fluoreszenzmikroskopiekanälen sowie den phänotypischen Annotationen überein.

2 Introduction

Currently, pharmacological drug development focuses on well-established biochemistry based approaches to find and optimize new drugs. However, the challenges these methods face are manifold. High cost related to drug failure rates during various clinical trials and commercialization bottleneck the industry. Another important aspect is the occurrence of adverse drug reactions subjecting patients to hospitalization, possibly ending up fatal. Therefore, the pharmaceutical industry is not only facing high financial risks but also humanitarian issues that strain the trust-based relationship between the industry, physicians and patients.[5]

Academia demonstrated computational tools to be employable to many health industry challenges, such as costs of drug target validation, drug safety, and commercialization.[6,7] Albeit, chemo- and bioinformatics are novel and complex disciplines recently fostered by technological advancements in high-throughput methods and big-data analysis. Thus, the health industry does not yet benefit from promises like computer-aided identification of drugs and drug targets on a large scale.

New high-throughput methods and automated microscopy gave rise to the development of high-content imaging, which is frequently used to record small compound perturbations inflicted on biological systems. High-content-imaging applies up to six fluorescent dyes allowing to portray up to five different compartments per cell simultaneously. This method also referred to as cell-painting (CP) captures compound perturbed biological systems and automatically resolves cellular characteristics. Computational models can interpret these in the context of a morphological fingerprint (or CP feature vectors).[3,8]

The raw images from high-content imaging are processed, mostly by the software CellProfiler[2] which extracts up to 1800 numerical features per image (e.g. nucleus shape, endoplasmatic reticulum (ER) texture, etc.). Features from CP assays can be interpreted as a morphological fingerprint, unique for each compound.[8] By now, many different CP assays have been conducted. A widely used CP data set was generated by Bray *et al.*[9] The images were recorded with sixfold fluorescence staining for imaging five crucial cellular organelles (further information see section 4.4).[3] The concept of CP is visualized in figure 2.1.

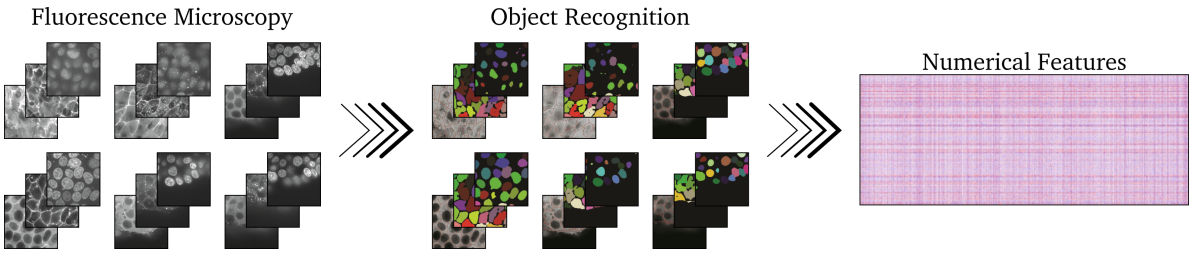


Figure 2.1: Visualization of a CP assay. First cellular images are generated by fluorescence microscopy, then cellular objects and compartments are recognized by CellProfiler from which a large data table is generated containing the morphological fingerprint for each compound. Figure from Rohban *et al.* and Carpenter *et al.*, modified.[2,10]

Recently, Gustafsdottir *et al.*[11] used CP data to link morphological states to mechanisms of action (MoAs) via gene expression data. Hierarchical clustering was used to find clusters of compounds, addressing the same set of genes and, therefore, the same biochemical pathway. The results obtained from this CP based approach mirrored the findings in the literature, which showed that CP data is directly correlated to cellular pathways responding to compound perturbations.

Nassiri and McCall[12] used different CP assay data[13] and compound perturbed gene expression data in the context of machine learning methods. They used a LASSO model to predict cell morphological features against similar gene expression profiles. In-depth analysis of the results revealed strong model predictiveness among compounds that steer gene expression in the same direction, suggesting common MoAs. In addition to linking CP data to MoAs they revealed direct relations between compounds' MoAs based on machine learning model performance.

Furthermore, Rohban *et al.*[10] transduced U-2 OS cells with lentiviral particles carrying cDNA constructs for gene overexpression.[14,15] In their approach, they conducted a CP assay to annotate the overexpressed genes with morphological fingerprints (numerical CP features). After calculating the Pearson correlation between each morphological fingerprint, they used hierarchical clustering resulting in 25 clusters for 110 overexpressed genes. The clusters generated from the CP data agglomerated genes that correspond to similar or identical pathways showing that genes can be connected using relatively inexpensive CP assays. Furthermore, they predicted an unknown relationship between the Hippo- and NF-κB-pathway.

Lapins and Spjuth[16] annotated compounds of the CP data from Bray *et al.*[3] and from the CMap[17] (gene expression profiles) with their MoA or target protein. The information about the compound-wise MoA and targets was obtained from the Drug Repurposing Hub or the Touchstone database. In total, they annotated 1484 compounds present in CP and CMap data with 234 MoAs or targets. As the third set of descriptors, structural fingerprints were generated. For several targets and MoAs a trained random forest classifier (RFC) could present significant discriminatory power (AUC-ROC>0.7). Furthermore, it was found that the three different descriptors were complementary to a certain degree, each excelling at different MoAs or targets.

Lapins and Spjuth not only showed that CP data could be used to predict compounds' MoA but also that a combination with other identifiers is likely to enhance their applicability domain.
Simm *et al.*[8] studied a CP assay specifically designed for glucocorticoid receptor (GCR) nuclear translocation. After treating H4 brain neuroglioma cells with $524\,371$ compounds, hydrocortisone was added to stimulate GCR translocation. Next, the treated cells were stained, imaged and processed analogously to the work of Gustafsdottir *et al.*[11] The $524\,371$ compounds were not only annotated with morphological information, but also with target activity information from 600 biochemical assays. Noticeably, most compounds were covered in few assays only, amounting to a fill rate of $1.6\,\%$. From this sparse activity matrix, they built a ML model with CP data as side information to predict all labels within the activity matrix. They evaluated the discriminatory power of their model and 34 bioassays (out of 600) showed high predictivity (AUC-ROC>0.9). One of the assays was part of an ongoing discovery project. Within this assay, the highest-ranking 342 compounds, by matrix factorization, were experimentally tested. 141 (41.2%) of these resulted in submicromolar hits which means a 60-fold enrichment over the initial high-throughput-screening (HTS). Another assay with an AUC-ROC greater 0.9 was part of an ongoing drug discovery project and could achieve a 250-fold hit enrichment over the initial HTS in an analogous way. Their work presented CP data as highly informative descriptors that might be repurposed for the prediction of sparse activity matrices. Additionally, they demonstrated their potential in ongoing drug discovery projects.[18]
Data science in drug development and pharmacology has great potential. However, some caveats require carefulness when working with CP data to access drug safety. The first one is imbalanced data. An assay testing compounds on a potential target will always feature fewer actives than inactive. Chawla *et al.*[19] proposed a technique that mitigates this effect called synthetic minority oversampling technique (SMOTE). This technique allows generating synthetic data that fit into the distribution of the real data points. Another obstacle is the high dimensionality of CP data. Only a comparably small number of features contains most of the information necessary to predict a given target. Thus, statistical tools are applied for sophisticated feature selection in CP studies. A CP data set containing too many features is bound to overfit the data and give overoptimistic predictions on the test set without generalizing particularly well. The project above of Rohban *et al.*[10] tackled this problem by principal component analysis (PCA). Thereby reducing the number of features from 2769 to 158 which comprise most of the variance.
In this explorative ML project, targets obtained from bioassays published on PubChem are predicted with regard to descriptors from the CP data set of Bray *et al.*[3] The PubChem assays are selected based on their relation to targets presumably contributing to cytotoxicity.[4] The results are compared to the small molecules' structural fingerprints, and the performance metrics are analyzed extensively. From this analysis, conclusions can be drawn, whether and when to use

CP data. Insights gained from this project might guide future decision making when it comes to the prediction of biological endpoints.

3 Scientific Aim

This project aims to generate heuristics that simplify working with CP data. Generally, CP data can be used as descriptors to predict targets obtained from compound associated biochemical readouts. The mechanistic relation between a biochemical readout and its predictivity via CP descriptors is poorly understood. Therefore, this work aims at understanding the results obtained from RFC prediction and linking the results to cellular mechanisms and the concept of cytotoxicity.

Conceptually, this means finding bioassays whose endpoints are related to toxicity for annotating the CP descriptors. Since this is a comparative approach, structural fingerprints (ECFPs) are used as another descriptor set. Both annotated data sets are entered into an ML model, and this model's discriminatory power is evaluated using generic statistical metrics. Another model is trained using the combined set of descriptors which is evaluated analogously.

The detailed procedure starts by preprocessing the CP raw image data into a ML-ready data frame. Next, the bioassays are selected from PubChem,[20] based on size and their relation to cytotoxicity. Every bioassay data frame is combined with the CP data frame to obtain annotated sets containing inputs as well as targets for prediction.

For comparison, structural descriptors are generated for all data sets using $\mathrm{sklearn}$-funtionalities.[21] To this end, the annotations in all data sets are highly imbalanced, comprising mostly samples labelled as inactive. Herein, this problem is tackled by applying SMOTE to the data sets in combination with undersampling of the majority label (here: inactive).

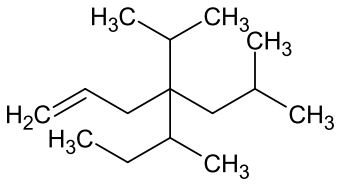
Two RFC models are trained on each data set, and their predictive power is evaluated. Furthermore, to examine if one descriptor's shortcomings can be mitigated by the other, selected features from both descriptors are combined, and another model is trained and evaluated. Since there is no apparent way to select features manually, statistical methods can be used to conduct features selection meaningfully. PCA, minimal-redundancy-maximal-relevance criterion (MRMR) and random forest feature importance are applied to score and select features from each set of descriptors for merging. Eventually, a fourth RFC model is trained and evaluated using the complete feature sets from both descriptors.

Based on the prediction evaluation and feature selection process, a rigorous analysis is conducted to explain the results with respect to cellular morphology and cytotoxicity. The focus lies in detecting and analyzing prediction similarities and dissimilarities between either descriptor sets, bioassays or groups characterized by their performance. Thereby, patterns can be detected that transform into heuristics which facilitate the application of CP data to ML problems in the future.

4 Theoretical Background

4.1 Simplified Molecular Input Line Entry Specification - SMILES

The simplified molecular input line entry specification (SMILES) refer to a specific formalism to generate identifiers for chemical compounds that are suited for chemists and computational input. The identifier, in this case, is deduced from a two-dimensional graph of the chemical structure. The result is a series of characters that contain mostly alphanumeric symbols, brackets and some other symbols. The selection of those symbols follows a specific order and a specific set of rules. The set of rules addresses six categories: atoms, bonds, branches, cyclic structures, disconnected structures and aromaticity. Also, SMILES considers stereochemical information. However, that is not mandatory since the initial approach to SMILES covers solely two-dimensional information.[22] Atoms are labelled by their element symbol. All elements of a SMILES string are written in square brackets with the exceptions of the organic subset, i.e. B, C, N, O, P, S, F, Cl, Br, and I. Hydrogen atoms have further specifications. They can appear implicitly with members of the organic subset. In that case, the remainder of the lowest valence is filled with hydrogen atoms. For example, [C] refers to CH_4. Explicit notation of hydrogen atoms occurs when they are attached to an element that is not part of the organic subset. Given a metal M, the nomenclature of four hydrogen atoms attached to that metal is [MH4]. Hydrogen can also be mentioned on its own in brackets [H], e.g. in its molecular form H_2. Charges are represented with a plus or minus with their respective count inside a bracket.[22] Bonds within the SMILES nomenclature are omitted if they are either aromatic or single covalent bonds. Double bonds are represented with '=', and triple bonds are represented by '#'. Ionic bonds are not explicitly denoted by the SMILES algorithm. An ion pair is written as two disconnected structures with formal charges to them. Tautomeric bonds are not explicitly denoted either. One of the possible structures is translated into the SMILES string, be it the enol or keto variation.[22]
Branches are depcited in parenthesis. 5-Methyl-4-(2-methylpropyl)-4-(propan-2-yl)hept-1-ene is an example of nested branching using nested parenthesis. The name of the structure is according to the convention of International Union of Pure and Applied Chemistry (IUPAC).[23] The structure and SMILES is shown in figure 4.1.



5-methyl-4-(2-methylpropyl)-4-(propan-2-yl)hept-1-ene

CC(C)CC(CC=C)(C(C)C)C(C)CC

Figure 4.1: On the left, side a model compound is shown as an example of nested branching in SMILES. On the right, the IUPAC name and its SMILES string are shown. The SMILES string features parenthesis that imply branching and nested branching.[22]

Cyclic structures are written linearly by breaking a single or aromatic bond within the cycles. Next, the broken bonds are arbitrarily labelled by writing the formerly connecting elements right in front of a number assigned to the broken bond. An illustrating example is shown in figure 4.2.

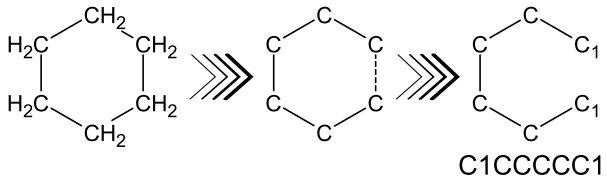
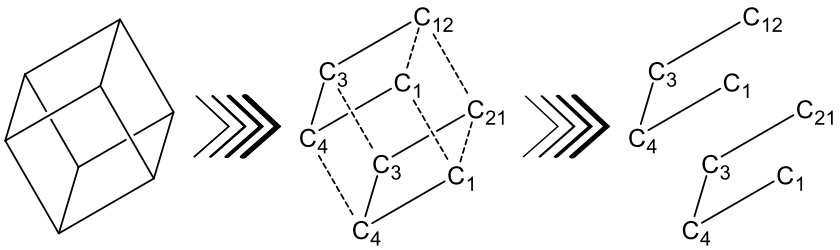


Figure 4.2: Cyclohexane as an example for a cyclic structure. First, the explicit hydrogens are exchanged for implicit ones, and the ring is linearized by conceptually breaking a bond implied by the dashed line. The carbons connected by the dashed line are being labelled, and the resulting SMILES string is shown below the right-hand structure.[22]

A single atom can be part of multiple rings which is then accounted for by using two or three single digits in sequence. However, for structures with more than ten rings, double digits are separated with a prefacing per cent sign. Also, a single digit can be reused for multiple broken bonds without creating ambiguity. A SMILES string is read from left to right, and a ring closes on the first repetition of a respective digit. Cubane is an example that has multiple rings. In figure 4.3 the generation of a SMILES string is shown with the usage of the digit '1'.[22]



C12C3C4C1C1C4C3C21

Figure 4.3: Cubane as an example of a structure that has multiple cycles. On the left, the structure is shown without explicit hydrogen atoms. In the middle picture, the bonds that are artificially broken to linearize the molecule for the SMILES string are shown in dashes. On the very right, the skeleton structure resembles the SMILES string, that is written below the molecular representations.[22]

A SMILES string is read from left to right, and a ring closes on the first repetition of a respective digit. Disconnected structures are written as individual SMILES strings separated by a comma.[22] Aromaticity is denoted by writing the atoms that are part of an aromatic cycle in lower case letters. Aromaticity is detected by applying an extended definition of Hückel's rule. Another noteworthy convention is the treatment of aromatic nitrogen atoms. A nitrogen atom that is embraced by two aromatic bonds has no valency left per default. However, for aromatic nitrogen that is connected to a hydrogen atom, the hydrogen atom is specified as shown in figure 4.4.[22]

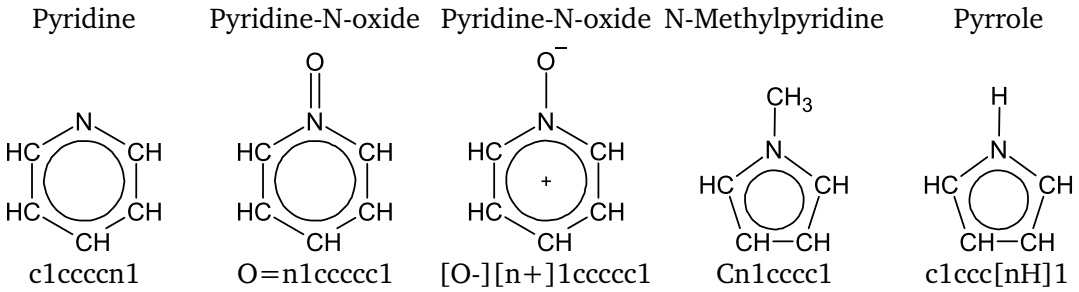
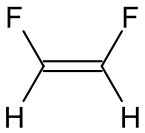


Figure 4.4: Different instances of aromatic nitrogen. Notice that the SMILES string of pyrrole contains an additional hydrogen atom that preceded the aromatic nitrogen. The aromaticity of an atom is denoted by writing it in lower case letters.[22]

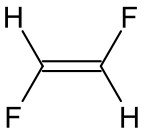
Furthermore, the SMILES algorithm introduces a convention for labelling double bond configurations and chirality. The double bond configuration is indicated by placing '/' or '\' between the atom constituting the double bond and their subsequent bonding partners. The indicators can be understood as a single bond type that gives information about their relative orientation. An example for (Z)-1,2-difluoroethene and (E)-1,2-difluoroethene is given in figure 4.5.[24]

(Z)-1,2-Difluoroethylen



F\C=C/F
F/C=C\F

(E)-1,2-Difluoroethylen

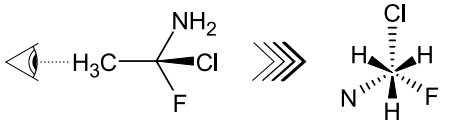


F\C=C\F
F/C=C/F

Figure 4.5: Example of double bond configuration in SMILES notation. On the left (Z)-1,2-difluoroethene is shown and on the right is (E)-1,2-difluoroethene with their SMILES notation. Both notations shown for each structure are valid.[24]

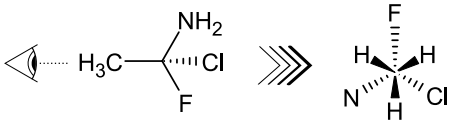
Chirality is assigned to chiral tetrahedral centres and any other chiral centre, e.g. allene-like or square planar centres. Herein, the SMILES notation is explained in the context of tetrahedral chirality centres, which is the most straightforward instance of chirality in organic chemistry. A chiral centre can not be the terminal node in a molecular representation since a terminal node is either only connected to hydrogen atoms if any at all. With that in mind, the convention for tetrahedral chirality is most easily explained by investigating an example which is (1S)-1-chloro-1-fluoroethan-1-amine which can be seen in figure 4.6. The whole molecule is viewed along the CC-bond. Necessarily, the SMILES string contains the central C-atom's binding partners in a specific order. This sequence can either correspond to the clockwise or anticlockwise binding partners' order when the molecule is viewed along the CC axis. Should the order be anticlockwise, an '@' is inserted after the central C-atom in brackets. The '@' is a visual mnemonic since it depicts an anticlockwise rotation around the central circle. For a clockwise order, '@@' is used instead of a single '@'.

(1S)-1-chloro-1-fluoroethan-1-amine



C[C@@](Cl)(F)N or C[C@](F)(Cl)N

(1R)-1-chloro-1-fluoroethan-1-amine



C[C@](Cl)(F)N or C[C@@](F)(Cl)N

Figure 4.6: Example of enantiomer SMILES strings.[24] Both molecules are pictured in the same way. Above each depiction is the name of the chemical formula followed by a schematic. The eye indicates the point of view along the CC axis. The resulting view of the structure is shown on the right of each subfigure. Written below are two equally adequate SMILES strings for each structure.

4.2 Canonical SMILES

In general, SMILES strings do not claim to be unique identifiers. There are many equivalent options to generate a SMILES string for a given structure. Nowadays, computational biochemical

research accesses structures from many different sources and databases, making the requirement of a unique identifier evident. The SMILES notation was developed with this objective in mind. So-called canonical SMILES strings fulfil this objective. They are based on the same set of rules described in section 4.1. The algorithm can be partitioned into two parts: the CANON part and the GENES part. The CANON part labels the atoms of the molecular structure canonically, i.e. a unique way based on the structural topology. The GENES part generates the unique SMILES from the aforementioned rule set (see section 4.1) and the canonical labelling.[25]

For finding a unique way of labelling atoms in a molecule, invariant structural properties are necessary. Thus, five properties are considered atomic invariants. Those would be (1) the number of connections, (2) the number of non-hydrogen bonds, (3) the atomic number, (4) the sign of the charge and (5) the number of attached hydrogens. They are called 'invariant' because they are invariant to atomic order changes within a structural notation, and they are exchangeable as long as this principle is not violated. The numbers in the parenthesis in front of each property correspond to a pre-defined prioritization. In summary, the atomic invariants assign five integers to every atom in a molecule. The so-called individual invariant can be obtained by simply combining these integers in order of their prioritization. Given the methyl carbon of 2-(acetyloxy)benzoic acid in figure 4.7 with the individual invariant 110603. From this individual invariant, it can be concluded that this atom has 1 connection, 1 bond to a non-hydrogen neighbour, an atomic number of 06, 0 charge and 3 attached hydrogen atoms. Other atomic properties like isotopic mass and local chirality can be added if these six properties are not sufficient to discriminate all distinguishable nodes from each other. It is noteworthy that some nodes are symmetric and require a tie-breaking function for absolute uniqueness (see next paragraph).[25]

After assigning every atom an individual variant, those are compared among all constituting atoms and are ranked by magnitude. The final atom labels depend on the topology, as well. Therefore, the nodes (atoms) rank is extended by its corresponding prime (for 1, 2, 3, the corresponding primes are 2, 3, 5). The so-called new invariant is obtained by multiplying the corresponding primes of all neighbours for every atom. Afterwards, ranks are assigned again, based on their current rank and new invariant. The procedure is repeated iteratively until the combined invariant is not changing anymore. Should there be constitutionally symmetric nodes present in the molecular graph, it becomes necessary to break ties since the symmetric groups make it impossible to find a ranking that offers a completely ordered set of nodes necessary for finding a canonical SMILES representation. For tie-breaking, all ranks are doubled, and the first instance of an asymmetric node is decremented by one. The resulting node ranking is considered a new invariant set that goes through the aforementioned iterative process of corresponding prime multiplication until it is no longer changing. After every rank is of the combined invariant is unique and not changing upon further iteration, the uniquely ordered

ranking has been accomplished.[25] The canonical labelling process for 2-(acetyloxy)benzoic acid is shown in figure 4.7.

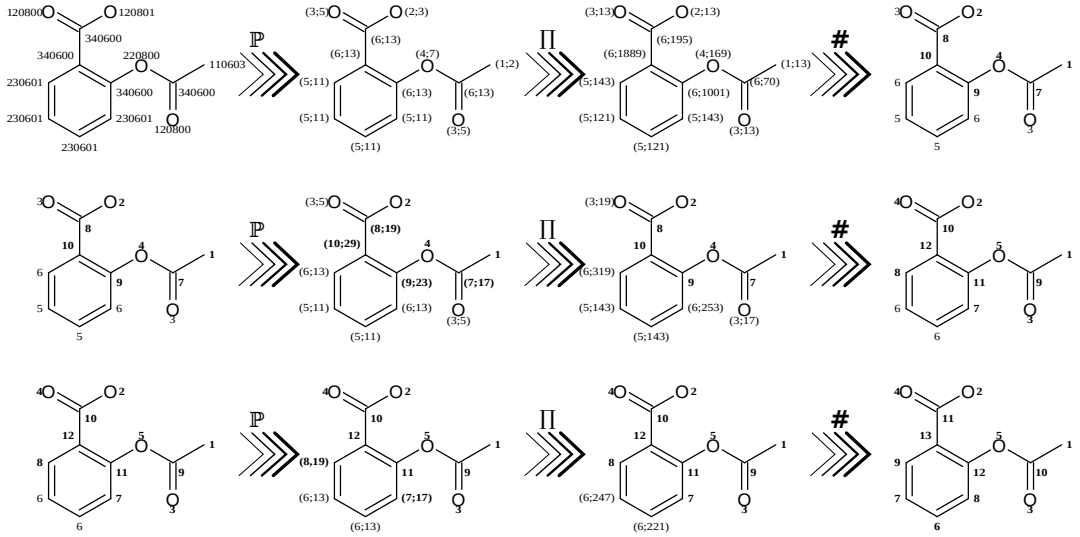


Figure 4.7: Canonical labelling with 2-(acetyloxy)benzoic acid. Every row corresponds to consecutive iterations of the CANON algorithm. The blackboard bold $\mathbb{P}$ denotes finding corresponding primes. The greek letter Π denotes taking the prime products of all atoms, and the hashtag denotes atoms' ranking. Bold numbers denote ranks that reached invariance.

THE GENES part of the algorithm can utilize the uniquely ordered ranking to chose the start node and prioritize at branching points, etc. As an entry point for the generation of canonical SMILES, the node with the lowest ranking is chosen. Branching decisions are made in the same fashion, i.e. the branching option with the lowest rank is chosen and followed until a dead end has been reached. A special rule applies when branching into a ring with a double or triple bond. To avoid opening the ring at any multi-bond, the algorithm will always branch towards the multi-bond. Also, the ring-opening digits must be in the order of ring-opening nodes.
Conclusively, a unique SMILES string can be assigned by first generating a unique invariant rank for every node that incorporates invariant atomic properties and topological information and then using these ranks as decision indicators for branching and cycles.[25]

4.3 Extended-Connectivity Fingerprints

The ECFP is a structural fingerprint that was developed to capture molecular features relevant for molecular activity.[26] ECFPs are also commonly referred to as Morgan-fingerprints since their development is partially based on the Morgan-algorithm which pursues rigorous canonicalisation similar to the CANON algorithm in section 4.2.[27] The procedure of the algorithm can be

distinguished into two parts: the initialization (or zeroeth iteration) and the iteration process. For initialization, all atoms in the molecule are given an identifier that is computed from six atomic invariants, similar to the ones in section 4.2 canonical SMILES algorithm (number of connections, number of bonds, atomic number, atomic mass, atomic charge, number of attached hydrogens).[24] However, a seventh invariant is taken into account: whether the atom is part of at least one ring structure. A hashing algorithm maps the atomic invariants to a 32-bit integer. Any functional hashing algorithm that reproducibly maps the input onto a 32-bit integer is a sensible choice since the only requirement is that a variant is uniquely mapped to the same 32-bit integer every time it is hashed. The 32-bit integer is also referred to as the core identifier. Within the initialization step, it corresponds to a substructure that contains information about one atom and its bonds and is appended to a list called ECFP-set. After completion of the algorithm, the ECFP-set is equal to the fingerprint.[26] Therefore, the result of the initialization are a set of core identifiers $[c_0^{(0)}, c_1^{(0)}, c_2^{(0)}, ..., c_n^{(0)}]$. The core identifier for atom 1 of the initialization (zeroeth iteration) would be $c_1^{(0)}$.

After the initialization the first iteration starts by randomly picking an atom, let that be atom 4. Next, the iteration number (1) and the core identifier, $c_4^{(0)}$, are appended to a temporary list. Afterwards, the neighbouring core identifiers are appended to the temporary list together with their respective bond order. b_{4j} denotes the bond order between atom 4 and its neighbour, j. The bond order can either be 1, 2, 3 or 4 for aromatic bonds. Let the resulting temporary list have the following format $[1, c_4^{(0)}, b_{34}, c_3^{(0)}, b_{45}, c_5^{(0)}, b_{46}, c_6^{(0)}]$. In this example the temporary list comprises eight entries which are inputted into a hashing function that returns a 32-bit integer. This integer is the core identifier of atom 4 for, $c_4^{(1)}$. Once this process is completed for every atom, all core identifiers are updated to the new core identifier simultaneously and are appended to the ECFP-set. The temporary lists are being discarded. After iteration 1, the ECFP-set comprises of the elements $[c_0^{(0)}, ..., c_n^{(0)}, c_0^{(1)}, ..., c_n^{(1)}]$.[26]

The second iteration is conducted in the same way as the first iteration. However, the core identifier of j of the first iteration contains information about its surrounding cores and their bonds. Therefore, information from up to two bonds is incorporated into the second iteration's core identifiers, $c_j^{(2)}$. Thus, from iteration to iteration, the core identifiers can be understood as the core atom within a larger and larger structural neighbourhood. $c_j^{(0)}$ is a 32-bit integer that encodes the atom j and its adjacent bonds. $c_j^{(1)}$ is a 32-bit integer that encodes atom j, its neighbouring atoms within one bond length, their bond orders and their adjacent bonds, and so on.[26]

After the specified number of iterations is completed, duplicate 32-bit integers are removed from the ECFP-set since they encode for the same subtructure. Butyramide is shown in figure 4.8 as a conceptual example of fingerprint generation.[26]

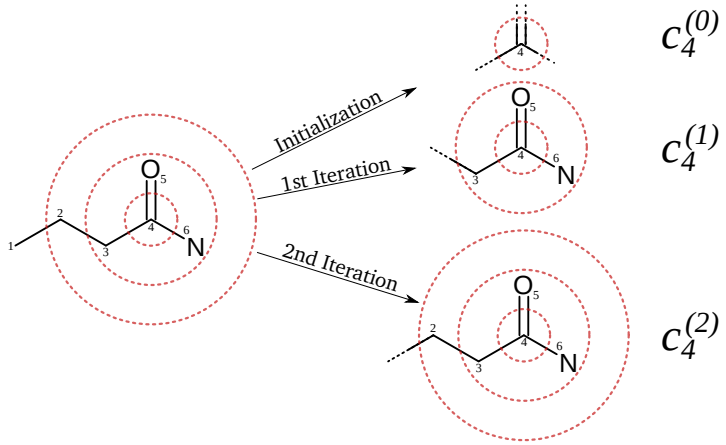
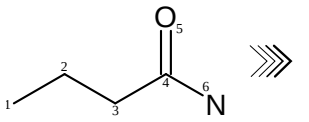


Figure 4.8: Fingerprint iterations with substructures for one atom. Atom 4 of butyramide iterated, denoted by the red circles. The smallest circle denotes initialization of atom 4, resulting in $c_4^{(0)}$ only containing information about the core atom and adjacent bonds. The first iteration is denoted by the intermediate circle and results in $c_4^{(1)}$. The second iteration is denoted by the outer circle and results in $c_4^{(2)}$.

So far ECFP-set contains 32-bit identifier. A substructure could be encoded by the integer, e.g. 1559650422, which would correspond to an ”on” bit within a bit set of 2^{32} bits. Since a hash space of 2^{32} is quite vast, the 32-bit integers are usually mapped onto a vector of 1024 (or 2048) bits by yet another hashing algorithm. Even though the bits in the new 1024-bit-vector cannot be directly decoded into molecular substructures, the identifiers and substructure pairs can be saved and subsequently accessed. In summary, ECFPs are generated by hashing atomic invariants into identifiers, which are then updated a specified number of times with information of their immediate surroundings. Eventually, the core identifiers are hashed into a bit-vector (usually 1024 or 2048 bits) that indicates present substructures by ”on” bits (1) and missing ones by ”off” bits (0). The procedure is visualized in figure 4.9.



r=0	r=1	r=2
734603939	734603939	734603939
1559650422	1559650422	1559650422
-1100000244	-1100000244	-1100000244
1572579476	1572579476	1572579476
-1074141655	-1074141655	-1074141655
	863188371	863188371
	-1793471910	-1793471910
	-1789102870	-1789102870
	-1708545601	-1708545601
	-932108173	-932108173
	2099970318	2099970318
		-87318679
		1112638790
		-627599602

(0,1,1,0,0,.....,0,0,0,1,0)

Figure 4.9: Butyramide as an example of ECFP generation. Starting from the structure, the identifiers are generated for radius 0, 1, and 2 and appended to the ECFP-set. From the identifiers with radius 2, a bit vector of a certain length is obtained.

4.4 Cell-Painting Assay

Cell-Painting (CP) refers to a high-content-screening method that generates cellular image data from high-throughput fluorescence microscopy experiments. A CP assay consists of several consecutive steps which result in tabulated raw image data. These steps consist of cell culture, treatment, staining and fixation, automated image acquisition and feature extraction.[3]
This project uses CP data generated by the Broad Institute.[3] U-2 OS cells were used as the target organism in the cell painting assay reported by Bray *et al.*[3] 1500-2000 cells were seeded into every well of multiple 384-well clear bottom plates and incubated at 37 °C for 24 hours.[28] Then, compounds were added to the well in quadruplicates of varying concentrations. In total, 30409 different compounds were added and incubated for 48 hours. The compounds used can be categorized as small molecules and were either taken from the Molecular Libraries Small Molecule Repository (MLSMR), the known bioactive compounds database of the Broad Institute, the Molecular Libraries Program (MLP) or compounds derived from diversity-oriented synthesis. Antibodies, enzymes and other biotherapeutics were not used in this bioassay.[3]
In total, six different fluorescent reagents were used to stain five different cell-organelles. Only two of the reagents were applied to the living cell culture; the remaining four were applied after the cells' fixation. A combination of two reagents was used to stain the F-actin cytoskeleton, plasma membrane and Golgi apparatus. Another reagent is used to stain the nucleoli and the cytoplasmatic ribonucleic acid (RNA). Additionally, individual reagents are used to stain the ER, the nucleus and the mitochondria, respectively. The six reagents are listed in table 4.1) with the cell organelles, they respond to and the catalogue number.

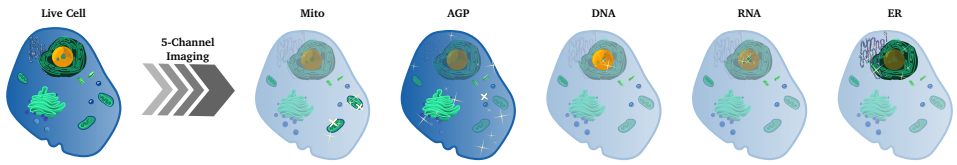


Figure 4.10: Concept of 5-channel imaging. The live cell is stained with fluorophors and the imaged in 5 different channels. Each highlighting different compartments in the cell. The highlighted compartments of each channel are opaque and light emission is implied by sparkles.

Table 4.1: The fluorescent dyes used in the CP assay are listed here. The list contains the names of the fluorophours, the cell organelle(s) that they are targeting and the catalogue number that refers to the Invitrogen catalogue.[28] The maxima of the excitation and emission wavelengths of the fluorophors are shown in nanometer.

Fluorescent reagents	Cell Organelle	$\hat{\lambda}_{ex}\text{nm}^{-1}$	$\hat{\lambda}_{em}\text{nm}^{-1}$	Invitrogen
Mitotracker Deep Red	Mitochondria	644	665	M22426
Wheat Germ Agglutinin, Alexa Fluor 594	F-actin cytoskeleton, plasma membrane, Golgi	589	615	W11262
Concanavalin A. Alexa Fluor 488	ER	495	519	C11252
Phalloidin, Alexa Fluor 594	F-actin cytoskeleton, plasma membrane, Golgi	581	609	A12381
Hoechst 33342	Nucleus	350	461	H3570
SYTO 14 green-fluorescent nucleic acid stain	Cytoplasmatic RNA, Nucleoli	521	547	S7576

After the compound treatment, a staining solution of Mitotracker and wheat germ agglutinin (WGA) was added and incubated for $30\,\text{min}$ at $37\,^{\circ}\text{C}$. Afterwards, cells were fixed using paraformaldehyde. Afterwards, staining solutions containing Phalloidin, Hoechst 33342, SYTO 14 and Concanavalin were applied to the cell containing wells and incubated for $30\,\text{min}$. Finally, the plates were thermally sealed and stored at $4\,^{\circ}\text{C}$.[9,28]

In the next step, images were generated via automated fluorescence microscopy. Five fluorescence channels were used, which scan the plates for different wavelengths emitted by the fluorophors tagging specific cell organelles. The channels are labelled DNA, RNA, AGP (F-actin cytoskeleton, Golgi and plasma membrane), Mito (mitochondria) and ER (Endoplasmatic Reticulum).[3]

After the automatic image acquisition was completed, the so-called CellProfiler[2,29] software generated numerical features from these images. CellProfiler has its standard pipeline to generate cellular features from fluorescence images. The concepts of this pipeline are visualized in figure 4.11. First, the images were aligned, cropped, and an illumination correction is applied, followed by the cell identification step. CellProfiler first identifies nuclei by searching for bright, well-dispersed and non-confluent so-called primary objects. Another important step in this recognition is to identify clumped primary objects, then find their dividing lines and remove them or merge them depending on certain measurements.[2] Taking the nuclei as a starting point, the secondary objects, like cell edges, the cytoplasm and nuclear membrane,

are identified next. After the cells have been identified, CellProfiler conducts different measurements to calculate various features related to cellular compartments and organelles. These features include the area, shape, texture and other more complex features.[2] The dataset used in this work comprises 1768 features with non-zero variance. Features that remain constant for every compound do not contain any information relevant to this work and are discarded. After the feature extraction via CellProfiler, the finalized data set is called raw image data.

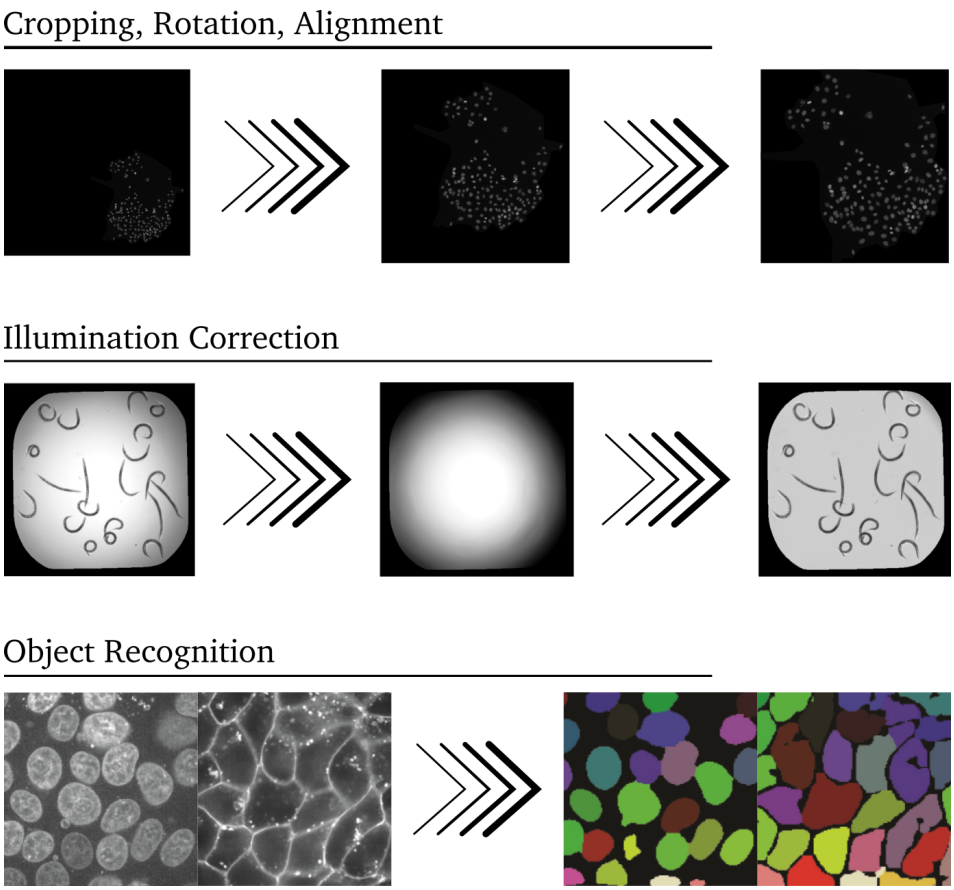


Figure 4.11: Conceptual visualization of the CellProfiler workflow. The first image shows fluorescent microscopy images of human cells. These are cropped and rotated to yield a better view of the cells.[30][31] In the second step, the illumination correction function is applied to a generic image.[31] In the last step (Object Recognition), the identification of stained nuclei and membranes of human pluripotent stem cells are shown as an example.[32] The obtained images were slightly modified.

4.5 Raw Image Data

CellProfiler extracts numerical features from cellular images. The resulting raw image data is a huge spreadsheet whose columns contain the features, and the rows correspond to wells

from which the original images were taken. Additionally, the rows are identified by the compounds used to treat the respective wells, exempt control wells treated with dimethyl sulfoxide (DMSO) only. Every compound is measured in quadruplicates as a minimum. Also, some are measured in octuplicates as well as in different concentrations. There are at least four rows corresponding to four wells in the raw image data spreadsheet for every compound. Compounds whose features have been extracted for only one concentration are referred to as single-concentration-compounds and the compounds that appear in multiple concentrations are called multi-concentration-compounds.
The spread sheet does not only contain numerical features extracted from CellProfiler but so-called metadata, too. Metadata refers to methodological information relevant for the experimental procedure. Hence, compound concentration, plate number, plate map number and other information are being categorized as metadata. Also, the information whether the row corresponds to a treated or control well, is stored within the first 17 columns before the listing of CellProfiler features from column 18 to 1801. From these 17 only five are important for the suceeding steps. Among plate, location, role and concentration these columns contain further information about the molecular structure of the respective compound as a SMILES string (see section 4.1). In table 4.2 the column header names are listed together with a brief description. The names in table 4.2 correspond to the ones from the original raw image data file.[3]

Table 4.2: Below, the names of the most important metadata column headers are listed verbatim from the source file. For every column header, a description is supplied.[3]

Column Name	Description
Metadata_Plate	Contains the plate number of respective well
Metadata_ASSAY_WELL_ROLE	States if the well was treated with a compound or just with DMSO
CPD_SMILES	Contains the compound as a SMILES string
Metadata_mmoles_per_liter	States the compound concentration for treated samples
Metadata_broad_sample	Identifier assigned by Broad Institute that varies inconsistently either with compound, concentration or plate number

4.6 PubChem-Assay

Within the subject of ML, targets or labels can be referred to as features of interest. An ML algorithm attempts to predict targets from a given input similar to a function that calculates y from a given x. Labelling cats and dogs images is a vivid example where the label would either be 'cat' or 'dog'. The inputs would be the individual pixels of an image or their numerical color values, to be precise. The same principle can be applied to bio- and chemoinformatics. Typical targets in this scientific area are 'active' and 'inactive' for a certain bioassay. Nevertheless, annotated data that can be used for labelling is scarce.
The database that supplies the targets for this project is the PubChem database.[33] The PubChem database contains information about chemical compounds and their bioactivities found in various assays. The bioassays in PubChem are assigned a unique assay identifier (AID) and possess a data page featuring descriptive information and the corresponding readout. The descriptive part contains, among others, information like the name and theoretical background, experiment procedure, data source and a readout explanation.[34]
In general, the depositor of a bioassay can provide as many detailed results as necessary.[35] However, PubChem requires the depositor to submit a summary result for each chemical sample. This summary result constitutes the numerical 'bioactivity score' and the categorical 'bioactivity outcome'. The bioactivity outcome can assume five mutually exclusive values: 'chemical probe', 'active', 'inactive', 'unspecified' and 'inconclusive'. The rationale behind the bioactivity outcome is usually provided in the assay comment section to enable a detailed interpretation of the users' results.[34] In the following the bioassay 720532 is described in detail as an example.

AID 720532

Kolokoltsov *et al.*[36] could prove that virus cell-entry strongly depends on the host-cell mediators. Mediators for cell-entry can be signalling factors, membrane attachment factors and endosomal and lysosomal factors.[37] By targeting the host-cell mediators, the emergence of drug-resistant virus mutants is less likely since the cellular mutation rates are up to 6 orders of magnitude smaller compared to viruses.[38] The bioassay AID 720532 is an assay that targets cell-entry of the Marburg virus. Non-pathogenic vesicular stomatitis virus (VSV) with the envelope glycoproteins of the Marburg virus were used as a pseudovirus referred to as VSV-MARV. The pseudovirus contains a Photinus luciferase reporter gene within its genome. Therefore, cell-entry is detected by changes in luminescence. The assay uses HEK293 cells in 1536-well plates. After applying compounds to the respective wells, the plate is incubated. Next, VSV-MARV is

applied in sub-saturating amounts. Luciferase signals reflect the virus titer able to infect the cells for the different compounds.[39][20] During a similar screening against Ebola virus entry (same family as Marburg virus), many signalling pathways relevant to cancer, gene regulation and cell cycle control were found to be relevant in preventing cell-entry.[38]

4.7 SMOTE - Synthetic Minority Oversampling Technique

SMOTE can be used to overcome problems associated with imbalanced data sets. The method uses given data as input and creates synthetic samples from it. The process is most easily described for a two-dimensional data set. For every point in that data set, the k nearest neighbours are found. New data points are then generated on the connecting lines in between the central point and its neighbours. The total number of new data points generated between the central point and its surrounding neighbours depends on the sampling strategy, i.e. how many data points the total semi-synthetic data set is supposed to have. The distance at which the synthetic points are inserted is chosen at random.[19] For a strong label imbalance, the ML algorithm performs well, even if it only classifies one label sufficiently. By generating data points that are presumably representative of the minority class, the ML algorithm has to shift its focus towards the minority class to achieve better performance.

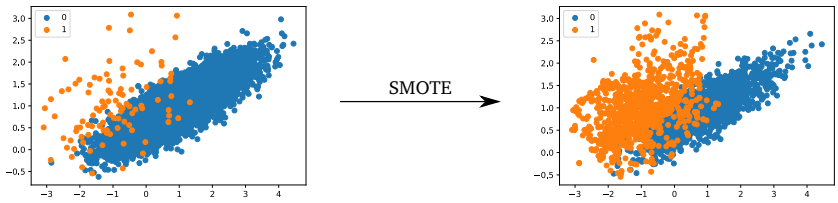


Figure 4.12: SMOTE applied to a 2D data set. The minority class (orange) is oversampled and the majority class (blue) is undersampled.

4.8 Random Forests

In ML classification denotes a (mathematical) rule that produces a label y from a set of input features X. One way of classification utilizes a decision tree. A decision tree is a common mathematical tool in stochastics, closely related to the probability tree. In ML, the nodes are used as entry points for data to be classified, and the paths after each node are called branches. The final nodes at which the classification process ends are called leaves. The decision tree splits the complete data set, D, into subsets, D_l and D_r, funnelling them to the left or right branch

starting at the root node. The information given by the feature, f, guides the splitting of the dataset at each node. Moving further down the branches, the subsets are split into smaller and smaller subsets until they reach the leaves. The leaves correspond to labels present in the data. A leaf assigns its label to every sample of a subset that arrives at the said leaf. A certain numerical rule, the splitting condition, defines how to split the data set, and each node splits the entering data set based on one certain feature. However, since the decision tree is sequential, the information from earlier nodes is always part of the decision process. Therefore, the final decision for the label at the leaves incorporates multidimensional data in a quite simple fashion. A decision tree can be visualized in the context of its data since it applies geometrical decision boundaries (see figure 4.13).[40]

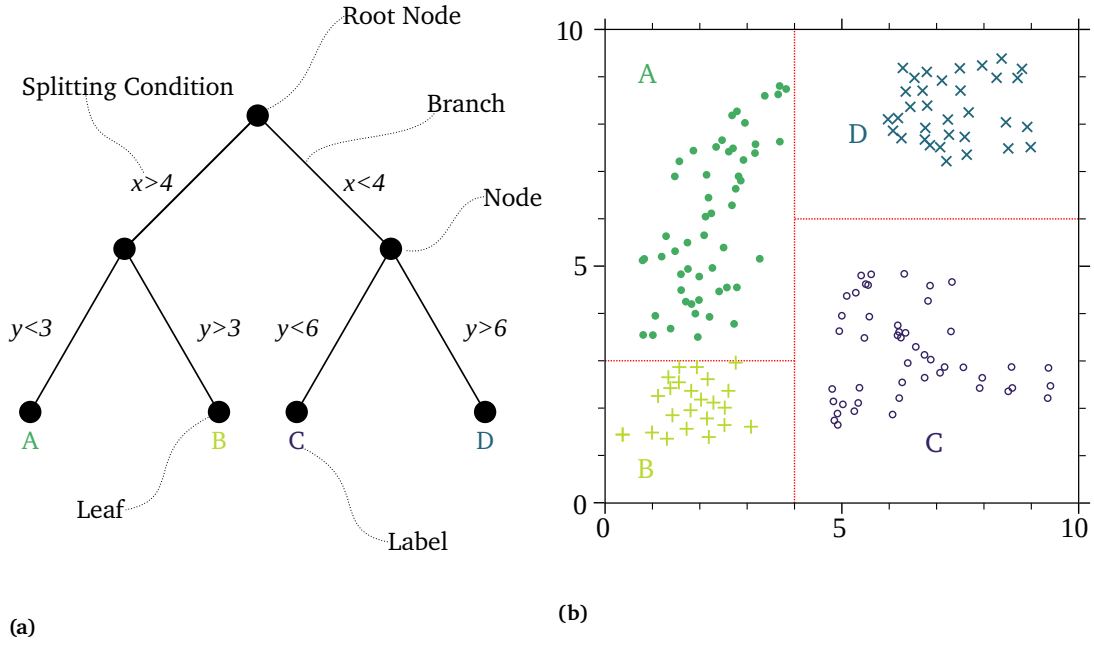


Figure 4.13: (a) Example of a decision tree with the description of the individual elements; (b) Data that is classified based on the exemplary decision tree; A decision tree and the geometrical representation of the classification process is shown for a simple 2-D data set comprising four different classes with the labels A, B, C and D. The data virtually enters the decision tree at the root node. The splitting condition defines if the subsets get funnelled into the right or left branch. At the second nodes, the other splitting criteria are applied, and eventually, the data reaches the leaves where the labels are assigned.[40]

Three cardinal algorithmic questions arise from the described procedure: How to choose the feature on which to split the data? How to choose the splitting condition? And how to assign class labels to the leaves?[40]

The simplest of these questions is the first one. The features of a decision tree are chosen at random. There are other options for generating a decision tree but this is the most common one. Interestingly, carefully choosing the features taken into account by different nodes does

not add great value to the classification process.[40] However if X contains many features that do not contribute to the classification problem (e.g., having a variance close or equal to zero), that approach can backfire. The probability that the decision tree chooses too many redundant features hinders its predictive capabilities.[40]
The second question about the splitting condition can be solved by applying information theory. There are several methods to determine a splitting condition. The information gain method is described here as an example. First, the splitting conditions are determined during the training phase or fitting of a decision tree. During training, all true labels, z, of the data set, are known to the algorithm. Therefore a good heuristic is to gain as much information as possible for a given split. Information gain refers to the enrichment of datapoints with common labels within each subset. To obtain the entropy $H(D_l)$ of the left subset D_l, the frequency of a class label c within this subset is multiplied with its binary logarithm and then summed up over all different class labels. The frequency of c within D_l is calculated by dividing the number of corresponding labels $n(c; D_l)$ by the total number of data points in D_l, $N(D_l)$. The entropy of the right subset after splitting is obtained accordingly.[40]

$$H(D_l) = \sum_c \left[\frac{n(c; D_l)}{N(D_l)} \log_2 \left(\frac{n(c; D_l)}{N(D_l)} \right) \right] \tag{4.1}$$

$H(D)$ corresponds to the number of bits necessary to classify a data point in the parent data set. Thus, $H(D_l)$ and $H(D_\mathrm{r})$ are the bits required to encode the labels within the left and right branch subsets. The information gain is defined as weighted entropies of the split subsets subtracted from the parent data set's entropy. Herein, the entropy of the split subsets $H(D_l)$ is weighted by the probability to find items in the corresponding pool (w_l or w_r).[40]

$$w_\mathrm{r} = \frac{N(D_\mathrm{r})}{N(D)} \qquad\qquad w_l = \frac{N(D_l)}{N(D)} \tag{4.2}$$

Finally, the information gain I of a certain node is given by equation (4.3).[40]

$$I = H(D) - w_\mathrm{r} \cdot H(D_\mathrm{r}) - w_l \cdot H(D_l) \tag{4.3}$$

In general, the greater the information gain, the better the split. From here, it is straightforward to obtain the optimal splitting condition. The inputs X of the data set D have a certain number of features f (usually corresponding to columns) and datapoints d (usually corresponding to rows). Within the assigned feature of the node f, there are d data points which means there are $d - 1$ possible splits that would change the composition of D_l and D_r. Hence, the information gain is computed for every $d - 1$ possible splits, and the threshold resulting in the best information gain is kept as a parameter for that node.[40] The concept of splitting is visualized in figure 4.14.

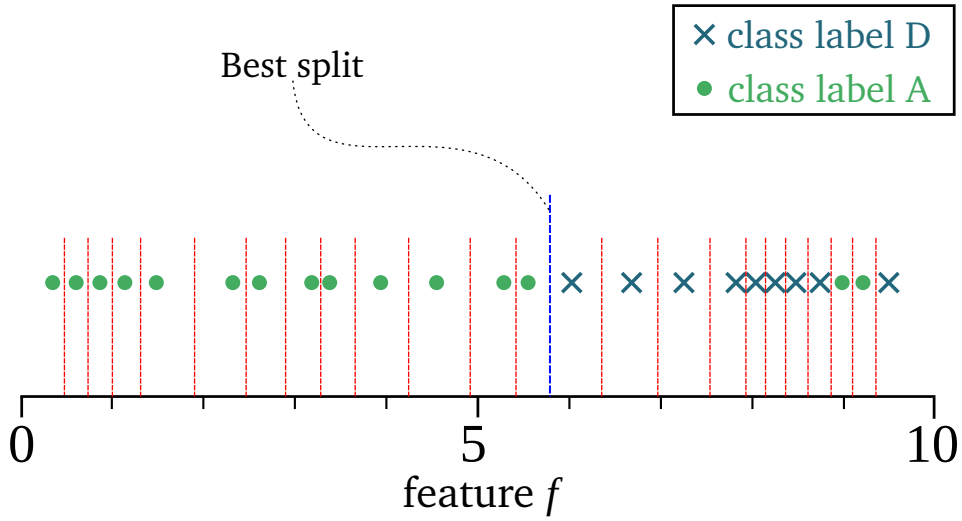


Figure 4.14: Visualization of the splitting condition. One feature of the data points is tested for optimal information gain. The two classes A and D are plotted along feature f, and the optimal split is marked in blue, whereas all other splits are denoted as red dotted lines.

Eventually, the leaves are reached and assign labels to the data points in the final subsets. One leave will assign the majority label, present in the final subset during training. One decision tree is considered a very poor classifier. Many decision trees, on the other hand, can become very powerful. A RFC comprises a large set of decision trees. A sample enters every pre-trained decision tree within the RFC, and every decision tree computes a label. The simplest RFC uses a majority vote to predict each target, i.e. the label most trees computed for that sample is being assigned.[40]

Apart from the splitting conditions, the feature selection and other parameters dictate a decision tree's behaviour and, therefore, the RFC's. Those parameters are referred to as hyperparameters. The hyperparameters control how many trees are included in a RFC, how deep the branches go, how many features they are allowed to use, to name a few.[21] As opposed to the splitting condition, which is chosen by mathematical optimization, the hyperparameters have to be entered by the operator.[40]

4.9 Cross Validation and Splitting

Usually, an advanced ML model has enough parameters to fit a given data set optimally, therefore 'memorizing' the inputs and their corresponding labels. If a dataset used for training in its entirety was also used to test the resulting model, the model's performance would be near perfect. However, the performance on unseen data would be inferior since the model would not abstract from the training dataset.[41]

Splitting the data and using one part for training and one part for performance estimation mitigates the selection bias. However, given the random splitting of the data, the model will not explore the complete data set, and the individual splits can have a biased label distribution. This bias would result in an underestimation of the performance. Both the issue of overfitting (or selection bias) and insufficient use of the data set can be circumvented by cross-validation (CV).[40]

During CV the data set is split in k subsets with equal sample count. These subsets are referred to as folds. Next, the model iterates through k cycles of training and evaluation. In every cycle, another fold is used as a validation set, whilst the remaining $k-1$ folds are used for training. All evaluations are then averaged to yield a better estimate of the true model performance. This method of splitting the data set into k folds and then cycling through each combination is called k-fold CV[41] A visual representation is shown in figure 4.15.

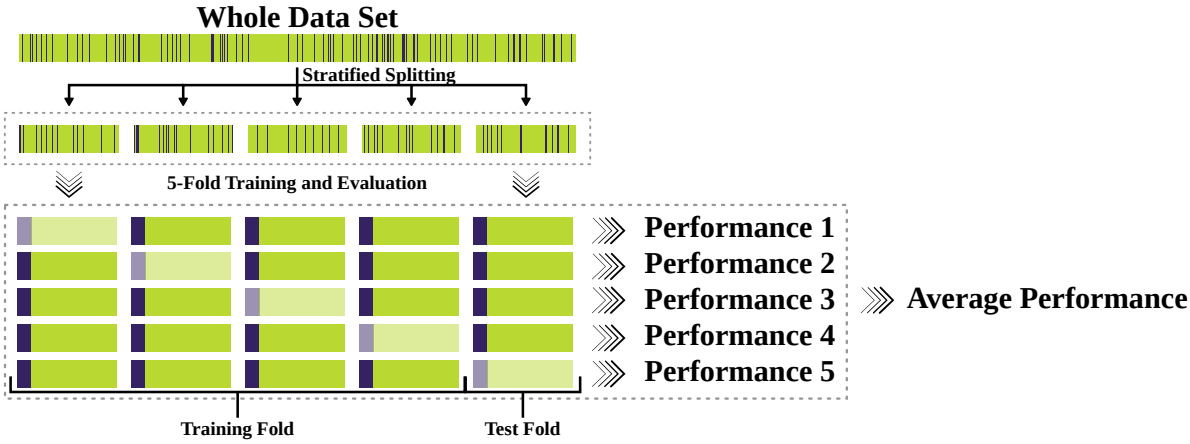


Figure 4.15: Visualization of CV. The whole data set contains 83% negative (green) data points and 17% positive (purple) data points. It is split up into five equally large subsets of equal distributions of positive and negative samples. For 5-fold CV the ML model is trained five times. Each iteration uses another subset for validation (indicated by the lighter shade) whilst the others are used for training. The individual performances are averaged to yield a good estimate of the predictive power of the model.

Selecting the hyperparameters is another caveat that results in performance overestimation. Hyperparameters were mentioned in section 4.8 which are parameters specified by the user that dictate the model's architecture (i.e. the number of decision trees, branching depth, etc.). Those parameters are usually optimized by applying an automatic sampling of different values for each parameter. One possibility is to supply a value list for each hyperparameter which is called a parameter grid. The algorithm uses every combination of values consecutively to find the hyperparameters that perform best. Thus, the hyperparameter selection itself exploits information in the data. Otherwise, the hyperparameter variation would not impact

the predictive performance. Therefore, optimizing the hyperparameters and using k-fold cross validation (KFCV) on the RFC parameter optimization only, results in exaggerated model performance. The application of KFCV to the hyperparameter optimization as well as to the training of the RFC solves this issue and is called nested KFCV. The concept is shown in figure 4.16.[41]

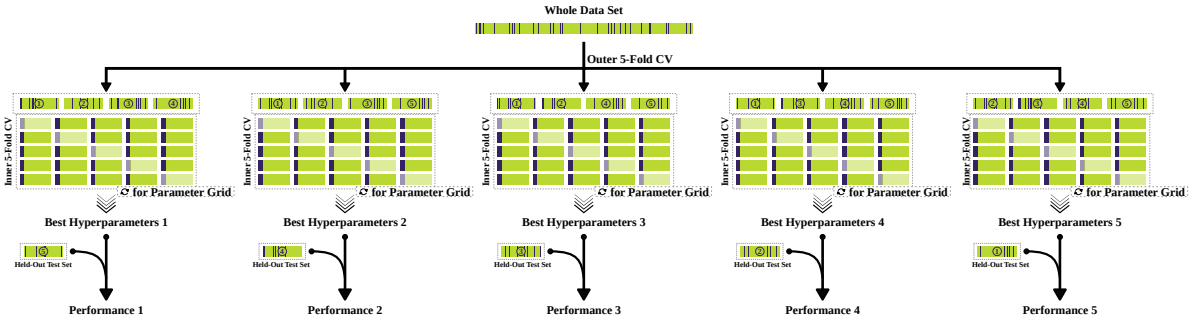


Figure 4.16: Visualization of nested CV. The whole data set contains 83% negative (green) and 17% positive (purples) samples. The outer CV splits the data into five subsets. Four of these subsets are used in the inner CV. The other one will be used as a held-out test set for performance evaluation. The outer CV results in five instances, each considering a different subset for evaluation. The inner CV starts by splitting the outer training set into five subsets. Next, those subsets are used for training and evaluating all hyperparameter combinations supplied in the parameter grid. The best parameters are then tested to estimate the model performance.

The splitting strategy that is chosen should be as unbiased as possible to diminish performance overestimation. However, the distribution of labels within each fold should be comparable to avoid underestimation of the performance. In the worst case scenario, one fold would contain all labels, which would lead to inferior prediction performance. The random-stratified-split-strategy splits the data set into randomly chosen subsets, each exhibiting an equal label distribution that counteracts pessimistic prediction performance.[42]

4.10 Performance Evaluation

There are several different metrics available for performance evaluation of an RFC. Since the work presented here is concerned with a binary classification problem (i.e. only two labels are possible per sample), the descriptions below are only valid for binary classification. The most fundamental performance assessment of a classifier is given by the confusion matrix, which compares the predicted labels with the true labels. From the confusion matrix more applied metrics can be calculated, namely the true positive rate (TPR), the true negative rate (TNR), the balanced accuracy, the Matthews correlation coefficient (MCC) and many more.

Furthermore, the receiver operating characteristic curve (ROC-curve) and AUC-ROC supply further information about the goodness of the fitted model.[43]

4.10.1 Confusion Matrix

A confusion matrix compares the predicted labels with the true labels of the classification problem. The confusion matrix is a quadratic matrix of the number of different classes (also referred to as the contingency table). Therefore, for a binary classification problem, the confusion matrix is a two by two matrix. The correctly identified instances are shown on the diagonal of the matrix, either referring to true positive (TP) or true negative (TN) instances. Off diagonal erroneously identified instances are presented, either false positive (FP) or false negative (FN) values. The general structure of a confusion matrix is shown in figure 4.17.

		Predicted Labels: True	Predicted Labels: False
True Labels	True	TP	FN
	False	FP	TN

(a) Structure of a confusion matrix

		Predicted Labels: True	Predicted Labels: False
True Labels	True	1274	810
	False	328	7588

(b) Confusion matrix with exemplary values.

Figure 4.17: The general structure of a binary confusion matrix is shown on the left and on the right exemplary values are inserted in the respective fields.

4.10.2 TPR, TNR, Balanced Accuracy and Matthews Correlation Coefficient

From the confusion matrix several other metrics can be computed. For example the TPR and TNR, more widely known as the sensitivity and the specficity, aswell as the BA and the MCC. The TPR describes the frequency of correctly positive-labelled samples whereas the false positive rate (FPR) depicts the frequency of incorrectly positive-labelled predictions.[43]

$$TPR = \frac{TP}{TP + FN} \tag{4.4}$$

$$FPR = \frac{FP}{TP + FN} \tag{4.5}$$

Analogously, the TNR describes the frequency of the correctly negative-labeled samples within the predictions.[43]

$$TNR = \frac{TN}{TN + FP} \tag{4.6}$$

The balanced accuracy, BA, is simply the average of the TPR and the TNR.[44]

$$BA = \frac{TPR + TNR}{2} \tag{4.7}$$

TPR, TNR and BA can all adapt values between zero and one. One correponds to a perfect model. The Matthews correlation coefficient, MCC, is a metric that scores high if the number of correctly predicted samples is significantly higher than the number of incorrectly assigned ones.[45]

$$MCC = \frac{TP \cdot TN - FP \cdot FN}{\sqrt{(TP + FP)\,(TP + FN)\,(TN + FP)\,(TN + FN)}} \tag{4.8}$$

The MCC can score between -1 and 1 where 1 represents very good performance, 0 refers to a random model and -1 describes consistently false predictions.[45]

4.10.3 ROC and AUC-ROC

RFCs assign discrete prediction labels z to their input samples. However, as stated in section 4.8, a majority voting of the total number of trees is conducted. The majority voting can be interpreted as a probability p for a certain class label by dividing the votes for a certain class label by the total number of votes. The resulting probability is between 0 and 1. A threshold t can be applied that determines the label depending on the probability.

$$z = \begin{cases} 0 & \text{if } p < t \\ 1 & \text{if } p \geq t \end{cases} \tag{4.9}$$

A majority voting refers to a threshold of 0.5. However, the threshold can be varied from 0 to 1. For a zero threshold, all labels will be predicted as 'positive' (i.e. '1') since no label will have p smaller than zero. For a threshold of one, all predictions will be 'negative'. Hence, for each threshold, a different confusion matrix is obtained and, therefore, a different TPR and FPR. For the ROC-curve the TPR is plotted on the y-axis and the FPR is plotted on the x-axis. The threshold mentioned above is iterated from 0 to 1 to obtain all different confusion matrices from which the TPR and FPR are calculated. The AUC-ROC corresponds to the goodness of the model depicted by the corresponding ROC-curve. An AUC-ROC of 1 is a perfect score since it corresponds to the aforementioned perfect ROC-curve. The benefit of the ROC-curve and AUC-ROC is that it contains more information than the other metrics and enable quick visual

confirmation.[43] In figure 4.18 different ROC-curve are shown that correspond to well or not so well-performing models.

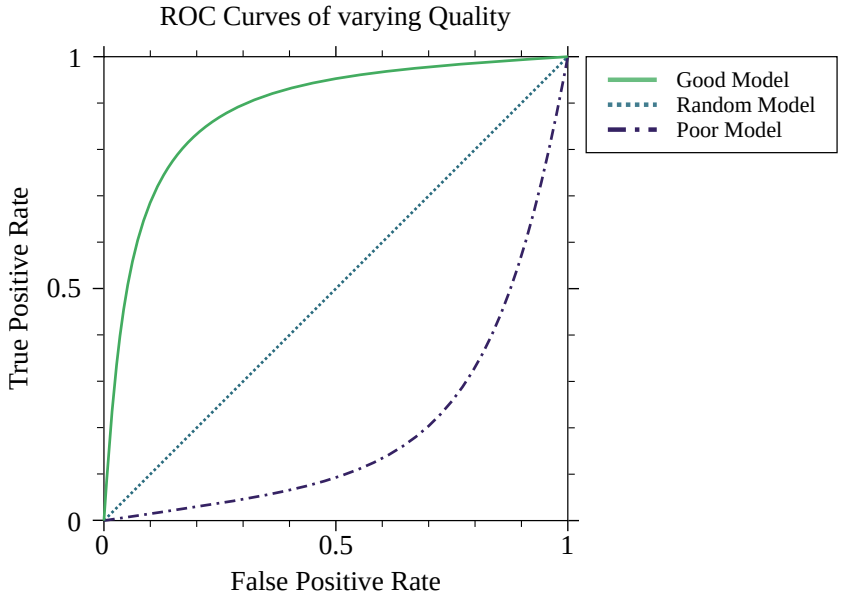


Figure 4.18: Examples of ROC-curves. The diagonal depicts a model that chooses labels randomly. If a curve is above the diagonal it predicts better than random, below indicates a bias towards the wrong label.

4.11 Feature Importance

There are three methods of choice to measure feature importance that are used in this project: PCA, random forest feature importance and the MRMR. PCA can be understood as a method of dimensionality reduction. The original data is mapped to a new coordinate system that maximizes the variation in each dimension.[46] Therefore, the number of new axes, referred to as principal components, is the same as in the original dimensions. However, the first few principal components usually account for a large part of the variance within the data set. In contrast, the lowest principal components account for no variance, which is referred to as noise in data science.[40] Transforming the coordinate system also gives information about the contribution of each feature to the principal components, which can be used to infer feature importance for the original data set.[46]

Random forest feature importance can be either measured by gini impurity or by entropy (or information gain, see section 4.8). The gini impurity is defined as the product of probabilities to encounter positive samples, p_1 and negative samples p_0 respectively at a given node k.[47]

$$G_k = 2p_1p_0 \tag{4.10}$$

From equation (4.10) follows, that a small gini impurity refers to a very succesful split from the parent node with the index $k-1$. If both descendant nodes contain one label only, the parent node has achieved the best split possible. The feature importance for a single decision tree $I_{\mathrm{d}}^{(f)}$ is therefore defined as the sum of reductions in gini impurity from every parent node k that uses feature f in its splitting condition to its descendants $k+1$.[47]

$$I_{\mathrm{d}}^{(f)} = \sum_{k} G_k^{(f)} - G_{k+1} \tag{4.11}$$

Notice, that G_{k+1} comprises the contribution of the left and right descendant nodes. The overall feature importance of feature f is calculated as the the average of tree-wise feature importances.[47]

$$I^{(f)} = \frac{1}{N_d} \sum_{d} I_{\mathrm{d}}^{(f)} \tag{4.12}$$

N_d is the number of trees in the RFC. Since the splitting at node k always incorporates the information from prior nodes, this method of feature importance accounts for non-linear feature interaction. One shortcoming of this method is that strongly correlated features tend to share their importance, which results in two difficulties. Firstly, critical features are scored lower since the importance measure is split within highly correlated feature clusters. Secondly, the final selection will contain many features that belong to the same highly correlated cluster. Adding many features that contain similar information has no beneficial effect on the model and fosters biased predictions based on a few clusters' information. To overcome these two problems, hierarchical clustering can be performed on the features beforehand, and from every cluster, only one feature is picked for further feature engineering.
MRMR is a feature selection algorithm that was proposed by Peng *et al.*.[48] It utilizes mutual information between features to calculate their redundancy and mutual information between features and each class label to calculate maximum relevance to the categorization problem. The algorithm then optimizes the subtraction of the features redundancy and the feature relevance to obtain a scoring for each feature. MRMR was found to be very suitable for data sets with more than 1000 numerical features.[48]

4.12 Gene Ontology Terms

The GO terms are keywords assigned to genes and their products by the Gene Ontology Consortium. The vocabulary is structured, precisely defined and controlled.[49]
Three main categories are distinguished within GO terms: biological processes, molecular function and cellular component. A biological process is defined as a process that involves

a chemical or physical transformation. Thus, something enters a process and is released as something else. High-level examples include GO terms like 'cell growth and maintenance'. A more specific example would be 'cAMP biosynthesis'.[49]

Molecular function refers to the biochemical activities of a gene product. That corresponds to chemical reactions, binding mechanisms and transport mechanisms, to name a few. The molecular function does not specify where or when it happens, only that the gene product holds the potential to execute this function. High-level GO terms are 'enzyme', 'ligand' and low-level terms are 'adenylate cyclase' and 'Toll receptor ligand'.[49]

The cellular component places a specific gene product within the cell. This placement does not necessarily correspond to a localized organelle. For example, 'proteome' is an adequate GO term within the cellular component. Other terms like 'Golgi apparatus' or 'nuclear memrane' refer to cellular components though.[49] Terms within GO can be connected hierarchically. A specific molecular function like 'oxidoreductase activity' is a descendant from 'catalytic activity', for example. The three main categories can also be linked by logical operators, which produces hierarchical networks containing abundant information about a given gene product.[49] Four exemplary hierarchical trees can be seen in figure 4.19.

The GO database includes information about a gene product if published experiments are available or if information can be inferred from sequence homology, which is mostly applied to less studied organisms. The information gained from these sources must fulfil evidence requirements set by the Evidence and Conclusion Ontology (ECO).[49,50]

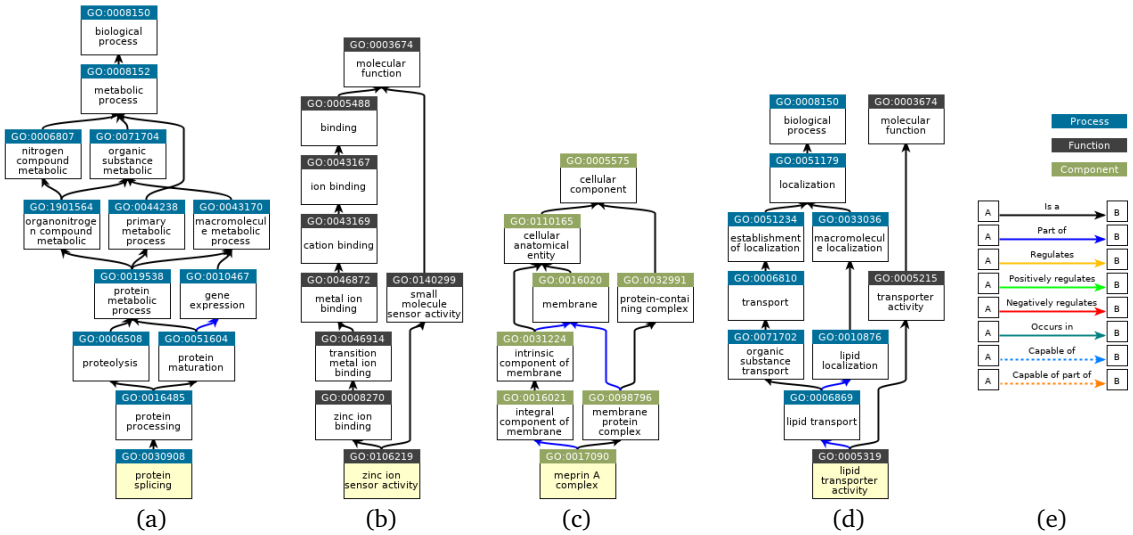


Figure 4.19: Examples of GO term hierarchies. (a) is an example of a biological process; (b) presents a molecular function GO tree; (c) is an example of a cellular component; (d) is a tree that is a mix of biological process and molecular function; (d) is an exhaustive legend defining colours as well as possible relationships.

5 Methods

In the following sections, the computational process is described. The implementation, including the data sets, will be available at https://github.com/Foly93/masterthesis. For programming, either Python or Bash was used. Jupyter notebooks were used as a user interface for python programming. Furthermore, python scripts and bash scripts were used as well.

5.1 Descriptors - CP and ECFPs

As inputs for the RFC two different descriptors are chosen. The first descriptor set comprises the morphological information that are extracted from the raw image data of the CP assay of Bray *et al.*[3] Before the raw image data can be inputted they need to be preprocessed, which is described in detail in section 5.3. The baseline descriptors that the CP data are compared against are ECFPs. For all compounds present in the final data sets (referred to as combined ML-ready data sets) the structural identifier was computed from the canonical SMILES utilizing the Chem package from the RDkit python library.[51] The obtained compound identifiers are 2048-bit vectors with radius 2. Every bit in this vector is considered a feature for the machine learning algorithm.[51]
First, the CP features and ECFP features are used separately as inputs. Afterwards, selected features from both features spaces are used as input together.

5.2 Targets

For creating annotations for the input vectors, the PubChem bioassay database was queried. PubChem comprises more than 1 200 000 bioassays.[52]The amount of information stored at the PubChem database is so vast that the process of finding data sets fitting for this project is a problem on its own. A stepwise filtering process was created to find relevant bioassays.
First, the eleven biggest folders are downloaded from the PubChem database, each containing up to 1000 bioassays.[53] Then the 100 assays with the most compounds are kept from each of the eleven folders resulting in 1100 bioassay data sets of large size. The next step was to find assays with an endpoint related to toxicity or cell morphology. For that purpose, two auxiliary files are generated. The first file was downloaded directly from https://pubchem.ncbi.nlm.nih.gov/ and contains detailed information about each of the 1100 assays. That includes the AID, the

assay name and a description of the assay and the endpoint tested. The second file is a list of protein targets enriched for cytotoxic and cytostatic phenotypes generated by Mervin *et al.*[4] A program searches the assay information file for instances from the protein targets list. It saves the AIDs that are related to said targets to another list. The resulting list of supposedly cytotoxic compounds consists of 671 assays. Next, the compound overlap with the raw image data needs to be found (see section 4.5). However, the compounds are annotated with their PubChem assigned compound identifier (CID) which is not a widely used identifier. Therefore a more general identifier was required for each compound that can be used to screen against the CP data set. The PubChem website offers functionality that generates a description for a list of CIDs. Part of that description is the international chemical identifier key (InChI-key), which is a more general, unique identifier that can be translated into other identifiers like SMILES strings. Therefore, the next step was to concatenate all compounds into a list and upload it to the PubChem website, to obtain the description of the CIDs for download. The CIDs in the 671 bioassays are then exchanged for the InChI-keys. The compound overlap with the CP compounds was conducted using the $\mathrm{Metadata\_broad\_sample}$ (which turned out to be suboptimal in section 5.3 and had to be corrected). Therefore, each assay's compounds were merged with the InChI-key annotations of the CP data set and only entries present in both data sets are kept. Next the InChI-keys are exchanged for their $\mathrm{Metadata\_broad\_sample}$ identifiers.
Not all 671 assays were further investigated. As mentioned in section 4.6, PubChem labels their compounds 'active', 'inactive' 'unspecified' or 'inconclusive'. If a dataset contains no actives or too little, machine learning applications will have trouble correctly categorising the data since the two classes (active and inactive class) are too imbalanced. Thus, in the next step, the threshold of at least 100 active compounds was applied as a filter, resulting in 52 bioassays. From these 52 bioassays, 'inconclusive' and 'unspecified' rated compounds were deleted. Notice that 100 active compounds can be a comparably small amount of actives since some of the 52 final bioassays have more than $20\,000$ compounds in total.
Conclusively, 52 spreadsheets were obtained, containing the metadata broad sample as a molecular identifier and the PubChem activity outcome as a classification target.

5.3 Preprocessing

As mentioned in section 4.5, the raw image data contains meta and data columns. The metadata columns of the CP data dictate the decision-making process during the preprocessing, and the data columns themselves are the subject of preprocessing. The data columns or CP features vectors are the inputs for machine learning applications described in the following chapters.
In brief, the preprocessing combined the bioassay data set and the raw image data into 52 fully

annotated and ML-ready data sets. Eventually, they contain information about the features, about the endpoint and some metadata information, e.g. for compound identification.
During the preprocessing, the Metadata_broad_sample turned out to be a suboptimal identifier because it is not unique for every compound (see section 4.5). Different Metadata_broad_sample values are used for the same compound if it corresponds to a different compound concentration or plate. Therefore, the Metadata_broad_sample was exchanged for the canonical SMILES string.
First, the individual 384-well plates were centred on the mock samples. For that purpose, the plate-wise average of the untreated samples was calculated for every morphological feature and then subtracted from the treated samples. The next step was to calculate compound-concentration-wise medians of each feature. However, some compounds appear in multiple concentrations. Therefore, a new metadata column was introduced that labelled each row either as a single-concentration-compound or a multi-concentration-compound. The single-concentration-compounds' medians were computed in a straight forward fashion for the whole raw image data set, whereas the process for multi-concentration-compounds is described in the next paragraph. The resulting preprocessed raw image data frame has 31 692 rows and 1768 feature vectors.
Next the preprocessed raw image data was merged with each of the 52 bioassays by keeping Metadata_broad_sample identifiers which are present in both data sets. Next, the compounds were properly annotated by transferring the Metadata_broad_sample into the canonical SMILES string. The canonical SMILES were computed from python's sklearn library. Afterwards, the multi-concentration-compounds, present in the combined data frames, were inspected since they required further consideration. The ML algorithm requires one morphological profile per compound and one label per compound. Thus, for each multi-concentration-compound, the feature medians were computed only for the most abundant concentration. The other concentrations were discarded. This computational process results in 52 combined ML-ready data sets with 1768 relevant features and varying row number since the compound wise overlap varies from assay to assay. A table of the resulting 52 assays with the number of active, inactive and total compounds can be found in table 5.1.

Table 5.1: This table gives an overview over the combined ML-ready data sets obtained from the preprocessing procedure. The AID from the original PubChem data set is given and the number of active, inactive and total compounds. All 52 assays combined have 371 978 inactive labels and 12 140 active lables.

AID	Inactives	Actives	Total	AID	Inactives	Actives	Total
1030	4804	832	5636	588334	6978	133	7111
1458	6547	487	7034	588458	8850	117	8967
1529	7794	150	7944	588852	8840	128	8968
1531	7818	122	7940	588855	7536	151	7687
1578	7816	146	7962	602340	21297	102	21399
1688	6814	158	6972	624202	8342	237	8579
1822	7822	141	7963	624256	8574	139	8713
2098	7719	132	7851	624296	6475	439	6914
2156	7868	149	8017	624297	7440	252	7692
2216	7328	154	7482	624466	8844	173	9017
2330	1752	131	1883	651610	18234	218	18452
2540	8015	127	8142	651635	8036	125	8161
2553	7908	109	8017	651658	18839	163	19002
2599	7913	229	8142	651744	297	207	504
2642	7821	196	8017	720504	8197	341	8538
2796	7837	345	8182	720532	1164	185	1349
485270	7992	190	8182	720582	8933	121	9054
485313	7497	491	7988	720635	248	126	374
485314	7589	172	7761	720648	8928	126	9054
504333	6598	526	7124	743012	315	195	510
504444	5296	275	5571	743014	315	188	503
504466	6909	260	7169	743015	320	211	531
504582	8022	110	8132	777	2831	911	3742
504652	7829	312	8141	894	4769	324	5093
504660	8094	131	8225	932	6399	420	6819
504847	9047	175	9222	938	2528	158	2686

5.4 Feature Engineering

The feature engineering for CP descriptors was performed using three methods further described in section 4.11. Firstly, the PCA is performed using the PCA method available from sklearn.[21] The method was applied to the CP-PubChem data sets to find the features that comprise most of the variance. The 100 features that account for most of the variance in the first principal component were added to the list of most important features.
The next step was to pick important features by applying gini impurity to a RFC algorithm. However, before the gini impurity feature importance can be applied, redundancy of similar features had to be reduced. For that reason, features were clustered based on their Spearman correlation with all other features. A distance cut-off was applied that resulted in 400 remaining clusters. From each cluster, one feature was picked at random. The resulting 400 presumably non-redundant features entered the random forest-based feature selection algorithm. This RFC used 250 estimators from which the features were scored using the gini impurity to select the 100 most important ones (see equation (4.12)).[55]
The last method used MRMR to extract the thirty most important features based on a maximum-relevance-minimum-redundancy criterion. The computational python implementation from Peng *et al.*[48] was used (https://pypi.org/project/pymrmr/).
Since the ECFPs are boolean features (either 0 or 1) Spearman-clustering, and MRMR will not work. Therefore, only the random forest feature importance of sklearn was used to score the structural fingerprint features. Instead of only using the gini impurity, the entropy-based feature selection was utilized as well. This resulted in 200 most important features from the 2048 features present in the original fingerprint data set for each of the 52 combined ML-ready data sets.[55]
In the next step, the most important CP and ECFP features are combined into one set of features for each of the 52 assays. The top features lists obtained from MRMR, PCA and GI can contain a small overlap. Therefore, the top CP features from each list are combined into one list, and duplicate features are removed. The same is done for the ECFP features. Finally, both descriptors' best features were combined and concatenated with the labels from the PubChem bioassays.

5.5 Prediction

For each endpoint represented by a PubChem assay, an RFC was developed and trained. Four different modelling cycles can be distinguished. Every cycle used the labels from PubChem assays as targets. The first cycle was solely concerned with the CP descriptors, the second cycle

was concerned with the ECFPs, whilst the third cycle used the feature engineered, combined set of descriptors. The last run omits feature engineering and incorporates all features from both descriptor sets.
Nested 5-Fold CV was used to train the model and tune the hyperparameters with a stratified split strategy. For the inner loop that fitted the parameter of the RFC, a random split strategy was used with 5-fold CV (see section 4.9). Before splitting the data in the inner loop, SMOTE was applied to increase the minority class label by $100\,\%$ effectively doubling its size, and random undersampling was applied as well (see section 4.7). The minority class amounted to $75\,\%$ compared to the majority class label after applying this sampling strategy. SMOTE was not applied to the held-out test set of the outer loop. Thus, the model was validated with real data points only.
The hyperparameters are optimized using a halving random search method from $\mathrm{sklearn}$.[21,54] For that purpose a parameter grid was used for each inner CV iteration. The parameters which were covered can be seen in table 5.2.

Table 5.2: Hyperparameters covered by the RFC

Hyperparameter	Values Covered
max_depth	10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20
max_features	40, 41, 42, 43, 44, 45, 46, 47, 48, 49, 50
min_samples_leaf	5, 6, 7, 8, 9, 10, 11, 12, 13
min_samples_split	4, 5, 6, 7, 8, 9, 10, 11, 12, 13
n_estimators	100, 200, 300, 400, 500
bootstrap	False, True
oob_score	False
criterion	gini, entropy
class_weight	None, balanced

From this parameter grid 500 combinations were randomly sampled for each inner CV iteration and only the best estimator was returned and evaluated in the outer CV iteration. Since a 5-fold CV was in use, five best estimators are evaluated in total by calculating the BA, MCC, TPR, TNR, ROC-curve and AUC-ROC.
This procedure is conducted for each of the 52 combined ML-ready data set. Afterwards, the ECFPs are generated and modelled in the same way. The feature engineered combination of both is used as descriptors, followed by the fourth modelling cycle that used the entire feature space.

6 Results and Discussion

For every PubChem assay, predictions were performed by using the CP descriptors first, then the ECFP descriptors and eventually by using the combined feature engineered descriptors. Additionally, to validate the feature selection, another prediction run was conducted using the complete feature space, i.e. all 2048 ECFP features and 1768 CP features.
The first section compares the performance of CP descriptors against the ECFPs among all PubChem assays. The second section adds the results from the modelling approach using the feature engineered data set. A comparative analysis is conducted that compares the resulting performance with those from the first and second prediction run (using CP data or ECFPs). The fourth prediction run is then evaluated. Several metrics are compared against the feature engineered approach to validate the advantages of feature engineering and to illuminate strengths and weaknesses of the CP data set.
An enrichment method is introduced that uses feature engineering information to connect the performance within an assay group to cellular morphology. Additionally, phenotypic keywords are manually generated to describe cellular processes concerning PubChem assays. The enrichment of specific keywords within assays grouped by their performance relates cellular processes to the predictive capabilities of CP descriptors. The phenotypic keywords suffer from limited knowledge about cellular processes and thus must be considered incomplete. However, there is no bias since they were generated before the performance results for each assay were generated.
A very similar intention is pursued by the final enrichment method. Herein, GO terms are assigned to each gene product featured in the PubChem assays. The GO also correspond to cellular processes and molecular functions with the advantage that extensive hierarchies of terms are available for a specific protein target. Enrichment of GO terms can link the predictive capabilities of CP descriptors to cellular mechanisms similar to the phenotypic annotations but with significantly less expert bias.

6.1 Comparative Analysis of ECFP and CP Predictions

The predictions were performed individually for each assay resulting in individual metrics that are compared among different descriptor sets. The first prediction run and the second prediction run are compared first. They use substantially different inputs, CP and ECFP descriptors, to predict the same bioassay targets. First, three performance metrics are evaluated: the AUC-ROC, BA and MCC. The results of both runs are presented in figure 6.1. figure 6.1a

contains the AUC-ROC and b and c contain the balanced accuracy and Matthews correlation coefficient. The results from ECFP and CP predictions are plotted in the same panel for the same metric. The PubChem assays are sorted by their AUC-ROC. Thus, assays which exert high predictive potential with CP data are listed first. This order is kept throughout this chapter.
The CP prediction run yielded AUC-ROCs from 0.4 to 0.8 as shown in figure 6.1a. The CP dscriptors outperform the ECFPs for 14 bioassays namely 720532, 651635, 624297, 2330, 743014, 588855, 743012, 624296, 743015, 1578, 504444, 651744, 1688 and 651610. However, 1688 and 651610 score an AUC-ROC below 0.7. The remaining 12 PubChem assay are henceforth referred to as 'high performing assays'. All other PubChem assays are referred to as low performing assays. In general, the ECFPs perform well with 27 AUC-ROC scoring higher than 0.7.
In figure 6.1b the balanced accuracy is shown for all assays. The balanced accuracy is calculated by means of the specificity and the sensitivity (see equation (4.7)). Measured by balanced accuracy, several assays perform better, others worse compared to their AUC-ROC. However, the high performing assays are still better scoring compared to the low performing assays. The ECFP predictions exhibit no apparent trends. However, they do not score a balanced accuracy higher than 0.7 which is achieved by most (83%) of the high performing assays when they are predicted using CP data.
For the CP predictions the Matthews correlation coefficient is not as consistent throughout the high performing assays (see figure 6.1c). The performances fluctuates from 0.2 to 0.6 depending on the assay. Five out of twelve achieve a Matthews correlation coefficient higher than 0.4. Within the low performing assays the CP predictions score very low. The trend is descreasing almost monotonously, mimicking the scoring by the AUC-ROC. A familiar trend is presented by the ECFPs. They score worse within the high performing assays but better within the low performing assays with a few exemptions. None of the ECFP predictions score a Matthews correlation coefficient higher than 0.4.

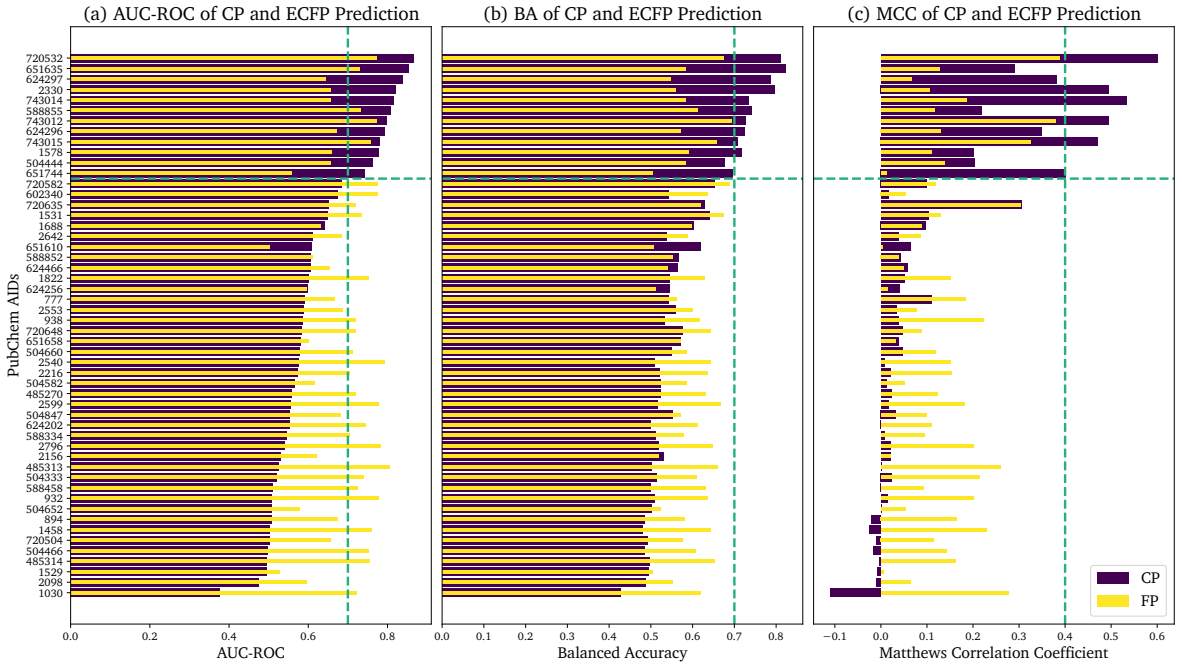


Figure 6.1: Performance comparison of CP and ECFP predictions. The AUC-ROC, balanced accuracy (BA) and Matthews correlation coefficient (MCC) are being compared in three bar plots. The CP predictions are shown in purple and ECFP in yellow with slimmer bars. The PubChem AIDs are listed on the y-axis. Two supporting lines are drawn in each subplot. One horizontal line segregates high and low performing assays and the vertical lines marks the threshold for high performance in each metric.

The first two prediction runs allowed to categorize the PubChem assays. A few performed consistently better via CP descriptors, and for others, the ECFPs showed higher predictive potential. Especially for balanced accuracy and Matthews correlation coefficient, two metrics that are more sensitive to label imbalance, the predictivity of CP descriptors was distinguished among the high performing assays. This finding can have many reasons since CP data and ECFPs are substantially different. However, one explanation might be rooted in SMOTE. This oversampling strategy is reported to perform better on numerical data, as provided by CP features. The ECFP features, on the other hand, are boolean and therefore less suitable.

6.2 Comparative Analysis of Modelling with Selected Features

The third prediction run was based on a mixed feature set containing CP as well as ECFP descriptors selected as described in section 5.4. This set of features will be referred to as selected features. When comparing the CP and ECFP with the selected features evaluation, the selected

features are expected to balance out the shortcomings of each identifier. By closely inspecting each metric's differences and revealing unexpected trends, further information about each identifier's shortcomings and strengths can be gained. The first panel in figure 6.2 shows the assay-wise difference between the CP and ECFP descriptors that was shown in absolute values in figure 6.1a. Here, the difference in performance between the high performing assays and low performing assays becomes even more apparent. The maximum performance gain from using CP over ECFP is almost 20 % in the case of AID 624297. On the other hand AID 1030 performes 30% worse in terms of AUC-ROC when predicted with CP.

The comparison between selected features and ECFP exhibits a very similar trend. The high performing assays score very similar, on the other hand the low performing assays score not as low. They perform up to 20% worse, in the case of AID 1030. Also, some assays that have a negative difference in the left panel score positive, i.e. better than ECFPs when predicted with selected features (AIDs 588852, 720635 and 2156).

The scale on the x-axis in figure 6.2c is the narrowest indicating more subtle changes when switching from CP to selected features. The high performing assays present mixed differences. Some assays are better predicted with CP descriptors only (e.g. 651744) and others are better predicted with the selected features (e.g. 720532). For that reason, the specific effect that the combination exerts on high performing assays remains inconclusive. Almost all low performing assays perform better when selected features are used for modelling. The only exemption is AID 1688. The improvements within the other PubChem assays reach as high as 20%.

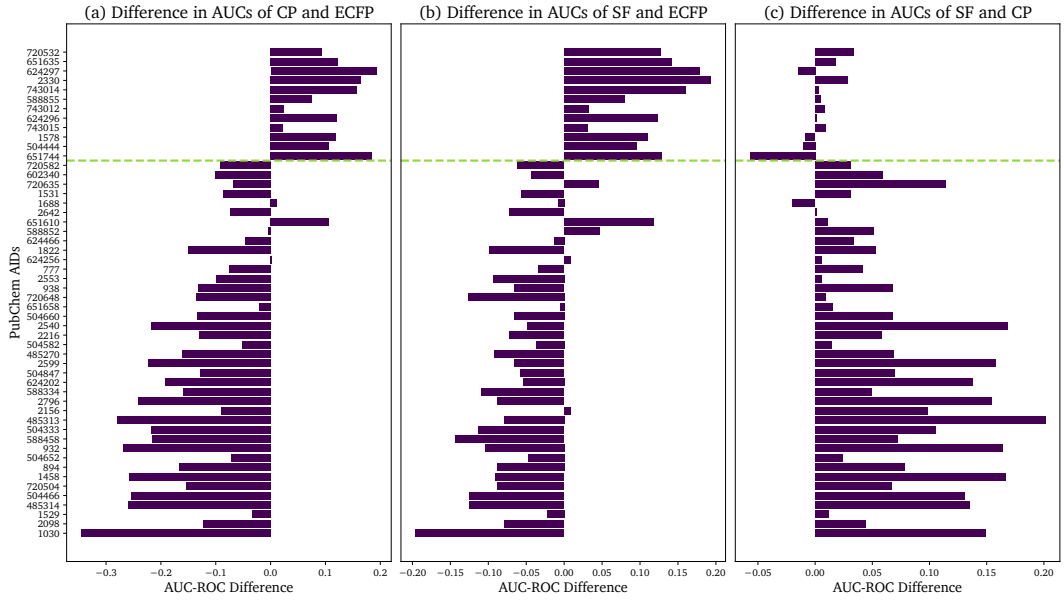


Figure 6.2: Difference in AUC between the CP, ECFP and selected features for all 52 PubChem assays. The dashed green line separates high and low performing assays. The performance metrics of the right descriptor set is subtracted from the ones of the left descriptor set (as seen in the figure titles) to obtain the corresponding differences.

The process of feature engineering fosters expectations on improvements in predictive capability and complementation of CP and ECFP descriptors. Upon inspection of figure 6.2c, CP improvements are mainly achieved within the low performing assays and can be attributed to the complementary information supplied by the ECFP features from feature engineering. However, apparent improvements can be found neither within the high performing assays for CP comparison nor within the low performing assays when comparing the selected features to ECFPs. This infers, that the feature selection process, might have removed descriptors that were vital for the ECFP performance on low performing assays and CP performance on high performing assays.
To solidify the findings drawn from figure 6.2 further metrics are compared and analyzed. The balanced accuracy takes TNR and TPR into account and the descriptor comparison is shown in figure 6.3.
Qualitatively the balanced accuracy between CP and ECFP in figure 6.3 shows the same trend as seen before. The ECFP perform slightly worse which can also be seen in figure 6.1b. The worst drop in balanced accuracy is presented by 1030 with 20%. Furthermore, seven low performing assays score higher balanced accuracys using CP instead of ECFP. The difference for high performing assays is almost identical to the difference in AUC-ROC concerning CP and ECFP comparison.
The selected features compared to the ECFP achieve consistently positive results within the high performing assays and mostly negative results for the low performing assays. Even though the

scoring within low performing assays is slightly higher compared to the AUC-ROC with eight assays that perform better using the engineered features.
In figure 6.3c the difference for balanced accuracy between CP data and selected features is shown. This panel shows the highest discrepancy compared to the AUC-ROC metric. The superiority of SF within the low performing assays is less definitive and the high performing assays lean more towards the CP as descriptors of choice, too. Within the low performing assays there are nine assays with a lower score using selected features. Then again, AID 651744 within high performing assays shows a drop in balanced accuracy of 8% after feature engineering.

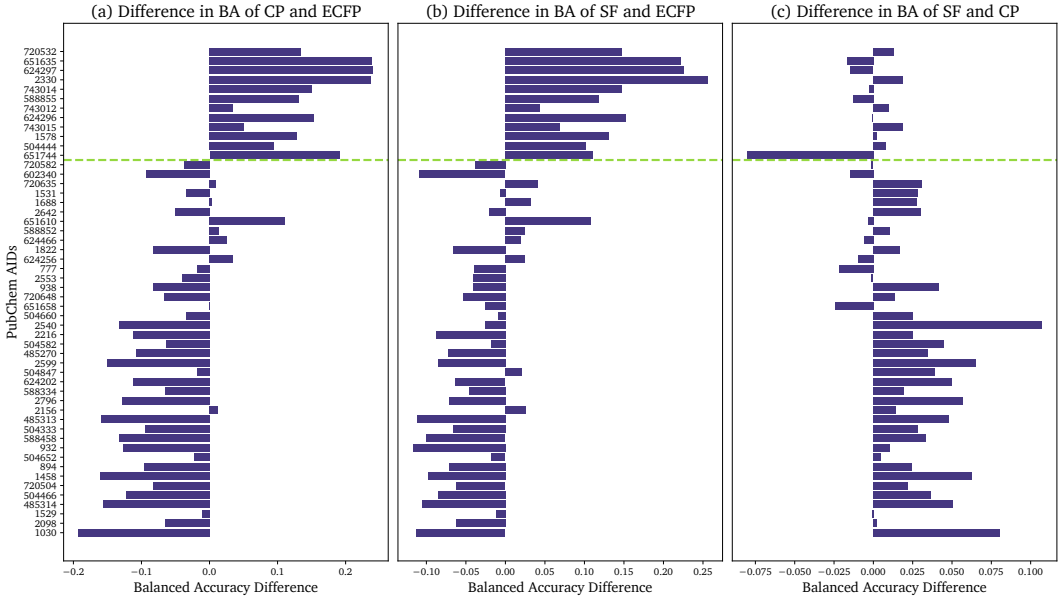


Figure 6.3: Difference in balanced accuracy between the CP, ECFP and selected features for all 52 PubChem assays. The dashed green line separates high and low performing assays. The performance metrics of the right descriptor set is subtracted from the ones of the left descriptor set (as seen in the figure titles) to obtain the corresponding differences.

Even though trends similar to the AUC-ROC, can be observed for the balanced accuracy, the findings from the balanced accuracy incentivize a stronger objection against the selected features' capabilities since CP data obtains better overall scores within high performing assays and low performing assays. Also, the margin between selected features and ECFP for balanced accuracy is still significantly large even if it is slightly smaller compared to the AUC-ROC comparison in figure 6.2b.
The Matthews correlation coefficient is a performance metric that is sensitive to label imbalance. This metric is consulted to confirm the findings in the balanced accuracy comparison.
The differences between the ECFP and CP descriptors mirror figure 6.3a. The high performing assays in figure 6.4a score higher with CP data opposed to low performing assays which score

generally better with ECFPs, apart from a few exemptions that are in agreement with those found for the respective balanced accuracy comparison.
In figure 6.4 the same trends that are apparent for the balanced accuracy reoccur for the Matthews correlation coefficient. The ECFPs perform consistently worse for high performing assays and behave conversely for the low performing assays with exemption of eight assays that show a positive difference indicating better performance with selected features.
The assay-wise performances in right panel of figure 6.4 behave almost identical to the balanced accuracy comparison between CP and selected features. The results show alternating performances within high performing assays slightly leaning towards the CP descriptors. Furthermore, the selected features outperform the CP on low performing assays which is in good agreement with the results from the AUC-ROC comparison and even more so with the results from the balanced accuracy.

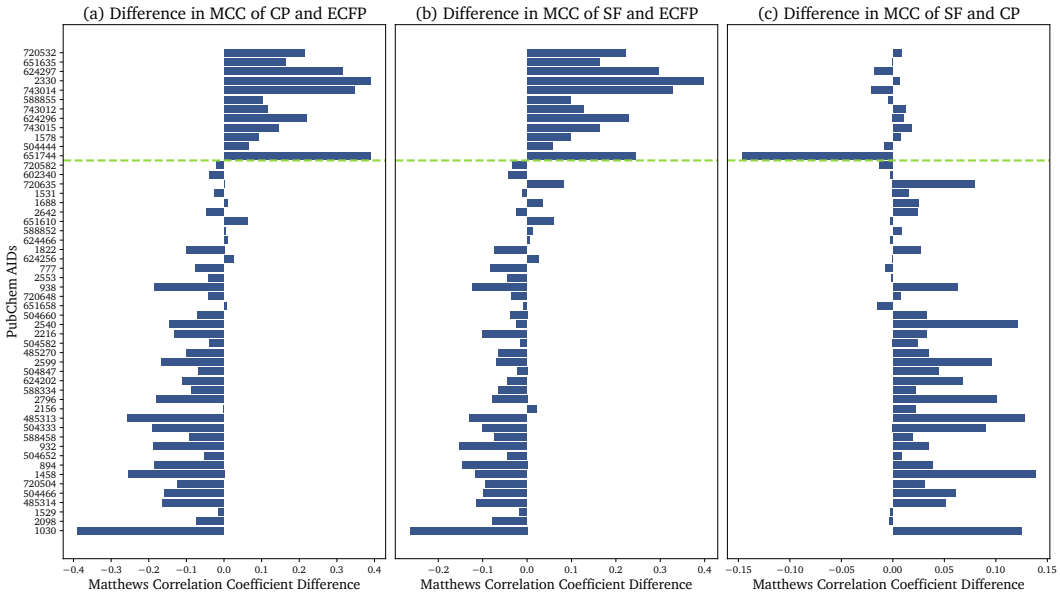


Figure 6.4: Difference in Matthews correlation coefficient between the CP, ECFP and selected features for all 52 PubChem assays. The dashed green line separates high and low performing assays. The performance metrics of the right descriptor set is subtracted from the ones of the left descriptor set (as seen in the figure titles) to obtain the corresponding differences.

Finally, the implications from the Matthews correlation coefficient solidify the results from the preliminary analysis. The selected features are not able to distinctly outperform neither CP features nor ECFPs within their adapted niche (high performing assays for CP and low performing assays for ECFP features). The reduction of noise by finding and eliminating redundant features via feature engineering is expected to elevate performance on the validation training set, even if initial descriptors perform well. Presumably, feature engineering eliminated

informative features since a performance increase within said niches of CP and ECFP is not observed. Nonetheless, they achieve higher performance when all assays are taken into account compared to CP and ECFP. Hence, each descriptor partly compensates for the other's weaknesses by combining the two feature spaces.

Compared to the prior metrics, the sensitivity or true positive rate (TPR) is less complex and cannot accomplish in-depth model validation. On the other hand, TPR is more accessible when it comes to interpreting results. The TPR indicates how many samples were correctly predicted as positives. It is also a measure for the count of samples falsely predicted as negatives (see equation (4.4)). ML models whose predictive power has been validated can be specialized in either detecting negative samples exceptionally well or positives. Analyzing the TPR for the different descriptors elucidates to which category the models at hand belong. The difference in TPR between CP descriptors and ECFPs exhibit the usual trends. Within the high performing assays the CP data excels and ECFPs data excels within its niche of low performing assays. However, upon closer inspection, twelve assays are detected that score a higher TPR within the low performing assays using CP features. The average difference in figure 6.5a in the low performing assays is $-7.1\,\%$ and for high performing assays the average difference is $27\,\%$. Qualitatively, those trends align with the prior performance metrics.

Figure 6.5b presents observations similar to figure 6.4b. Within the low performing assays fourteen assays outperform ECFPs by using selected features. On average the TPR difference between ECFPs and selected features for low performing assays is $-7.6\,\%$. The selected features outperform the ECFPs for all high performing assays but one (AID 743012). The improvement that comes with feature engineering amounts to $26.8\,\%$ on average.

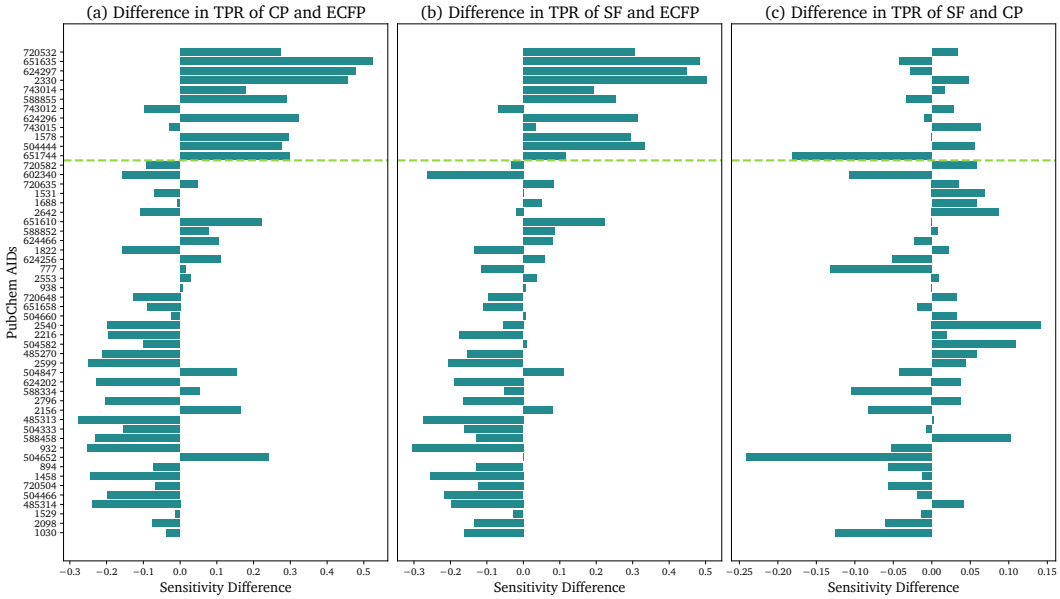


Figure 6.5: Difference in true positive rate between the CP, ECFP and selected features (SF) for all 52 PubChem assays. The dashed green line separates high and low performing assays. The performance metrics of the right descriptor set is subtracted from the ones of the left descriptor set (as seen in the figure titles) to obtain the corresponding differences.

Two different groups of assays must be observed when comparing the TPR for selected features and CP. Firstly, in the high performing assays there is no coherent trend. The average difference of $-0.4\,\%$, i.e. in favor of CP descriptors, is standard when compared with corresponding averages (see table 6.1). On the contrary, the low performing assays for figure 6.5c show a new trend. Instead of clear predicitive dominance of the selected features the TPRs are more or less balanced. The average TPR difference between selected features and CP for low performing assays is $-0.5\,\%$. Therefore, the TPR is the only metric where selected features are on a par with CP descriptors within low performing assays.

The TPRs of low performing assays do not benefit from feature engineering to the same extend as the AUC-ROC. Instead of consistently better scores of selected features a lot of variation is found (see figure 6.5c). One possible explanation is, that ECFPs do not add relevant information to CP descriptors regarding the TPR. If that was true, CP data should obtain better TPR scores than ECFPs within low performing assays which is not the case. Thus, the feature engineering is bound to be the reason for this discrepancy.

From figure 6.5 it can be seen that within the high performing assays the TPR from CP descriptors exceeds the ECFPs' by up to $50\,\%$ and $27\,\%$ on average. This implies that within assays that can be well categorized by CP data, the positive samples are particularly well classified.

The specificity or true negative rate (TNR) was also calculated for all prediction runs. The TNR quantifies the correctness of predicting negative labels. It complements the aforementioned

TPR and is not suitable for evaluating overall model performance. Furthermore, an assessment of the feature engineering and the capability to correctly predict negative values can be conducted by comparing all descriptors among all PubChem assays. This comparison is presented in figure 6.6

In the difference between CP and ECFP it is most apparent that the high performing assays do not score as high as in prior metric comparisons (see figure 6.6). Instead of CP data scoring consitently and significantly higher than ECFPs, CP is outperformed on several occasions. Within the high performing assays the average TNR difference is only $2.6\,\%$ compared to $27.2\,\%$ for TPR (see table 6.1). Within low performing assays the trend follows abovementioned metrics. CP descriptors score lower than ECFPs with a few exemptions and an average of $-6.1\,\%$ is computed which is comparable to results obtained above.

The comparison of selected features and ECFPs shows novel trends as well. The high performing assays are performing inconsistently, analogous to the behaviour described for figure 6.6a. The average difference for high performing assays in the middle panel of figure 6.6 is $2.0\,\%$. The low performing assays behave in an unseen way as well. Previously, this part of metric comparison was dominated by the ECFPs. For the TNR this is not the case. Alternating difference scores are reported with an average TNR difference of $-0.6\,\%$ which is roughly one magnitude smaller than usual (see table 6.1).

In figure 6.6c the differences in TNR for selected features and CP are presented. The high performing assays alternate around zero and the low performing assays show better predictive performance with respect to the TNR via selected features.

Novel insights with respect to the TNR can be gained from the first two panels of figure 6.6 in particular. It is apparent that CP features are less suitable for correctly predicting inactive compounds. They perform comparably worse especially in high performing assays where they should excel at theoretically. The feature engineering and therefore the additon of ECFPs leads to improvements mostly restricted to low performing assays that are on a par with the ECFP-only modelling approach. These findings suggest that information present in the ECFPs improves the TNR of low performing assays and is retained after feature engineering. The TNR is the only metric which is modelled equally well by ECFPs and SF for low performing assays.

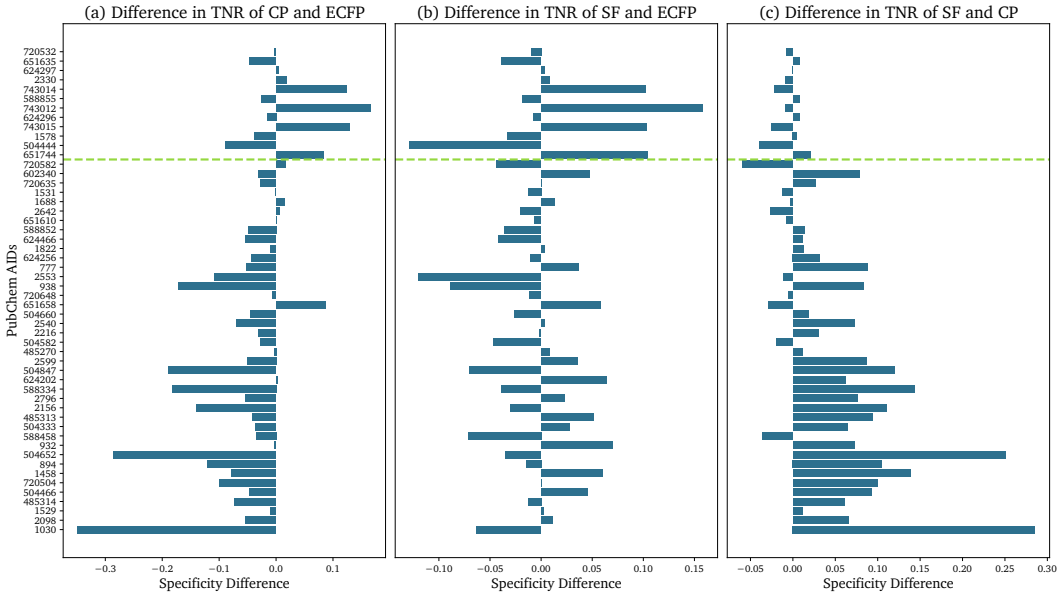


Figure 6.6: Difference in true negative rate between the CP, ECFP and selected features (SF) for all 52 PubChem assays. The dashed green line separates high performing assays and low performing assays. The performance metrics of the right descriptor set is subtracted from the ones of the left descriptor set (as seen in the figure titles) to obtain the corresponding differences.

In table 6.1 the average difference between the possible descriptor combinations can be seen for the high performing assays and low performing assays. The general trends hint towards better performance of CP in high performing assays and ECFP performing better in low performing assays (hence the names). The same can be said for the comparison between the combined feature space and the ECFP with the difference that the combined features do not score as poorly in low performing assays. In the high performing assays CP only performs generally better than the combined features and in the low performing assays the combined perform better than CP. Exceptions from these general trends can be especially seen for TPR and TNR as described above.

Table 6.1: Average evaluation metrics sorted by high performing assays and low performing assays. 'CP' denote the prediction run that used the cell-painting descriptors, 'FP' denotes the run with the structural fingerprints and 'SF' the combined, selected features. The listed values are the differences in the respective evaluation metric.

	high performing assays			low performing assays		
Metric	CP vs FP	SF vs FP	SF vs CP	CP vs FP	SF vs FP	SF vs CP
AUC	0.1155	0.1167	0.0012	-0.1337	-0.0611	0.0726
BA	0.1488	0.1440	-0.0048	-0.0662	-0.0412	0.0250
MCC	0.2132	0.2019	-0.0113	-0.0926	-0.0546	0.0380
TPR	0.2721	0.2678	-0.004	-0.0713	-0.0763	-0.0051
TNR	0.0255	0.0202	-0.0053	-0.0613	-0.0061	0.0552

The findings obtained from the analysis of CP, ECFPs and selected features raises the question, if a different approach to feature selection can improve the performance. Especially the deficiencies that occured for the TPR within low performing assays which could not be mitigated by feature engineering (see figure 6.5). To investigate this further, a fourth modelling approach is explored that uses all features (3816 in total). The results are compared analogous to the prior paragraphs. The AUC-ROC comparison is shown in figure 6.7.

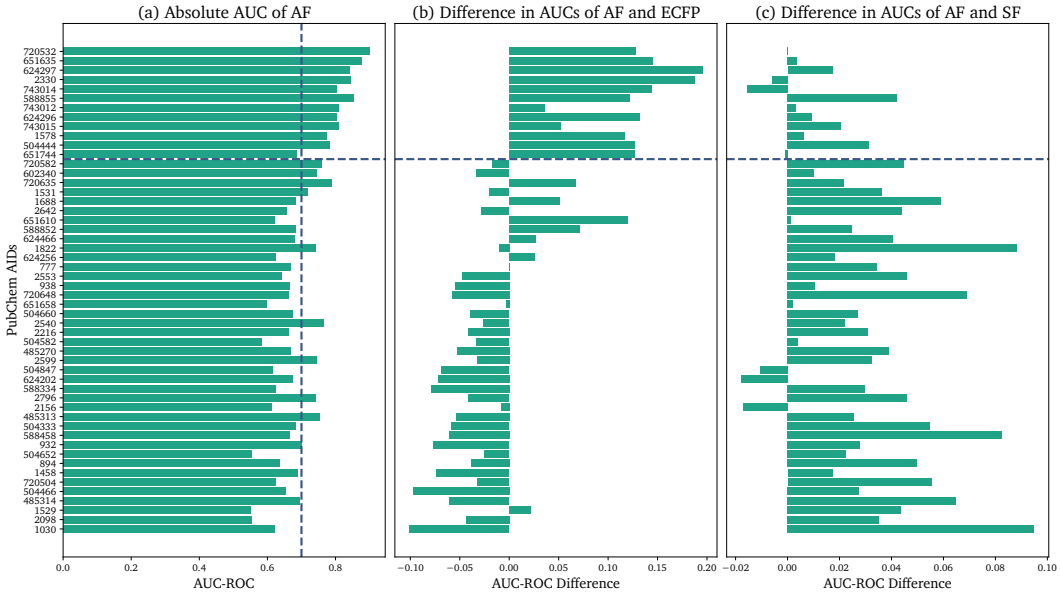


Figure 6.7: AUC-ROC scores from the modelling with all features (AF). Subplot (a) shows the abolute AUC-ROCs whereas (b) and (c) show the difference of all features to the ECFPs and selected features. The blue, horizontal, dashed line separates high and low performing assays and the vertical line in subplot (a) marks a performance of 0.7. The performance metrics of the right descriptor set is subtracted from the ones of the left descriptor set (as seen in the figure titles) to obtain the corresponding differences. The y-axis is sorted in the usual manner.

In the left panel the absolute AUC-ROCs are shown and twenty assays score a AUC-ROC higher than 0.7. In the middle panel the differences of all features and ECFP predictions are visualized. For high performing assays modelling with all features scores significantly higher compared to ECFPs and for low performing assays ECFPs perform better. The right panel shows the difference in AUC-ROCs for all features and selected features. All features score higher in almost all PubChem assays regardless of high or low performing assays.
The selected features did not perform well with respect to TPR especially in low performing assays which could indicate deficiencies within the features engineering process. To validate this assumption the TPRs from the modelling apporach with all features are shown in figure 6.8. The absolute TPRs are shown in the left panel and the results for all assays are in agreement with their order (which corresponds to the AUC-ROC from the CP-only modelling). In figure 6.8b the difference in between all features and ECFPs-only is shown. The high performing assays perform significantly better with all features and the low performing assays do not show an unambiguous trend. Eventually, the TPR comparison between all features and selected features reveals a strong dominance of all features within low performing assays. This trend was missing when selected features where compared to CP data. Within the high performing assays however, the selected features outperform all features on most occasions.

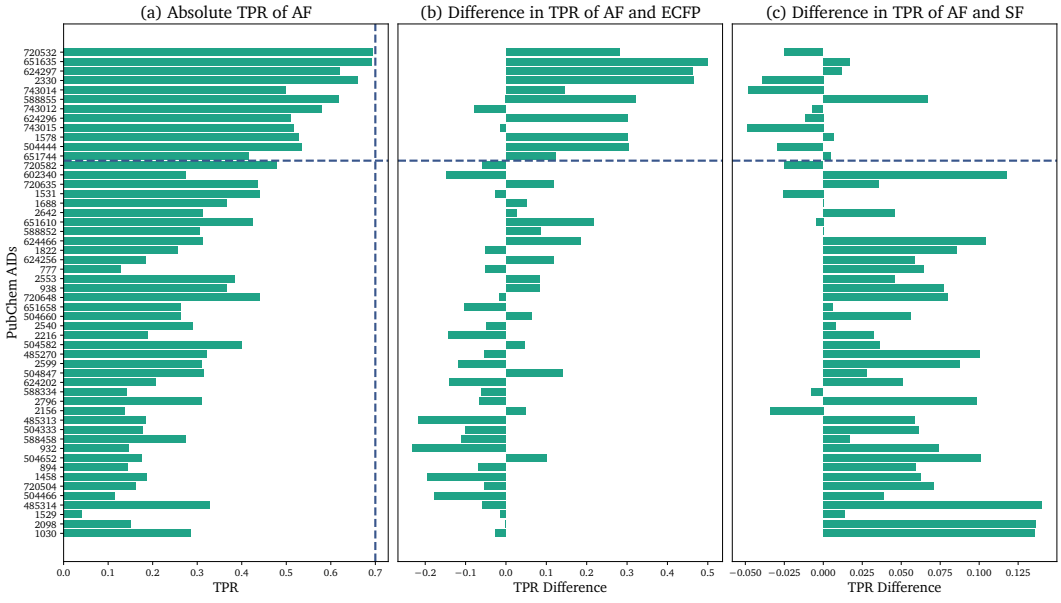


Figure 6.8: TPR scores from the modelling with all features (AF). Subplot (a) shows the abolute TPRs whereas (b) and (c) show the differences of all features to the ECFPs and selected features. The blue, horizontal, dashed line separates high and low performing assays and the vertical line in subplot (a) marks a performance of 0.7.

The conclusion that can be drawn from these results is that combining ECFPs and CP data without feature engineering allows improving the performance significantly for TPR as well

as for the AUC-ROC. However, for assays that CP is already able to predict confidently, the TPR is diminished by a rigorous feature combination without engineering. As a summary, the combination of features from CP and ECFP never leads to a general performance increase. Either the increase is achieved within high performing assays or within low performing assays, respectively. The performance of CP-only modelling within high performing assays as well as the performance of ECFP-only modelling within low performing assays remains unmatched by combinatory approaches.

6.3 Channel Enrichment Analysis for Important Features within High and Low Performing PubChem Assays

The feature engineering of the CP data might be able to further illuminate why some assays are better predictable with CP descriptores and others are not. For clarification, in this section the notation of high performing assays and low performing assays introduced in section 6.1 is still in use.

The features in the CP data set are experimentally obtained by fluorescence microscopy. The staining and image generation process described in section 4.4 utilizes six fluorescent staining agents and thereby measures five different fluorescence channels corresponding to five different cellular organelles or compartments, namely DNA, RNA, Mito, ER and AGP (see table 4.1).

If CP data can reliably predict a given bioassay, unusual activity within the fluorescent channels should be noticeable. Therefore, features belonging to specific channels should be more critical for the assay's predictivity. For example, a genotoxic assay should have higher contributions from the RNA and DNA channels and, in turn, lower contributions from the remaining channels. On the other hand, if an assay is not well predictable by CP descriptors, the importance of distinct features and channels are supposed to be less enriched. In the worst case, features and their corresponding channels are randomly selected and are therefore uniformly represented. If this hypothesis holds, each channel's normalised standard deviation should be high within the high performing assays and low within the low performing assays. The feature channels f_c can be inferred from the dataset's metadata[3] and are counted among the important features. Their frequencies $\nu_c^{(a)}$ are calculated by dividing f_c by the total number of important features (from feature selection) of the corresponding assay $N_f^{(a)}$.

$$\nu_c^{(a)} = \frac{f_c}{N_f^{(a)}} \tag{6.1}$$

Afterwards, the frequencies are normalized for each channel with respect to all bioassays for easier comparison. The normalization was performed using the standard scaler from the

sklearn preprocessing library.[21] This scaler transforms each channel's frequencies to adopt an average of zero and a standard deviation of 1. Afterwards, every value is multiplied by 100 for easier visual interpretation, resulting in $\widetilde{\nu}_c^{(a)}$.

$$\widetilde{\nu}_c^{(a)} = \text{transform}(\nu_c^{(a)}) \cdot 100 \tag{6.2}$$

The channel-wise average standard deviation within high performing assays, $\widetilde{\sigma}_c^{\text{hpa}}$, and low performing assays, $\widetilde{\sigma}_c^{\text{lpa}}$, is calculated by the formula given in equation (6.3).

$$\widetilde{\sigma}_c^{\text{hpa}} = \sqrt{\frac{1}{N_{\text{hpa}}} \cdot \sum_{a}^{hpa} \left(\widetilde{\nu}_c^{(a)} - \langle \widetilde{\nu}_c \rangle_{\text{hpa}}\right)^2} \qquad \widetilde{\sigma}_c^{\text{lpa}} = \sqrt{\frac{1}{N_{\text{lpa}}} \cdot \sum_{a}^{lpa} \left(\widetilde{\nu}_c^{(a)} - \langle \widetilde{\nu}_c \rangle_{\text{lpa}}\right)^2} \tag{6.3}$$

For easier notation $\widetilde{\sigma}_c^{\text{hpa}}$ and $\widetilde{\sigma}_c^{\text{lpa}}$ are referred to as the assay-group standard deviations. The results for each channel are shown graphically in figure 6.9 and also in table 6.2 with their ratio for better comparison. It can be seen, that the DNA, RNA, Mito and ER channels exhibit a ratio bigger than one, which corresponds to a higher assay-group standard deviation for high performing assays.

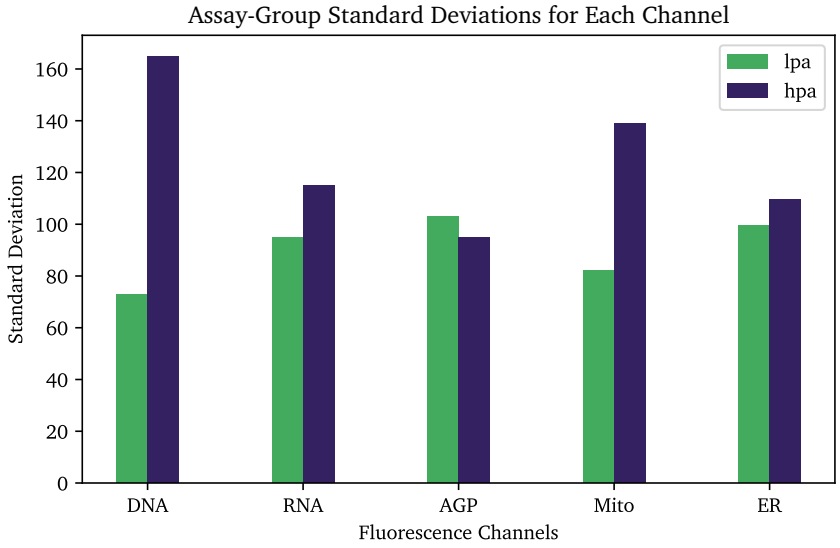


Figure 6.9: Assay-group standard deviations for low and high performing assays. For each fluorescence channel the standard deviations are compared. The low performing assays are shown in green and high performing assays are shown in purple.

The ratio in standard deviation varies from channel to channel. AGP features a higher standard deviation for low performing assays and has, therefore, the smallest ratio. The ER channel standard deviation is slightly enriched for high performing assays and has the second-lowest ratio. Next is the Mito channel and the RNA channel. The most significant ratio in standard deviation by far is exhibited by the DNA channel.

Table 6.2: Assay-normalized standard deviation per channel for the low performing assays and high performing assays. In the last row the ratio of the two is shown as well. A ratio greater 1 means, that the channel is enriched in the high performing assays compared to the low performing assays which is the case for DNA, RNA, Mito and ER.

Assay Group	DNA-std	RNA-std	AGP-std	Mito-std	ER-std
$\widetilde{\sigma}_c^{\text{lpa}}$	72.8	94.7	103.1	82.0	99.7
$\widetilde{\sigma}_c^{\text{hpa}}$	164.8	115.1	95.0	139.1	109.5
$\widetilde{\sigma}_c^{\text{hpa}}/\widetilde{\sigma}_c^{\text{lpa}}$	2.26	1.22	0.92	1.70	1.10

The enrichment strategy presented here showed a clear distinction between PubChem assays categorized as high and low performing assays. Four out of five fluorescence channels presented a higher assay-group standard deviation for high performing assays in contrast to low performing assays. The features used to generate the channel counts in the first place were selected by feature importance and generated by cellular imaging. Therefore, the enrichment of specific channels corresponds to their importance within the assay group. Conclusively, by performing this enrichment analysis, a connection is generated between cellular morphology and an assay's predictive capabilities via CP data.

6.4 Phenotypic Annotations Analysis of High Performing PubChem Assays

In an attempt to annotate the PubChem assays, the descriptions of the PubChem assays were manually screened for further information. The goal was to find terms or keywords relating bioassay endpoints to cellular morphology and cytotoxicity. The terms that could be filtered out are shown in table 6.3. The presented term list is not comprehensive, but the terms refer to phenomena that are most likely visible on a cellular scale and might be affected by small compounds.

Table 6.3: Phenotypic terms that can be associated with individual PubChem assays. These terms were manually filtered from the descriptions of the PubChem assays available at https://pubchem.ncbi.nlm.nih.gov/.[52]

Acronym	Associated Phenotypic Terms
Adipgen	Adipogenesis, Obesity
Ang-gen	Angiogenesis
C-Grwth	Cell Growth, Cell Viability
C-Death	Apoptosis, Cell Death
Im-Resp	Immune Response
Inflamm	Inflammation
Snscnce	Senescence
Mei/Mit	Meiosis, Mitosis
C-Stres	Xenobiotics, Toxins, Cell Stress
Signall	Signalling, Secretion, Hormones
CNS/NDD	CNS, Epilepsy, Depression, NDD
C-Entry	Invasion, Cell Entry
Mitotox	Mitotoxicity
A-Cancr	Anti-Cancer
Genome	Genome Integrity, DNA-Repair, genotoxicity
Prteom	Ubiquitinylization, Protein Regulation, Proteome influencing

In figure 6.10 the PubChem assays with their annotations are shown. White squares imply that the phenotypic term is related to the corresponding PubChem assay, and a blue square is assigned if the term cannot be associated with confidence. The decision-making process is intuitive up to a certain amount and prone to human error. For example, genotoxicity is related to cell death. If a large portion of the DNA is damaged, the cell initiates apoptosis and dies. However, the AID 2540 probes inhibitors for a protein called SENP8 that moderates the maturation of Nedd8, which plays a crucial role in DNA-repair. Herein, SENP8 is considered to be a modulator of DNA-Repair and is not directly connected to apoptosis. Therefore AID 2540 has a white square at 'Genome' and a blue square at 'C-Death' even though the two are inseparable in practice.

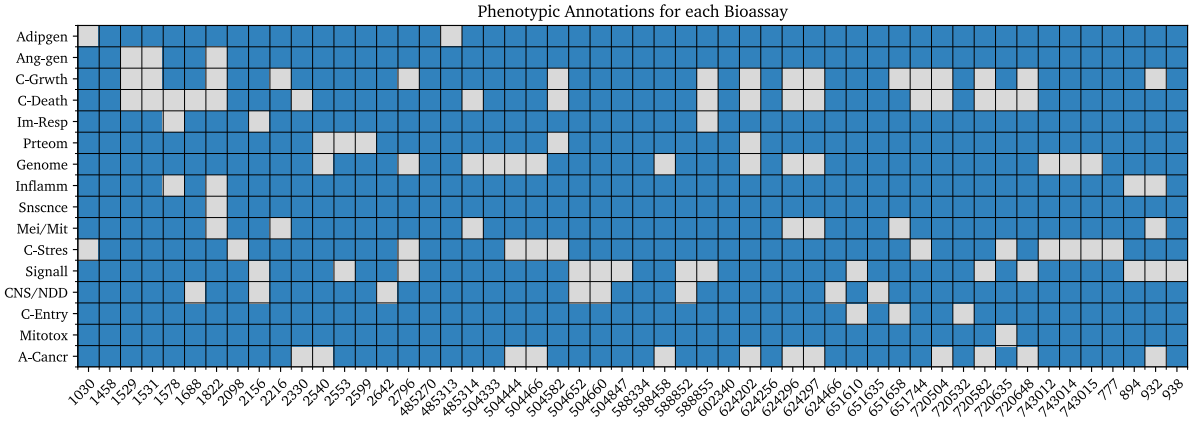


Figure 6.10: Phenotypic terms that can be associated with individual PubChem assays. These terms were manually filtered from the descriptions of the PubChem assays available at https://pubchem.ncbi.nlm.nih.gov/. White fields correspond to presence and blue entries correspond absence of a term within an assay.

It is noteworthy that the complete annotation matrix was created before the prediction performances were recorded and can be considered unbiased. This matrix allows testing for enriched annotations within high performing assays and low performing assays respectively.

The state for a phenotypic term within an assay can either be 'present' or 'absent', denoted by a 1 or 0. To calculate the partial abundance of a given phenotypic term within a specific assay group A_p^{xpa} these binary states for all relevant assays $s_p^{(a)}$ are summed up.

$$A_p^{\text{lpa}} = \sum_a^{\text{lpa}} s_p^{(a)} \quad A_p^{\text{hpa}} = \sum_a^{\text{hpa}} s_p^{(a)} \tag{6.4}$$

The relative abundances or frequencies a_p^{xpa} with an assay group can be calculated by dividing the partial abundances by the total abundance of the respective phenotypic term.

$$a_p^{\text{lpa}} = \frac{A_p^{\text{lpa}}}{A_p^{\text{lpa}} + A_p^{\text{hpa}}} \quad a_p^{\text{hpa}} = \frac{A_p^{\text{hpa}}}{A_p^{\text{lpa}} + A_p^{\text{hpa}}} \tag{6.5}$$

The enrichment of phenotypic term within an assay group needs to incorporate the total size of that group N_{xpa}. Therefore, the frequencies are divided by the corresponding group size to obtain the term related enrichment R_p^{xpa}.

$$R_p^{\text{lpa}} = \frac{a_p^{\text{lpa}}}{N_{\text{lpa}}} \quad R_p^{\text{hpa}} = \frac{a_p^{\text{hpa}}}{N_{\text{hpa}}} \tag{6.6}$$

The resulting measure is the phenotypic term frequency per assay and describes the enrichment of a phenotypic term within the corresponding assay group.
In table 6.4 the enrichment for the high performing assays and low performing assays as well as the difference of the two is shown. A negative difference means that this phenotypic term

is enriched in the low performing assays and vice versa. It can be seen that phenotypic enrichment is positive for 'C-Grwth', 'C-Death', 'Im-Resp', 'Genome', 'Inflamm', 'Mei/Mit', 'C-Stres', 'C-Entry', and 'A-Cancr'. 'Im-Resp', 'Genome' and 'C-Death' are the three terms showing the largest enrichment in high performing assayss.

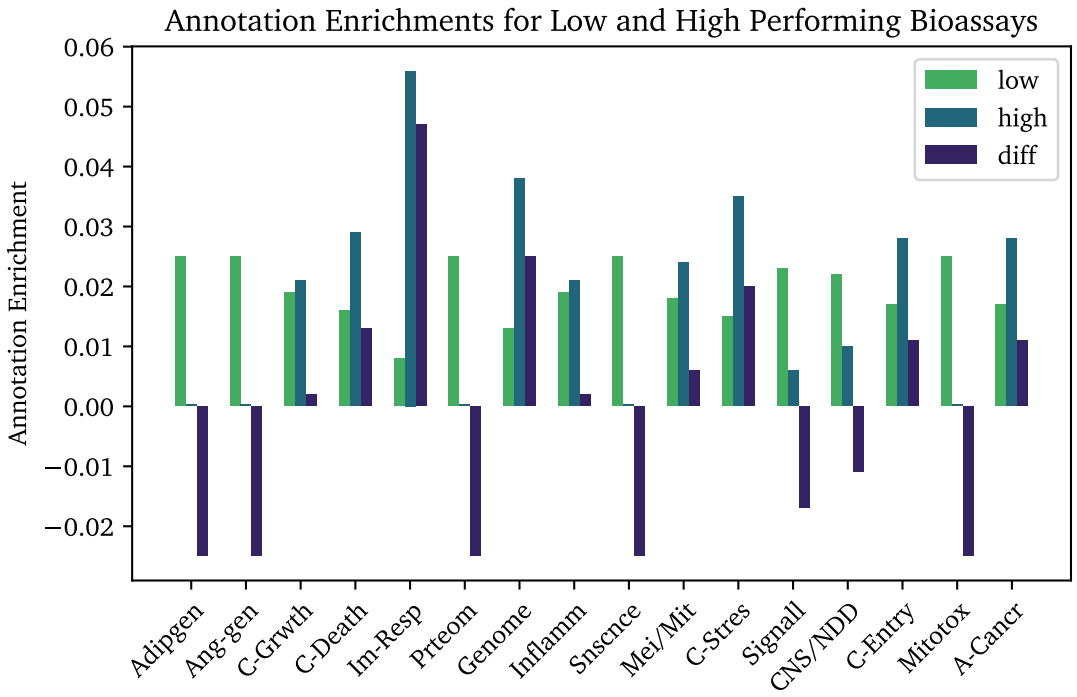


Figure 6.11: Annotations Enrichment in high performing assays and low performing assays and the difference of the two.

Table 6.4: Enrichment of phenotypic terms in high performing assays and low performing assays and the difference between the two. A high number corresponds to a higher frequency of the corresponding term within the group of assays. The difference clarifies if the relative frequency is higher or lower in the high performing assays and quantifies that enrichment in a comparable manner.

Group	Adipgen	Ang-gen	C-Grwth	C-Death	Im-Resp	Prteom	Genome	Inflamm
low	0.025	0.025	0.019	0.016	0.008	0.025	0.013	0.019
high	0.0	0.0	0.021	0.029	0.056	0.0	0.038	0.021
diff	-0.025	-0.025	0.002	0.013	0.047	-0.025	0.025	0.002
Group	Snscnce	Mei/Mit	C-Stres	Signall	CNS/NDD	C-Entry	Mitotox	A-Cancr
low	0.025	0.018	0.015	0.023	0.022	0.017	0.025	0.017
high	0.0	0.024	0.035	0.006	0.01	0.028	0.0	0.028
diff	-0.025	0.006	0.02	-0.017	-0.011	0.011	-0.025	0.011

From this analysis, it can be concluded that assays that probe endpoints related to these

specified phenotypes exhibit higher predictive capability with CP descriptors. The fact that genome integrity and DNA-repair score high values is also in agreement with the channel enrichment analysis. As shown in table 6.2, the DNA channel has the highest ratio among all five.

6.5 Gene Ontology Term Analysis

GO terms are keywords referring to molecular functions or cellular processes associated with proteins. Each protein target is assigned a terminal GO term that is part of a hierarchical network. Hence, every GO term can be associated with parent GO terms until the highest level of generalization is reached, leading to a tree of GO terms that are associated with each known protein.

As an addition to the manually generated phenotypic annotations GO terms are generated for each bioassay which probes a protein target. Twelve PubChem assays do not probe protein targets, six of which are part of the high performing assays group, and they are further discussed below. The terminal GO terms were obtained from $\mathrm{https://www.ebi.ac.uk/interpro}$ for each target and the entire hierarchy is extracted from $\mathrm{http://geneontology.org}$. Since GO terms are hierarchical, a protein target with several unique terminal GO terms can have duplicate high order GO terms. For the analysis frequency information, i.e. how often a GO term appears within a given protein target is discarded. Afterwards, the relative abundance of GO terms within high performing assays and low performing assays are computed and compared. GO terms that are only present once throughout all assays are considered too rare to be included in this comparative analysis and are therefore discarded. Since the GO terms directly describe mechanisms within the cell, a deeper understanding of the performances is anticipated. The concept is analogous to the manually developed phenotypic annotations because certain cellular mechanisms are anticipated to be more abundant within a certain group of assays.

All GO terms found for the 52 PubChem assays and their relative abundances within high and low performing assays are shown in figure 6.12. Roughly a third of the GO terms are associated with the high performing assays and are also more abundant in this group. Furthermore, some of the GO terms are only present for high performing assays at all. A majority of the GO terms found in total are present within protein targets of low performing assays. A considerable amount of these is not present for high performing assays.

One problem that must be addressed concerns the small sample size. The high performing assays only features six protein targets. That means the relative abundances are either $1/6$, $2/6$, etc. The low performing assays suffer from this to a lower extent. Therefore, this analysis has to be interpreted with the lack of data in mind.

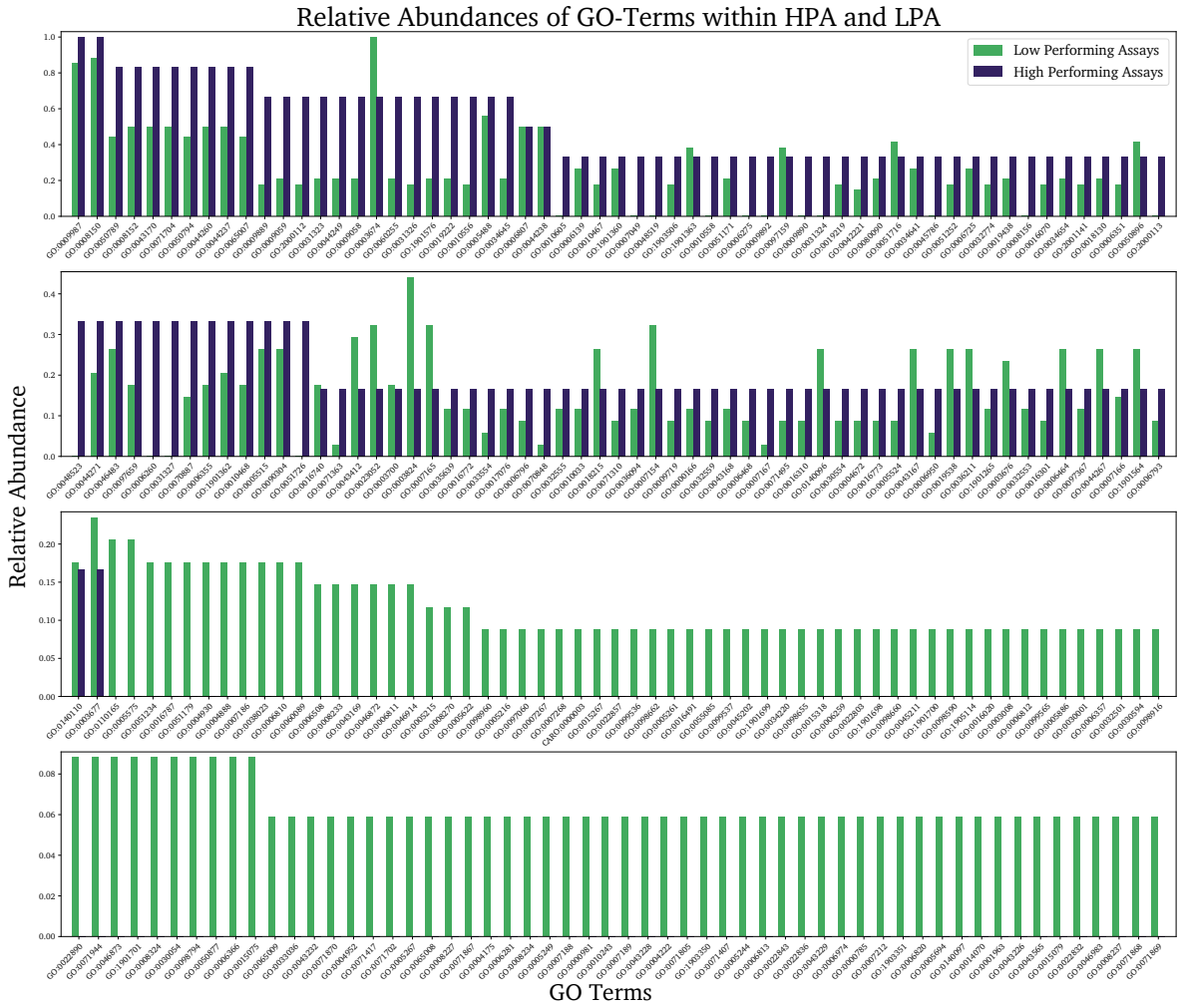


Figure 6.12: Relative abundances of GO Terms in each assay group

The figure 6.12 serves the sole purpose of visualization since it is only annotated with the GO term identifier and not with its description. The descriptions for GO terms enriched in high or low performing assays can be found in table 8.1 in the appendix. The GO terms for low performing assays are very diverse and many instances related to synapses, ion transport and G-protein coupled receptors can be found. For high performing assays the terms are enriched for metabolic and biosynthetic processes of macromolecules as well as the regulation and processing of DNA and RNA. This is in agreement with findings from section 6.3 where the most important CP features are analyzed for enrichment of channel variance. The DNA channel of high performing assays is enriched by 226 % compared to low performing assays. Also in section 6.4 the phenotypic annotation connected to genotoxicity and genetic regulation is highly enriched for high performing assays. CP descriptors seem to be particularly sensitive to bioassays primarily related to the regulation and processing of DNA and RNA. Nevertheless,

the nucleus is the largest organelle in the cell. And the CP descriptors are rooted in fluorescence microscopy which is dependend on signal resolution. Since most of the signal that is eventually imaged stems from the nucleus and is therefore related to DNA and RNA, this conclusion seems logical.

Additionally, the high performing assays that do not probe gene products are inspected in further detail. The assays 651635, 651744, 720532, 743014, 743012 and 743015 are involved. Five out of these six probe complex cellular processes instead of single gene products. 651635 is the only assay that probes a gene product with no GO terms available by now. In section 4.6 the assay 720532 was explained in further detail as an example because it is the bioassay scoring the highest AUC-ROC via CP descriptors. This assay probed the inhibition of virus entry by targeting cellular processes exploited by the virus. 651744 probes cytotoxicity within NIH3T3 cells via luminescence. 743012, 743014 and 743015 are parts of the same experiment where genotoxicity is screened for different mutants of the DT40 cell line.

Remarkably, five out of seven bioassays within high performing assays probe for complex cellular processes. On the one hand, this hinders the annotation by GO terms. On the other hand, this indicates that CP contains information that is particularly applicable for more complex cellular processes going beyond molecular functionality. This finding agrees with the richness in information that was also found by Simm *et al.* [8] when their CP based model scored an AUC-ROC greater than 0.9 for 34 out of 600 assay. The authors do not specify what endpoints these assays probe in detail. The execution of an analysis similar to this one could validate the notion that CP is most suitable for modelling complex biological mechanisms.

7 Conclusion and Outlook

In this novel approach, the CP data by Bray *et al.* were used to predict certain PubChem assays that were selected for their relation to cytotoxicity. Nassiri and McCall[12] used model performance as an indication for a common MoA. A similar approach is considered in this project. By comparatively analyzing performance metrics from an RFC model among different descriptors, the relation between bioassays and cellular morphology as captured by the CP assay is inferred. Within a comparative analysis of the different modelling approaches, twelve bioassay exhibit elevated predictive potential via CP descriptors which were able to outperform ECFPs on nearly every metric. However, the specificity and sensitivity showed different behaviour in comparison to AUC-ROC, balanced accuracy and MCC. Especially the TPR was elevated within the group of high performing assays. That leads to the conclusion that CP descriptors have a higher chance to label a positive compound correctly. This characteristic is beneficial for toxicity prediction since the ability to correctly predict toxic (positive) compounds can prevent unnecessary testing and harm. The TNR was found to be better predicted using either ECFPs or the feature engineered descriptor set. However, the selected features could not achieve an improvement over the individual predictive capabilities. It was also shown that ECFPs and CP descriptors complement each other, which is in agreement with the findings from Lapins and Spjuth.[16] For drug safety projects, it is suggested to examine if CP descriptors characterize the endpoint well. If so, feature engineering can be omitted for ML approaches comparable to the one presented here. Predicting the TNR is presumably better achievable via ECFPs.

A new enrichment metric developed from feature importance analysis is presented here. It measures the fluorescence channel-wise assay group standard deviation. This enrichment metric connects the results from feature engineering with cellular morphology, and it was found that for four out of five fluorescent channels, enrichment could be measured. This approach can be used as the first entry point when working with CP data to categorize bioassays before prediction runs and focus on the assays that are estimated to be well characterized by CP data. Lapins and Spjuth[16] used MoA and targets to annotate compounds in their work and used CP, structural fingerprints and gene expression profiles as descriptors. A similar approach was chosen in the sense that structural fingerprints and CP features were used as descriptors. Furthermore, their use of genetic information as a descriptor led to the idea that phenotypic annotations and GO terms could be used to investigate if certain biological mechanisms are responsible for performance differences within the PubChem bioassays. Opposite to Lapins et Spjuth,[16] the genetic information was not used as a descriptor but to illuminate the information that facilitates the high CP predictivity.

The phenotypic annotations have been manually generated and connect the bioassays readouts to cellular processes (e.g. signalling, proteome regulation, genotoxicity etc.) that are likely to result in morphological changes. By calculating the phenotypic enrichment for high performing assays and low performing assays, it was found that endpoints that relate to genome integrity, DNA-repair, genotoxicity, cell-death, apoptosis, cell stress, toxins and immune response are generally better described by CP descriptors.
Since the results agreed with the findings from the channel enrichment, the phenotypic analysis can be considered a confirmatory argument. However, its generalizability is hindered by its irreproducibility and imperfect knowledge. In an attempt to mitigate these shortcomings, the GO term analysis was conducted yielding results affirmative of the channel enrichment. Many GO terms found for the high performing assays are connected to DNA, RNA and macromolecule synthesis. The caveat for GO term analysis stems from the relatively small sample size. Only six bioassays within the high performing assays are probing gene products, therefore qualifying for this analysis. Nonetheless, it is suggested that GO terms should be included for a mechanistic and cellular understanding of the prediction performance when CP data is in use, especially if a large number of endpoints are included in the analysis.

Bibliography

[1] Singh, S.; Khanna, V. K.; *In Vitro Toxicology*; Elsevier, 2018; pp 1–19.

[2] Carpenter, A. E.; Jones, T. R. CellProfiler: image analysis software for identifying and quantifying cell phenotypes *Genome Biology* **2006**, *7*, R100.

[3] Bray, M.-A. et al. A dataset of images and morphological profiles of 30 000 small-molecule treatments using the Cell Painting assay *GigaScience* **2017**, *6*.

[4] Mervin, L. H.; Cao, Q. Understanding Cytotoxicity and Cytostaticity in a High-Throughput Screening Collection *ACS Chemical Biology* **2016**, *11*, 3007–3023.

[5] Katara, P. Role of bioinformatics and pharmacogenomics in drug discovery and development process *Network Modeling Analysis in Health Informatics and Bioinformatics* **2013**, *2*, 225–230.

[6] Myers, S.; Baker, A. Drug discovery-an operating model for a new era *Nature Biotechnology* **2001**, *19*, 727–730.

[7] Nelson, M. R.; Bacanu, S.-A.; Genome-wide approaches to identify pharmacogenetic contributions to adverse drug reactions *The Pharmacogenomics Journal* **2008**, *9*, 23–33.

[8] Simm, J. et al. Repurposed high-throughput images enable biological activity prediction for drug discovery **2017**,

[9] Bray, M.-A.; Singh, S. et al. Cell Painting, a high-content image-based assay for morphological profiling using multiplexed fluorescent dyes *Nature Protocols* **2016**, *11*, 1757–1774.

[10] Rohban, M. H.; Singh, S. et al. Systematic morphological profiling of human gene and allele function via Cell Painting *eLife* **2017**, *6*.

[11] Gustafsdottir, S. M.; Ljosa, V. et al. Multiplex Cytological Profiling Assay to Measure Diverse Cellular States *PLoS ONE* **2013**, *8*, e80999.

[12] Nassiri, I.; McCall, M. N. Systematic exploration of cell morphological phenotypes associated with a transcriptomic query *Nucleic Acids Research* **2018**, *46*, e116–e116.

[13] Wawer, M. J.; Jaramillo, D. E. et al. Automated Structure-Activity Relationship Mining *Journal of Biomolecular Screening* **2014**, *19*, 738–748.

[14] Wiemann, S.; Pennacchio C. The ORFeome Collaboration: a genome-scale human ORF-clone resource *Nature Methods* **2016**, *13*, 191–192.

[15] Yang, X.; Boehm, J.S. et al. A public genome-scale lentiviral expression library of human ORFs *Nature Methods* **2011**, *8*, 659–661.

[16] Lapins, M.; Spjuth, O. Evaluation of Gene Expression and Phenotypic Profiling Data as Quantitative Descriptors for Predicting Drug Targets and Mechanisms of Action **2019**,

[17] Subramanian, A.; Narayan, R. et al. A Next Generation Connectivity Map: L1000 Platform and the First 1,000,000 Profiles *Cell* **2017**, *171*, 1437–1452.e17.

[18] Simm, J.; Klambauer, G. et al. Repurposing High-Throughput Image Assays Enables Biological Activity Prediction for Drug Discovery *Cell Chemical Biology* **2018**, *25*, 611–618.e3.

[19] Chawla, N. V.; Bowyer, K. W. et al. SMOTE: Synthetic Minority Over-sampling Technique *Journal of Artificial Intelligence Research* **2002**, *16*, 321–357.

[20] Kim, S.; Chen, J. et al. PubChem in 2021: new data content and improved web interfaces *Nucleic Acids Research* **2020**, *49*, D1388–D1395.

[21] Pedregosa, F.; Varoquaux, G. et al. Scikit-learn: Machine Learning in Python *CoRR* **2012**, *abs/1201.0490*.

[22] Weininger, D. SMILES, a chemical language and information system. 1. Introduction to methodology and encoding rules *Journal of Chemical Information and Modeling* **1988**, *28*, 31–36.

[23] *Pure and Applied Chemistry*; De Gruyter, 2014.

[24] Inc., D. C. I. S. Daylight Theory Manual. https://www.daylight.com/dayhtml/doc/theory/index.pdf, **2011**; *last time opened: 24.02.2021*.

[25] Weininger, D. et. al SMILES. 2. Algorithm for generation of unique SMILES notation *Journal of Chemical Information and Modeling* **1989**, *29*, 97–101.

[26] Rogers, D.; Hahn, M. Extended-Connectivity Fingerprints *Journal of Chemical Information and Modeling* **2010**, *50*, 742–754.

[27] Morgan, H. L. The Generation of a Unique Machine Description for Chemical Structures-A Technique Developed at Chemical Abstracts Service *Journal of Chemical Documentation* **1965**, *5*, 107–113.

[28] Wawer, M. J. et al. Toward performance-diverse small-molecule libraries for cell-based phenotypic screening using multiplexed high-dimensional profiling *Proceedings of the National Academy of Sciences* **2014**, *111*, 10911–10916.

[29] Kamentsky, L.; Jones, T. R. et al. Improved structure, function and compatibility for CellProfiler: modular high-throughput image analysis software *Bioinformatics* **2011**, *27*, 1179–1180.

[30] Moffat, J. et al. A Lentiviral RNAi Library for Human and Mouse Genes Applied to an Arrayed Viral High-Content Screen *Cell* **2006**, *124*, 1283–1298.

[31] Institute, B. CellProfiler example images and pipelines. https://cellprofiler.org/examples, **2020**; https://cellprofiler.org/examples, *Last accessed: 26.02.2021*.

[32] McQuin, C.; Goodman, A. et al. CellProfiler 3.0: Next-generation image processing for biology *PLOS Biology* **2018**, *16*, e2005970.

[33] Kim, S.; Thiessen, P. A. et al. PubChem Substance and Compound database *Nucleic Acids Research* **2015**, *44*, D1202–D1213.

[34] Wang, Y.; Bolton, E. et al. An overview of the PubChem BioAssay resource *Nucleic Acids Research* **2009**, *38*, D255–D266.

[35] Wang, Y.; Xiao, J. et al. PubChems BioAssay Database *Nucleic Acids Research* **2011**, *40*, D400–D412.

[36] Kolokoltsov, A. A.; Deniger, D. et al. Small Interfering RNA Profiling Reveals Key Role of Clathrin-Mediated Endocytosis and Early Endosome Formation for Infection by Respiratory Syncytial Virus *Journal of Virology* **2007**, *81*, 7786–7800.

[37] Hofmann-Winkler, H.; Kaup, F. et al. Host Cell Factors in Filovirus Entry: Novel Players, New Insights *Viruses* **2012**, *4*, 3336–3362.

[38] Kolokoltsov, A. A.; Saeed, M. F. et al. Identification of novel cellular targets for therapeutic intervention against Ebola virus infection by siRNA screening *Drug Development Research* **2009**, *70*, 255–265.

[39] National Center for Biotechnology Information (2021), A. PubChem Bioassay Record for AID 720532. Source: National Center for Advancing Translational Sciences (NCATS), 2021; https://pubchem.ncbi.nlm.nih.gov/bioassay/720532.

[40] Forsyth, D. *Applied Machine Learning*; Springer-Verlag GmbH, 2019.

[41] Raschka, S. Model Evaluation, Model Selection, and Algorithm Selection in Machine Learning *CoRR* **2018**, *abs/1811.12808*.

[42] Kohavi, R. A Study of Cross-Validation and Bootstrap for Accuracy Estimation and Model Selection. Proceedings of the 14th International Joint Conference on Artificial Intelligence - Volume 2. San Francisco, CA, USA, 1995; p 1137–1143.

[43] Fawcett, T. An introduction to ROC analysis *Pattern Recognition Letters* **2006**, *27*, 861–874.

[44] Kelleher, J. *Fundamentals of machine learning for predictive data analytics : algorithms, worked examples, and case studies*; The MIT Press: Cambridge, Massachusetts, 2015.

[45] Boughorbel, S.; Jarray, F.; El-Anbari, M. Optimal classifier for imbalanced data using Matthews Correlation Coefficient metric *PLOS ONE* **2017**, *12*, e0177678.

[46] Jolliffe, I. T. *Principal Component Analysis*; Springer-Verlag GmbH, 2002.

[47] Cha Zhang, Y. M., Ed. *Ensemble Machine Learning*; Springer-Verlag GmbH, 2012.

[48] Peng, H.; Long, F.; Ding, C. Feature selection based on mutual information criteria of max-dependency, max-relevance, and min-redundancy *IEEE Transactions on Pattern Analysis and Machine Intelligence* **2005**, *27*, 1226–1238.

[49] Ashburner, M. et al. Gene Ontology: tool for the unification of biology *Nature Genetics* **2000**, *25*, 25–29.

[50] Carbon, S.; Douglass, E. et al. The Gene Ontology resource: enriching a GOld mine *Nucleic Acids Research* **2020**, *49*, D325–D334.

[51] Landrum, G. *RDKit Documentation Release 2019.09.1*; Creative Commons: o Creative Commons, 543 Howard Street, 5thFloor, San Francisco, California, 94105, USA, 2019.

[52] of Medicine, N. L. PubChem. https://pubchem.ncbi.nlm.nih.gov/, Last Access: 04.03.2021.

[53] PubChem Database. ftp.ncbi.nlm.nih.gov/pubchem/, Last Access: 05.03.2021.

[54] Bergstra, J.; Bengio, Y. Random Search for Hyper-Parameter Optimization *J. Mach. Learn. Res.* **2012**, *13*, 281–305.

[55] Scikit-Learn, Feature importances with forests of trees. https://scikit-learn.org/stable/auto_examples/ensemble/plot_forest_importances.html, Last Access: 04.03.2021.

8 Appendix

Table 8.1: Lists of GO terms more abundant in each assay group.

GO-Terms More Abundant in Low Performing Assays	
'cellular_response_to_stimulus'	'anterograde_trans-synaptic_signaling'
'organic_cyclic_compound_binding'	'nervous_system_process'
'heterocyclic_compound_binding'	'cation_transmembrane_transport'
'DNA_binding'	'cellular_response_to_nitrogen_compound'
'cellular_protein_modification_process'	'cellular_response_to_oxygen-containing_compound'
'transferase_activity'	'trans-synaptic_signaling'
'macromolecule_modification'	'connected_anatomical_structure'
'catalytic_activity'	'ion_transmembrane_transport'
'organonitrogen_compound_metabolic_process'	'regulation_of_transcription_by_RNA_polymerase_II'
'protein_metabolic_process'	'postsynaptic_neurotransmitter_receptor_activity'
'signal_transduction'	'response_to_nitrogen_compound'
'catalytic_activity_acting_on_a_protein'	'inorganic_cation_transmembrane_transport'
'cell_communication'	'inorganic_ion_transmembrane_transport'
'DNA-binding_transcription_factor_activity'	'cell_surface_receptor_signaling_pathway_involved_in_cell-cell_signaling'
'ion_binding'	'system_process'
'cellular_protein_metabolic_process'	'postsynaptic_membrane'
'protein_phosphopantetheinylation'	'passive_transmembrane_transporter_activity'
'nucleic_acid_binding'	'synaptic_membrane'
'transcription_regulator_activity'	'response_to_organonitrogen_compound'
'signaling'	'chromatin'
'protein_modification_process'	'response_to_dopamine'
'cellular_component'	'organic_substance_transport'
'G_protein-coupled_receptor_signaling_pathway'	'catalytic_activity_acting_on_DNA'
'G_protein-coupled_receptor_activity'	'response_to_catecholamine'
'molecular_transducer_activity'	'cellular_response_to_catecholamine_stimulus'
'signaling_receptor_activity'	'non-membrane-bounded_organelle'
'establishment_of_localization'	'metalloendopeptidase_activity'
'transmembrane_signaling_receptor_activity'	'response_to_organic_cyclic_compound'
'hydrolase_activity'	'cysteine-type_peptidase_activity'
'ion_transport'	'adenylate_cyclase-modulating_G_protein-coupled_receptor_signaling_pathway'
'peptidase_activity'	'metallopeptidase_activity'
'metal_ion_binding 'cation_binding'	'DNA_repair'
'proteolysis'	'potassium_channel_activity'
'transition_metal_ion_binding'	'cellular_response_to_monoamine_stimulus'
'intracellular_anatomical_structure'	'DNA-binding_transcription_factor_activity_RNA_polymerase_II-specific'
'zinc_ion_binding'	'dopamine_neurotransmitter_receptor_activity'
'transporter_activity'	'response_to_monoamine'
'cell_periphery'	'cellular_response_to_dopamine'
'response_to_oxygen-containing_compound'	'gated_channel_activity'
'chemical_synaptic_transmission'	'chromosome'
'transmembrane_transporter_activity'	'voltage-gated_channel_activity'
'multicellular_organismal_process'	'adenylate_cyclase-activating_G_protein-coupled_receptor_signaling_pathway'
'cation_transport'	'organelle'
'inorganic_cation_transmembrane_transporter_activity'	'sequence-specific_DNA_binding'
'plasma_membrane_region'	'cellular_response_to_DNA_damage_stimulus'
'oxidoreductase_activity'	'cellular_response_to_organic_cyclic_compound'
'ion_channel_activity'	'protein_dimerization_activity'
'ion_transmembrane_transporter_activity'	'intracellular_organelle'
'plasma_membrane'	'potassium_ion_transmembrane_transport'
'synaptic_signaling'	'voltage-gated_potassium_channel_activity'
'inorganic_molecular_entity_transmembrane_transporter_activity'	'macromolecule_localization'
'cell_junction'	'synaptic_transmission_dopaminergic'
'channel_activity'	'cellular_response_to_organonitrogen_compound'
'neurotransmitter_receptor_activity'	'voltage-gated_ion_channel_activity'
'cation_transmembrane_transporter_activity'	'potassium_ion_transmembrane_transporter_activity'
'membrane'	'potassium_ion_transport'
'metal_ion_transport'	'dopamine_receptor_signaling_pathway'
'cation_channel_activity'	'regulation_of_molecular_function'
'metal_ion_transmembrane_transporter_activity'	'endopeptidase_activity'
'DNA_metabolic_process'	'intracellular_non-membrane-bounded_organelle'
'synapse'	'voltage-gated_cation_channel_activity'
'transmembrane_transport'	'anion_transport'
'transcription_by_RNA_polymerase_II'	'regulation_of_biological_quality'
'cell-cell_signaling'	'G_protein-coupled_amine_receptor_activity'
'chemical_synaptic_transmission_postsynaptic'	
'postsynapse'	

GO-Terms More Abundant in High Performing Assays	
'cellular_process'	'cellular_response_to_chemical_stimulus'
'cellular_macromolecule_metabolic_process'	'negative_regulation_of_DNA_replication'
'regulation_of_cellular_process'	'regulation_of_cell_cycle'
'regulation_of_cellular_biosynthetic_process'	'negative_regulation_of_macromolecule_metabolic_process'
'regulation_of_cellular_macromolecule_biosynthetic_process'	'regulation_of_nucleobase-containing_compound_metabolic_process'
'cellular_macromolecule_biosynthetic_process'	'protein_binding'
'biosynthetic_process'	'aromatic_compound_biosynthetic_process'
'regulation_of_cellular_metabolic_process'	'cellular_nitrogen_compound_metabolic_process'
'macromolecule_biosynthetic_process'	'regulation_of_transcription_DNA-templated'
'organic_substance_biosynthetic_process'	'DNA_replication'
'regulation_of_biosynthetic_process'	'cellular_aromatic_compound_metabolic_process'
'regulation_of_macromolecule_biosynthetic_process'	'cellular_nitrogen_compound_biosynthetic_process'
'cellular_biosynthetic_process'	'negative_regulation_of_cellular_biosynthetic_process'
'regulation_of_macromolecule_metabolic_process'	'regulation_of_nitrogen_compound_metabolic_process'
'nucleobase-containing_compound_biosynthetic_process'	'negative_regulation_of_macromolecule_biosynthetic_process'
'organic_cyclic_compound_metabolic_process'	'nucleobase-containing_compound_metabolic_process'
'negative_regulation_of_cellular_process'	'RNA_metabolic_process'
'regulation_of_primary_metabolic_process'	'heterocycle_metabolic_process'
'RNA_biosynthetic_process'	'nucleic_acid-templated_transcription'
'regulation_of_nucleic_acid-templated_transcription'	'negative_regulation_of_cell_cycle'
'negative_regulation_of_cellular_metabolic_process'	'organic_cyclic_compound_biosynthetic_process'
'regulation_of_DNA_replication'	'heterocycle_biosynthetic_process'
'negative_regulation_of_cellular_macromolecule_biosynthetic_process'	'regulation_of_RNA_biosynthetic_process'
'negative_regulation_of_metabolic_process'	'transcription_DNA-templated'
'negative_regulation_of_biological_process'	'regulation_of_RNA_metabolic_process'
'gene_expression'	'negative_regulation_of_biosynthetic_process'
'cell_cycle'	'regulation_of_gene_expression'
'nucleic_acid_metabolic_process'	

List of Figures

List of Tables

Abbreviations

AF all features

AID assay identifier

AUC-ROC area under the ROC curve

BA balanced accuracy

CID compound identifier

CP cell-painting

CV cross-validation

DMSO dimethyl sulfoxide

DNA deoxyribonucleic acid

ECFP extended-connectivity fingerprint

ER endoplasmatic reticulum

FN false negative

FP false positive

FPR false positive rate

GCR glucocorticoid receptor

GI gini impurity

GO gene ontology

hpa high performing assays

HTS high-throughput-screening

InChI-key international chemical identifier key

IUPAC International Union of Pure and Applied Chemistry

KFCV k-fold cross validation

lpa low performing assays

MCC Matthews correlation coefficient

ML machine learning

MLP Molecular Libraries Program

MLSMR Molecular Libraries Small Molecule Repository

MoA mechanism of action

MRMR minimal-redundancy-maximal-relevance criterion

PCA principal component analysis

RFC random forest classifier

RNA ribonucleic acid

ROC-curve receiver operating characteristic curve

SF selected features

SMILES simplified molecular input line entry specification

SMOTE synthetic minority oversampling technique

TN true negative

TNR true negative rate

TP true positive

TPR true positive rate

VSV vesicular stomatitis virus

WGA wheat germ agglutinin