Universiteit Leiden

Master Computer Science

A statistical approach to classification of mechanistic computational models of Parkinson's Disease

Name: Lalithasushma Chakravadhanula
Student ID: s2028891
Date: 14/08/2019

Specialisation: Computer Science: Computer Science and Advanced Data Analytics

1st supervisor: Dr. Michael Emmerich
2nd supervisor: Professor Fons Verbeek

Master Thesis in Computer Science

Leiden Institute of Advanced Computer Science
Leiden University
Niels Bohrweg 1
2333 CA Leiden
The Netherlands

1 Abstract

Parkinson's Disease (PD) is a progressive neurodegenerative disorder. It is the second most common disease that affects the central nervous system. PD manifests with both motor symptoms such as tremors, rigidity and non-motor symptoms like depression and fatigue. Although the pathogenesis of PD remains a puzzle, several genetic and environmental factors are known to affect the progression of the disease. This research focuses on the discovery of metabolic changes due to the PINK1 gene mutation. PINK1 is a mitochondrially targeted, serine/threonine-protein kinase PTEN-induced kinase 1 (PINK1) that protects cells from stress-induced mitochondrial dysfunction. We compare the effects of PINK1 gene mutation by inhibiting complex 1 and complex 5 on the inner membrane of mitochondria, with healthy controls. Generic human genome-scale metabolic models are used to create metabolic models of a context-specific dopaminergic neuron. The steady-state solution spaces of these constraint-based models of dopaminergic neurons are sampled for flux distributions. The objective of this work is three folds, firstly, the sampled flux distributions are checked for uniformity. Secondly, an algorithm is proposed to classify diverse (uniform, normal, truncated, etc.) flux distributions, to reduce the challenge of statistical analysis. Thirdly, individual reactions are compared to highlight dissimilarities in their distributions. The approach proposed in this paper allows us to study the changes in metabolic rates between the inhibited, and healthy control models by categorizing corresponding flux distributions. Thereby, leading to a better understanding of the pathogenesis of PINK1-PD.

2 Introduction

Parkinson's Disease is the second most common neurodegenerative disorder, affecting about 4-10 million people every year[24, 17]. It is a progressive disease, with increasing severity of symptoms with the aging of the person. The symptoms include a group of motor deficits, like bradykinesia (slowness in movement)[7], postural instability[10], rigidity and tremors, which could lead to immobility. Some other symptoms are non-motor such as loss of smell, depression, and fatigue[34, 17]. Although the effect of neurodegeneration is evident in different parts of the brain, the primary problem lies in the loss of dopaminergic neurons in the substantia nigra pars compacta (SNpc)[17, 39], causing problems in movements. However, the mechanism that leads to the death of

dopaminergic neurons, and therefore causes the progression of PD, is still not completely understood. Multiple hypotheses were formed for unraveling the puzzle of this progressive disorder, such as proteostasis, oxidative stress, mitochondrial dysfunction, neuroinflammation[39]. It is predicted that the number of patients suffering from PD would increase to 9 million by the year 2030, in Europe alone. Studying the progression of the disease through a systems approach could lead to a better (preventative) treatment or cure, giving patients a better quality of life.

Several medications have been suggested for the treatment of Parkinson's Disease[55]. The first and most commonly used medication is levodopa. Discovered in 1996, levodopa is administered as a tablet or liquid. The medication is absorbed by the nerve cells and turned into chemical dopamine, and used for transmitting messages between parts of the brain and nerves that control movement, thereby improving levels of dopamine and reducing movement problems. However, in some cases, levodopa is prescribed along with other medications such as benserazide or carbidopa, in patients with side effects such as nausea and vomiting[33, 34]. Other treatments include dopamine agonists, monoamine oxidase-B inhibitors, and catechol-O-methyltransferase inhibitors. Dopamine agonists are induced to act as substitutes for dopamine in the brain, having a similar effect as levodopa, but milder. However, the positive effects of agonists may diminish over time[19]. Monoamine oxidase-B inhibitors, including rasagiline and selegiline, are alternative treatments in the early stages of the disease. These act by blocking the break down of dopamine, thereby increasing dopamine levels[45], while catechol-O-methyltransferase inhibitors are used to prevent levodopa in the later stages of PD[61]. The available options for treatment are shown in Figure 1. Although the medication currently used seems to reduce the effects of the symptoms caused by PD, the root cause of the disease seems to remain a puzzle. Hence, there is no proven cure. Observing the progression in detail could lead to a preventive cure.

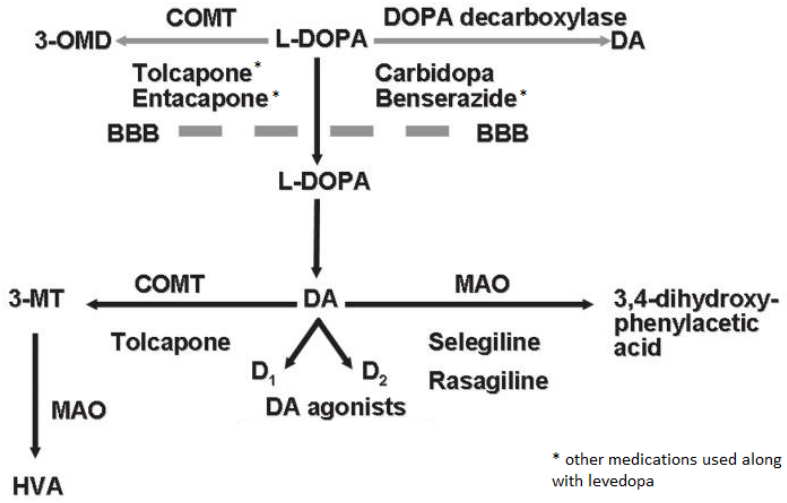

Figure 1. Pharmacologic treatment options available for PD. Abbrivations: BBB, blood-brain barrier; COMT, catechol-O-methyl-transferase; DA, dopamine; L-DOPA, 3,4 dihydroxy-L-phenylamine; HVA, homovanillic acid; 3-MT, 3-methoxytramine; MAO, monoamine oxidase, inspired by [33].

Although the exact cause of the disease is not known, both genetic and environmental factors have been identified to play significant roles in the pathology of the disease. Early observations revealed relations between mitochondrial dysfunction with the progression of PD[37]. Mutations in the genes encoding mitochondria-related proteins, seem to have a considerable effect on the early on-set of PD[58]. Our research focuses on one such gene mutation[1]. PINK1 is a mitochondrially targeted, serine/threonine-protein kinase PTEN-induced kinase 1 (PINK1) that protects cells from stress-induced mitochondrial dysfunction. This protein is localized in mitochondria [28, 18]. It is linked to quality control of mitochondria[47], protecting it from damage. PINK1 is continuously processed and degraded by proteases in healthy mitochondria. When mitochondria are damaged, proteolysis is stopped, causing PINK1's accumulation on the outer membrane of mitochondria. Parkin is then recruited from the cytosol to the depolarized, damaged mitochondria and mediates the process of mitophagy. PINK1 acts as an enzyme to relocate Parkin and carry out mitophagy, thus becoming a dimer in the removal of damaged mitochondria and quality control [44] (Figure 2). When PINK1 protein is mutated, it causes to fail the process of recruitment of Parkin. Thereby, inhibiting the process of mitophagy. Thereby, accumulating damaged mitochondria in the cell, ultimately leading to cell death. Hence, it is interesting to observe biochemical changes of PINK1 gene mutations

and comparing these changes to healthy controls.

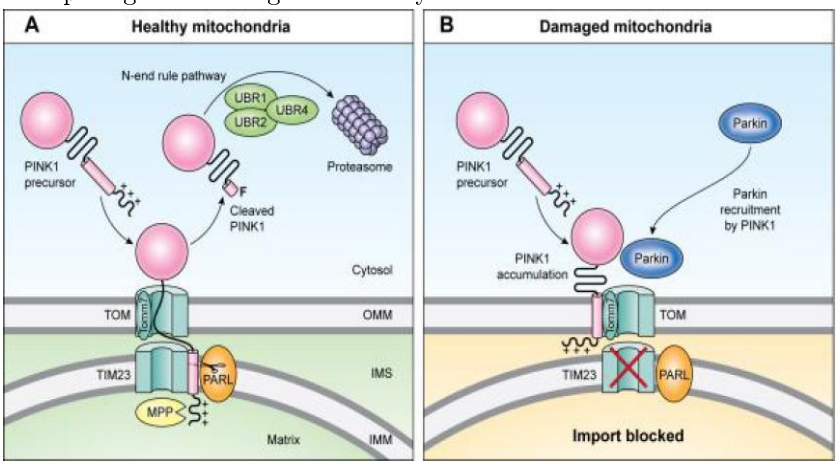

Figure 2. Depolarization of mitochondria or blocking mitochondrial import causes PINK1 to accumulate on the outer mitochondrial membrane[48].

PINK1 gene mutation also impairs the respiratory chain and ATP production of a cell's mitochondria[66]. In the respiratory chain, several enzyme-complexes take part to provide cells with the required energy[49]. Complex 1 plays the role of the initial enzyme used to catalyze the process of electron transfer to a liquid soluble electron carrier (ubiquitin). Thereby initiating the multi-step process of producing ATP to the cell[20]. While complex 5 is the last step for producing ATP to the cell. Previous research has shown evidence that exposure to high amounts of rotenone or oligomycin, would mutate complex 1 and complex 5 in mitochondria affecting the process of oxidative phosphorylation leading to mitochondrial dysfunction. As discussed before damaged mitochondria lead to Parkinson's disease[37]. Therefore leaps of differences seen when either complex 1 or complex 5 are inhibited in healthy mitochondria. Hence, in this research, two pathways are inhibited in a healthy dopaminergic neuronal model to observe the changes made by their mutations.

To comprehend how the dysfunctional pathway interacts with the rest of the network to result in neurodegeneration, it requires an interdisciplinary system's approach[68]. An iterative cycle of mathematical model formulation, computational modeling, and quantitative experimental measurements is the process of the systems approach. Mathematical model formulation and computational modeling are formal representations of biochemical knowledge used for proposing a hypothesis, design experiments, and interpret results. Conversely, quantitative experimental measures are used to test the hypotheses, and to generate

data used to build the models. The ultimate aim of applying a systems approach to PD is to formulate the non-trivial hypothesis for discovering metabolic changes in both in-silico and in-vitro, leading to specific biomarkers[39]. This work is part of a SysMedPD consortium, an H2020 project funded by the EU, to implement the systems approach to Parkinson's Disease. [1].

This research seeks to compare metabolic changes between complex 1 and complex 5 inhibitions, in healthy dopaminergic neuronal cells derived from human neuroepithelial stem cells through system's approach. A generic genome-scale human metabolic model is used to create candidate-specific dopaminergic neuron models. The steady-state solution spaces of these constrained-models are uniformly sampled to obtain flux distributions in the reactions involved. These sampled points are checked for uniformity. Through this research, a classification algorithm is presented, which categorizes diverse flux distributions (uniform, normal, truncated, etc.) to observe dissimilarities. These dissimilarities of flux distributions are used to analyze metabolic phenomena such as phase shifts, and log fold changes between reactions of different models.

In Chapters 2 to 4, the computational approaches taken to identify dissimilarities in flux distributions of reactions in the models are discussed.

Chapter 2 describes the process of building a candidate-specific genome-scale dopaminergic neuron model using constraint-based modeling. This chapter aims to explain genome-scale metabolic models and their importance in predicting biomarkers for diseases. It also discusses various generic genome-scale metabolic models (Recon3, HMR 2.0), and an approach of making them candidate-specific. Diving deeper, the method used for building these dopaminergic neuronal models is described with examples from the current research. An introduction to unbiased and biased sampling approaches is given, while defining the method chosen. The concept of the popular sampling technique of CHRR sampler is discussed. Further, the uniformity of sampled points for flux distributions is studied using the Gap-ratio algorithm. Variance and standard deviation measures are studied for further insights on the spread of population sampled. Different sets of points are generated and are checked for uniformity.

Sampled points from the constrained-based solution spaces have various types of distributions. The reasons for these distributions are analyzed. To compare these distributions, and observe changes quantitatively, a classification algorithm is proposed in Chapter 3. This novel algorithm aims to categorize the observed distributions. Each part of the algorithm is separately discussed. The derivation of the classification algorithm is explained for the robustness and

results specific to the research.

Each reaction is treated as a different dimension, and after running the distributions through the classification algorithm, the reaction is compared between complex 1 inhibited models, complex 5 inhibited models and healthy control models. These reactions are compared using fold changes between the models in Chapter 4. The results are quantified for the most changed reactions for complex 1 and complex 5 inhibited models when compared to healthy controls.

In Chapter 5, overall conclusions and comparison of *in-silico* insights with *in vitro* are stated. Chapter 6 describes future works. Chapter 7 includes an appendix.

Chapter 1 : Constraint-based modelling and flux based sampling

The integration of biochemical knowledge with physiolgy, enzyme kinetics, stoichiometry of enzyme-reaction relationships lead to mathematical reconstrunction of metabolic networks. Genome-scale metabolic networks are integral part of biotechnology, allowing an understanding of the phenotypic behaviour of all living organisms, including humans[64]. Genome-scale metabolic networks are comprehensive representations of all chemical reactions while containing stoichiometric representations of all reactions of any living organism.

In this chapter, candidate-specific dopaminergic neurons are built using genome-scale metabolic networks. These networks are constrained using information from human neuroepithelial stem cell-derived dopaminergic neurons *in vitro* cell cultures. Multiple steps are involved in building a constrained-based genome-scale model for candidate specific dopaminergic neurons, to observe metabolic changes[5]. The constrained model's steady-state solution spaces are sampled for flux distributions of corresponding reactions.

Components, such as metabolites and enzymes, play the most important role in biological proceses. They vary in time and are constrained by thermodynamical laws. Components are the building blocks of reactions, *links,* that form the metabolic network. Enzymes, encoded by genes, enable thermodynamically infavourable metabolite conversions required to sustain metabolic functions, *functional states*, of the organisms. Given enough time, the network reaches homeostasis, steady-state, where components no longer vary in time, but the constant flux through the reactions is observed (Figure 3). Geome-scale models are based on the steady-state assumption, and consist of all metabolites and reactions observed to be present in a given organims.

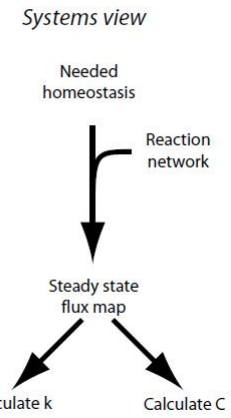

Figure.3. System view of the reactions. The concentrations of the network components(C), and kinetic properties of links in the network (K).[46]

As the number of functional states increases the solution space also increases exponentially, and the maintenance of a single solution for each reactions' flux becomes infeasible. Therefore, constraining the conditions under which a cell operates and evolves against are easier to state, use, and identify. Constraint-based approaches for analyzing complex biological networks are proven to be very useful[32] (Figure 4.a). Cells are subjected to hard constraints based on the components and their associated mass and energy balances[52], giving an allowable range of states for the network[23]. The states that are suitable for the network are kept, while unwanted states are removed by implementing regulatory networks (Figure 4.b). The allowability of changing states and phenotypic behaviour is regulated by the gene and its bi-products at that point in time.

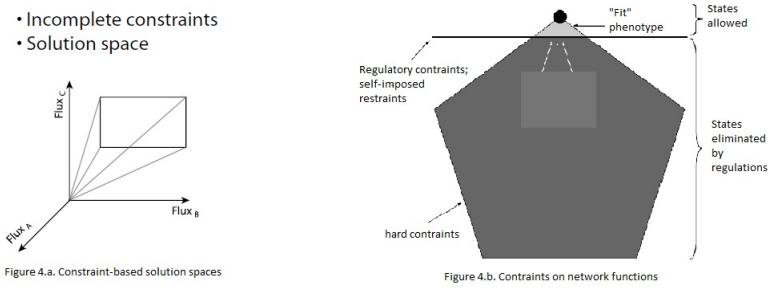

Figure 4. Illustration of constraints on solution spaces and network func-

tions. (left) allowable range of solutions, (right) viable contraints on network functions for different states and phenotypes, inspired by [46].

3 iNESC2DN model

3.1 Reconstruction:

In this research, Generic human metabolic models are used to generate a steady-state dopaminergic neuronal model by using manual curations[51]. These models were built in MATLAB using a mathematical framework called *COnstraint-Based Reconstruction and Analysis (COBRA) toolbox*[31]. COBRA toolbox provides a mechanistic computational framework for intergation and analysis of experimental, quantitative prediction of biochemical and physiochemically feasible phenotypic states. COBRA toolbox is an open-source library framework which intergrates biochemical information from various databases to enable mathematical representations of genome-scale human metabolic networks (Figure 6), and allows uniform sampling of high dimensional sampling spaces. It takes a mathematical approach in representing relationship between genotype and phenotype by modelling constraints. It also highlights gaps in reconstructions for specific genomes to get a clear picture of the models.

Two of the most widely used generic human metabolic model reconstructions are HMR 2.0[50] and Recon3D[31]. HMR 2.0 is a intergration of multi-omics data widely used for cell-specific data analysis. It contains reactions, related compounds and annotation informations. Initially it was built using the Edinburgh human metabolic network [38], Recon2 [64] and HepatoNet[35] as well as external reaction databases: KEGG[35], HumanCyc[57], BRENDA (18), HMDB[69], ChEBI[35], LMSD[62] and PubChem[11], and annotation data was combined based on Ensembl[26] and UniProt [50].

Recon3D, is the most comprehensive generic human metabolic model reconstruction. It is used to re-build context-specific models by gaining knowledge from a combination of manual model curations and integration of different omics data. It is an updated, expanded metabolic reconstruction that integrates pharmacogenomic associations, large-scale phenotypic data, and structural information for both proteins and metabolites. It contains over 6000 more reactions from the previous models, manually curated for redundency[14].

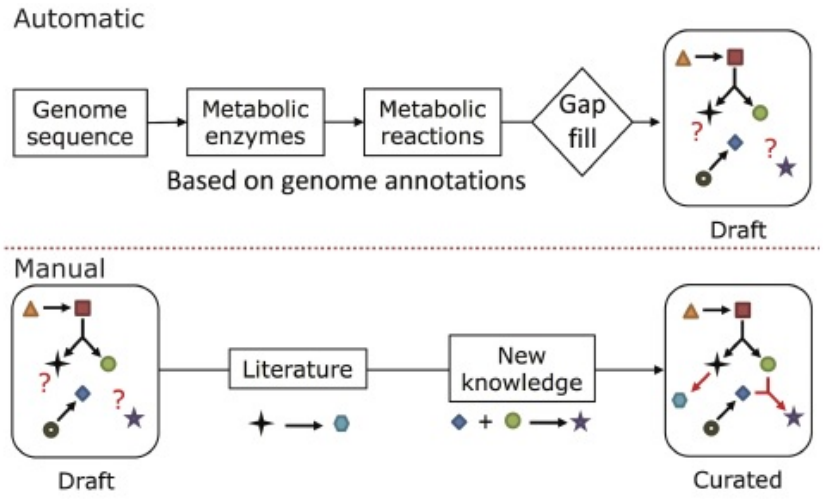

Figure 5. Genome-scale metabolic reconstruction

Figure 5. Genome-scale metabolic reconstruction in Recon3D[31]

3.2 Model Generation:

To build context-specific dopaminergic neuronal models that are stoichiometrically and flux consistant, manual curations of NESC-derived dopaminergic neurons *in vitro* are used[51]. Constraints based on the transcriptomics data and manual curation were integrated with the Recon3D using COBRA toolbox, and the model was created using FASTCORE algorithm. Figure 6, shows the workflow of generation of dopaminergic neuron for analysis developed by Preciat et al [51]. Models were refined by comparing biochemical literature with the results of Flux Balance Analysis.

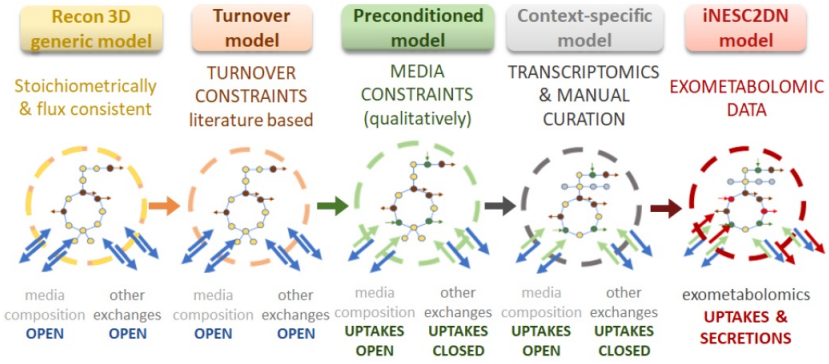

Figure 6. Overview of the model generation pipeline[51]

Generic human metabolic model, Recon3D:

Generic human metabolic recontruction, Recon2[60] was updated using manual curations for literature specifics of dopaminergic neurons, and included in the Recon3D[31], highlighting the enzyme-gene correspondence. More manual curations were performed to define active and inactive reactions, transport reactions, degradation pathways, and quantitative constraints necessary to represent requirements for molecular turnover in a non-growing dopaminergic neuron. The biomass requirements for turnover constraints were ensured by initally taking all the needed biomass, and slowly understanding the progression of degradation by precurser dysfunction. The rates of exchange reactions and reversible extracellular transport reactions, including water, CO2, and oxygen were constrained by defined fresh cell culture medium.

Generation of the turnover model:

On the generic model, constraints of turnover rates of key constituents of dopaminergic neurons were imposed to create a turnover model. These constraints are derived by literature. A 25% relaxation of the lower bounds from the estimated degradation rate was used as standard to account for uncertainty in the data[63]. If more than one reaction could be degraded, the total sum of degradation was set as greater than 0.75 times the degeradation rate d.

$$v_1 + v_2 + ... + v_n \geq 0.75 * d \tag{1}$$

Generation of the preconditioned model:

The preconditioned model was generated by applying the maximum uptake constraints on the, otherwise reversible, exchange reactions. The uptake of metabolites was measured in *in vitro* cell cultures of dopaminergic neurons. If metabolite was uptaken *in vitro* the corresponding exchange reaction's lower boundary was set to the measured value, otherwise the lower boundary was set to zero.

Generation of context-specific model:

This model is an integration of omics data with preconditioned model to achieve a flux-consistent network. A metabolic network formed from the set of core reactions alone is not necessarily flux consistent. Therefore, FASTCORE algorithm along with COBRA toolbox was used to integrate different omics data to generate a network, not only consisting of core reactions, but support reactions aswell to ensure flux-consistency of the models.

4 Derivation of reactions

There are several ways to study properties of genome-scale networks once the solution space is formed. The conical solution space can be studied by uniform random sampling to generate representative solutions of correspondoing flux distributions. Properties of the representative solutions can be studied to understand the behaviour of the original solution space. This method is known as the unbiased assessment. On the contrary, biased assessment involves studies of network states of interest with an objective function. Computationally, this approach is based on linear optimization, where solutions can be obtained by the popular procedure of *Flux-Balance Analysis (FBA).* Biased methods allow to obtain a single solution for the objective function, however they do not provide an information about the flux distributions within the solution space of the rest of the network[46].

To study changes between solution spaces of different models, unbiased approach was used. The iNESC2DN model was used to study the effect of inhibition of two key reactions in energy metabolism: mitochondrial complex I, and mitochondrial complex V. The solution spaces of the iNESC2DN model affected by each of these inhibitions were uniform randomly sampled. Total number of active reactions for these models are 1789, where the constrained-solution space has 439 dimensions, and 184 exchange reactions.

Constraint-based models can be formulated as optimization problems to derive and predict the nature of reactions. This optimization problem can be mathematically written as:

$$min_{\{v \in \mathbb{R}n\}} \ \psi(v) \tag{2}$$

$$s.t. Sv = 0, \text{ and } l \leq v \leq u, \tag{3}$$

where $S \epsilon \mathbb{R}^{\mathbb{M}\mathbb{X}\mathbb{N}}$is a stiochiometric matrix of *m metabolites and n reactions representing a network,* $v \epsilon \mathbb{R}^{\mathbb{M}}$is the vector representing the flux through reactions(Figure.7). The set of feasible steady-state flux vectors form a polyhedral convex solution space, defined by equality and inequality of constraints set on the Equation (2). Internal reactions are balanced by mass and charge. Exchange reactions are described as sink, demand and exchange reactions (Figure 7.a.) and are characterised by having only one non-zero element in the corresponding column of the stoichiometrix matrix. The lower and upper bounds of contraints for distributions are l and u respectively.

The metabolic network reconstruction can be represented as a stoichiometric matrix, S (Figure 7.b.), where rows correspond to each reaction (n), and columns corresponds to the metabolites (m) involved in those reactions. An element in the matrix, S_{nm}, represents the stoichiometric requirement of a metabolite (m) in that reaction (n). If $S_{nm} \geq 0$ the metabolite (m) is a product, $S_{nm} \leq 0$ it is a substrate of the reaction, and if $S_{nm} = 0$ then metabolite (m) is not involved in the reaction. The linearity in Equation (3) represents the mass balance for all the metabolites. The reaction achieves an equilibrium between metabolite consumption and production in a steady state model.

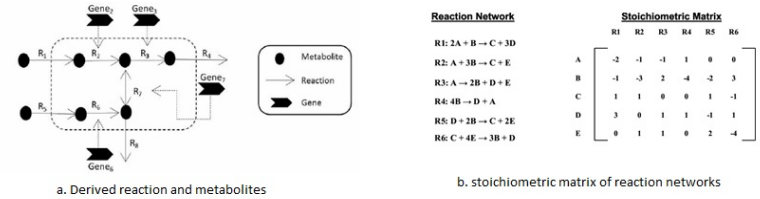

Figure 7. Derived reactions and stoichiometric matrix of reactions.7.a. shows derivation of reaction and metabolites[9],7.b.stoichiometric matrix of reaction networks.

Constraining the solution space generates a polytope with all the feasible function states. In the feasible function states, the projected dimensions are described to be hard constraints, eliminating dependent reactions,i.e., those that have similar functionality, and dimensions of the vector space are the independent reactions that define overall representations. This polytope consists of optimal solutions for each reaction in the network. Uniform random sampling of the solution space can be performed resulting in a representative dataset containing a feasible flux distribution for each reaction in the network. There are several algorithms available for uniform sampling of high-dimensional spaces, that follow the Markov chain Monte Carlo sampling methods.

5 Linear optimization

Monte Carlo methods rely on repeated random sampling to obtain numerical results. This method is often used to describe highly complex probability distributions that are difficult to handle with traditional analytical approaches. Given N independent and ideally distributed (i.i.d) samples $x_i \sim p(x|D)$ from the posterior distribution over a given dataset D, the method aims towards a

desired quantity of interest Φ by considering a sample mean such that,

$$\lambda = {}^{1}/_{N}\, N \sum i = 1\, \phi(x_i) \tag{4}$$

such that, $\lambda \approx E[\phi(x)|D]$,and measure for posterior probability, for an element of $(m, n) \in UxV$.

$$p(m|n) = \int_v p(m, v|n)dv \tag{5}$$

Two central properties justify this approach, the i.i.d. estimator λ is unbiased and the arithematic mean of observations converges to the expected value (law of strong large numbers)[56]. A stocastic process that satisfies Markov property is a Markov process. Markov property is the conditional probability of future states in the process depends only on the present state and not influenced by sequence of previous events. This can be mathematically represented as,

$$P(X_i = x_{ij} X_0 = x_0, ..., X_i{=}1 = x_i{=}1) = P(X_i = x_{ij} X_i{=}1 = x_i{=}1)$$

Transition between each state is modelled using a transition matrix and probabilities of transition must be equal to 1. If the process is confined to a state space, then it is called a Markov chain[13].

When each of the chain states can be a starting point for the markov process, the initial distribution is denoted by $\pi_0^T \epsilon \mathbb{R}^{\mathbb{N}}$where N represents the dimensionality of the n-dimensional state vector. The time step t can be modelled, and hence the output represents the probability distribution of over Markov chain in the t-th time step. This distribution presents the probability for each of the chain states to be the current state at time t. Such a distribution is called stationary if

$$\pi = \pi.P \tag{6}$$

A combination derivation of posterior probability from Markov chain and Monte carlo methods to obtain a stationary distribution π of a given markov chain that approximates a presumed posterior probability $p(x|D)$. The combined algorithm is a part of many popular sampling algorithms[41, 12]. Here, an implementation of the algorithm is discussed, for sampling high-dimenstional solution spaces generated by constraint-based models.

5.1 CHRR sampler

Hit-and-Run (HR) algorithm mitigates the problem of rejection sampling, where if a sample generated is out of the boundaries of constrained solution space, then that sample is rejected. It is not viable to accumulate hard boundaries on solutions spaces, and generate feasible samples. Hence reject sampling is not iteratively used. Hit-and-Run algorithm samples directly from solution space. It starts at a point $\overrightarrow{x}_0$in the contrained space, and it travels along an arbitrary path $\overrightarrow{u_1}$to a point chosen on the boundary of uniformly distributed sphere $\mathbb{R}^{\mathbb{N}}$. The distance travelled between the starting point and $\overrightarrow{u_1}$ determines the maximum distance. A small step $\lambda_1$is taken in the negative direction towards $\overrightarrow{u_1}$, such that no boundary contraints are exceeded. The next sampled point$\overrightarrow{x}_1$is reached after travelling the distance ofλ_1, and iteration through this process obtains Markov chain of consecutive samples[13].

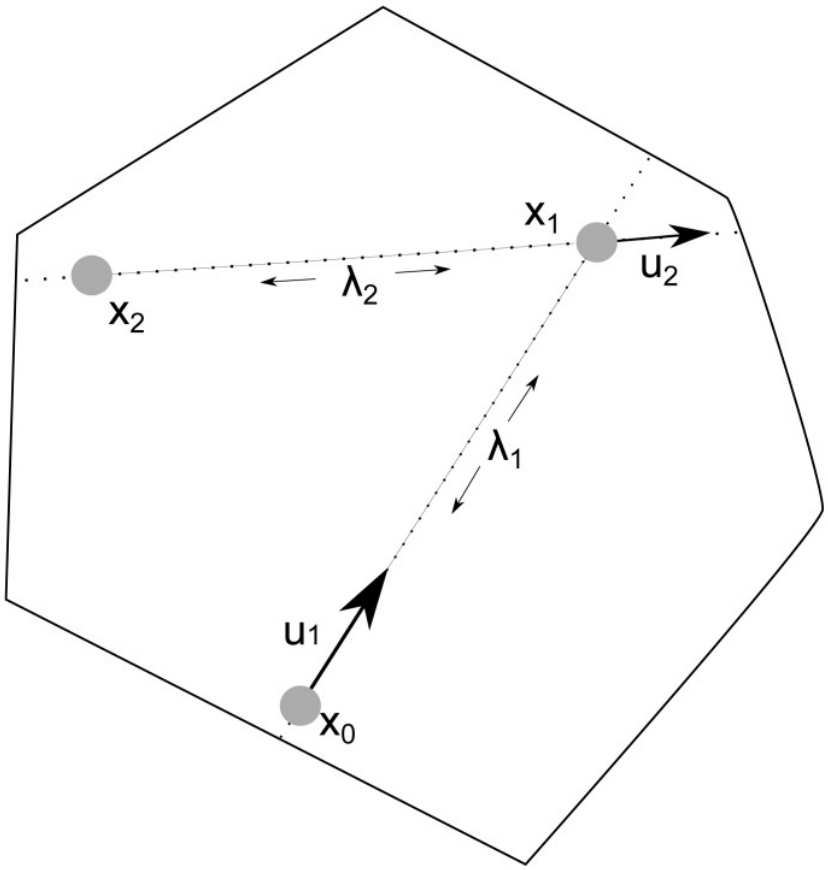

Figure 8. Hit-and-Run algorithm[41].

Coordinate Hit-and-Run algorithm (CHRR) is used to uniformly sample a constrained solution space using volume ellipsoid algorithm as rounding algorithm. This algorithm has two parts, rounding of the polytope $P = \{x \epsilon \mathbb{R}^{\mathbb{N}} | Ax \leq b\}$preprocessing, generating samples using coordinate HR[29].

5.1.1 Rounding

To ensure the efficient convergence, the polytope formed by solution space is rounded. This helps to achieve uniform random samples, with fewer steps. However, there are two types of roundness: well-roundedness and isotropy.

For a polytope $P = \{x \epsilon \mathbb{R}^{\mathbb{N}} | Ax \leq b\}$ is said to be R-rounded if $B_n \subseteq P \subseteq B_n.R$. So the body in observation is in the radius of R and the radius of 1. It is well-rounded if $R = O * (M - msn)$ and will converge in $O * (n^3)$.

Another approach for well-roundedness is to observe if polytope P has an isotropic position. A polytope is in isotropic position if its centre of mass is the origin and its covarience matrix is the its identity. If P is isotropic, then its radius is given by $O * (\sqrt{n})$.

Since both approaches mentioned above are computationally expensive, here maximum volume ellipsoid algorithm is used[72](Figure 9). Results show that the body is n-rounded[29].

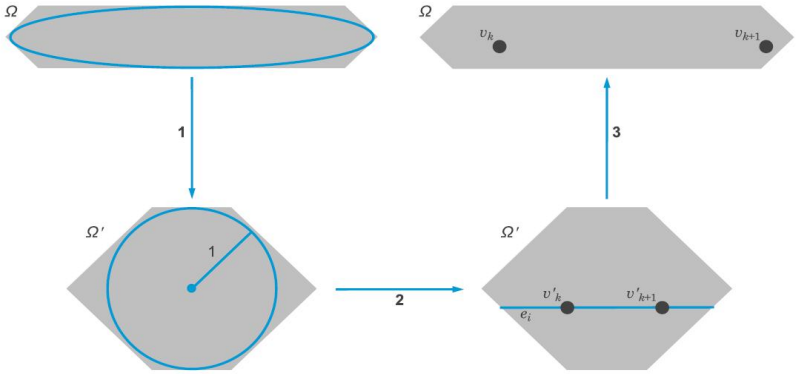

Figure 9. CHRR sampler

5.1.2 Coordinate hit-and-run

Coordinate Hit-and-Run follows the similar process that of Hit-and-Run. Instead of choosing a random direction from the current point $\overrightarrow{x}_0$ to $\overrightarrow{x}_1$ in $\overrightarrow{u_1}$, it selects a uniformly random direction from the current direction. It take $O * n^2$steps to converge with CHRR, while it takes $O * n^3$steps to converge with

Hit-and-Run algorithm.

6 Results

The set of sampled points for each reaction needs to be verified for uniform spread over the solution space(uniformity). Variance and standard deviation of the solution samples are also checked. It is also hypothesized, that larger the number of samples, more uniformity of the distribution[29]. To check these hypothesis, following algorithms are used:

1. Gap-ratio algorithm for checking the uniformity of distribution[8],
2. Chi-square test for checking standard deviation[4].

6.1 GA algorithm:

For testing the spread of points on the polytope, gap-ratio algorithm is used. The distance between each consecutive ordered point is measured, and divided by the difference and checked if the ratio is less than or equal to 2. Reactions that have maximum points with the difference equal to 2 or less, are labelled as uniform distributions. If the majority of points deviate from the difference, then the reactions are labelled as non-uniform distributions.

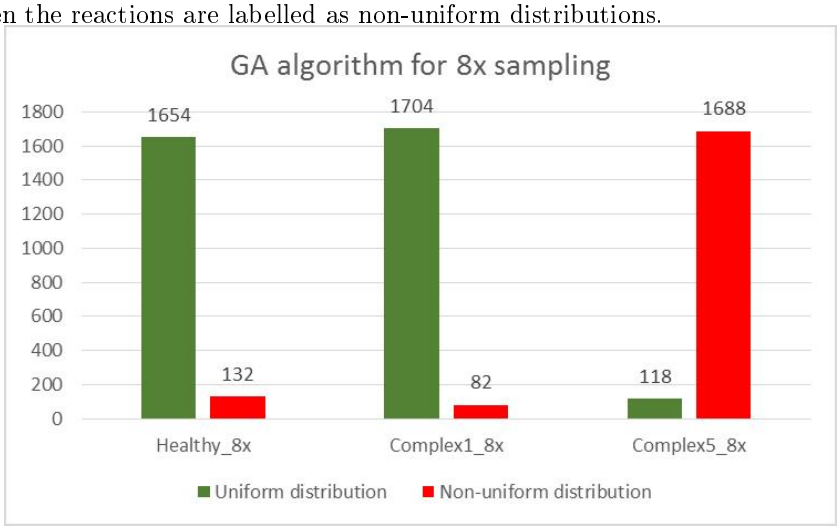

Figure 10. Uniformity of 8x sampling.

In Figure 10, sampled points of size 3512 are measured (8 times the number of independent reactions in the model). The total size of samples is 3512, and number of reactions is 1786. In healthy controls, 1654 are uniform flux distributions and 132 are non-uniform flux distributions. For complex 1 inhibited models, 1704 reactions have uniform distributions, while only 82 are non-uniform. Although for complex 5, a change is observed. There are 1688 non-uniform distributions, and 118 uniform distributions. This could have occured due to skipping of sampled points that travel out of the constrained solution space.

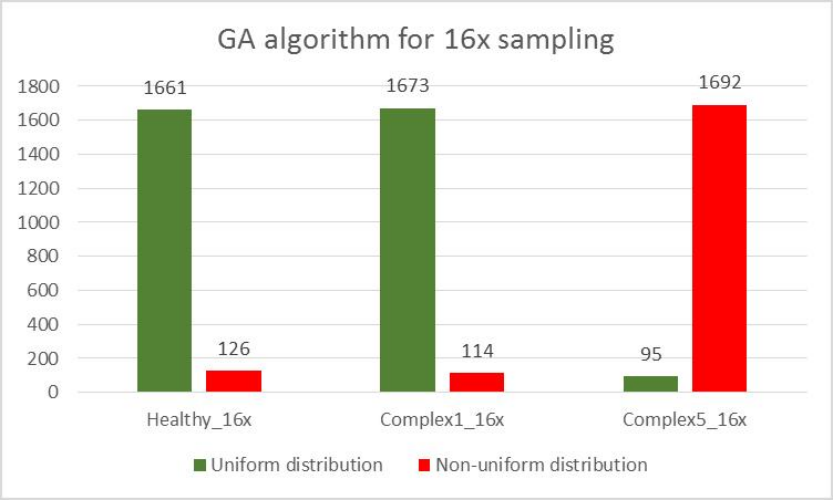

Figure 11. Uniformity of 16x sampling.

With larger number of points sampled for the distribution, higher the chances of covering the solution space, and larger the uniformity of points. In Figure 11, sampled points of size 16 times the number of independent reactions are measured(439). The total size of samples is 7024, and number of reactions is 1786. In healthy controls, 1661 are uniform flux distributions and 126 are non-uniform flux distributions. For complex 1 inhibited models, 1673 reactions have uniform distributions, while only 114 are non-uniform. For complex 5, similar change is observed as in Figure 9. There are 1692 non-uniform distributions, and 95 uniform distributions.

6.2 Chi-sqaured test:

For testing the varience of the population, two sided chi-squared test is used. The two sided version of tests against the varience and standard deviation of the samples normal distribution. The null hypothesis is tested for equal variences

for the population under question and the 5% standard deviation of normal distribution. If the varience is equal the null hypothesis is accepted, else, it is checked if the varience is greater or lesser than the ideal population, accepting alternative hypothesis.

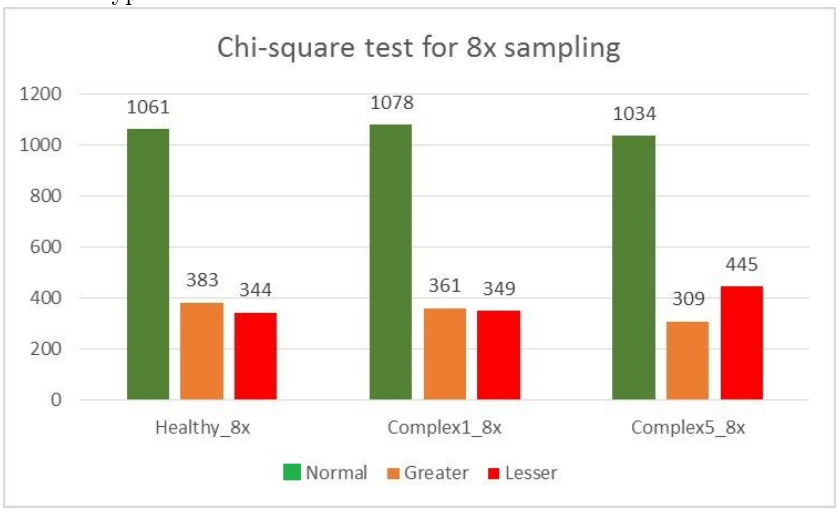

Figure 12. Chi-square test varience of 8x sampling.

In Figure 12, sampled points of 1786 reactions are tested for varience. In healthy controls, 1061 samples have same varience, 383 reactions have varience greater normal distribution, and 344 samples have lesser varience. Complex 1 inhibited models, have same varience for 1078, 361 samples greater than, and 349 reactions less than normal distributions. In complex 5 inhibited models 1034 samples have same varience, 309 and 445 samples with varience greater and lesser than normal distribution respectively.

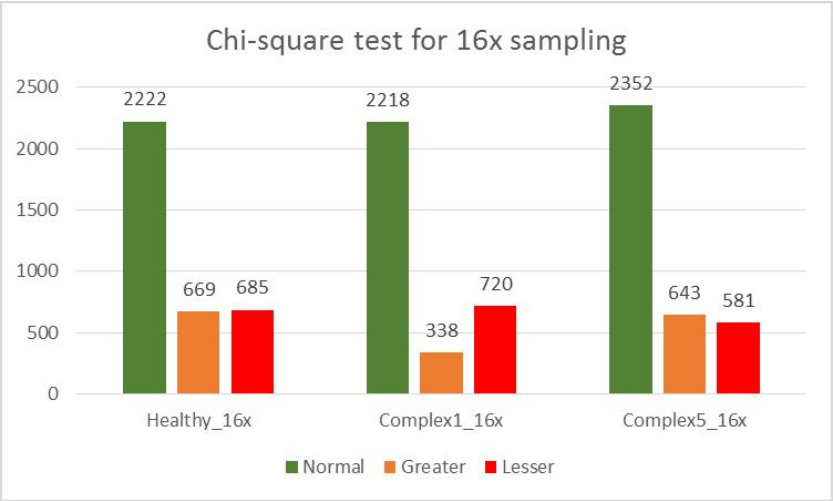

Figure 13. Chi-square test varience of 16x sampling.

In Figure 13, sampled points of 1786 reactions are tested for varience. In healthy controls, 2222 samples have same varience, 669 samples have varience greater normal distribution, and 685 samples have lesser varience. Complex 1 inhibited models, have same varience for 2218, 338 reactions greater than, and 720 samples less than normal distributions. In complex 5 inhibited models 2352 samples have same varience, 643 and 581 samples with varience greater and lesser than normal distribution respectively.

7 Conclusions

Contrained-based modelling has many applications in the field of bioinformatics. Genome-scale modelling and flux balance analysis(FBA) have played crucial roles in discovery of cures for various diseases[65, 53]. Here, genome-scale modelling is used for understanding the progression of Parkinson's Disease. Recon3D was used to build the required patient-specific dopaminergic neuronal model with the aquired information from *in vitro* cell cultures. The procees of data integration and building of model were explained. The constrained solution spaces were sampled using CHRR sampler. The uniformity of population and varience are tested, as shown in the results.

It can be observed from the obtained results that the percentage of uniformity of distributions is not high. For healthy control model, points that were sampled 8x the polytope size had 92.9% of uniform distributions, and 16x had 93.4% of uniform distributions. Similarly, Complex 1 inhibition model's

reactions have 95.3% of uniform distributions for 8x sampling, while for 16x reduces the amount of reactions with uniform distributions to 93.6%. Contrary to healthy controls and complex 1 inhibition models, complex 5 model's reactions have higher percentage of non-uniform distributions. For 8x sampling, complex 5 reactions have 94.4%, and for 16x, 94.6% of non-uniform distributions. For testing the varience of the samples chi-squared test was used. As seen in the results, there is high varience in both 8x and 16x sampling for healthy controls, complex 1, and complex 5 inhibited models.

Uniformity of a population represents the gap between the sampled points. Here, sampled points are of flux distributions, in a steady-solution space generated by constrained models of dopaminergic neurons. These points represent distributions for each reactions of the genome-scale model. As observed in results, there are more uniform distribution in complex 1 and healthy control models than in complex 5 models. It can be infered from this result that complex 1 and healthy control models have reactions have hard-limits over their constrained solution space, there by having uniformity in their sampled flux distributions. Whereas, same is not the case in complex 5 model. It has more reactions with non-uniform samples than uniform, infering that the boundaries applied on the solution spaces are not definitive. Theoratically, 8x sampling points achieves better performance[29]. This has been proven in the results obtained.

However, it is also observed that the varience is similar for all three models. Even with high varience, the number of uniform distributions is minimal in complex 5 models. This could be due to a number of reasons. The main property of any sampling algorithm is the step size. The step size determines how wide spread sampled points are. The larger the step size the higher the chances are of hitting the contraints, thus ignoring the sampled point. Another reason for non-uniformity is to be stuck at an local-optima solution, because of over sampling. The most common reason for non-uniformity is skipping the points that are sampled outside the constrained solution space.

Chapter 2 : Classification algorithm

From the CHRR sampler (Chapter 1), points representing each reaction are uniform randomly sampled. These sets of randomly sampled points are checked for uniformity with the Gap-Ratio (GA) algorithm. Each reaction's set of points are continuous. The statistical description of each population is derived to understand the shape of the distribution. Based on these inferences, various types of distributions are classified using a novel algorithm.

8 Types of flux distributions

Continuous distributions of each reaction provide an insight of the flux distributions of the solution space for individual reaction. These distributions have a cumulative distribution function that is absolutely continuous. A population set is said to be continuous if,

$$P[a \leq X \leq b] = \int_a^b f(x)dx \tag{7}$$

where [a,b] are the lower and upper bound of the reaction's contrainted solution space. The probability density function of a population represents the probabilities of occurence of solution for that reaction. A probability indicates the likelihood that a value will fall under a certain interval.

The area cover under the plot of probability distribution must always equal to 1. The proportion of the area under the curve that falls within a range of values along the X-axis represents the likelihood that a value will fall within that range.

There are multiple approaches to understand types of distributions. Graphically, Q-Q plots represent observed points of data, against the normally distributed data for the same mean and standard deviation. Statistically, Kolmogorov-Smirnov test compares the cumulative distribution function with step function of the same data. Different types of distributions observed in the data provided are represented in Figure 14.

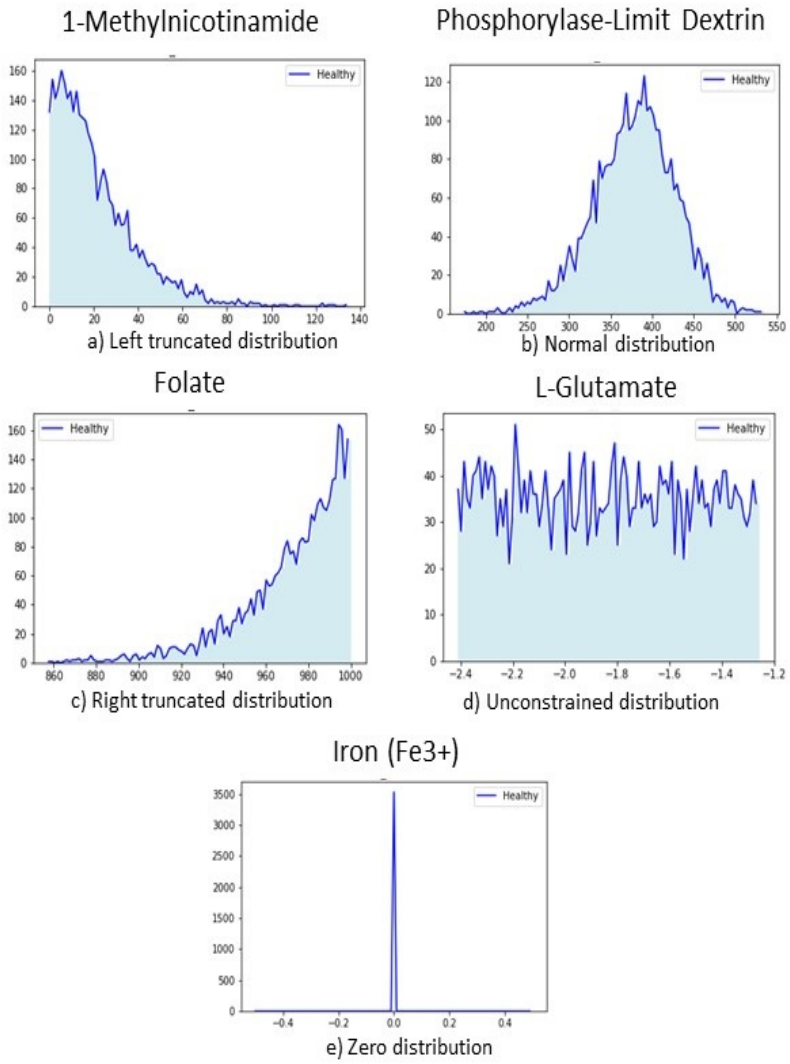

Figure 14: Types of flux distributions from the solution space.

When there is a disruption in any pathway, the uptake and secretion rates of corresponding reactions change. Characterisation of the flux distributions of individual reactions will lead to a better understanding of their changes between different models.

9 Kolmogorov Smirnov test

The sampled data points for each reaction from the solution space, are assumed to have a normal distributions. To check this hypothesis, Kolmogorov-Smirnov test for goodness of fit was used[40]. The null hypothesis of this parametric test assumes that the distribution in question correlates with a normal distribution for the same parameters. Alternatively, if the there is very little correlation between both distributions, null hypothesis is rejected.

For a deeper understanding, a cumulative distribution function of a reaction F(x), the cumulative step-function of a random sample of N observations is expected to be close to normal distribution function. If they do not correlate, then both the distributions vary considerably. Mathematically, if CDF(x) is the cumulative distribution function of x reaction, and $S_N(x)$is the step function of the same population, such that$S_N(x) = k/N$, where k is the number of observations less than or equal to x, then the sampling distribution of

$$d = max|F_0(x) - S_N(x)|$$

is known, is independent of$F_0(x)$ is continuous as shown in Figure 15.

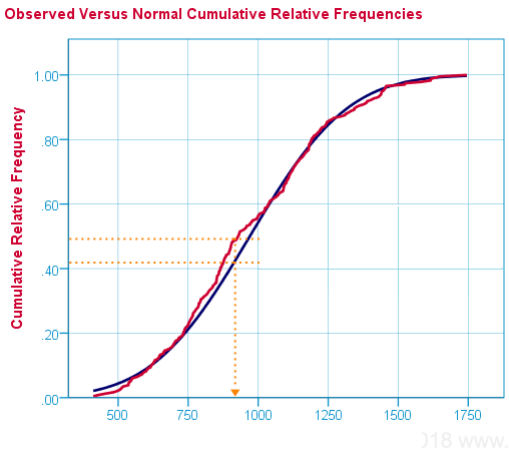

Figure 15. KS test cumulative relative frequency[2].

For the classification algorithm, 3497 points were sampled for each reaction. Based on the central limit theorem (CLT), the mean of sample points are used to plot the probability distributions from the acquired points. These sampled points for each reaction are run through KS-test for goodness of fit, between the sample distribution and normal distribution. Intuitively, if d is significantly

small, the compared distributions are more similar.

10 Effects of truncation on normal populations

Probability density function of a truncated normal variate can be mathematically given as:

$$f(x;\mu,\sigma,a,b) = \frac{1}{\sqrt{(2\pi)}\sigma} e^{\frac{-1}{2}(\frac{x-\mu}{\sigma})^2} \; -\infty < x < \infty; -\infty < \mu < \infty; \sigma > 0 \quad (8)$$

where,

$$C^{-1} = (2\pi)^{\frac{-1}{2}} \int_a^b e^{\frac{-1t^2}{2}} dt \quad (9)$$

The cumulative distribution function for a population derived from Equation (1) can be written as,

$$\widetilde{X} = (1/n) \sum_{i=1}^{n} X_i \quad (10)$$

and the quantile function can be represented by,

$$Z_{\{\alpha\}}/\sqrt{(n)} := Pr(X \leq x)/\sqrt{n} \quad (11)$$

Hypothesis tests derived from Equation(3) that, for a normal distribution with sample size n, the mean(μ) and varience(σ^2), one can assume standard deviation $\sigma^2 = 1$, without loosing generality.

Hypothesis	Conditions	Outcomes
H_0	$\mu = 0$	Normal distribution
	$\mu < 0$	
H_a	$\mu > 0$ if $\widetilde{X} > Z_{\{\alpha\}}/\sqrt{(n)}$tends to 1	left truncated distribution
	$\mu > 0$ if $\widetilde{X} > Z_{\{\alpha\}}/\sqrt{(n)}$ tends to 0	right truncated distribution

If H_a closer to1 the distribution is left truncated, and closer to 0 the distribution is right truncated as the sample size increases.

11 Skewness

The third moment of Equation (2) measures the skewness of the distribution. The nonnormality of a population can be described by using its central moments differing from ideal values. The third moment can be calculated as below:

$$\sqrt{\beta_1} = \frac{E(X-\mu)^3}{[E(X-\mu)^2]^{3/2}} = \frac{E(X-\mu)^3}{\sigma} \tag{12}$$

If the value of $\sqrt{\beta_1}$equal to 0, then it reflects symmetry. If it is great than zero, then it is skewed to the right, and if it is less than zero, it is skewed to the left(Figure 16).

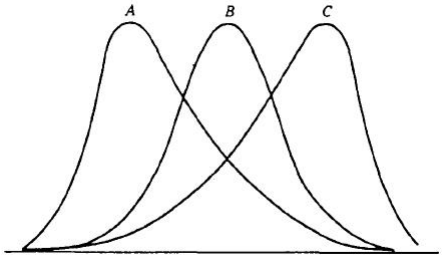

Figure 16: Differing distributions in Skewness; A: Right skewed distribution, B: Normal distribution, C: Left skewed distribution.[30]

Hypothesis	Conditions	Types
H_0	$\sqrt{\beta_1} = 0$	Normal distribution
H_a	$\sqrt{\beta_1} > 0$	Right skewed distribution
	$\sqrt{\beta_1} < 0$	Left skewed distribution

This property was discovered by [56, 22], to observe non-normal distributions formed by the populations. Popular algorithms such as D'Agostino and Stephans[21] have moments as their basis. For this algorithm, each reaction's sampled points are tested for skewness. Skewness observes the Expected value operator (E) over the cube of standard deviation. Implying, variation in the population influences the statistic of skewness. If the standard deviation is high, less expected value operator(E), and hence left skewness, and vice-versa. The larger the skewness on both sides, the more skewness.

12 Classification algorithm

The classification algorithm built allows a statistical description of flux distributions in the multi-dimensional solution space is described. In figure 17, classified distributions at different stages of the algorithm are picturized. In the first step, sample's population data is provided, for each model. Points for individual reactions are run through KS test for goodness of fit algorithm (section 2.2). In this step, zero dimensions and normal distributions along with indexes are recognized and subsetted. Reactions which are not subsetted in the first step are fed to effects of truncation on normal populations (section 2.3). Individual labels are given as "Left truncated", "Right truncated", and "normal distributions". On the same step, skewness(section 2.4) of these distributions are determined, and labelled. Upon comparing the labels, distributions with "left" and "right" truncation are subsetted. All the other distributions that do not fall under the three categories are labelled as "uniform distributions" for uncontrained reactions.

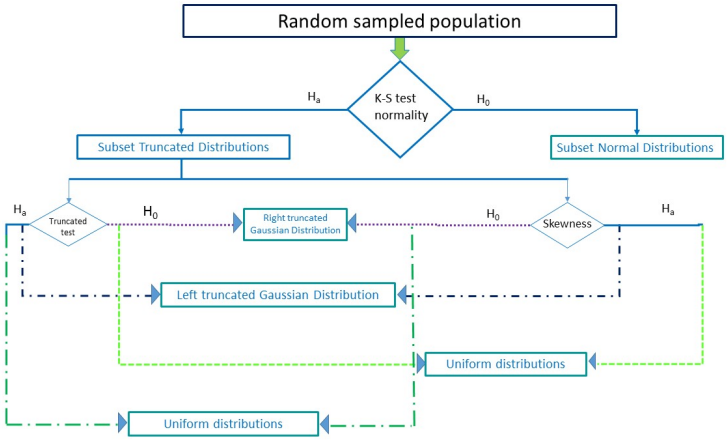

Figure 17. Classified distributions at different stages of the algorithm.

Algorithm 1 Classification algorithm

Input: List of samples for individual reactions per each row; Reaction names; number of samples(n)
Output: List of Exchange reactions along with samples and labels.

1. Integrate the list of reaction names with the matrix of sampled points for flux distributions from the solution space.
2. KS test for normality:
3. Cumulative distribution function(CDF) for individual reaction by taking mean and standard deviation of the sample.
4. IF p-value of CDF(x) $<$ 0.0000005: Fail to accept null hypothsis for normal distribution
   (a) Else: Fail to reject null hypothesis for normal distribution
   (b) Comparison index of distribution that failed ks test:
      i. IF p-value in 2.a.ii is 0: Zero dimension
      ii. Else: normal distribution
5. Effects of truncation on normal distribution:
   (a) Determine CDF(x)
      i. sum_norm = Sum of CDF(x)
      ii. len_norm = 1/n
      iii. $\widetilde{X}$ =len_norm * sum_norm
   (b) Inverse cumulative function(Z):
      i. Pr(x)
      ii. $Z_\alpha/\sqrt{n} = Pr(X \leq x)/\sqrt{n}$
      iii. IF $Z_\alpha < \widetilde{X}$:
         A. IF $\widetilde{X} \geq 0.5$: left truncated distribution
         B. Else : Right truncated distribution
      iv. Else : Normal distribution
6. Skewness:
   (a) Difference = []
      i. Skew(x)
      ii. Inverse cumulative distribution function : PPF(x)
      iii. Difference = skew(x) - PPF(x) : append "Difference list"
   (b) IF Difference(x) $>$ 0 : left skewed
      i. Elif Difference(x) = 0 : normal distribution
      ii. Else: Right skewed
7. Comparison of Skewness and effects of truncation on normal distribution:
   (a) IF Difference = Z_α: label the reaction same distribution
   (b) Else: Difference != Z_α(Truncated) is label for the reaction
   (c) Difference != Z_α(Skewness) is label for the reaction

13 Results

Classification algorithm was used to categorise 184 exchange reactions from three models. The distributions are classified into five catergories, "Normal distributions","Zero dimensions", "left skewed", "right skewed", and "unconstrained". The five categories of distributions are counted for all the three models.

	Complex 1 inhibitions	Complex 5 inhibitions	Healthy controls
Normal distributions	6	6	6
Zero dimensions	6	6	6
Left skewed	123	122	125
Right skewed	5	5	5
Unconstrained distributions	44	45	43

Table 1. Number of classified reactions into respective categories.

Catergorized distribution data is plotted as bar graph in figure 18. Highest number of flux distributions are left skewed for all three models, 123 for healthy controls, 122 for complex 1 inhibition and 125 for complex 5 inhibition. The second largest category is uncontrained reactions, with 44 reactions in healthy controls, 45 for complex 1, and 43 for complex 5. While there is a slight shift between left skewed distributions to unconstrained distributions, other three categories remain constant. Six reactions have normal distribution, six zero dimension reactions and five right skewed reactions for all three models.

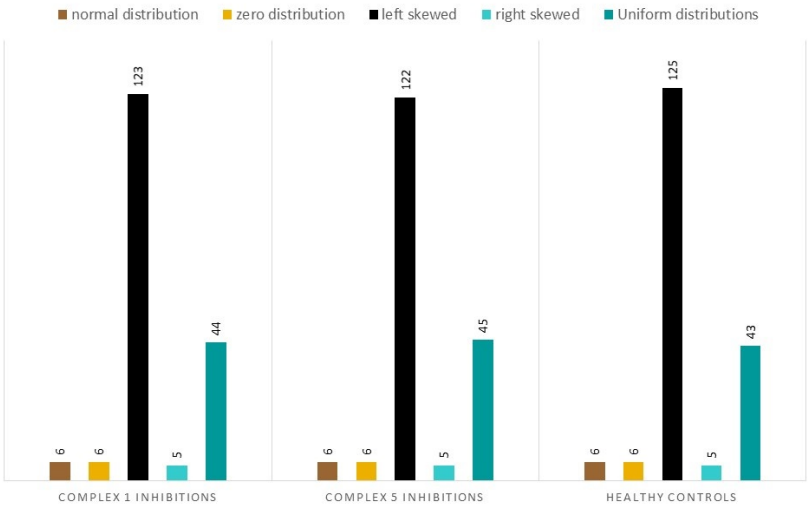

Figure 18. Display of categorised distributions

14 Conclusions

Comparing Flux distributions for a reaction is proven to be a difficult task for all three models. With no prior label of distribution type represented by the reaction in the solution space, important features such as phase shifts, or direction changes cannot be obeserved. Eventhough, comparing algorithms like f-test, t-test inform sufficiently about the statistics of the populations, crucial inferences for observing metabolic changes may not be infered. Hence, a classfication algorithm which effectively labels reactions with their corresponding distribution types, based on statistical inferences is very important for observing flux-based changes in genome-scale models.

From the results, an important observation is made, there are reactions that shift distributions between left truncated gaussians to uniform distributions and vice-versa. This would imply that certain metabolites, eventhough have higher secretion rates, the constraints of these reactions do not have hard lower-bounds, hence becoming unconstrained reactions. Taking this misallocation into account, the accuracy of predicted labels for flux distribution is 90%. Intutively, when plotted the labels are clearly accurate.

Chapter 3 : Comparison of distributions

From the classification algorithm the sampled distributions resulted into five classes, that is normal, left-truncated, right-truncated,uncontrained distributions and zero dimensions. The labelled reactions are listed, to compare same reactions from the three models. Each reaction is compared to observe the changes in flux distributions and the secretion/uptake rates of metabolites. These distributions are compared by observing the fold changes.

To compare reactions between two models, their distribution labels are taken into account. These labels are checked for smililarities. When a similar distribution is encountered, the metabolite is considered to have the same activity in both models. When the labels are dissimilar then there is a change in the metabolite activity. For example, if a certain reaction has a left skewed distribution in complex 1 inhibited model, where as in healthy control model the reaction is of uniform distribution, then there is a clear deviation between the metabolite's activity in the two models. So reactions with same labelled distributions have the same shape, while distributions with different labels represent the most changed reactions.

Each reaction in a genome-scale metabolic model is constrained using lower and upper bounds oberved in cell-cultures(chapter 1). The lower and upper bounds represent the uptake, and secretion rates of metabolites. If the flux distribution is symmetrical around zero, it is a normal distribution. Normal distributions have equal uptake and secretion rates. When the flux distribution reaches zero from negative ranges, by left or right truncation, the activity of reaction decrease. Implying uptake/consumption of the metabolite. While the flux distribution moves away from zero to positive ranges, the reaction is more active, there by implying secretion/production. Uniform distributions represent reactions with no hard boundaries, while zero dimensions are support reactions for the metabolic networks.

The comparison between complex 1 model and healthy control model's exchange reactions have 134 reactions with the same labels as shown in Table 2. The break-down of each category is also shown. As observed, there are 38 reactions with dissimilar distributions, where 23 are right truncated gaussians and 15 are left truncated gaussians. The dissimilar distribution between complex 1

and healthy controls are stated in Table 3, with reaction and metabolite. The metabolite's flux changes are not similarly distributed, implying a change in their activity. Hence these reactions could be interesting for further investigations.

Non-equal reactions are distributions that the algorithm was not able to decide a definitive categorisation. Hence these reactions are labelled to have uniform distributions, 14 of them fall under this category when compared between complex 1 and healthy control models.

Distribution types	Equal distributions		Non-equal distributions	
Similarities	Similar(134)	Non-similar(38)	Similar(6)	Non-similar(8)
Normal distributions	6	-	-	-
Zero dimensions	6	-	-	-
Left truncated distributions	119	-	-	-
Right truncated distributions	3	-	-	-
Right truncated	-	23	4	2
Left truncated	-	15	2	6
Right skewed	-	-	-	-
Left skewed	-	-	-	-
Not equal	38	6	8	-

Table 2: Similar distributions between complex 1 and healthy control model's reactions.

Reactions	Metabolites
EX_fucacngalacglcgalgluside_hs[e]	Iv3-A-Neu5Ac, Iii4-A-Fuc-Lc4Cer
EX_HC00229[e]	Isomaltose
EX_phe_L[e]	L-Phenylalanine
EX_nh4[e]	Ammonia
EX_xolest_hs[e]	Cholesterol Ester
EX_xoltri24[e]	7-Alpha, 24(S)-Dihydroxycholesterol
EX_xylt[e]	Xylitol

Table 3: list of maximum changed reactions between complex 1 and healthy control model's reactions.

Most changed reactions have been represented in figure 19 below.

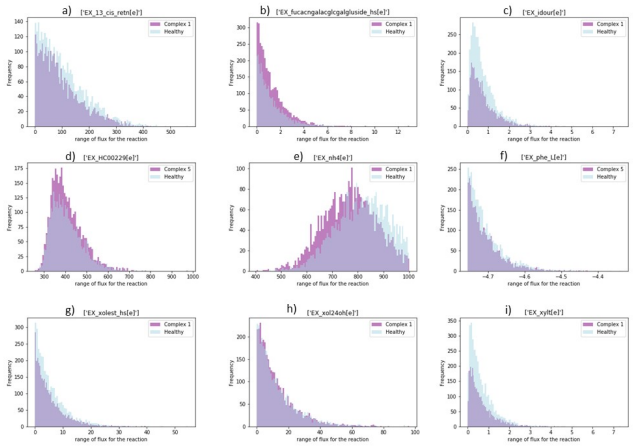

Figure 19: flux distribution plots for most changed reactions between complex 1 and healthy controls.

Comparison between complex 5 model and healthy control model's exchange reactions have 130 similar distribution labels as shown in Table 3. The breakdown of each category is as observed, there are 36 reactions with dissimilar distributions, with 12 right truncated gaussians and 23 left truncated gaussians. The dissimilar distribution between complex 1 and healthy controls are stated in Table 5, with reaction and metabolite names. The metabolite's flux changes are not similarly distributed, implying a change in flow of those metabolites.=

Non-equal reactions are distributions that the algorithm was not able to decide a definitive categorisation. Hence these reactions are labelled to have uniform distributions, which are 18 of them when compared between complex 5 and healthy control models.

Distribution types	Equal distributions		Non-equal distributions	
Similarities	Similar(130)	Non-similar(36)	Similar(9)	Non-similar(9)
Normal distributions	4	-	-	2
Zero dimensions	6	-	-	-
Left truncated distributions	116	-	-	6
Right truncated distributions	4	-	-	1
Right truncated	-	12	8	-
Left truncated	-	24	1	-
Right skewed	-	-	-	-
Left skewed	-	-	-	-
Not equal	36	9	9	-

Table 4: Similar distributions between complex 5 and healthy control model's reactions.

Reactions	Metabolites
EX_dxtrn[e]	Phosphorylase-Limit Dextrin
EX_gal[e]	D-Galactose
EX_glcn[e]	D-Gluconate
EX_glcur[e]	D-Glucuronate
EX_h2o[e]	Water
EX_h2o2[e]	Hydrogen Peroxide
EX_HC00229[e]	Isomaltose
EX_hco3[e]	Bicarbonate
EX_hxan[e]	Hypoxanthine
EX_lac_L[e]	L-Lactate
EX_leuktrB4[e]	Leukotriene B4
EX_nicrnt[e]	Nicotinic acid mononucleotide
EX_o2[e]	Oxygen
EX_pydxn[e]	Pyridoxine
EX_so4[e]	Sulfate
EX_thymd[e]	Thymidine
EX_xolest_hs[e]	Cholesterol Ester

Table 5: list of maximum changed reactions between complex 5 and healthy control model's reactions.

Most changed reactions have been represented in figure 20 below.

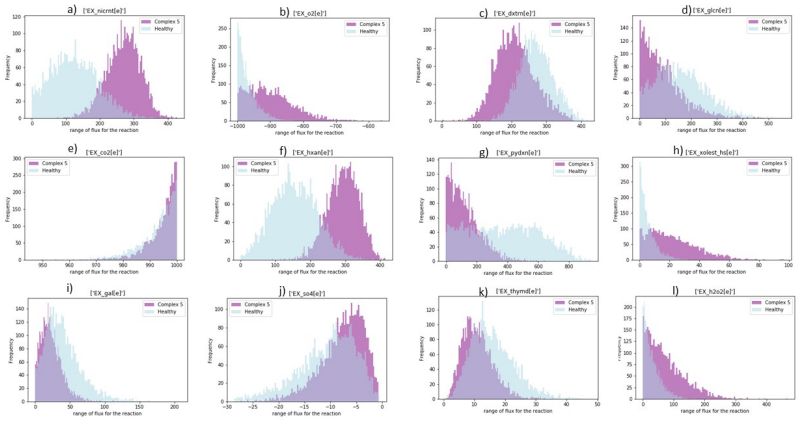

Figure 20: flux distribution plots for most changed reactions between complex 5 and healthy controls.

Generally, to observe changes between two models and their most changes reactions log of ratio between populations are derived. Fold change is the ratio between mean of the distribution of inhibited model and the control model over each reaction. This can be mathematically represented by,

$$FC = \frac{M_{inhibition}}{M_{healthycontrols}} \tag{13}$$

where M= mean of distribution.

Log fold change is represented by $log_2(FC)$ of Equation (1). Log fold changes are used to easily interpret the fold changes between two models. In the biological perspective, if the log fold change is less than zero, then there is a change in inhibition model. If the value is equal to or greater than zero, then there is change in control model.

Fold changes of exchange reactions of three models in question are calculated, along with their log fold changes. The fold changes between reactions that are greater than between complex 5 and healthy control models are subsetted, and plotted as a heatmap shown in Figure 22. The colour scheme represented shows that larger values have darker shades and smaller values(less than zero) have lighter shades. It can be observed that most of the reactions have similar means for healthy control model and complex 5, while complex 1 reaction means show many differences compared to control. Figure 21 represents the heatmap for fold changes between complex 1 and healthy control models. It can be observed that

there are similarities in complex 5 inhibited model means and healthy control means, while complex 1 changes.

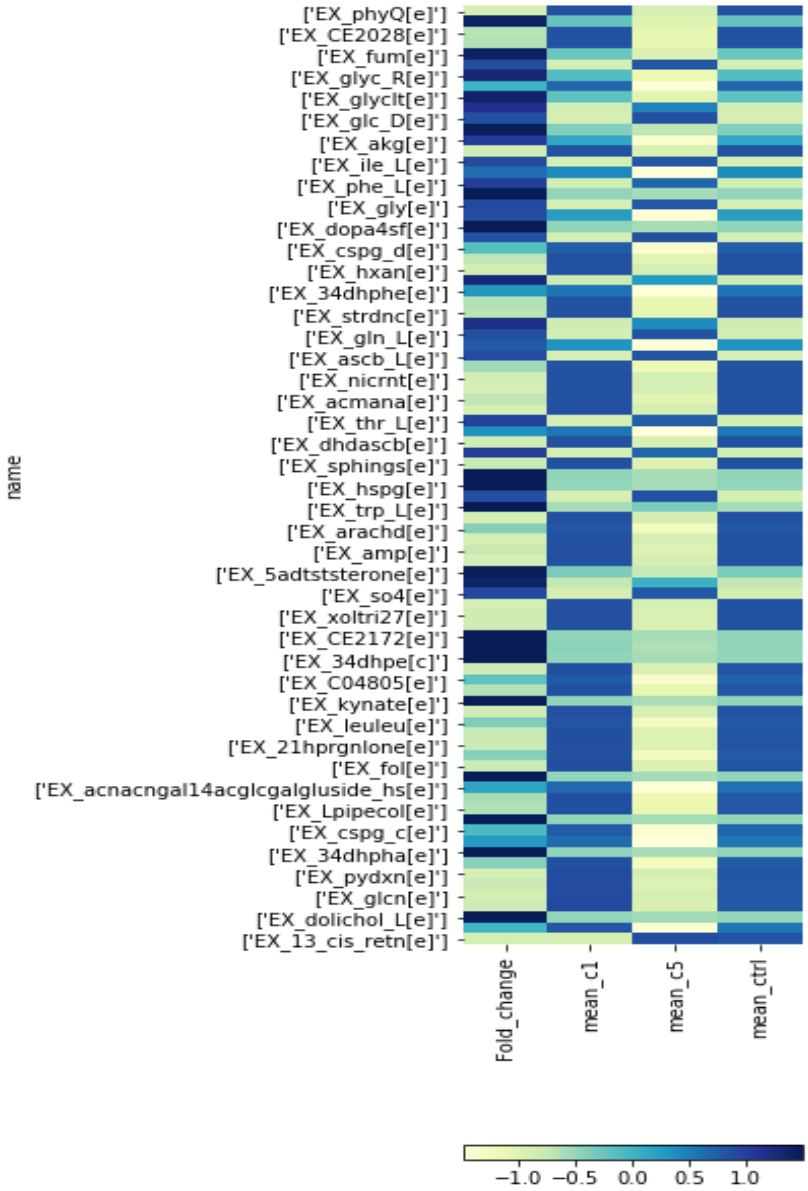

Figure 21: Fold changes between reactions of complex 5 and healthy control models for similar reactions.

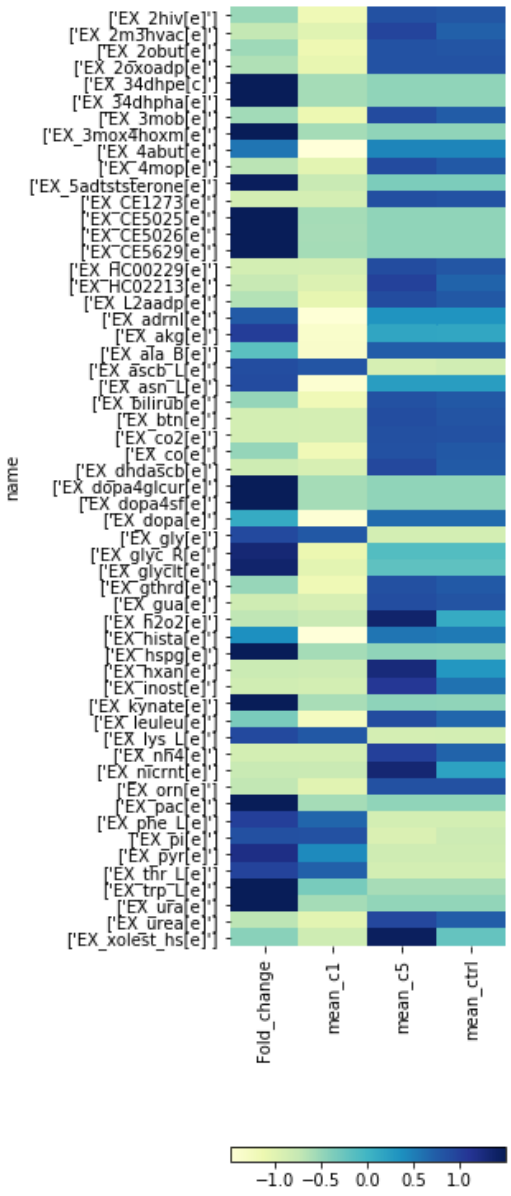

Figure 22: Fold changes between reactions of complex 1 and healthy control models for similar reactions.

Figures 19 and 20 show the most varied reactions in the complex 1 and complex 5 inhibitions in comparison with healthy controls respectively. It can be observed there are more varying reactions in complex 5 inhibitions than in complex 1.

Conclusion Classified flux distributions need to be compared to identify significant changes. These significant changes may lead to discovering important pathways that cause neurodegeneration thereby leading to Parkinson's Diease. To observe flux changes, labelled distributions are compared between inhibition models and healthy control models. Reactions with dissimilar labels seems to have maximum changes in flux distributions, as per the obtained results.

In complex 5 inhibited model's exchange reactions for Oxygen, L-Lactic acid, Phosphate, Water, Dehydroascorbic acid, Carbondioxide, Isomaltose, Ascorbic acid, and L-Cystine have shown major changes when compared with healthy control models. These are positive results since, complex 5 is a enzyme-complex used to produce ATP for the energy of cells. When this complex is inhibited, there needs to be flux changes for oxygen, water, carbondioxide, which are important reactions for release Adinosine triphosphate, which provides the cells with energy. Similar changes have been shown in the heatmap. Hence, classification algorithm has considerable accuracy in predicting and comparing dissimilar flux distributions between complex 5 and heatlhy controls.

Similarly, complex 1 inhibited model's reactions for L-Lactic acid, L-Phenylalanine, Cholesterol Ester, Hydrogen Peroxide, Isomaltose, Ammonia have shown major changes when compared to healthy controls. All these metabolites play important roles in TCA cycle which initiates the product for electron tranfer in mitochondria. Eventhough most of the common observed changes are part of TCA cycle which is altered by complex 1 inhibitions, complex 5 inhibition has shown more alterations in reactions. This conclusively proves that inhibiting complex 5 has more effect over neuron. It can be expected to see deviations in flux distributions for these metabolites. Therefore, classification algorithm displays positive results for categorize differences in flux distributions, there by allowing comparisons.

Discussion

Overall conclusions

Parkinson's Disease is a neurodegenerative disease, that effects up to 10 million people every year. It is a progressive disease with motor symptoms such as tremors, bradykinesia (slowness in movement)[7], postural instability[10], rigidity and tremors, which could lead to immobility. Some other symptoms are non-motor such as loss of smell, depression, and fatigue[33, 17]. Evidence has proven that the neurodegeneration, is caused due to loss of dopaminergic neurons in substantia nigra part of the brain, responsible for movement in the body. However the pathology of the death of dopaminergic neurons in PD is still incompletely understood. Multiple hypotheses are aiming at unravelling the cause of this progressive disorder, such as proteostasis, oxidative stress, mitochondrial dysfunction, neuroinflammation[39]. Hence, this research looks into the effects of PINK1 gene mutations on dopaminergic neurons. Complex 1 mutation and complex 5 mutations are observed to obtain the maximum changes cause to mitochondria in comparison to healthy controls. These changes are observed under the consortium of SysMedPD which takes a system's approach for predicting biomarkers for Parkinson's Disease.

System's approach to observe progression of Parkinson's Disease is a novel approach. Constrained-based modelling is used to build candidate-specific genome-scale dopaminergic neuron model. The constrained steady state solution spaces are uniform randomly sampled for flux distributions of each reactions. These continuous flux distributions can be of various types(normal, uniform, truncated). This research provides a novel classification algorithm, which effectively classified the variety of flux distrbutions, and labels the reactions with their corresponding distribution type. Thereby, allowing an informed observation of maximum changed reactions, in the uptake and secretion rates. These are compared using log fold changes and maximum deviated reactions represented as heatmap.

In Chapter 1, steps taken to attain a constrained-based model of dopaminergic neuron is discussed. It is observed that there are 1789 reactions for the currently used model for healthy controls. This model is inhibited for complex 1 pathways constraining the uptake and secretion rates for corresponding reactions and the same for complex 5 pathway. These pathways generate a steady

state solution space which when randomly sampled produces flux distributions of each reaction. These population of flux distributions are checked for uniformity and varience by using gap-ratio algorithm and chi-square test respectively. From the results it can be observed that 8x sampling of the solution space gives an average uniformity of 70% over all the three models, avoiding getting stuck in a local optima as is the case for 16x sampling. This conclusively proves that 8x the dimensions of the polytope gives better results than 16x sampling. The varience tested is quiet high for both times of sampling, hence it can be concluded that sampling algorithm produces various solutions for the same reaction.

In Chapter 2, the classification algorithm has been introduced. Each algorithm used as a part of classification is explained in detail. Exchange reactions from the three models are run through the classification algorithm to categorize them as normal distributions, zero dimensions, unconstrained distributions, left truncated distributions and right truncated distributions. An accuracy of 90% is achieved in categorising flux distributions for exchange reactions. It can be concluded that the classification algorithm is useful to statistically categorise the flux distributions, with high accuracy.

In Chapter 3, comparison of the distributions between complex 1, complex 5 and healthy controls were stated. For each reaction the labels of their corresponding reactions are compared to obtain similarities and most importantly dissimilarity of distributions. As observed, there are more reactions with similar distributions between complex 1 and healthy controls than complex 5 and healthy controls. Fold changes are compared and a heatmap plotted to observe phase shifts between complex 5 and healthy controls in relation with complex 1. Exhange reactions for metabolites for L-Lactic acid, L-Phenylalanine, Cholesterol Ester, Hydrogen Peroxide, Isomaltose, Ammonia have shown major in both inhibitions when compared to healthy controls.

One of the most commonly changed metabolite when comparing complex 1 and complex 5 inhibited models with healthy controls is Isomaltose. It is a disaccharide similar to maltose, with an alpha linkage. Isomaltose is used as a product of digestion, there by producing the energy to the cell. When there is an alteration of this metabolite to the right, the usage of it has increased. Eventhough it is not a direct biomarker pointing towards PD, the energy degradation process that it is a biproduct of, is a known path for the progression of the disease[15]. Hence it can be proposed as a biomarker for observing the progression of the disease.

Other metabolites that showed alterations are cholesterol ester, and hydro-

gen peroxide. The electronic properties of hydrogen peroxide were monitored for progression of Parkinson's Disease. Increase in the metabolite has shown shifts to left truncations, thereby leading higher levels of deep brain variations. This could be caused by oxidative stress to the brain[36]. Implying from the changes seen in this research and [36], hydrogen peroxide could be a varient in observing dopamine secretion rates in different parts of the brain.

Lipids have lead considerable hypothesis for the study of progression of the disease. A number of control-studies have suggested lower prevalence of PD with an increase in serum levels of cholesterol[59, 42].While some studies showed evidence that this phenomenon occurs only in male metabolism[6]. Although recent studies have shown that a decrease in cholesterol ester would increase the risk of PD[43]. As observed in this research, there has been considerable changes in cholesterol ester's secretion rates, which could be factor for progression of PD, or an element to decrease the prevalence of the disease.

Overall there are considerable changes seen in metabolites that are proven biomarkers. Some of the other biomarkers observed while comparing individual inhibitions on the networks with healthy controls are: xylitol[71, 73] and ammonia[16] for complex 1 inhibitions, water[27], oxygen[3, 67, 70], phosphorylase-limit[25], dextrin[54] and carbondioxide[27] for complex 5. These metabolites have been proven to progression of parkinson's disease. One of the most important metabolite that varied in both inhibitions is in dompamine. In both the models, it was represeting a left truncated distribution implying higher usage, and lesser production. Hence, it is proven that the classification algorithm labels each reaction with corresponding flux distributions and they can be observed to show maximum changes.

Future work

In the futuristic perspective, the main objective would be to embed the classification algorithm into model generation in Matlab. By fixing this algorithm in the process of model generation(Chapter 1), the flux distributions are labelled and classified as a part of the output. This could lead to catergorized comparisons of reactions, thereby hypothesizing efficiently. These hypothesis would lead to predicting biomarkers for early-onset of Parkinson's Disease.

Secondly, the results are comparisons of exchange reactions alone. Implementing these comparisons for all the reactions and conclusively represent biomarkers for progression of Parkinson's disease due to PINK1 gene mutation.

These comparisons would have better effects when compared by the ratio standard deviations of sampled populations rather than means. This could also be another futuristic goal.

Thirdly, more efforts can be put into testing different sampling algorithms to achieve higher uniform distributions, without disrupting the varience of results. Increasing efficiency of sampling algorithms reduces the stress on hardware, there by expidite the computations.

References

[1] Sysmedpd-new directions for better disease detection and therapy for parkinson's disease.

[2] 2019.

[3] James D Adams Jr and Ifeoma N Odunze. Oxygen free radicals and parkinson's disease. *Free Radical Biology and Medicine*, 10(2):161–169, 1991.

[4] Tihomir Asparouhov, Bengt Muthén, and BO Muthén. Robust chi square difference testing with mean and variance adjusted test statistics. *matrix*, 1(5):1–6, 2006.

[5] GJ Baart and DE Martens. Genome-scale metabolic models: reconstruction and analysis. *Methods in molecular biology (Clifton, NJ)*, 799:107, 2012.

[6] Shuang Bai, Yi Song, Xin Huang, Lidan Peng, Jie Jia, Yu Liu, and Hong Lu. Statin use and the risk of parkinson's disease: an updated meta-analysis. *PLoS One*, 11(3):e0152564, 2016.

[7] Alfredo Berardelli, JC Rothwell, PD Thompson, and M Hallett. Pathophysiology of bradykinesia in parkinson's disease. *Brain*, 124(11):2131–2146, 2001.

[8] Arijit Bishnu, Sameer Desai, Arijit Ghosh, Mayank Goswami, and Subhabrata Paul. Uniformity of point samples in metric spaces using gap ratio. *SIAM Journal on Discrete Mathematics*, 31(3):2138–2171, 2017.

[9] Anna S Blazier and Jason A Papin. Integration of expression data in genome-scale metabolic network reconstructions. *Frontiers in physiology*, 3:299, 2012.

[10] Bastiaan R Bloem, Yvette AM Grimbergen, Monique Cramer, Mirjam Willemsen, and Aeilko H Zwinderman. Prospective assessment of falls in parkinson's disease. *Journal of neurology*, 248(11):950–958, 2001.

[11] Evan E Bolton, Yanli Wang, Paul A Thiessen, and Stephen H Bryant. Pubchem: integrated platform of small molecules and biological activities. In *Annual reports in computational chemistry*, volume 4, pages 217–241. Elsevier, 2008.

[12] James G Booth and James P Hobert. Maximizing generalized linear mixed model likelihoods with an automated monte carlo em algorithm. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 61(1):265–285, 1999.

[13] Pierre Brémaud. *Markov chains: Gibbs fields, Monte Carlo simulation, and queues*, volume 31. Springer Science & Business Media, 2013.

[14] E Brunk, S Sahoo, DC Zielinski, A Altunkaya, A Draeger, N Mih, F Gatto, A Nilsson, GAP Gonzales, MK Aurich, et al. Recon3d: A resource enabling a three-dimensional view of gene variation in human metabolism. *Nat Biotech*, 2017.

[15] David Sancho Cantus, Natalia Santiesteban López, María Cuerda Ballester, Silvia Solera Gómez, and José Enrique de la Rubia Ortí. Stress in parkinson's disease. cortisol and amylase biomarkers. systematic review. *Revista Científica de la Sociedad de Enfermería Neurológica (English ed.)*, 2019.

[16] DR Chadwick. Emissions of ammonia, nitrous oxide and methane from cattle manure heaps: effect of compaction and covering. *Atmospheric environment*, 39(4):787–799, 2005.

[17] Chou Chai and Kah-Leong Lim. Genetic insights into sporadic parkinson's disease pathogenesis. *Current genomics*, 14(8):486–501, 2013.

[18] Ira E Clark, Mark W Dodson, Changan Jiang, Joseph H Cao, Jun R Huh, Jae Hong Seol, Soon Ji Yoo, Bruce A Hay, and Ming Guo. Drosophila pink1 is required for mitochondrial function and interacts genetically with parkin. *Nature*, 441(7097):1162, 2006.

[19] Barbara S Connolly and Anthony E Lang. Pharmacological treatment of parkinson disease: a review. *Jama*, 311(16):1670–1683, 2014.

[20] Katrina Cowan, Oleg Anichtchik, and Shouqing Luo. Mitochondrial integrity in neurodegeneration. *CNS neuroscience & therapeutics*, 2019.

[21] Ralph B D'Agostino. *Goodness-of-fit-techniques*, volume 68. CRC press, 1986.

[22] Ralph B D'agostino, Albert Belanger, and Ralph B D'Agostino Jr. A suggestion for using powerful and informative tests of normality. *The American Statistician*, 44(4):316–321, 1990.

[23] Roy Danchick. *The Delphic Boat: what genomes tell us*. Harvard University Press, 2002.

[24] Lonneke ML De Lau and Monique MB Breteler. Epidemiology of parkinson's disease. *The Lancet Neurology*, 5(6):525–535, 2006.

[25] Salvatore DiMauro and Costanza Lamperti. Muscle glycogenoses. *Muscle & Nerve: Official Journal of the American Association of Electrodiagnostic Medicine*, 24(8):984–999, 2001.

[26] Paul Flicek, M Ridwan Amode, Daniel Barrell, Kathryn Beal, Konstantinos Billis, Simon Brent, Denise Carvalho-Silva, Peter Clapham, Guy Coates, Stephen Fitzgerald, et al. Ensembl 2014. *Nucleic acids research*, 42(D1):D749–D755, 2013.

[27] Jay M Gorell, CC Johnson, BA Rybicki, EL Peterson, and RJ Richardson. The risk of parkinson's disease with exposure to pesticides, farming, well water, and rural living. *Neurology*, 50(5):1346–1350, 1998.

[28] Jessica C Greene, Alexander J Whitworth, Isabella Kuo, Laurie A Andrews, Mel B Feany, and Leo J Pallanck. Mitochondrial pathology and apoptotic muscle degeneration in drosophila parkin mutants. *Proceedings of the National Academy of Sciences*, 100(7):4078–4083, 2003.

[29] Hulda S Haraldsdóttir, Ben Cousins, Ines Thiele, Ronan MT Fleming, and Santosh Vempala. Chrr: coordinate hit-and-run with rounding for uniform sampling of constraint-based models. *Bioinformatics*, 33(11):1741–1743, 2017.

[30] W Keith Hastings. Monte carlo sampling methods using markov chains and their applications. 1970.

[31] Laurent Heirendt, Sylvain Arreckx, Thomas Pfau, Sebastian N Mendoza, Anne Richelle, Almut Heinken, Hulda S Haraldsdottir, Jacek Wachowiak, Sarah M Keating, Vanja Vlasov, et al. Creation and analysis of biochemical constraint-based models using the cobra toolbox v. 3.0. *Nature protocols*, 14(3):639, 2019.

[32] François Jacob. Evolution and tinkering. *Science*, 196(4295):1161–1166, 1977.

[33] Joseph Jankovic. Parkinsonâs disease: clinical features and diagnosis. *Journal of neurology, neurosurgery & psychiatry*, 79(4):368–376, 2008.

[34] Joseph Jankovic and L Giselle Aguilar. Current approaches to the treatment of parkinsonâs disease. *Neuropsychiatric disease and treatment*, 4(4):743, 2008.

[35] Minoru Kanehisa, Susumu Goto, Yoko Sato, Miho Furumichi, and Mao Tanabe. Kegg for integration and interpretation of large-scale molecular data sets. *Nucleic acids research*, 40(D1):D109–D114, 2011.

[36] CV Krishnan, M Garnett, and B Chu. Spatiotemporal oscillations in biological molecules: Hydrogen peroxide and parkinsonâs disease. *Int. J. Electrochem. Sci*, 3:1364–1385, 2008.

[37] J William Langston, Philip Ballard, James W Tetrud, and Ian Irwin. Chronic parkinsonism in humans due to a product of meperidine-analog synthesis. *Science*, 219(4587):979–980, 1983.

[38] Hongwu Ma, Anatoly Sorokin, Alexander Mazein, Alex Selkov, Evgeni Selkov, Oleg Demin, and Igor Goryanin. The edinburgh human metabolic network reconstruction and its functional analysis. *Molecular systems biology*, 3(1), 2007.

[39] Longfei Mao, Averina Nicolae, Miguel AP Oliveira, Feng He, Siham Hachi, and Ronan MT Fleming. A constraint-based modelling approach to metabolic dysfunction in parkinson's disease. *Computational and structural biotechnology journal*, 13:484–491, 2015.

[40] Frank J Massey Jr. The kolmogorov-smirnov test for goodness of fit. *Journal of the American statistical Association*, 46(253):68–78, 1951.

[41] Wout Megchelenbrink, Martijn Huynen, and Elena Marchiori. optgpsampler: an improved tool for uniformly sampling the solution-space of genome-scale metabolic networks. *PloS one*, 9(2):e86587, 2014.

[42] Yoshihiro Miyake, Keiko Tanaka, Wakaba Fukushima, Satoshi Sasaki, Chikako Kiyohara, Yoshio Tsuboi, Tatsuo Yamada, Tomoko Oeda, Takami Miki, Nobutoshi Kawamura, et al. Case–control study of risk of parkinson's disease in relation to hypertension, hypercholesterolemia, and diabetes in japan. *Journal of the neurological sciences*, 293(1-2):82–86, 2010.

[43] ROBERTO Musanti, EUGENIO Parati, ELENA Lamperti, and G Ghiselli. Decreased cholesterol biosynthesis in fibroblasts from patients with parkinson disease. *Biochemical medicine and metabolic biology*, 49(2):133–142, 1993.

[44] Derek Narendra, Atsushi Tanaka, Der-Fen Suen, and Richard J Youle. Parkin is recruited selectively to impaired mitochondria and promotes their autophagy. *The Journal of cell biology*, 183(5):795–803, 2008.

[45] C Warren Olanow, Olivier Rascol, Robert Hauser, Paul D Feigin, Joseph Jankovic, Anthony Lang, William Langston, Eldad Melamed, Werner Poewe, Fabrizio Stocchi, et al. A double-blind, delayed-start trial of rasagiline in parkinson's disease. *New England Journal of Medicine*, 361(13):1268–1278, 2009.

[46] Bernhard Ø Palsson. *Systems biology: properties of reconstructed networks*. Cambridge university press, 2006.

[47] Jeehye Park, Sung Bae Lee, Sungkyu Lee, Yongsung Kim, Saera Song, Sunhong Kim, Eunkyung Bae, Jaeseob Kim, Minho Shong, Jin-Man Kim, et al. Mitochondrial dysfunction in drosophila pink1 mutants is complemented by parkin. *Nature*, 441(7097):1157, 2006.

[48] Alicia M Pickrell and Richard J Youle. The roles of pink1, parkin, and mitochondrial fidelity in parkinsonâs disease. *Neuron*, 85(2):257–273, 2015.

[49] Steve R Pieczenik and John Neustadt. Mitochondrial dysfunction and molecular pathways of disease. *Experimental and molecular pathology*, 83(1):84–92, 2007.

[50] Natapol Pornputtapong, Intawat Nookaew, and Jens Nielsen. Human metabolic atlas: an online resource for human metabolism. *Database*, 2015, 2015.

[51] German. Preciat. Mechanistic model-driven exometabolomic characterisation of human dopaminergic neuronal metabolism. *in preparation*, 2019.

[52] Nathan D Price, Jennifer L Reed, and Bernhard Ø Palsson. Genome-scale models of microbial cells: evaluating the consequences of constraints. *Nature Reviews Microbiology*, 2(11):886, 2004.

[53] Vytautas Raškevičius, Valeryia Mikalayeva, Ieva Antanavičiūtė, Ieva Ceslevičienė, Vytenis Arvydas Skeberdis, Visvaldas Kairys, and Sergio Bordel. Genome scale metabolic models as tools for drug design and personalized medicine. *PloS one*, 13(1):e0190636, 2018.

[54] Rajendra Kumar Rath, S Subramanian, and JS Laskowski. Adsorption of dextrin and guar gum onto talc. a comparative study. *Langmuir*, 13(23):6260–6266, 1997.

[55] Philippe Rizek, Niraj Kumar, and Mandar S Jog. An update on the diagnosis and treatment of parkinson disease. *Cmaj*, 188(16):1157–1165, 2016.

[56] Christian Robert and George Casella. A short history of markov chain monte carlo: Subjective recollections from incomplete data. *Statistical Science*, pages 102–115, 2011.

[57] Pedro Romero, Jonathan Wagg, Michelle L Green, Dale Kaiser, Markus Krummenacker, and Peter D Karp. Computational prediction of human metabolic pathways from the complete human genome. *Genome biology*, 6(1):R2, 2005.

[58] AHV Schapira, JM Cooper, D Dexter, JB Clark, P Jenner, and CD Marsden. Mitochondrial complex i deficiency in parkinson's disease. *Journal of neurochemistry*, 54(3):823–827, 1990.

[59] Anthony HV Schapira. Mitochondria in the aetiology and pathogenesis of parkinson's disease. *The Lancet Neurology*, 7(1):97–109, 2008.

[60] Jan Schellenberger, Richard Que, Ronan MT Fleming, Ines Thiele, Jeffrey D Orth, Adam M Feist, Daniel C Zielinski, Aarash Bordbar, Nathan E Lewis, Sorena Rahmanian, et al. Quantitative prediction of cellular

metabolism with constraint-based models: the cobra toolbox v2. 0. *Nature protocols*, 6(9):1290, 2011.

[61] Anette Schrag. Entacapone in the treatment of parkinson's disease. *The Lancet Neurology*, 4(6):366–370, 2005.

[62] Manish Sud, Eoin Fahy, Dawn Cotter, Alex Brown, Edward A Dennis, Christopher K Glass, Alfred H Merrill Jr, Robert C Murphy, Christian RH Raetz, David W Russell, et al. Lmsd: Lipid maps structure database. *Nucleic acids research*, 35(suppl_1):D527–D532, 2006.

[63] Ines Thiele, Nathan D Price, Thuy D Vo, and Bernhard Ø Palsson. Candidate metabolic network states in human mitochondria impact of diabetes, ischemia, and diet. *Journal of Biological Chemistry*, 280(12):11683–11695, 2005.

[64] Ines Thiele, Neil Swainston, Ronan MT Fleming, Andreas Hoppe, Swagatika Sahoo, Maike K Aurich, Hulda Haraldsdottir, Monica L Mo, Ottar Rolfsson, Miranda D Stobbe, et al. A community-driven global reconstruction of human metabolism. *Nature biotechnology*, 31(5):419, 2013.

[65] Beste Turanli, Cheng Zhang, Woonghee Kim, Rui Benfeitas, Mathias Uhlen, Kazim Yalcin Arga, and Adil Mardinoglu. Discovery of therapeutic agents for prostate cancer using genome-scale metabolic modeling and drug repositioning. *EBioMedicine*, 42:386–396, 2019.

[66] Patrik Verstreken. *Parkinson's disease: molecular mechanisms underlying pathology*. Academic Press, 2016.

[67] A Vidal-Madjar, J-M Désert, A Lecavelier Des Etangs, G Hébrard, GE Ballester, D Ehrenreich, R Ferlet, JC McConnell, M Mayor, and CD Parkinson. Detection of oxygen and carbon in the hydrodynamically escaping atmosphere of the extrasolar planet hd 209458b. *The Astrophysical Journal Letters*, 604(1):L69, 2004.

[68] Peter Wellstead and Mathieu Cloutier. *Systems biology of Parkinson's disease*. Springer Science & Business Media, 2012.

[69] David S Wishart, Dan Tzur, Craig Knox, Roman Eisner, An Chi Guo, Nelson Young, Dean Cheng, Kevin Jewell, David Arndt, Summit Sawhney, et al. Hmdb: the human metabolome database. *Nucleic acids research*, 35(suppl_1):D521–D526, 2007.

[70] Leslie I Wolfson, Klaus L Leenders, Lucy L Brown, and Terry Jones. Alterations of regional cerebral blood flow and oxygen metabolism in parkinson's disease. *Neurology*, 35(10):1399–1399, 1985.

[71] Anna Wuolikainen, Pär Jonsson, Maria Ahnlund, Henrik Antti, Stefan L Marklund, Thomas Moritz, Lars Forsgren, Peter M Andersen, and Miles Trupp. Multi-platform mass spectrometry analysis of the csf and plasma metabolomes of rigorously matched amyotrophic lateral sclerosis, parkinson's disease and control subjects. *Molecular BioSystems*, 12(4):1287–1298, 2016.

[72] Yin Zhang and Liyan Gao. On numerical solution of the maximum volume ellipsoid problem. *SIAM Journal on Optimization*, 14(1):53–76, 2003.

[73] Yair Zlotnik, Yacov Balash, Amos D Korczyn, Nir Giladi, and Tanya Gurevich. Disorders of the oral cavity in parkinsonâs disease and parkinsonian syndromes. *Parkinsonâs disease*, 2015, 2015.

Appendix

Gap ratio algorithm

```
import os
import pandas as pd
import scipy.spatial
from scipy.spatial import distance_matrix,distance
import numpy as np
from sklearn.decomposition import PCA
from sklearn.metrics.pairwise import euclidean_distances
import matplotlib.pyplot as plt
from sklearn import preprocessing
from collections import Counter



os.chdir('C:\\Users\\Padma\\Desktop\\Master Thesis\\16x\\16x')
os.getcwd()

#datasets
complex1_8x = pd.read_csv("c1_8.csv",delimiter=',')
complex5_8x = pd.read_csv("c5_8.csv",delimiter=',')
healthy_8x = pd.read_csv("H_8.csv",delimiter=',')

#selecting the dataset
data = complex5_16x

df = pd.DataFrame(data.iloc[:,:])
```

```
dist_mat_ = distance_matrix(df.values,df.values)
df_dist = pd.DataFrame(dist_mat_) #distance matrix


#plot
for val in data.index:
    print(val)
    v_b = data.iloc[val,:]
    fa,binns,patches = plt.hist(v_b, bins=100,align='left')
    #plt.legend(name)
    u = fa
    xp = binns[0:100]
    fig = plt.plot(xp,u,'r')
    plt.plot()
    plt.show()


#gap ratio algorithm implementations
gap_ratios = []
for kp in range(len(df.iloc[:,1:])-1):
    print(kp)
    p = 0
    for i in range(len(df.iloc[:,1:])-1):
        p_ = p +1
        rp = ((df.iloc[kp,p]-df.iloc[kp,p_])/2)
        Rp = abs(distance.minkowski(df.iloc[0,p],df.iloc[0,p_], p = float('inf')))
        gap_ratio = rp/Rp
        p = p_
        gap_ratios.append("For index {0} gap ratio is {1}".format(kp,gap_ratio))
samples_for_dim = [gap_ratios[samp:samp+1787] for samp in range(0,len(gap_ratios),1787)]
uniform_split_index = [gapr.split(' ')[3] for gapr in samples_for_dim[1]]
```

```
distribution_type = []
for ik in range(len(samples_for_dim)):
    print(ik)
    uniform_split = [gapr.split(' ')[6] for gapr in samples_for_dim[ik]]
    epp = []
    for o in range(len(uniform_split)):
        ep = float(uniform_split[o])
        nu = float(2 * ep)
        de = float(1 + float(ep))
        if de == 0:
            epsilon = 0
        else:
            epsilon = float(nu) / float(de)
        if (ep <= epsilon <= 2):
            epp.append("True")
        else:
            epp.append("False")
    false = []
    true = []
    for k in range(len(epp)):
        false_ = 0
        true_ = 0
        if epp[k] == "False":
            false.append(false_+1)
        else:
            true.append(true_+1)
    if len(false) < len(true):
        print("uniform distribution")
        distribution_type.append("uniform distribution")
```

```
    else:
        print("non-uniform distribution")
        distribution_type.append("non-uniform distribution")
```

Classification algorithm

```
reactions = pd.read_csv("rxns.csv", delimiter = ",")
complex_5 = pd.read_csv("S_C5.csv", delimiter = ",")
control = pd.read_csv("S_iNESC2DN.csv", delimiter = ",")
complex1 = pd.read_csv("S_C1.csv",delimiter=",")

react = np.array(reactions.iloc[:,0])
c5 = complex_5
c5.set_index(react,inplace=True)
#c5.to_csv("c5.csv",sep = ',')

ctrl = control
ctrl.set_index(react,inplace=True)
#ctrl.to_csv("ctrl.csv",sep=',')

complex1.set_index(react,inplace=True)
#complex1.to_csv("c1.csv",sep=",")
#np.where(complex1[complex1.index == 'EX_t'])

matching = [s for s in react if "EX_" in s]

index_match_c1 = []
for i in range(len(complex1.index)):
```

```
    rea = complex1.index[i]
    for k in range(len(matching)):
        if np.all(rea == matching[k]):
            index_match_c1.append(i)
exchange_react_c1 = complex1.iloc[index_match_c1,:]

index_match_ctrl = []
for i in range(len(ctrl.index)):
    rea = ctrl.index[i]
    for k in range(len(matching)):
        if np.all(rea == matching[k]):
            index_match_ctrl.append(i)
exchange_react_ctrl = ctrl.iloc[index_match_ctrl,:]

index_match_c5 = []
for i in range(len(c5.index)):
    rea = c5.index[i]
    for k in range(len(matching)):
        if np.all(rea == matching[k]):
            index_match_c5.append(i)
exchange_react_c5 = c5.iloc[index_match_c5,:]

dataset = exchange_react_ctrl

#ks test for complex 5
reject_null_hyp_ks = []
accept_null_hyp_ks = []
accept = []
pval_ = []
```

```
for ks in range(len(dataset.index)):
    print(ks)
    data = dataset.iloc[ks,:]
    ks_,pval = stats.kstest(data, 'norm', args = (np.mean(data), np.std(data)))
    pval_.append(pval)
    #print(ks_)
    if np.any(pval < 0.000001):#ctrl=0.005,c1=0.000001,c5=0.00000005
        reject_null_hyp_ks.append("for {} reject null hypothesis".format(ks))
    else:
        accept_null_hyp_ks.append("for {} accept null hypothesis".format(ks))
        accept.append("{}".format(ks))

accept_ = []
accept_index = []
zero_dimension = []
zero_dimension_index = []
for i in accept:
    ip = int(i)
    k = pval_[ip]
    if np.all(math.isnan(k)):
        print(dataset.index[ip])
        zero_dimension.append(dataset.index[ip])
        zero_dimension_index.append(ip)
    else:
        print("{}:{}".format(ip,k))
        accept_.append(dataset.index[ip])
        accept_index.append(ip)

for i in accept_index:
```

```
    k = dataset.iloc[i,:]
    plt.hist(k,bins=100)
    plt.show()


c5_without_zero =  dataset.drop(index=zero_dimension)
c5_zero = dataset.iloc[zero_dimension_index,:]
c5_without_normal =  c5_without_zero.drop(index=accept_)
c5_normal = dataset.iloc[accept_index,:]


#truncated test
x_bar_c5 = []
for val in c5_without_normal.index:
    print(val)
    norm_cdf = scipy.stats.norm.cdf(c5_without_normal.loc[val,:],loc=0,scale=1) # calculate the cdf - also
discrete
    norm_cdf_ = (sum(norm_cdf))
    norm_len = 1/len(c5_without_normal.columns)
    num_ = norm_len*norm_cdf_
    x_bar_c5.append(num_)


uniform_c5 = []
for i in range(len(c5_without_normal.index)):
    critt = scipy.stats.norm.ppf(q =
0.05,loc=np.mean(c5_without_normal.iloc[i,:]),scale=np.std(c5_without_normal.iloc[i,:]))
    z_alpha = critt/np.sqrt(len(c5_without_normal.columns))
    if z_alpha < x_bar_c5[i]:
        if x_bar_c5[i] >= 0.5:
            print("index {} : left skewed".format(i))
            uniform_c5.append("index {} : left skewed".format(i))
```

```
        else:
            print("index {} : right skewed".format(i))
            uniform_c5.append("index {} : right skewed".format(i))
    else:
        print("index {} : normal".format(i))
        uniform_c5.append("index {} : normal".format(i))
#skewess
skewed = []
for i in c5_without_normal.index:
    print(i)
# generate univariate observations
    data = c5_without_normal.loc[i,:]
    # normality test
    stat= stats.skew(data)
    #print('Statistics=%.3f, p=%.3f' % (stat, p))
    statss= stats.norm.ppf(0.05, loc=np.mean(data), scale=np.std(data))
    diff = stat - statss
    #print('Statistics=%.3f' % (statss))
    skewed.append("{0} : {1}".format(i, stat))

t_skew = [lname.split(':')[1] for lname in skewed]
t_skew_ind = [iname.split(':')[0] for iname in skewed]

skew = []
for s in range(len(c5_without_normal.index)):
    k = float(t_skew[s])
    indk = t_skew_ind[s]
    if k > 0 :
        print("index {} : left skewed".format(indk))
```

```
        skew.append("index {} : left skewed".format(indk))
    elif k == 0:
        print("index {} : normal".format(indk))
        skew.append("index {} : normal".format(indk))
    else:
        print("index {} : right skewed".format(indk))
        skew.append("index {} : right skewed".format(indk))

uniform_splitc5 = [unam.split(':')[1] for unam in uniform_c5]
skew_splitc5 = [sname.split(':')[1] for sname in skew]

algorithm_comp = []
list_true = []
list_false_uc5 = []
list_false_sc5 = []
algo_index = []
false_index = []
true_index = []
for ji in range(len(uniform_splitc5)):
    a = uniform_splitc5[ji]
    print(a)
    b = skew_splitc5[ji]
    print(b)
    if np.any(a == b):
        print("TRUE")
        algorithm_comp.append("TRUE")
        algo_index.append("TRUE for {}".format(ji))
        true_index.append("{}".format(ji))
        list_true.append(uniform_splitc5[ji])
```

```
    else:
        print("FALSE")
        algorithm_comp.append("FALSE")
        algo_index.append("FALSE for {}".format(ji))
        false_index.append("{}".format(ji))
        list_false_uc5.append(uniform_splitc5[ji])
        list_false_sc5.append(skew_splitc5[ji])


zero_list = ["zero distributions" for x in range(len(zero_dimension_index))]
normal_list = ["normal distribitions" for x in range(len(accept_index))]


zero_dimensions_data = dataset.iloc[zero_dimension_index,:]
zero_dimensions_data["type"] = zero_list
normal_dimensions_data = dataset.iloc[accept_index,:]
normal_dimensions_data["type"] = normal_list
true_index_dimensions = c5_without_normal.iloc[np.array(true_index).astype(np.float),:]
true_index_dimensions["type"] = list_true
false_index_dimensions = c5_without_normal.iloc[np.array(false_index).astype(np.float),:]
false_index_dimensions["type_uni"] = list_false_uc5
false_index_dimensions["type_skew"] = list_false_sc5
classified_c5 = pd.concat([c5_zero,c5_normal,true_index_dimensions,false_index_dimensions])
classified_c5.to_csv("classified_ctrl_exchange.csv",sep=",")
```

Comparison

```
classified_c1 = pd.read_csv("classified_c1_exchange.csv",delimiter = ',')
classified_c5 = pd.read_csv("classified_c5_exchange.csv",delimiter = ',')
classified_ctrl = pd.read_csv("classified_ctrl_exchange.csv",delimiter = ',')

concate_c1ctrldf = pd.concat([classified_c1,classified_ctrl], ignore_index=True).fillna(0)
div = [rec_df for rec,rec_df in concate_c1ctrldf.groupby('Unnamed: 0')]
#comparing the means for fold changes
k_c1_ctrl = []
k_c1_ctrl_index = []
mean_c1_ctrl = []
mean_c1_ = []
mean_ctrl_ = []
log_2cha = []
for u in range(len(div)):
    c1 = div[u].iloc[0,1:3497]
    ctrl = div[u].iloc[1,1:3497]
    mean_c1 = np.mean(c1)
    mean_ctrl = np.mean(ctrl)
    dst = mean_c1/mean_ctrl
    log_2fold = abs(np.log2(dst))
    if (mean_c1 == 0 or mean_ctrl == 0):
        mean_c1_ctrl.append(0)
        log_2cha.append(0)
    else:
        mean_c1_ctrl.append(dst)
```

```
        log_2cha.append(log_2fold)
    #mean_c1_ctrl.append(dst)
    mean_c1_.append(mean_c1)
    mean_ctrl_.append(mean_ctrl)
    k_c1_ctrl.append(np.unique(div[u].iloc[:,0]))
    k_c1_ctrl_index.append(u)

df_k_c1_ctrl = pd.DataFrame({'index':k_c1_ctrl_index,
                             'name': k_c1_ctrl,
                             'fold_change' : mean_c1_ctrl,
                             'mean_c5':mean_c1_,
                             'mean_ctrl':mean_ctrl_,
                             'log2_FC_c5':log_2cha}).fillna(0)

k_gh = df_k_c1_ctrl.sort_values(by='fold_change',ascending=True)
#df_k_c1_ctrl.to_csv("sort_c5_ctrl_std.csv",sep=',')
k_il = k_gh['index'].tolist()

#equality of distributions
eq_eq_zero = []
eq_eq_t_zero = []
eq_eq = []
not_eq = []
not_eq_trun_not_eq = []
not_eq_skew_eq = []
not_eq_skew_not_eq = []
not_eq_trun_eq = []
not_eq_t = []
eq_eq_not_t_zero = []
```

```
not_eq_trun_eq_t = []
not_eq_trun_not_eq_t = []
not_eq_skew_not_eq_t = []
not_eq_skew_eq_t = []
for le in range(len(k_il)):
    inde = int(k_il[le])
    if np.any(div[inde].iloc[1,3497] == div[inde].iloc[0,3497] == 0):
        print(inde)
        #print(div[inde].iloc[:,0])
        eq_eq_zero.append(inde)
        if np.any(div[inde].iloc[1,3498] == div[inde].iloc[0,3498]):
            #print(inde)
            eq_eq_t_zero.append(inde)
            continue
        elif np.any(div[inde].iloc[1,3498] != div[inde].iloc[0,3498]):
            #print(inde)
            #print(div[inde].iloc[0,3498])
            eq_eq_not_t_zero.append(inde)
            continue
    elif np.any(div[inde].iloc[1,3497] == div[inde].iloc[0,3497] != 0):
        #print(inde)
        eq_eq.append(inde)
    elif np.any(div[inde].iloc[1,3497] != div[inde].iloc[0,3497] == 0):
        not_eq_t.append(inde)
        if np.any(div[inde].iloc[1,3498] == div[inde].iloc[0,3498]):
            #print(inde)
            not_eq_trun_eq.append(inde)
            continue
        elif np.any(div[inde].iloc[1,3498] != div[inde].iloc[0,3498]):
```

```
            #print(inde)
            not_eq_trun_not_eq.append(inde)
            continue
            if np.any(div[inde].iloc[1,3499] != div[inde].iloc[0,3499]):
                not_eq_skew_not_eq.append(inde)
                continue
            elif np.any(div[inde].iloc[1,3499] == div[inde].iloc[0,3499]):
                not_eq_skew_eq.append(inde)
    elif np.any(div[inde].iloc[1,3497] != div[inde].iloc[0,3497] != 0):
        not_eq.append(inde)
        if np.any(div[inde].iloc[1,3498] == div[inde].iloc[0,3498]):
            not_eq_trun_eq_t.append(inde)
            continue
        elif np.any(div[inde].iloc[1,3498] != div[inde].iloc[0,3498]):
            not_eq_trun_not_eq_t.append(inde)
            continue
            if np.any(div[inde].iloc[1,3499] != div[inde].iloc[0,3499]):
                not_eq_skew_not_eq_t.append(inde)
                continue
            elif np.any(div[inde].iloc[1,3499] == div[inde].iloc[0,3499]):
                not_eq_skew_eq_t.append(inde)

zero = df_k_c1_ctrl.iloc[eq_eq,:].set_index('name')
zero.to_csv("sort_c1_ctrl_std.csv",sep=',')

c1_ctrl = pd.read_csv("sort_c1_ctrl_zerostd.csv",delimiter = ',',index_col='name')
c1_the_ = c1_ctrl.sort_values(by=['fold_change'],ascending = False)
c1_the = c1_the_[c1_the_['fold_change'] > 1]
c1_log = c1_the[c1_the['log2_FC'] < 0.5]
```

```
c5_ctrl = pd.read_csv("sort_c5_ctrl_zerostd.csv",delimiter=',',index_col='name')
c5_the_ = c5_ctrl.sort_values(by=['fold_change'],ascending = True)
c5_the = c5_ctrl[c5_ctrl['fold_change'] > 1.00005]
c5_log = c5_the[c5_the['log2_FC'] < 0.5]

c1_c5_ctrl = c1_the_.merge(c5_the_,how='outer',left_index=True,right_index=True).fillna(0)
c1_c5_ctrl= c1_c5_ctrl.drop(columns = ['index_x','index_y'])

df_norm_row= c1_c5_ctrl.iloc[:,[0,2,4,6,7]].sub(c1_c5_ctrl.iloc[:,[0,2,4,6,7]].mean(axis=1), axis=0)
# 2: divide by standard dev
df_norm_row=df_norm_row.div(c1_c5_ctrl.iloc[:,[0,2,4,6,7]].std(axis=1), axis=0 )
plt.figure(figsize=(5,15))
plt.xticks(rotation=0)
plt.yticks(rotation=90)
sns_ = sns.heatmap(df_norm_row,annot=False,cmap="YlGnBu",yticklabels='auto',cbar=False,
                   cbar_kws=dict(use_gridspec=False,location="top"))
plt.colorbar(sns_.get_children()[0], orientation = 'horizontal')
# locate colorbar ticks
plt.cax.xaxis.set_ticks_position('top')
fig = sns_.get_figure()
fig.savefig("compare_c15.pdf")
```