1. Multiple Substrates and Inhibitor Models

In this lecture, we continue our discussion of enzyme kinetics. Today we will:

- Discuss competitive and non-competitive inhibitor models
- Introduce phenomenological and mechanistic multi-substrate enzyme kinetics models
- Incorporating different levels of information into the bounds constraints

2. Competitive, non- and uncompetitive inhibitors

Inhibitors are critical research and therapeutic tools; they are often a go-to strategy for manipulating biological function (and are widely used in the allosteric regulation context).

- Clinical application: Literature search on small-molecule checkpoint inhibitors

Let's revisit the previous feedback inhibition examples, getting more granular about the type of inhibition (and inhibitor) involved. There are three types of inhibitors:

- **Non-competitive inhibition**: A non-competitive inhibitor reduces the enzyme's activity and binds equally well to the enzyme whether or not it has already bound the substrate. This type of inhibition reduces the maximum rate of a chemical reaction without changing the apparent binding affinity of the catalyst for the substrate. No chemistry is possible for the E:I or E:S:I complexes. Derivation for non-competitive inhibition here can be found here.
- **Competitive inhibition**: A competitive inhibitor prevents the binding of the substrate to the enzyme. In competitive inhibition, the maximum reaction velocity V_{max} is unchanged. At the same time, the apparent affinity of the substrate to the binding site is decreased (decreasing affinity leads to increased K_m). Derivation for competitive inhibition can be found here.
- **Uncompetitive inhibition**: An uncompetitive inhibitor binds only to the enzyme-substrate complex. Uncompetitive inhibition typically occurs in reactions with two or more substrates or products. Both V_{max} and K_m decreased by the inhibitor. Derivation for uncompetitive inhibition can be found here.

Example: Non-competitive inhibition. The form for an enzyme catalyzed reaction of a single substrate S to product P by enzyme E in the presense of a non-compeitive inhibitor I is given by:

$$v = V_{max}^{app}\left(\frac{S}{K_m + S}\right)$$

where the *appearant* maximum reaction velocity is given by:

$$V_{max}^{app} = \frac{k_{cat}E}{1 + I/K_I}$$

where K_I denotes an inhibition constant (units: Inhibitor concentration), and k_{cat} denotes the turnover number for the enzyme of interest (units: 1/time).

Example: Competitive inhibition The form for an enzyme-catalyzed reaction of a single substrate S to a product P by enzyme E in the presence of a competitive inhibitor I is given by:

$$v = V_{max}\left(\frac{S}{K_m^{app} + S}\right)$$

where the *appearant* saturation constant is given by:

$$K_m^{app} = K_m(1 + I/K_I)$$

where K_I denotes an inhibition constant (units: Inhibitor concentration).

```
let

    # setup some parameters -
    kcat = 13.7 # units s^-1
    Eₒ = 2.0    # units: *mol/L
    Km = 5.0    # units: *mol/L
    KI = 20.0   # units: *mol/L
    n = 10.0

    # set the substrate and inhibitor concentration,
    and initialize some space -
    I_array = range(0.0,step = 10,stop = 50.0) |>
    collect
    S_array = range(0.0,step = 0.01,stop = 100.0) |>
    collect
    Nₛ = length(S_array);
    Nᵢ = length(I_array)

    # initialize -
    v_array = Array{Float64, 2}(undef, Nₛ, Nᵢ)

    # main loop -
    for (i,S) ∈ enumerate(S_array)
        for (j,Iₒ) ∈ enumerate(I_array)

            # compute the apparent Vmax -
            Vmax = (kcat*Eₒ)
            Km_app = Km*(1+Iₒ/KI)
            sat_term = (S^n)/(Km_app^n+S^n)
            v_array[i,j] = Vmax*sat_term
        end
    end

    # plots -
    for (j,Iₒ) ∈ enumerate(I_array)
        if (j == 1)
            plot(S_array,v_array[:,j], c=:blue,lw=2,
            label="I = $(Iₒ) mmol/L",
                legend=:bottomright)
        else
            plot!(S_array,v_array[:,j], c=:red,lw=2,
            label="I = $(Iₒ) mmol/L")
        end
    end

    xlabel!("Substrate S (mM)", fontsize=18)
    ylabel!("Rate (*mol/L-time)", fontsize=18)
end
```

2.1 Discrete state models of inhbitors

2.1.1 Theory

Suppose we model the rate v_j as the product of a kinetic limit (a simple model of the rate) and a correction term that accounts for the missing regulation:

$$v_j = r_j \theta(\dots)_j$$

where v_j denotes the overall rate (units: μM/time), r_j denotes the kinetic limit i.e., the maximum rate of conversion (units: μM/time) and $0 \leq \theta(\dots)_j \leq 1$ (units: dimensionless) is a control function that describes the influence of effector molecules.

Suppose an enzyme E can exits in one of $s = 1, 2 \dots, \mathcal{S}$ possible microstates, where each microstate s has some pseudo energy ϵ_s. Some microstates will lead to activity (the ability to carry out the chemical reactions, while others will not). For each microstate s, let's assign a pseudo energy ϵ_s, where by definition $\epsilon_1 = 0$; we assume the base state has the lowest energy. Next, suppose the probability that enzyme E is in microstate s follows a Boltzmann distribution which says:

$$p_i = \frac{1}{Z} \times f_i \exp\left(-\beta \epsilon_i\right) \qquad i = 1, 2, \dots, \mathcal{S}$$

where p_i denotes the probability that enzyme E is in microstate $i = 1, 2, \dots, \mathcal{S}$, f_i denotes a state-specific factor $f_i \in [0, 1]$, β denotes the thermodynamic beta and Z denotes a normalization factor (called the Partiton function in the statistical physics community). We can find Z using the summation law of discrete probolity e.g., $\sum_s p_s = 1$ which gives:

$$Z = \sum_{s=1}^{\mathcal{S}} f_i \exp\left(-\beta \epsilon_i\right)$$

which gives:

$$p_i = \frac{f_i \exp\left(-\beta \epsilon_i\right)}{\sum_{s=1}^{\mathcal{S}} f_i \exp\left(-\beta \epsilon_i\right)} \qquad i = 1, 2, \dots, \mathcal{S}$$

.

Finally, we relate the probability that enzyme E is in microstate s back to the θ control function by computing the overall probability that the desired event happens, e.g., enzyme E catalyzes the reaction of interests. We know if $\Omega = \{1, 2, \dots, \mathcal{S}\}$, then we can define the subset $\mathcal{A} \subseteq \Omega$ in which the desired event happens. Given $\mathcal{A}$, the θ function becomes:

$$\theta = \sum_{s \in \mathcal{A}} p_s$$

2.1.2 A DSM model for non-competitive inhibition

Enzyme E, inhibited by a non-competitive inhibitor I, can exist in one of four possible microstates:

- **s = 1**: No substrate S is bound. No chemical reaction is possible.
- **s = 2**: Substrate S is bound to enzyme E, but inhibitor I is not bound. A chemical reaction is possible.
- **s = 3**: Both the substrate S and inhibitor I are bound to enzyme E. No chemical reaction is possible
- **s = 4**: Inhibitor I is bound to enzyme E. No chemical reaction is possible

There are four possible states $\Omega = \{1, 2, 3, 4\}$, while the subset of states resulting in a chemical reaction $\mathcal{A} \subseteq \Omega$ is given by $\mathcal{A} = \{2\}$. Thus, the θ funtion is given by:

$$\theta = p_2$$

or:

$$\theta = \frac{f_2 \exp\left(-\beta \epsilon_2\right)}{\sum_{i=1}^{\mathcal{S}} f_i \exp\left(-\beta \epsilon_i\right)}$$

```
let

    # setup constants -
```

```
kcat = 13./              # units: sec^-1
Eₒ = 2.0                 # units: *mol/L
R = 8.314                # units: J/mol-K
T = (273.15 + 25.0)      # units: K
β = (1/(R*T))            # units: mol/J
K₂ = 25.0                # units: mmol/L
K₃ = 10.0                # units: mmol/L
K₄ = 20.0                # units: mmol/L
n₂ = 1.0                 # units: dimensionless
n₃ = n₂                  # units: dimensionless
n₄ = 1.0                 # units: dimensionless

# setup ϵ array -
ϵ_array = [
    0.0         ; # 1 s₁ units: J/mol
    -5000.0     ; # 2 s₂ units: J/mol
    -1000.0     ; # 3 s₃ units: J/mol
    -1000.0     ; # 4 s₄ units: J/mol
];

# compute the W_array -
W_array = exp.(-β*ϵ_array)

# set the inhibitor concentration, and initialize
some space -
I_array = range(0.0,step = 10,stop = 50.0) |>
collect
S_array = range(0.0,step = 0.01,stop = 100.0) |>
collect
Nₛ = length(S_array);
Nᵢ = length(I_array)

# initialize -
θ_array = Array{Float64, 2}(undef, Nₛ, Nᵢ)

# main loop -
for (i,S) ∈ enumerate(S_array)
    for (j,Iₒ) ∈ enumerate(I_array)

        # compute fᵢ -
        f₄ = ((Iₒ/K₄)^(n₄))/(1 + (Iₒ/K₄)^(n₄))
        f₃ = (Iₒ/K₄)*(((S/K₂)^(n₃))/(1 +
        (S/K₂)^(n₃)))
        f₂ = (((S/K₂)^(n₂))/(1 + (S/K₂)^(n₂)))
        f₁ = 1

        # populate the f array -
        f_array = [f₁,f₂,f₃,f₄];

        # compute θ -
        D = dot(f_array,W_array)
```

```
            N = f₂*W_array[2]
            θ_array[i,j] = kcat*Eₒ*(N/D)
        end
    end

    # plots -
    for (j,Iₒ) ∈ enumerate(I_array)
        if (j == 1)
            plot(S_array,θ_array[:,j], c=:blue,lw=2,
            label="I = $(Iₒ) mmol/L",
                legend=:bottomright)
        else
            plot!(S_array,θ_array[:,j], c=:red,lw=2,
            label="I = $(Iₒ) mmol/L")
        end
    end

    xlabel!("Substrate S (mM)", fontsize=18)
    ylabel!("Rate (*mol/L-time)", fontsize=18)
end
```

2.1.3 A DSM model for competitive inhibition

Enzyme E, inhibited by a competitive inhibitor I, can exist in one of three possible microstates:

- **s = 1**: No substrate S is bound. No chemical reaction is possible.
- **s = 2**: Substrate S is bound to enzyme E, but inhibitor I is not bound. A chemical reaction is possible.
- **s = 3**: Inhibitor I is bound to enzyme E. No chemical reaction is possible

There are three possible states $\Omega = \{1, 2, 3\}$, while the subset of states resulting in reaction $\mathcal{A} \subseteq \Omega$ is given by $\mathcal{A} = \{2\}$. Thus, the θ funtion is given by:

$$\theta = p_2$$

or:

$$\theta = \frac{f_2 \exp\left(-\beta\epsilon_2\right)}{\sum_{i=1}^{\mathcal{S}} f_i \exp\left(-\beta\epsilon_i\right)}$$

```
let

    # setup constants -
    kcat = 13.7             # units: sec^-1
    Eₒ = 2.0                # units: *mol/L
    R = 8.314               # units: J/mol-K
    T = (273.15 + 25.0)     # units: K
    β = (1/(R*T))           # units: mol/J
    K₂ = 10.0               # units: mmol/L
    K₃ = 5.0                # units: mmol/L
    n₂ = 10.0               # units: dimensionless
    n₃ = 1.0                # units: dimensionless

    # setup ϵ array -
    ϵ_array = [
        0.0         ; # 1 s₁ units: J/mol
        -4000.0     ; # 2 s₂ units: J/mol
        -100.0      ; # 3 s₃ units: J/mol
    ];

    # compute the W_array -
    W_array = exp.(-β*ϵ_array)

    # set the inhibitor concentration, and initialize
    some space -
    I_array = range(0.0,step = 10,stop = 50.0) |>
    collect
    S_array = range(0.0,step = 0.01,stop = 100.0) |>
    collect
    Nₛ = length(S_array);
    Nᵢ = length(I_array)

    # initialize -
    θ_array = Array{Float64, 2}(undef, Nₛ, Nᵢ)

    # main loop -
    for (i,S) ∈ enumerate(S_array)
        for (j,Iₒ) ∈ enumerate(I_array)
```

3. Kinetics of multiple substrate reactions

3.1 Power-law kinetics and biochemical systems theory (BST)

Power-law kinetics are a flexible tool to describe multiple substrate kinetics. Suppose reaction v_i depends upon $j = 1, 2, \ldots, \mathcal{F}$ factors. These factors can be concentration e.g., substrates or products well as other type of data e.g., categorical data:

$$v_i = \alpha_i \prod_{j=1}^{\mathcal{F}} X_j^{f_{ij}} \qquad i = 1, 2, \ldots, \mathcal{R}$$

where α_j denotes the rate constant for reaction j, X_j denotes the abundance of factor j and f_{ij} denotes the kinetic order of factor j in reaction j. Power-law kinetics are a prominent feature of Biochemical Systems Theory (BST). Biochemical systems theory has been developed since the 1960s by Michael Savageau, Eberhard Voit, and others for the systems analysis of biochemical processes:

- Atkinson MR, Savageau MA, Myers JT, Ninfa AJ. Development of genetic circuitry exhibiting toggle switch or oscillatory behavior in Escherichia coli. Cell. 2003 May 30;113(5):597-607. doi: 10.1016/s0092-8674(03)00346-5. PMID: 12787501.
- Alvarez-Vasquez F, Sims KJ, Cowart LA, Okamoto Y, Voit EO, Hannun YA. Simulation and validation of modeled sphingolipid metabolism in Saccharomyces cerevisiae. Nature. 2005 Jan 27;433(7024):425-30. doi: 10.1038/nature03232. PMID: 15674294.

- Goel G, Chou IC, Voit EO. Biological systems modeling and analysis: a biomolecular technique of the twenty-first century. J Biomol Tech. 2006;17(4):252-269.

3.2 General multisubstrate kinetics

Suppose the irreversible rate v_i is dependent upon susbtrates $S_j, j = 1, 2, \ldots, \mathcal{S}$, then the multiple saturation kinetic form is given by:

$$v_i = V_{max,i} \left[\frac{\prod_j \frac{S_j}{K_j}}{\prod_j \left(1 + \frac{S_j}{K_j}\right) - 1} \right] \qquad i = 1, 2, \ldots, \mathcal{R}$$

where $V_{max,i}$ denote the maximum reaction rate (units: concentration/time), S_j denotes the substrate concentration (units: concentration) and K_j denotes the saturation constant for substrate j.

- Liebermeister W, Klipp E. Bringing metabolic networks to life: convenience rate law and thermodynamic constraints. Theor Biol Med Model. 2006;3:41. Published 2006 Dec 15. doi:10.1186/1742-4682-3-41

3.3 Ping–pong, sequential- and random-order mechanistic models

Multi substrate enzyme kinetics models can be broadly broken down into classes:

- **Ping-pong mechanisms**: In a ping-pong mechanism, also called a double-displacement reaction, the enzyme is transformed into an intermediate state where the first substrate to product reaction occurs, followed by a second reaction. Enzymes with ping–pong mechanisms include oxidoreductases, transferases, and serine proteases such as trypsin, chymotrypsin and several enzymes of the blood clotting cascade.
- **Random and sequential order mechanisms**: In a random or sequential order model both substrates bind to the enzyme before the chemistry can occur. Products can be released in

random or sequential order.

4. Kinetics and flux balance analysis calculations

Kinetic rate equations (and the associated regulation of enzyme activity, enzyme state, and overall biochemical state) are included in the flux balance analysis calculation through the flux bounds equation. There are different levels of information included in the bounds. These levels are partitioned (broadly) into metabolite and metabolite-free groups.

4.1 General bounds model

Flux bounds constrain the values that each reaction in a metabolic network can take. A general model for these bounds is given by:

$$-\delta_j \left[V^{\circ}_{max,j} \left(\frac{e_j}{e^{\circ}} \right) \theta_j(\ldots) f_j(\ldots) \right] \leq v_j \leq \left[V^{\circ}_{max,j} \left(\frac{e_j}{e^{\circ}} \right) \theta_j(\ldots) f_j($$

where $V^{\circ}_{max,j}$ denotes the maximum reaction velocity (units: flux) computed at some characteristic enzyme abundance (units: concentration), the ratio e/e° is a correction for enzyme abundance (units: dimensioness), $\theta_j(\ldots) \in [0, 1]$ is the fraction of maximial enzyme activity (a function or measurement producing units: dimensionless), and $f_j(\ldots)$ is a function describing the substrate dependence of the reaction rate j (units: dimensionless). Both $\theta_j(\ldots)$ and $f_j(\ldots)$ could have associated parameters, e.g., saturation or binding constants, etc. Finally, the quanity $\delta_j \in \{0, 1\}$ is a *binary* variable:

- If reaction j is **reversible** $\delta_j = 1$ or,
- If reaction j is **irreversible** $\delta_j = 0$

4.2 Metabolite free bounds models

In metabolite-free models, we, as the simulation engineer, do not have access to any information about the concentrations of the substrates that appear in the various form of the kinetics. In these cases, $f_j(\ldots) = 1, \forall j$ and $\theta_j = 1, \forall j$ which gives bounds of the form:

$$-\delta_j\left[V^{\circ}_{max,j}\left(\frac{e_j}{e^{\circ}}\right)\right] \leq v_j \leq \left[V^{\circ}_{max,j}\left(\frac{e_j}{e^{\circ}}\right)\right]$$

Finally, if we do not have access to enzyme abundance data, then $(e/e^{\circ}) = 1$ and $V^{\circ}_{max,j}$ is calculated using a *characteristic* enzyme abundance.

4.3 Metabolite-dependent bounds models

In this class of bounds, we assume that we have some (or full) information about the concentrations of the metabolites. Assuming general multi-substrate kinetics, the bounds would take the form (shown for irreversible):

$$0 \leq v_j \leq V^{\circ}_{max,i}\left(\frac{e_j}{e^{\circ}}\right)\theta_j(\cdots)\left[\frac{\prod_k \frac{S_j}{K_j}}{\prod_k\left(1+\frac{S_j}{K_j}\right)-1}\right]$$

where the products are carried out over the reactants of the reaction j. Note: we could have chosen a different form for the kinetics, e.g., a power-law formulation, and the choice of enzyme activity model is up to the simulation designer.

4.4 Sources of data for bounds models

- Chang A, Jeske L, Ulbrich S, Hofmann J, Koblitz J, Schomburg I, Neumann-Schaal M, Jahn D, Schomburg D. BRENDA, the ELIXIR core data resource in 2021: new developments and updates. Nucleic Acids Res. 2021 Jan 8;49(D1): D498-D508. doi: 10.1093/nar/gkaa1025. PMID: 33211880; PMCID: PMC7779020.
- Milo R, Jorgensen P, Moran U, Weber G, Springer M. BioNumbers–the database of key numbers in molecular and cell biology. Nucleic Acids Res. 2010 Jan;38(Database issue): D750-3. doi: 10.1093/nar/gkp889. Epub 2009 Oct 23. PMID: 19854939; PMCID: PMC2808940.
- Park JO, Rubin SA, Xu YF, Amador-Noguez D, Fan J, Shlomi T, Rabinowitz JD. Metabolite concentrations, fluxes and free energies imply efficient enzyme usage. Nat Chem Biol. 2016 Jul;12(7):482-9. doi: 10.1038/nchembio.2077. Epub 2016 May 2. PMID: 27159581; PMCID: PMC4912430.

5. Summary and Conclusions

In this lecture we:

- Discussed competitive and non-competitive inhibitor models
- Introduced phenomenological and mechanistic multi-substrate enzyme kinetics models
- Incorporated different levels of information into the bounds constraints

6. Next Time

In-class problem set in which we estimate the flux through the Urea cycle with different types of bounds constraints.

```
TableOfContents(title="📚 Table of Contents", indent=true, depth=5, aside=true)
```

```
begin

    # load -
    using PlutoUI
    using Plots
    using LinearAlgebra
    using CSV
    using DataFrames

    # setup paths -
    _PATH_TO_ROOT = pwd()
    _PATH_TO_DATA = joinpath(_PATH_TO_ROOT, "data")

    # show -
    nothing
end
```