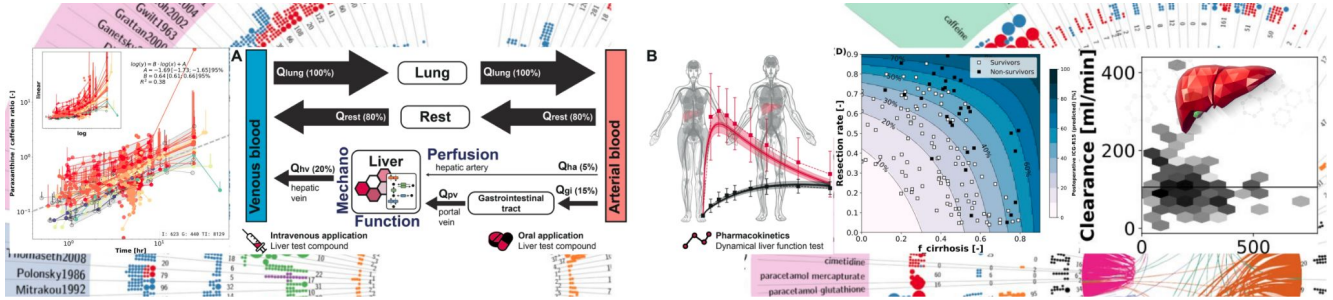


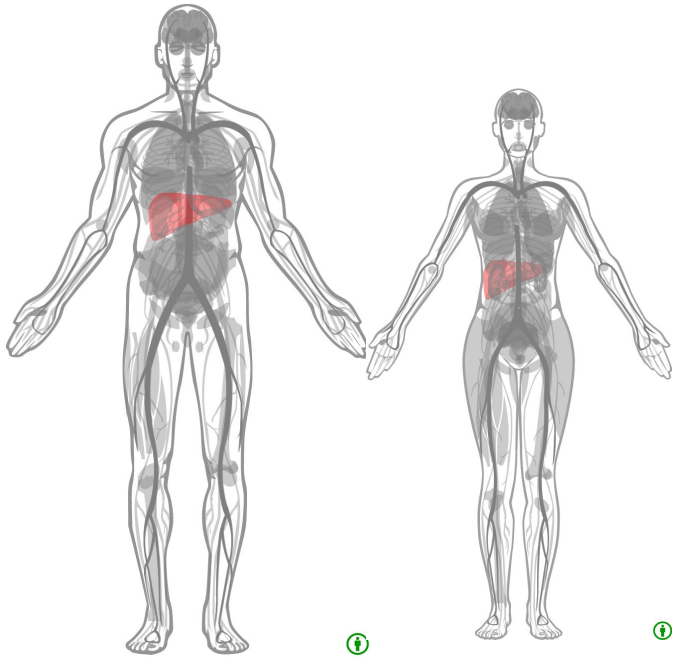
Pharmacokinetics Modelling Course: 1. Structural model, ODE, simple systems

Matthias König, MB19, SS2023
Humboldt-University Berlin, Systems Medicine of the Liver Lab
https://livermetabolism.com
konigmatt

Course summary

Pharmacokinetic modelling is the study of how drugs are absorbed, distributed, metabolized, and excreted in the body. The course on pharmacokinetic modelling will cover topics such as pharmacokinetic principles, drug distribution, clearance, and elimination, and the factors that affect these processes. Students will learn about different models used to describe pharmacokinetics, such as compartmental models and physiologically-based pharmacokinetic models, and how to use these models to predict drug concentrations and optimize dosing regimens. Other topics that might be covered include pharmacodynamics, drug-drug interactions, and the use of pharmacokinetic modelling in drug development and clinical practice. Overall, a course on pharmacokinetic modelling will provide students with a comprehensive understanding of the principles and techniques used to describe the movement of drugs through the body, and how this knowledge can be applied to improve drug therapy.

For more information see: https://livermetabolism.com.

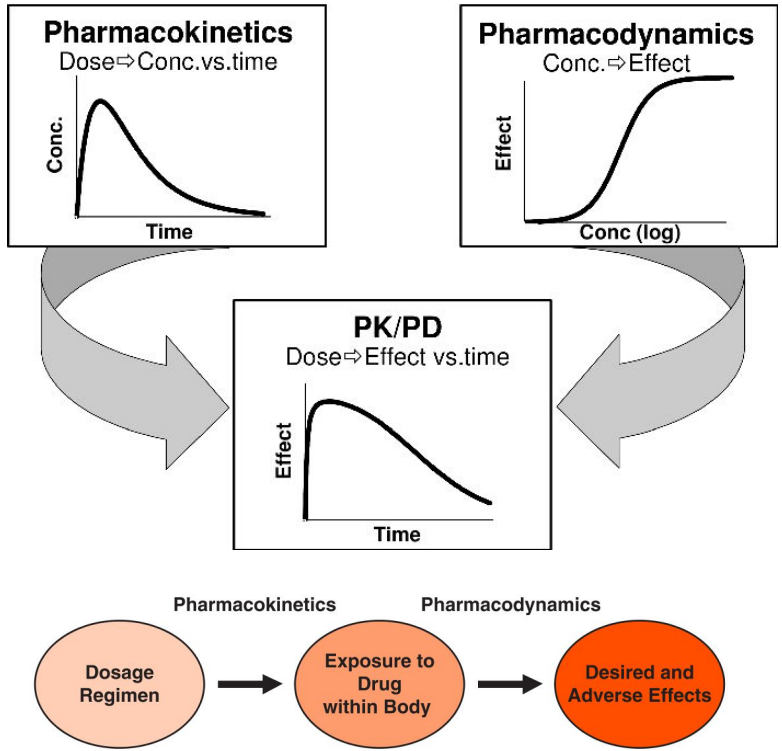


Requirements for passing

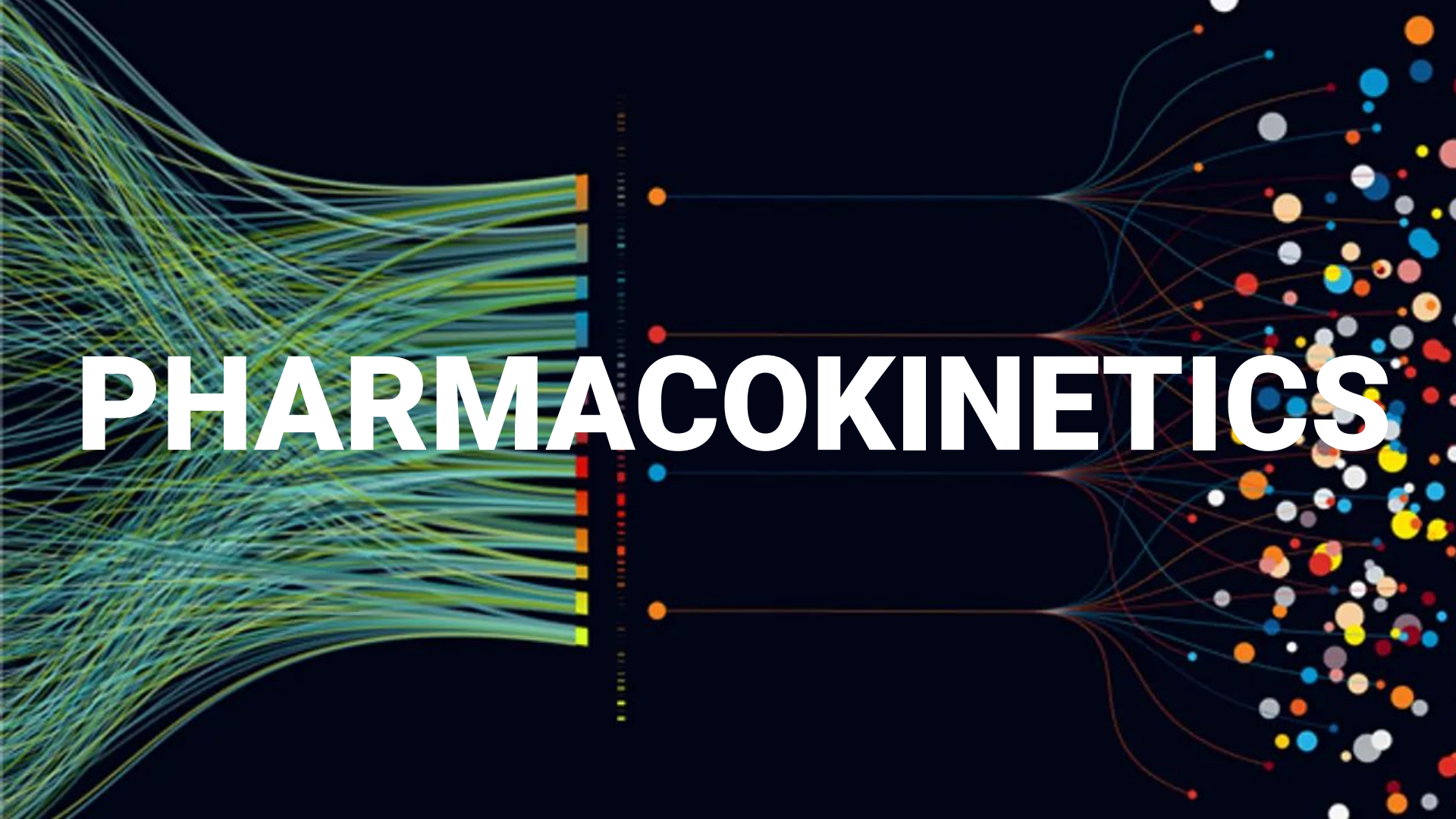
- subscribed to course list on moodle: https://moodle.hu-berlin.de/course/view.php?id=118385#section-2
- ≥ 3 courses on DataCamp
- participation in the course
- finish daily tasks

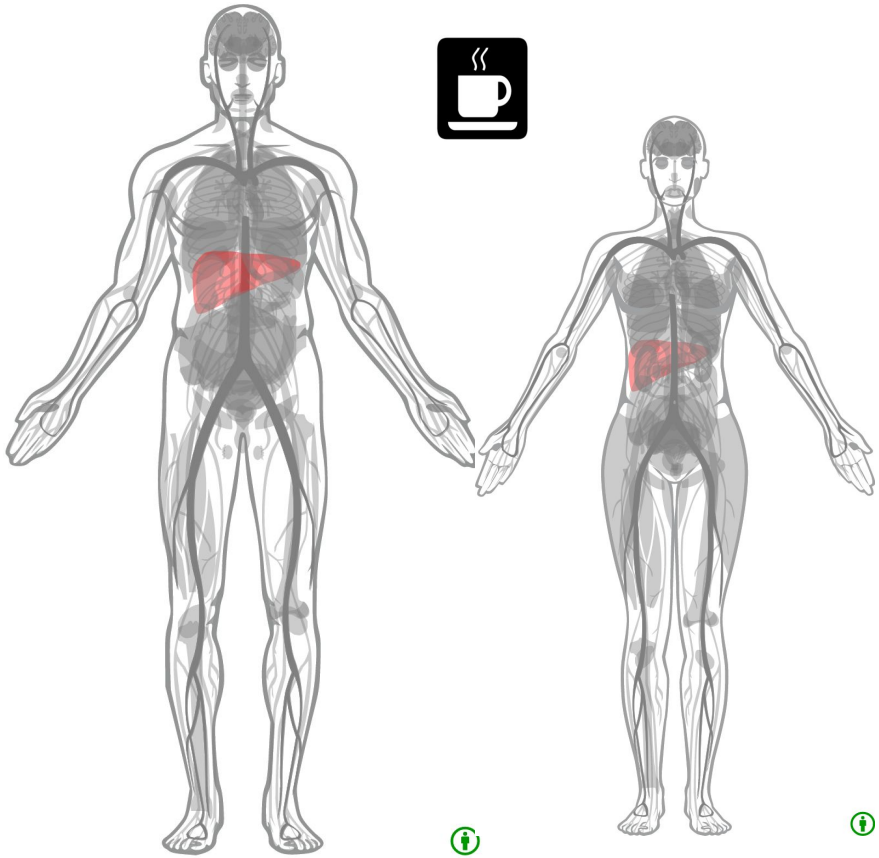
Pharmacokinetics (PK) & pharmacodynamics (PD)

- **Pharmacokinetics** is **what the body does to the drug**
  - study of the time course of drug absorption, distribution, metabolism, and excretion
  - **drug disposition**
- **Pharmacodynamics** is **what the drug does to the body**
  - **desired** (and adverse) **effects**

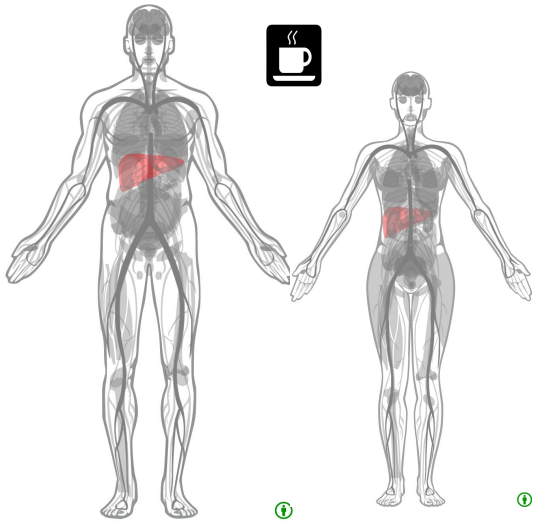


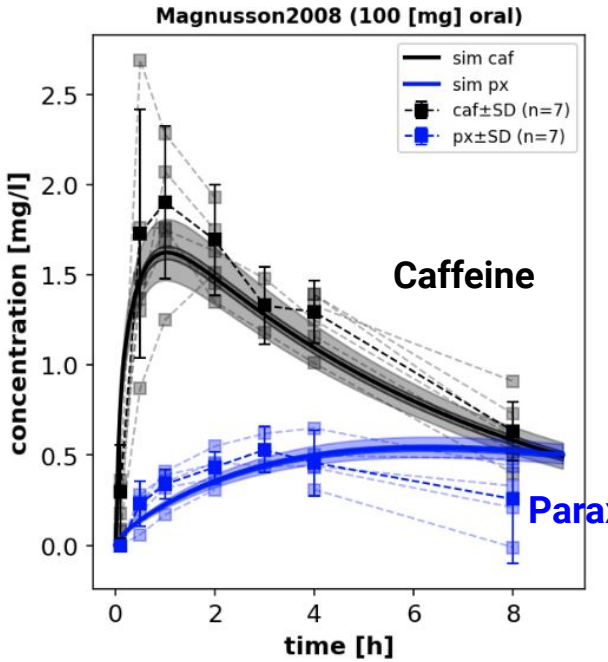
Tozer TN, Rowland M. **Essentials of pharmacokinetics and pharmacodynamics.** Second edition


PHARMACOKINETICS

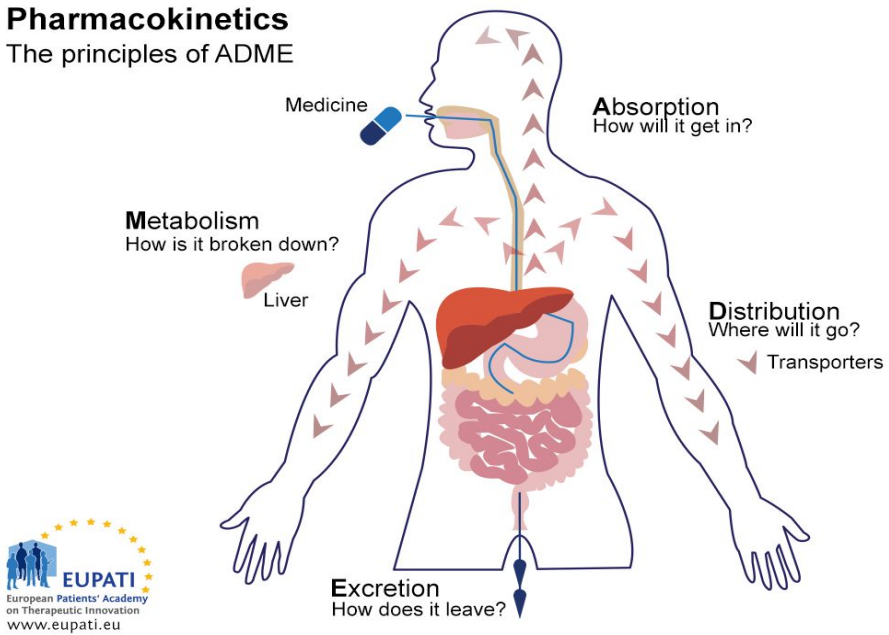


100 mg oral caffeine





ADME



ADME processes determine pharmacokinetics

- **A**bsorption
- **D**istribution
- **M**etabolization
- **E**limination

Pharmacokinetic parameter

- **C_{max}** : Maximal concentration
- **T_{max}** : time of maximal concentration
- **AUC** : area under the curve
- **k_{el}**: elimination rate
  fitting linear part of terminal phase (log)
- **$t_{1/2}$** : half-life (= ln2/k_{el})
  time for concentration to fall to half
- **Vd**: volume of distribution
  (= CL/k), dilution space
- **CL**: clearance (=Dose/AUC, =Dose/$C(0)_{extrapolated}$)

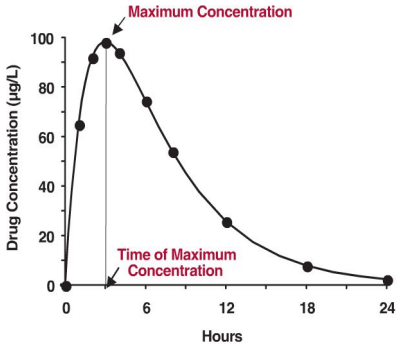


FIGURE 2-1. Drug concentration–time curve following a single oral dose showing the maximum systemic exposure (C_{max}) and the time of its occurrence (t_{max}). The concentration could represent drug in whole blood, plasma, or serum.

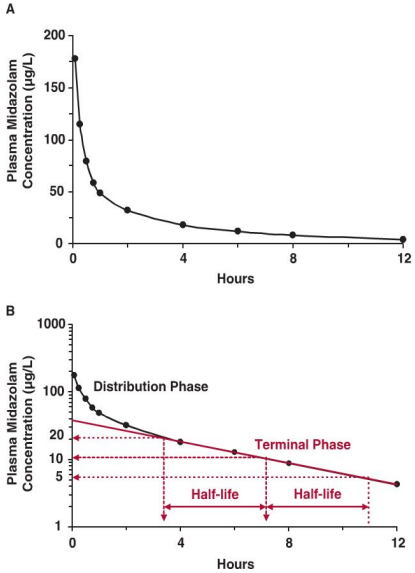


FIGURE 3-4. **A.** Plasma concentration of midazolam with time in an individual after an 8.35-mg i.v. bolus dose of midazolam hydrochloride (7.5 mg of the base) in a healthy adult. **B.** The data in A are redisplayed as a semilogarithmic plot. Note the short distribution phase. (From: Pentikäinen PJ, Välisalmi L, Himberg JJ, Crevoisier C. Pharmacokinetics of midazolam following i.v. and oral administration in patients with chronic liver disease and in healthy subjects. J Clin Pharmacol 1989;29: 272–277.)

Tozer TN, Rowland M. **Essentials of pharmacokinetics and pharmacodynamics.** Third edition

Variability between drugs

- large differences in physico-chemical properties between compounds
- large differences in pharmacokinetic parameters

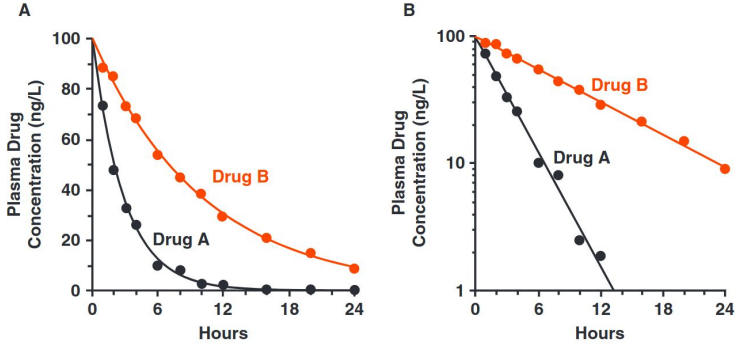


FIGURE 5-1 Drugs A (*black circle*) and B (*colored circle*) show the same initial (peak) exposure, but have different half-lives and total exposure–time profiles (AUC). Regular (cartesian) plot (*left*). Semilogarithmic plot (*right*). Doses of both drugs are the same.

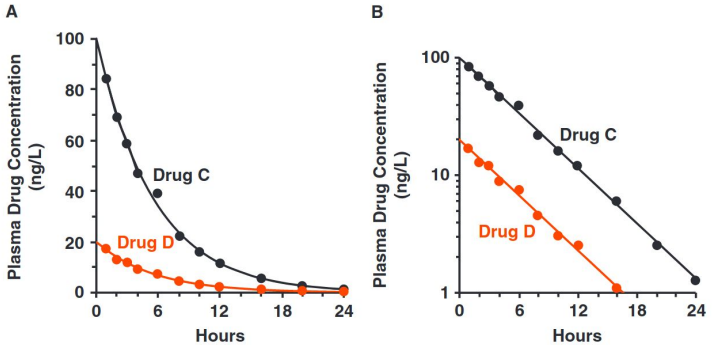


FIGURE 5-2 Drugs C (*black circle*) and D (*colored circle*) have the same half-life but different initial and total exposure–time (AUC) profiles. Regular (cartesian) plot (*left*). Semilogarithmic plot (*right*). Doses of both drugs are the same.

Tozer TN, Rowland M. **Essentials of pharmacokinetics and pharmacodynamics.** Second edition

Variability between drugs

- large variability in pharmacokinetic parameters between drugs

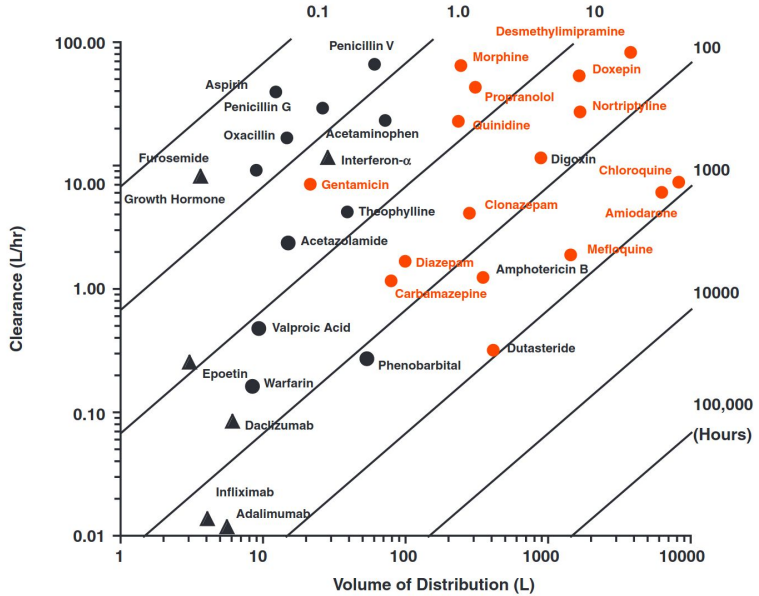
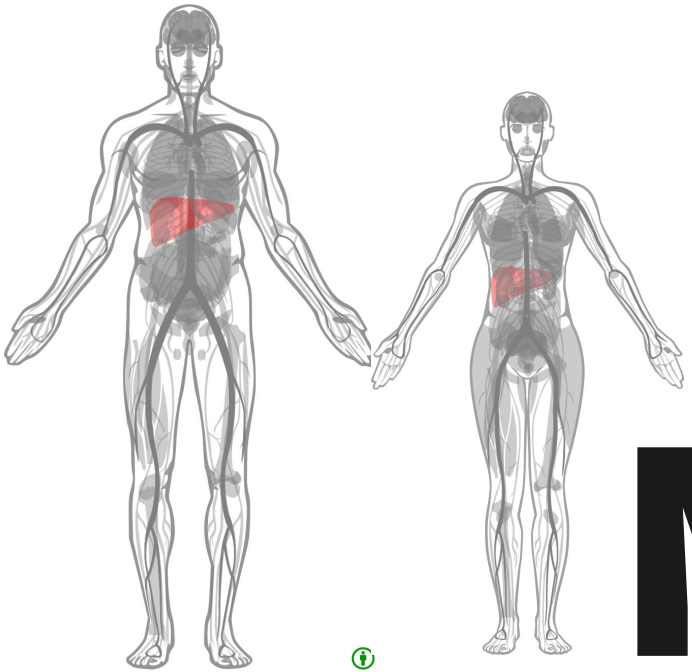


FIGURE 5–8 Clearance (*ordinate*) and volume of distribution (*abscissa*) of selected acidic (*black circle*) and basic (*colored names*), as well as protein (*black triangle*), drugs vary widely. Diagonal lines on the fully logarithmic plot show the combinations of clearance and volume values with the same half-lives (hours). Note that drugs with very low clearance and very large volumes (*lower right-hand quadrant*) are uncommon; their half-lives are often too long for these drugs to be used practically in drug therapy. Note also that large protein drugs have volumes of distribution close to plasma volume (3 L) and that basic compounds tend to have larger volumes of distribution than acids. Digoxin and dutasteride are neutral compounds, while amphotericin B is both a weak acid and a weak base.

Tozer TN, Rowland M. **Essentials of pharmacokinetics and pharmacodynamics.** Second edition



MODELS

Structural models as algebraic equations

- simplest representation is an **algebraic equation** representing a **one-compartment model**
- drug being administered as single intravenous dose
- relationship between
  - independent variable time (t)
  - dependent variable concentration (C)
  - C depends on Dose, clearance (CL), and distribution volume (V)

$$C_{(t)} = \frac{\text{Dose}}{V} e^{-\frac{CL}{V}.t}$$

Mould DR, Upton RN. **Basic concepts in population modeling, simulation, and model-based drug development.** CPT Pharmacometrics Syst Pharmacol. 2012 Sep 26;1(9):e6. doi: 10.1038/psp.2012.4. PMID: 23835886; PMCID: PMC3606044.

Structural models as algebraic equations

- Parametrization with clearance (CL) and volume of distribution (Vd)
- Simulation of concentration time course after given iv Dose

$$C_{(t)} = \frac{\text{Dose}}{V} e^{-\frac{CL}{V}.t} \qquad (1)$$

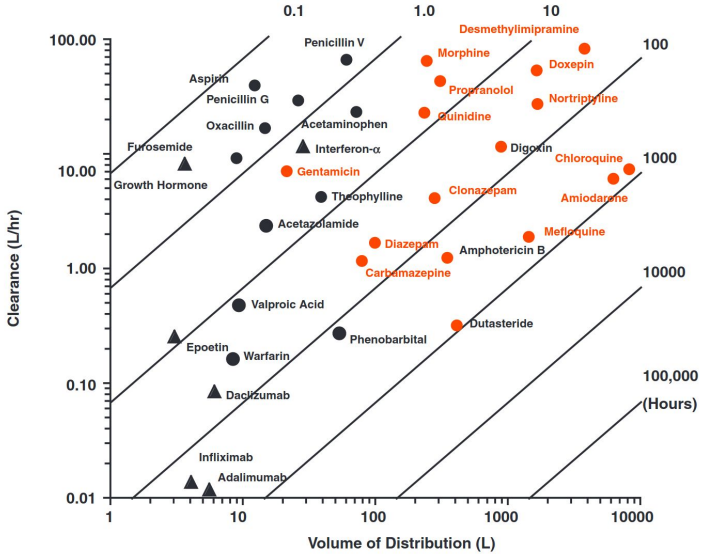


FIGURE 5-8 Clearance (*ordinate*) and volume of distribution (*abscissa*) of selected acidic (*black circle*) and basic (*colored names*), as well as protein (*black triangle*), drugs vary widely. Diagonal lines on the fully logarithmic plot show the combinations of clearance and volume values with the same half-lives (hours). Note that drugs with very low clearance and very large volumes (*lower right-hand quadrant*) are uncommon; their half-lives are often too long for these drugs to be used practically in drug therapy. Note also that large protein drugs have volumes of distribution close to plasma volume (3 L) and that basic compounds tend to have larger volumes of distribution than acids. Digoxin and dutasteride are neutral compounds, while amphotericin B is both a weak acid and a weak base.

Mould DR, Upton RN. **Basic concepts in population modeling, simulation, and model-based drug development.** CPT Pharmacometrics Syst Pharmacol. 2012 Sep 26;1(9):e6. doi: 10.1038/psp.2012.4. PMID: 23835886; PMCID: PMC3606044.

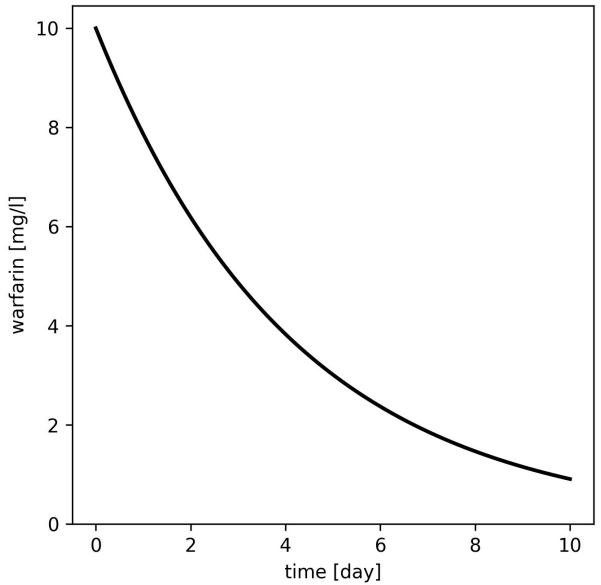
Structural models as algebraic equations

```python
from matplotlib import pyplot as plt
import numpy as np

# Warfarin
V = 10  # [l]
CL = 0.1  # [L/hr]
Dose = 100  # [mg]
t = np.linspace(start=0, stop=10*24, num=200)  # [hr]
C = Dose/V * np.exp(-CL/V * t)  # [mg/l]

# plot
f, ax = plt.subplots(nrows=1, ncols=1, figsize=(5, 5), dpi=300)
ax.plot(t/24.0, C, label="warfarin", color="black", linewidth=2.0)
ax.set_xlabel("time [day]")
ax.set_ylabel("warfarin [mg/l]")
ax.set_ylim(bottom=0)
plt.show()
```

$$C_{(t)} = \frac{\text{Dose}}{V} e^{-\frac{CL}{V}.t} \qquad (1)$$



Mould DR, Upton RN. **Basic concepts in population modeling, simulation, and model-based drug development.** CPT Pharmacometrics Syst Pharmacol. 2012 Sep 26;1(9):e6. doi: 10.1038/psp.2012.4. PMID: 23835886; PMCID: PMC3606044.

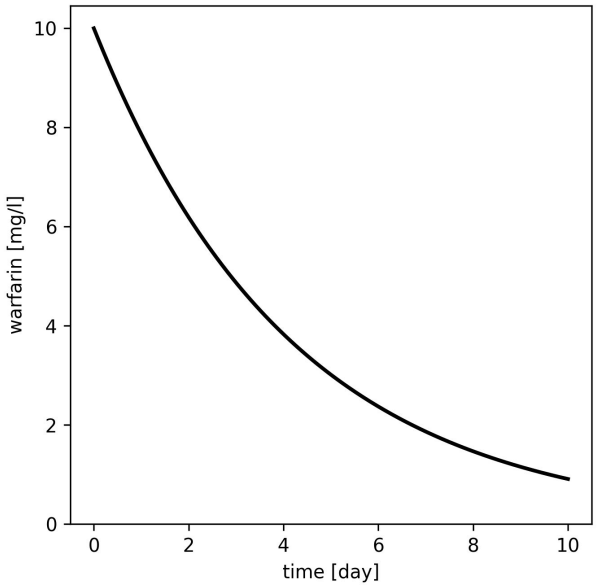
Structural models as algebraic equations

```python
from matplotlib import pyplot as plt
import numpy as np

# Warfarin
V = 10  # [l]
CL = 0.1  # [L/hr]
Dose = 100  # [mg]
t = np.linspace(start=0, stop=10*24, num=200)  # [hr]
C = Dose/V * np.exp(-CL/V * t)  # [mg/l]

# plot
f, ax = plt.subplots(nrows=1, ncols=1, figsize=(5, 5), dpi=300)
ax.plot(t/24.0, C, label="warfarin", color="black", linewidth=2.0)
ax.set_xlabel("time [day]")
ax.set_ylabel("warfarin [mg/l]")
ax.set_ylim(bottom=0)
plt.show()
```

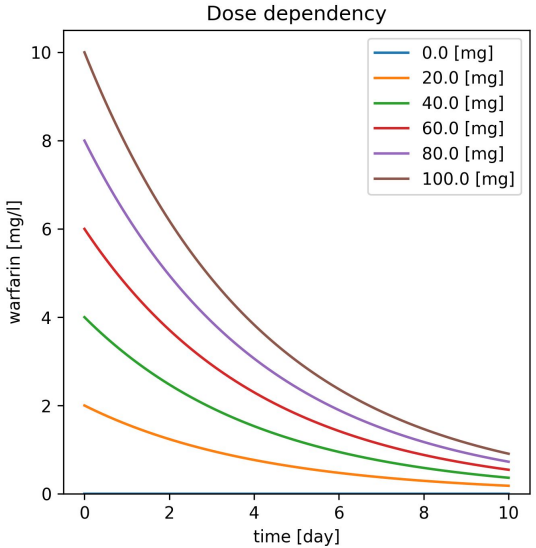
$$C_{(t)} = \frac{\text{Dose}}{V} e^{-\frac{CL}{V}.t} \qquad (1)$$

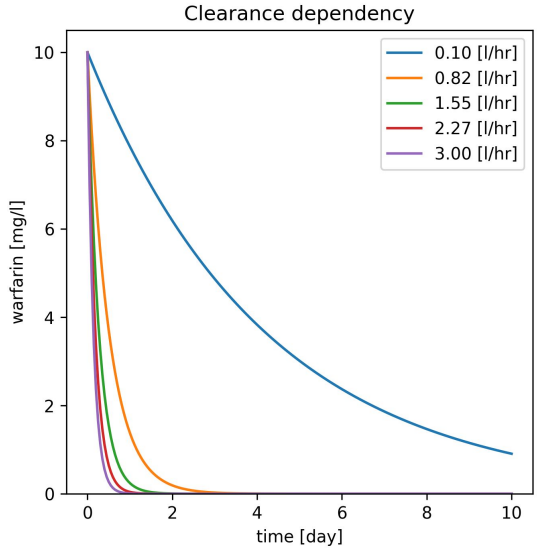


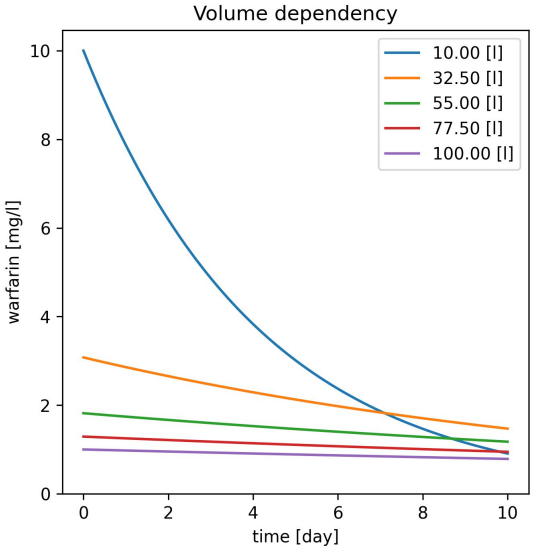
Mould DR, Upton RN. **Basic concepts in population modeling, simulation, and model-based drug development.** CPT Pharmacometrics Syst Pharmacol. 2012 Sep 26;1(9):e6. doi: 10.1038/psp.2012.4. PMID: 23835886; PMCID: PMC3606044.

Parameter scans

$$C_{(t)} = \frac{\text{Dose}}{V} e^{-\frac{CL}{V}.t} \qquad (1)$$







Ordinary differential equations (ODE)

- a differential equation describes the rate of change of a variable
- dC/dt denotes the rate of change of the concentration over time
- differential equations require specification of the initial value (C0)

$$C_{(t)} = \frac{\text{Dose}}{V} e^{-\frac{CL}{V}.t} \quad (1)$$

$$\frac{dC}{dt} = -\frac{CL}{V} * C \quad (2)$$

$$C_0 = \frac{\text{Dose}}{V}$$

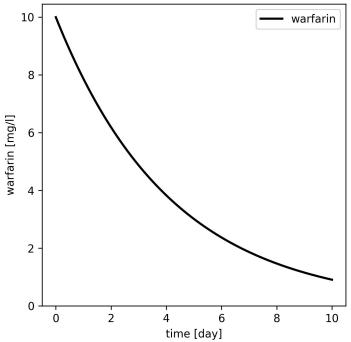
Mould DR, Upton RN. **Basic concepts in population modeling, simulation, and model-based drug development.** CPT Pharmacometrics Syst Pharmacol. 2012 Sep 26;1(9):e6. doi: 10.1038/psp.2012.4. PMID: 23835886; PMCID: PMC3606044.

Numerical integration

- ODEs can be solved via numerical integration
- e.g. Euler method as simplest case
- solving a system of equations is computationally intensive

$$\frac{dC}{dt} = -\frac{CL}{V} * C$$

$$C_0 = \frac{Dose}{V}$$




```python
from scipy.integrate import odeint
from matplotlib import pyplot as plt
import numpy as np

# Parameter
V = 10  # [l]
CL = 0.1  # [L/hr]
Dose = 100  # [mg]


new *
def ydot(y, t):
    """ODE system: dx/dt"""
    C = y[0]
    return np.array([-CL/V * C])


# initial condition
y0 = np.array([Dose/V, ])  # [mg/l]

#Numerical integration
t = np.linspace(start=0, stop=10*24, num=200)  # [hr]
C = odeint(ydot, y0, t)

f, ax = plt.subplots(nrows=1, ncols=1, figsize=(5, 5), dpi=300)
ax.plot(t/24.0, C[:, 0], label="warfarin", color="black", linewidth=2.0)
ax.set_xlabel("time [day]")
ax.set_ylabel("warfarin [mg/l]")
ax.set_ylim(bottom=0)
ax.legend()
plt.show()
```

Compartment models

- Pharmacokinetics can be modeled via compartment models
- Simple pharmacokinetic models have proven useful in many applications
- Main processes (**ADME)**
  - **A**bsorption
  - **D**istribution
  - **M**etabolization
  - **E**xcretion

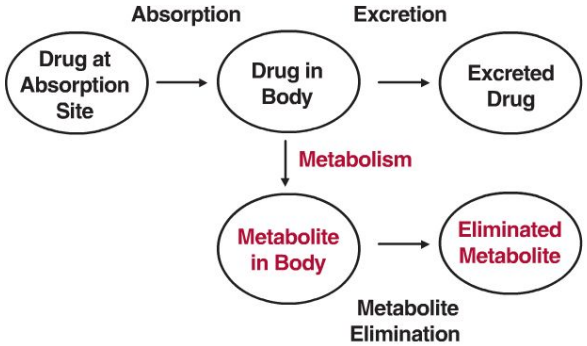


FIGURE 2-5. A drug is simultaneously absorbed into the body and eliminated from it, by excretion and metabolism. The processes of absorption, excretion, and metabolism are indicated with arrows and the compartments with ovals. The compartments represent different locations and different chemical species (color = metabolite). Metabolite elimination may occur by further metabolism or excretion.

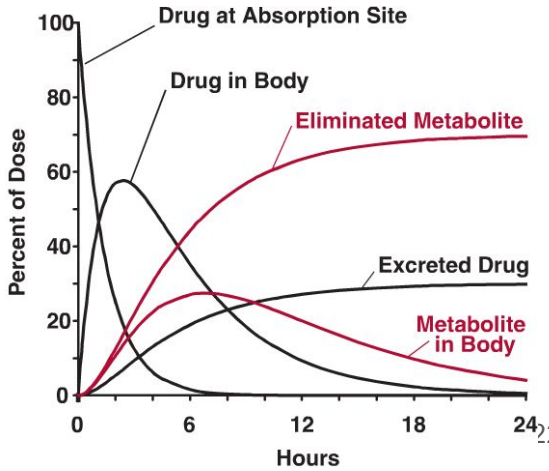


FIGURE 2-6. Time course of drug and metabolite in each of the compartments shown in Fig. 2-5. The amount in each compartment is expressed as a percentage of the dose administered. In this example, all the dose is absorbed. At any time, the sum of the molar amounts in the five compartments equals the dose.

Example of compartment model

- system of ODEs
- solved numerically
- A(D)ME
  - Absorption (v_a)
  - Metabolism (v_m)
  - Elimination ($v_{u,A}$, $v_{u,B}$)
- Mass action equations with rate constants k_a, k_m, k_e

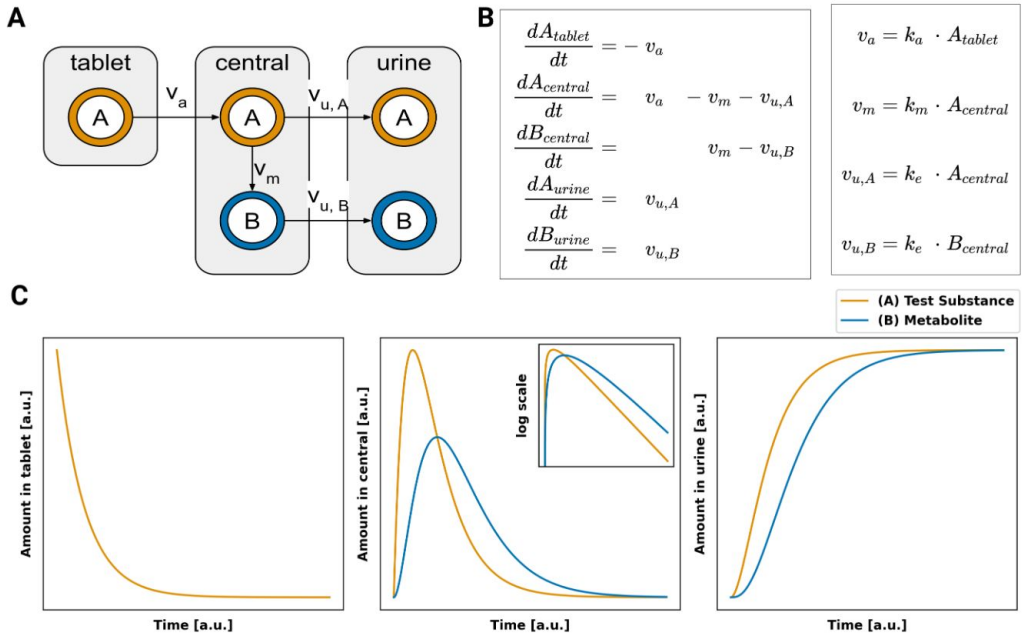
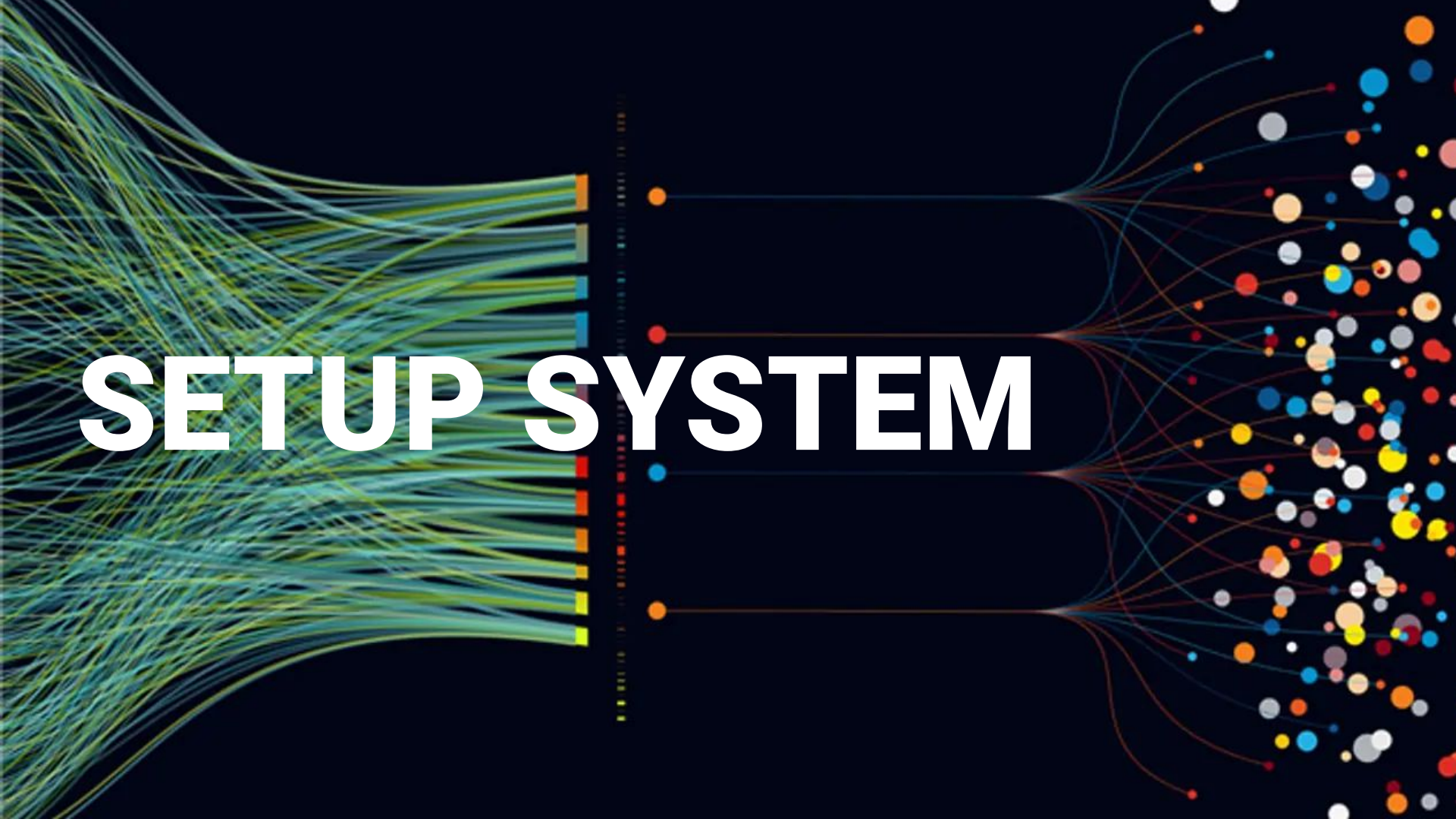


FIGURE 2.3: **Simple ODE-based pharmacokinetics model. A)** The system consists of three compartments (tablet, central, urine) that are connected via transport reactions. The model contains two substances the test substance A (orange); and the metabolite B (blue). The test substance A is metabolized to metabolite B in the central compartment. **B)** The resulting system of ordinary differential equations (ODEs). The rate of absorption, metabolism, and excretion (v_a, v_m, $v_{u,A}$, $v_{u,B}$) are modeled via irreversible mass-action kinetics. **C)** With an initial amount of $A_{tablet} = 10$ and rates $k_a = 1$, $k_m = 1$, and $k_e = 1$, all in [a.u.], the resulting amounts over time of the substances in the tablet, central, urine compartments are depicted.

Physiologically based pharmacokinetic (PBPK) modeling for dynamical liver function tests and CYP phenotyping. Jan Grzegorzewski (supervisor: **Matthias König**). PhD Thesis, Jan 2023

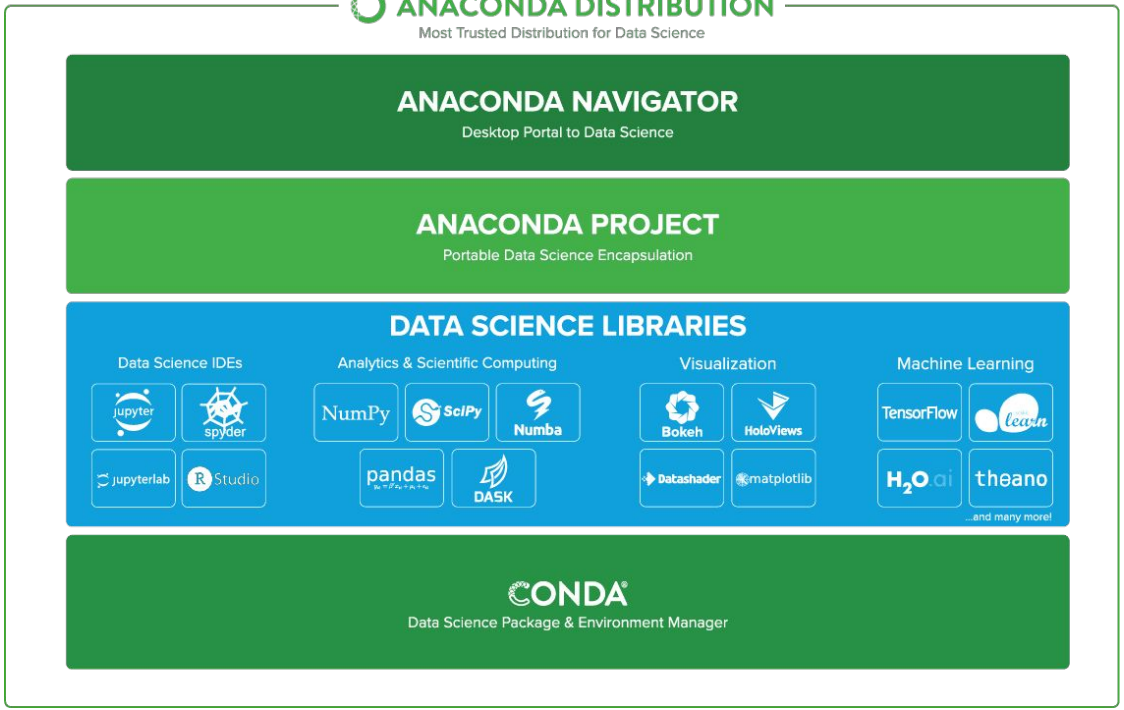

SETUP SYSTEM

Install Anaconda

Install Anaconda Distribution
https://docs.anaconda.com/free/anaconda/#installation

Anaconda® Distribution is a free Python/R data science distribution that contains:

- conda - a package and environment manager for your command line interface
- Anaconda Navigator - a desktop application built on conda, with options to launch other development applications from your managed environments
- 250 automatically-installed packages that work well together out of the box
- access to the Anaconda Public Repository, with 8000 open-source data science and machine learning packages



Setup conda environment

- create conda environment `mb19`
  - open terminal
  - create environment
    `conda create -n mb19`
  - install packages
    `pip install numpy scipy matplotlib pandas`

Activating/Deactivating environments

- To see a list of environments: **conda env list**

```
vperezg@login1:/home/vperezg>conda env list
# conda environments:
#
base                  *  /prod/apps/conda/3
bio-computation          /prod/apps/conda/3/envs/bio-computation
machine-learning         /prod/apps/conda/3/envs/machine-learning
machine-learning-gpu     /prod/apps/conda/3/envs/machine-learning-gpu
prosado                  /prod/apps/conda/3/envs/prosado
qiime2-2019.7            /prod/apps/conda/3/envs/qiime2-2019.7
quantum-chem             /prod/apps/conda/3/envs/quantum-chem
```

- To load an env: **conda activate <env_name>**

```
vperezg@login1:/home/vperezg>conda activate bio-computation
(bio-computation) vperezg@login1:/home/vperezg>
```

- To unload: **conda deactivate**

```
(bio-computation) vperezg@login1:/home/vperezg>conda deactivate
vperezg@login1:/home/vperezg>
```

Install Spyder

- available via the **anaconda navigator**