Team QuantumDose: Hybrid Quantum and ML PK-PD Modeling and Prediction Pipeline

Jack Diab[a] **Byoungwoo Kang**[b] **Nothando Khumalo**[a] **Aman Mehta**[c] **William Munizzi**[a] **Scott Nie**[a] **Prineha Narang**[d]

[d] *Division of Physical Sciences, College of Letters and Science, University of California, Los Angeles, Los Angeles, CA, 90095, USA*

[a] *Department of Chemistry and Biochemistry, University of California, Los Angeles, Los Angeles, CA, 90095, USA*

[c] *Electrical and Computer Engineering Department, University of California, Los Angeles, CA, 90095, USA*

[b] *Department of Physics, University of California, Los Angeles, CA, 90095, USA*

E-mail: jdiab@ucla.edu, byoungwookan901@ucla.edu, nkhumalo@g.ucla.edu, mehtaaman@ucla.edu, wmunizzi17@ucla.edu, nies@ucla.edu, prineha@ucla.edu

Abstract: Early-phase clinical trials face a critical challenge: accurately predicting optimal drug dosages from sparse, variable patient data. Traditional pharmacokinetic/pharmacodynamic (PK/PD) modeling approaches often struggle with the limited datasets characteristic of Phase 1 and 2 trials, leading to suboptimal dosing decisions that can compromise trial success. We propose a novel three-stage hybrid quantum-classical framework that leverages quantum-inspired algorithms and quantum machine learning to address this fundamental limitation. In Stage 1, we employ supervised machine learning trained on synthetic PK/PD trajectories to rapidly identify the most appropriate model structure and provide initial parameter estimates. Stage 2 refines these estimates using Quantum-Inspired Hamiltonian Monte Carlo (QIHMC), which introduces stochastic momentum fluctuations analogous to quantum particle behavior, enabling more efficient exploration of complex posterior distributions compared to classical Bayesian methods. Finally, Stage 3 implements a GPU-accelerated quantum kernel regression model that maps patient covariates to a high-dimensional quantum Hilbert space, capturing subtle non-linear correlations that classical models may overlook in sparse data regimes. By integrating quantum-enhanced parameter estimation with quantum neural networks for predictive modeling, our framework aims to significantly improve dosage prediction accuracy even from single-phase trial data, potentially reducing pharmaceutical development risk and accelerating the path from laboratory to clinic. Our team (QuantumDose) has unique expertise with cutting-edge methodologies for quantum computing, open and closed systems simulation, state preparation beyond variational approaches, classical optimization, and AI for science. We will leverage our long-standing relationship with NVIDIA in addition to Quantinuum as a confirmed hardware partner to contribute to circuit implementation/embedding and hardware access/deployment. Working closely with NVIDIA and NQCP, our team brings existing momentum to the project via access to quantum simulators and emulators native to next-generation GPUs to validate, verify, and work synergistically with our calculations on quantum hardware.

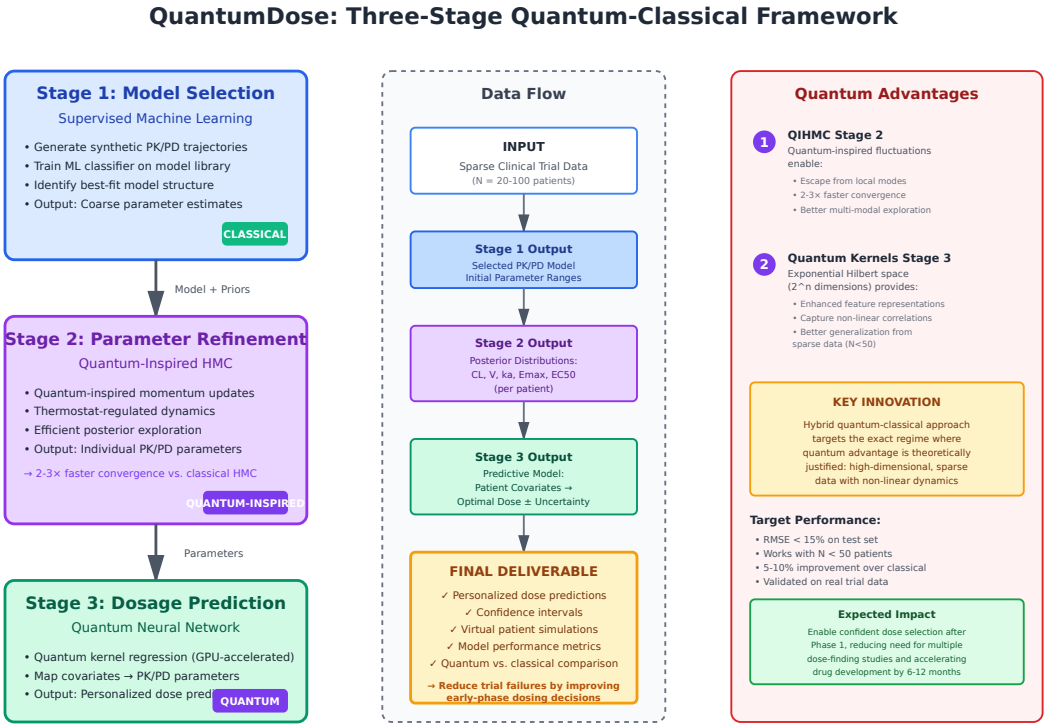
Contents

1 Introduction **1**
- 1.1 Project Timeline 3
- 1.2 Team Expertise 3

2 Methodology **5**
- 2.1 Stage 1: Model Selection Using Synthetic Data and Supervised Machine Learning 5
  - 2.1.1 Work Plan 5
  - 2.1.2 Resource Estimation 6
  - 2.1.3 Stage 1 Preliminary Results 6
- 2.2 Stage 2: Parameter Refinement via Quantum-Inspired Hamiltonian Monte Carlo 7
  - 2.2.1 Work Plan 9
  - 2.2.2 Resource Estimation 9
  - 2.2.3 Stage 2 Preliminary Results 10
- 2.3 Stage 3: Quantum Neural Network for Dosage Prediction 10
  - 2.3.1 Resource Estimation 11
  - 2.3.2 Work Plan 11
  - 2.3.3 Stage 3 Preliminary Results 13

1 Introduction

Optimizing drug dosage during early-phase clinical trials is a critical, high-stakes challenge central to pharmaceutical development. Pharmacokinetic and Pharmacodynamic (PK/PD) models are the primary tools for this task, but their predictive accuracy is often constrained by the sparse and variable data characteristic of Phase 1 and 2 trials. This limitation can lead to suboptimal dosage selection, increasing the risk of trial failure due to a lack of efficacy or unforeseen toxicity. We propose a novel hybrid quantum-classical framework for PK/PD parameter estimation and dosage prediction to address this fundamental challenge. Our approach focuses on learning the map between the system of differential equations that define a PK/PD model and an effective quantum Hamiltonian. We leverage quantum machine learning algorithms to fit this Hamiltonian to sparse clinical data, and hypothesize that the enhanced generalization capabilities of quantum models can derive more robust and accurate parameter estimates than are achievable using classical methods.

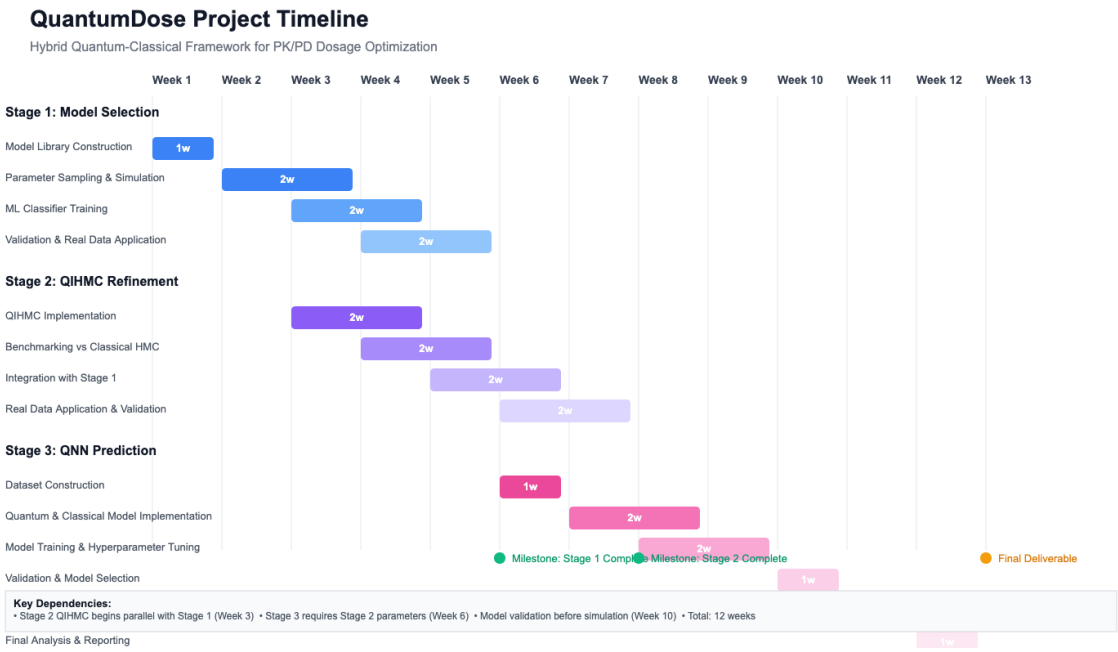


Following the acquisition of all relevant parameters, we believe that addressing the issue of limited trial data requires thoughtful predictive analysis. For advanced, data-driven predictions, neural-networks (NNs) stand out as the gold standard for recognition of complex relationships in high-dimensional data. While classical neural networks are notoriously 'data hungry' when training, quantum neural networks (QNNs) map data directly to a quantum kernel, allowing us to create a regression model that learns the underlying data relationships with even very limited datasets. Using the individual characteristics paired with their respective PK/PD parameters, we can construct a dataset that will allow for the training of a predictive QNN with which new patients can be characterized. This, alongside the improved calculation of the PK/PD parameters themselves is expected to significantly aid in the dosage estimation of new patients after only one trial phase.

We have organized our quantum and AI-inspired enhancement according to the PK/PD modeling process beginning with stage 1 where we use ML and quantum circuits to choose the correct model, i.e. PK/PD equations, and map it to a Hamiltonian form. Stage 2 describes our enhancement to the modeling process where we employ cutting-edge quantum Monte Carlo methods to fit the model to the trial data, recovering each individuals' PK/PD parameters. Finally, after all parameters are found, we move to stage 3 where a GPU-accelerated predictive quantum neural network is constructed with the entire trial dataset and parameters from stage 2.

1.1 Project Timeline

Here we show our projected project timeline. Detailed work plans for each stage are included in their respective sections.



1.2 Team Expertise

Our team has relevant background in computer science, chemistry, biochemistry, physics, and electrical engineering. Our collective research experience spans the development of quantum sensors for biological targets, advanced quantum algorithm design, and fundamental physics research. This diverse skill set provides the ideal blend of domain knowledge in the life sciences and deep technical expertise in quantum computing required to successfully execute this project. Our unique expertise has been recognized through our recent success in winning phase 1 of both the quantum computing and quantum sensing NIH technology challenges.

UCLA will lead the overall program and oversee efforts related to application development, workflows, quantum algorithms, scalability analysis, validation, and implementation. **Dr. Prineha Narang**, Prof at UCLA with experience across academia and industry in quantum science and technology will lead the program (Head-Marsden, Flick, et al., 2021; Head-Marsden, Krastanov, et al., 2021). **Dr. William Munizzi** is a postdoctoral scholar at UCLA with research expertise in mathematical physics, quantum foundations, and applications for near-term quantum devices ([1–4]). NarangLab graduate student **Jack Diab** has a broad range of experience in chemistry, spectroscopy, computational modeling and quantum sensing. Diab has worked with NASA JPL on computational approaches to modeling

complex chemical systems before starting his PhD focusing on developing quantum sensing methods (Diab, et al., 2022). Graduate Student **Byoungwoo Kang** has experience in utilizing Machine Learning for Quantum Control in Trapped Molecular Ion systems, and is highly skilled in Reinforcement Learning and Transformer Models for processing sequential data. Prior to his PhD, he also has experience in fabricating Semiconductor Quantum dots and running physical simulations. Graduate student **Aman Mehta** is a second-year PhD candidate in Electrical Engineering, with research experience in Quantum Benchmarking, Open Quantum Systems, and Quantum Information [5, 6], alongside three years of industry experience at a technology start-up. As a team we have tackled questions at the interface of computational chemistry, physics, and engineering. **Scott Nie** has continued to work on quantum algorithms for quantum chemistry applications. Scott also has a previous background in chemical biology and protein interactions (Dutta et al., 2024).

We capitalize on our breadth of expertise to solve critical questions in interdisciplinary domains, harnessing our full complement of available talent. Quantinuum is a confirmed hardware partner and will contribute to circuit implementation/embedding, and hardware access/deployment. NVIDIA and NQCP (where Dr. Narang is a Visiting Professor of Physics) will provide access to quantum simulators, emulators native to next-generation GPUs to validate, verify, and work synergistically with our calculations on quantum hardware. Letters of support are available upon request.

Capabilities and Facilities Available to the Team: UCLA Quantum Innovation Hub (QIH): UCLA has started a collaborative academic/industry research institute called QIH, which will be housed in its own off-campus building. UCLA leadership has committed to making QIH lab space available for this project particularly to facilitate industry interactions. In this program, we bring together U.S. research programs at UCLA and private, and non-profit organizations that conduct critical research in quantum computing for biology, provide training and career opportunities, and engage with the public at large through meaningful and long-term impact strategies. The goal of this ecosystem is to promote the optimal use of our limited scientific resources in the search for creative solutions to complex, interdisciplinary problems in quantum computing for drug discovery and human health. IPAM, a key partner via Narang, fosters the interaction of mathematics with a broad range of science and technology, builds new interdisciplinary research communities, and promotes mathematical innovation. IPAM fulfills its mission through workshops and other programs that connect mathematics and other disciplines. CNSI Tech Transfer CNSI offers a unique opportunity to support the academic and industrial research community in translation biomedical science. Magnify, the CNSI startup incubator, supports entrepreneurship and commercialization of advanced technologies.

2 Methodology

2.1 Stage 1: Model Selection Using Synthetic Data and Supervised Machine Learning

Stage 1 addresses the fundamental challenge of choosing the most appropriate PK/PD structural model for a given dataset. Traditional approaches rely heavily on expert intuition and iterative fitting, which is both time-consuming and prone to bias. Recent studies have explored the integration of machine learning into pharmacometric workflows to support model selection [7], parameter prediction, and decision-making, demonstrating that data-driven approaches [8] can significantly enhance and accelerate this process. We propose a data-driven framework that leverages synthetic training data to train a machine learning classifier capable of identifying the best-fit model class and plausible parameter ranges directly from observed trajectories.

To achieve this, we will construct a library of canonical PK/PD models (e.g., one-compartment, two-compartment, indirect response models, Emax/sigmoid-Emax), each parameterized by biologically plausible ranges drawn from empirical literature (e.g., clearance [CL], volume of distribution [V], half-life [$t_{1/2}$] ranges bounded by known pharmacological constraints). Prior work has shown that simulation studies commonly employ parameter sampling within physiologically plausible ranges to ensure generated profiles reflect realistic pharmacological behavior [9–11]. Building on this precedent, we will perform parameter sampling for each model, simulate the corresponding ODE trajectories, and down-sample them at clinically realistic observation times to mimic trial data.

This synthetic dataset comprising model class labels, parameter sets, and sampled trajectories will then be used to train a supervised machine learning model. We utilize a Transformer Encoder structure for the model to effectively deal with the sequential trajectories. The model will learn to map observed concentration-time, biomarker-time profiles to both the most probable underlying PK/PD model class, and initial parameter estimates consistent with observed dynamics.

The central hypothesis of Stage 1 is that by training on a sufficiently diverse library of simulated profiles, the ML model will be able to generalize to unseen patient data and provide strong initial guesses that can be used as initial guess or a seed for more rigorous Bayesian inference in subsequent stages.

2.1.1 Work Plan

- **Synthetic Data Generation**
  - *Task 1.1 - Model Library Construction:* Define the set of candidate PK/PD models (e.g., 1C/2C linear, nonlinear elimination, indirect response).
  - *Task 1.2 - Parameter Sampling:* Sample parameters (e.g., CL, V, k_a, $t_{1/2}$) from literature-informed distributions.

- *Task 1.3 - Trajectory Simulation:* Numerically solve ODEs under each parameter set and sample at realistic time intervals.

- **Supervised Model Training**
  - *Task 1.4 - Dataset Assembly:* Construct feature matrices from simulated trajectories and associate with model/parameter labels.
  - *Task 1.5 - Classifier Training:* Implement ML models (Neural Networks, Transformer Encoders) to predict the model class and parameter estimates.
  - *Task 1.6 - Validation:* Evaluate accuracy using held-out synthetic data and robustness against noise injection.

- **Application to Real Data**
  - *Task 1.7 - Model Inference:* Apply the trained ML pipeline to early-phase patient datasets.
  - *Task 1.8 - Aggregation Across Patients:* Infer the most likely global model structure by aggregating per-patient predictions.

2.1.2 Resource Estimation

Stage 1 is primarily computational and requires modest resources.

- **Computational:** Trajectory simulations (ODE solvers) and ML model training can be performed on mid-range GPU-enabled servers. Estimated usage is 100-200 GPU hours (for training the Neural Network classifiers) and ∼200 CPU hours (for ODE-based synthetic data generation of ∼3M dataset).

- **Data:** No clinical data are required for initial development; simulations provide the training dataset. For validation, small-scale public PK/PD datasets [12] will be used.

2.1.3 Stage 1 Preliminary Results

We code a Transformer Encoder equipped Machine Learning model that receives sequential data and outputs a prediction of the model class and parameter values related to the sequence. We show that such a model trained with a synthetic dataset of 7,500 sequences sampled with 4 example models (one compartment concentration model, two compartment concentration model, indirect response with one compartment, indirect response with two compartment), and 9 different types of parameters (k_a, CL, V, V_c, V_p, Q, k_{in}, k_{out}, $IC50$), can be trained up to an accuracy on the synthetic test dataset of 76.9% for model classification. This is achieved after 100 epochs of training with a loss graph of Fig. 1. We use this model to classify the observed

dataset and show that for the biomarker sequential data, all of them are correctly classified to one of the "response models" as shown in Fig. 2. Also, the prediction of the parameter ranges for the 23 biomarker sequence sets classified as a "indirect response with one compartment" model, are shown in Fig.3. With more time and computational resources, we hope to increase the number of PK/PD models we use to generate a synthetic dataset, and also add other time-dependent variables, such as "actual dose amount administered" to our input data. Finally, extensive hyperparameter tuning to find the best learning rate and model parameters are needed. We anticipate that such measures will increase both the accuracy of the model classification task on the synthetic dataset, and also give us a detailed and accurate picture of what PK/PD model and parameters we should use for Stage 2.

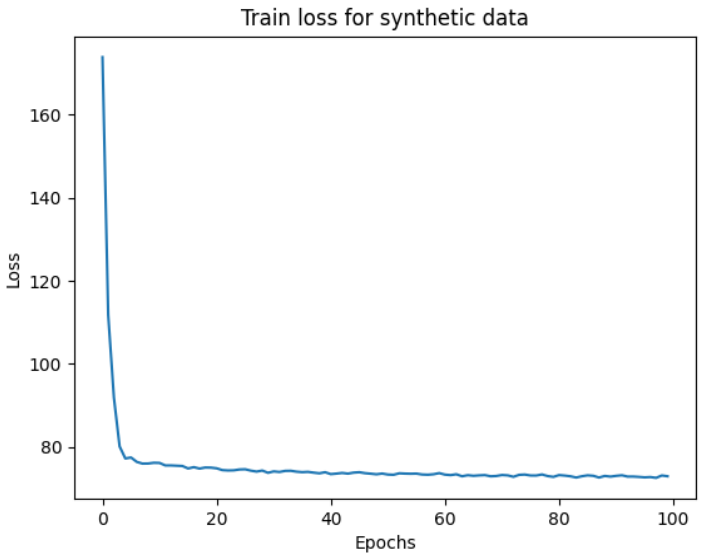


Figure 1. Training loss for the Transformer Encoder Supervised Learning model for a 7500 synthetic dataset with 100 epochs.

2.2 Stage 2: Parameter Refinement via Quantum-Inspired Hamiltonian Monte Carlo

Stage 2 builds upon the coarse model and parameter estimates generated in Stage 1 by applying Quantum-Inspired Hamiltonian Monte Carlo (QIHMC)[13, 14] for rigorous Bayesian inference of PK/PD parameters. Classical Hamiltonian Monte Carlo (HMC) is widely regarded as a gold standard for Bayesian parameter estimation due to its use of Hamiltonian dynamics to explore posterior distributions efficiently. However, standard HMC can suffer from slow mixing and long autocorrelation times[15, 16] when confronted with complex, high-dimensional posterior landscapes common in pharmacometrics.

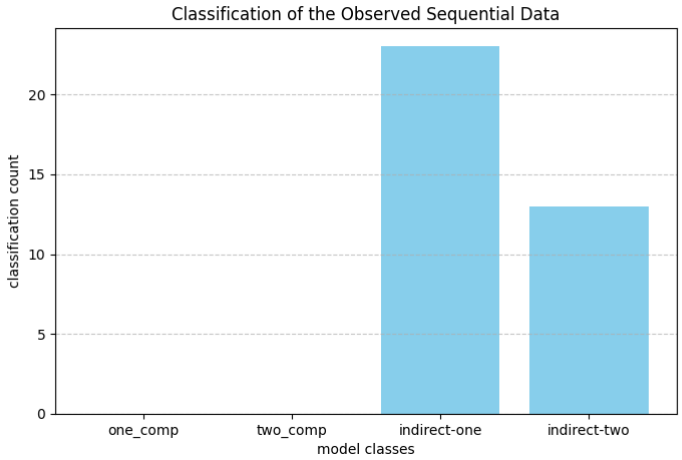


Figure 2. Example results of Stage 1 Model Classification of the 36 subject PK/PD dataset.

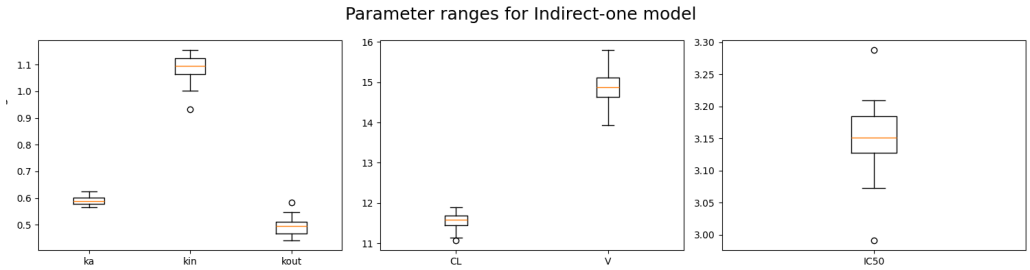


Figure 3. Example results of Stage 1 Parameter range search of the 23 subject PK/PD dataset identified as "Indirect One" Model.

Quantum-inspired HMC extends classical HMC by introducing stochastic momentum fluctuations and thermostat dynamics that mimic the probabilistic nature of quantum particles exploring an energy landscape. Here, the momentum of each parameter is perturbed with controlled noise, analogous to quantum fluctuations, while a thermostat variable regulates the system's effective temperature to maintain stability. These modifications allow the sampler to traverse parameter space more freely, avoiding local posterior modes and accelerating convergence relative to conventional deterministic HMC.

The outputs of Stage 1, in the form of coarse maximum-likelihood or machine-learning-informed parameter estimates, provide informative priors for QIHMC. This ensures that the sampler begins in a physically plausible region of parameter space, reducing burn-in and enhancing sampling efficiency. The resulting posterior distributions over individual pharmacokinetic and pharmacodynamic parameters—including CL_i, V_i, k_a, E_{max}, and other patient-specific covariates—capture high-resolution individualized uncertainty, forming a solid foundation for predictive modeling and down-

stream decision-making in Stage 3.

2.2.1 Work Plan

- **Sampler Design and Testing**
  - *Task 2.1 – QIHMC Implementation:* Develop a quantum-inspired HMC sampler in Python. The sampler will use stochastic momentum perturbations and thermostat-regulated dynamics to emulate quantum fluctuations, enabling efficient exploration of high-dimensional posterior landscapes of PK/PD parameters.
  - *Task 2.2 – Benchmarking Against Classical HMC:* Evaluate the QIHMC sampler on synthetic PK/PD datasets. Compare convergence rates, effective sample sizes, and posterior fidelity against classical HMC on synthetic PK/PD data.
- **Integration with Stage 1**
  - *Task 2.3 – Prior Incorporation:* Use Stage 1 ML outputs or coarse maximum likelihood estimates to define informative priors for each subject, ensuring the sampler initiates in high-probability regions of parameter space.
  - *Task 2.4 – Real Data Application:* Apply QIHMC to real patient datasets to refine individualized PK/PD parameter estimates. The stochastic momentum updates allow the sampler to overcome posterior multi-modality and local traps efficiently.
- **Validation and Reporting**
  - *Task 2.5 – Posterior Validation:* Validate estimated posteriors against known pharmacological ranges and simulation benchmarks.
  - *Task 2.6 – Deliverables:* Produce patient-level posterior distributions as inputs to Stage 3 regression framework. Visualizations of posterior predictive trajectories will illustrate sampling efficiency and coverage.

2.2.2 Resource Estimation

Stage 2 requires higher computational resources due to Bayesian sampling.

- **Computational:** QIHMC sampling will require $\sim$5000 CPU hours for synthetic data benchmarking and $\sim$200 GPU hours for parallelizable components (generation, trajectory evaluations). Compared to standard HMC, we anticipate $\sim$2-3$\times$ faster convergence, reducing total compute costs.

- **Data:** Stage 1 outputs provide priors and structural models. The provided dataset, along with the open-source data ($N \approx 50 - 100$ individuals) will be used for a proof-of-concept.

2.2.3 Stage 2 Preliminary Results

We show here how the qutip-based quantum Monte Carlo method is used to sample the predicted concentration in the pk-function then Markov-Chain Monte Carlo (MCMC) is used to improve the model.

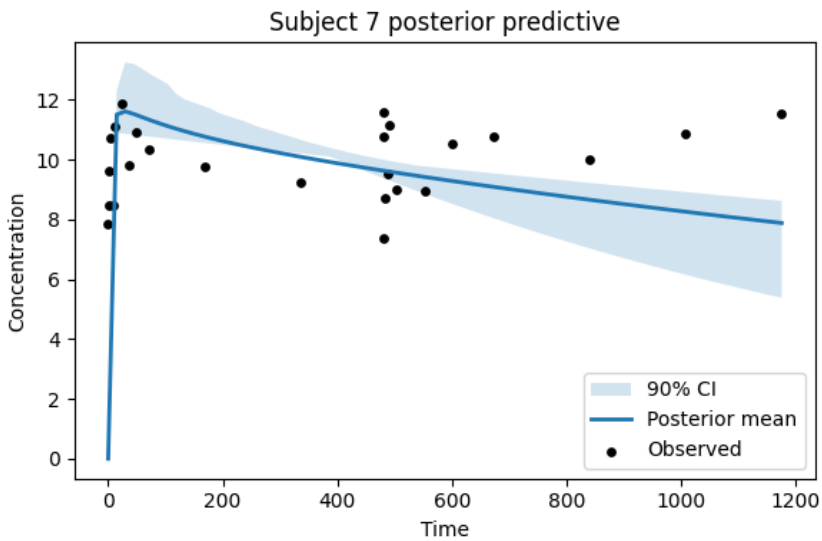


Figure 4. Predicted model for an individual's trial data (ID:7) with 90% confidence interval.

With more time, we will utilize more complex modeling and physical understanding of the trial data to inform our code. However, we have shown that this technique is possible and can potentially yield complex and meaningful results.

2.3 Stage 3: Quantum Neural Network for Dosage Prediction

Stage 3 directly addresses the critical challenge of predicting individual drug responses from the sparse data typical of early-phase trials. To construct a predictive engine that generalizes well from limited information, we will implement a quantum kernel-based regression framework using a GPU-accelerated approach [17–19]. The training dataset will be formed by pairing patient covariates (e.g., body weight, concomitant medication) as input features with their corresponding individual PK/PD parameters derived in Stage 2 as the target variables. Central to our approach is the design of a parameterized quantum circuit, or 'feature map,' to encode each patient's covariate vector into a high-dimensional quantum Hilbert space. This quantum embedding is hypothesized to capture subtle, non-linear correlations

between covariates—patterns that classical models may overlook in a sparse data regime. We will then compute the corresponding quantum kernel matrix, which quantifies patient similarity within this feature space. This matrix will serve as the input for a classical Support Vector Regressor (SVR) to construct an optimal predictive model, ultimately mapping a new patient's baseline characteristics to their estimated PK/PD parameters and informing initial dosage by predicting their drug clearance, or the rate at which the individual's body eliminates the drug.

Our choice of methodology in Stage 3 addresses the core challenge of generalizing from a small data set. Quantum kernels map classical data into an exponentially large quantum feature space, which is hypothesized to be exceptionally effective at uncovering complex, non-linear correlations between patient covariates. The SVR is a powerful and mature classical algorithm designed to work natively with such kernels[19]. In low-data regimes, classical models like neural networks are prone to overfitting, learning noise rather than the true biological signal. By leveraging the rich expressive power of a quantum feature space, our model is designed to capture the underlying patient-to-patient variability more effectively. This allows for more robust predictions, which is the central challenge in translating Phase 1 results to broader populations.

A main concern for Stage 3 is that the quantum kernel model may not show a significant predictive advantage over powerful classical methods, such as Gradient Boosted Trees. Our work plan directly mitigates this by building and validating these classical models in parallel. Should a quantum advantage not be realized, the project will fall back to the best-performing classical algorithm, ensuring the core objective of delivering a high-quality predictive engine is met.

2.3.1 Resource Estimation

Currently we dont expect CPU hours to go beyond 100s of CPU hours for smaller circuits of not more than 10 qubits. As most circuit simulators are CPU-bound, estimation of the exact number of GPU hours needed for this model implementation if moved to a GPU-based workflow is difficult but likely on the order of 10s to 100s of hours [17]. Datasets will likely not exceed the size of the trial data yet will need to be processed many times to achieve statistically significant results and thus we expect computational times to be manageable on most modern multi-core workstations where parallel computing is available.

2.3.2 Work Plan

- **Data Assembly and Model Implementation**
  - *Task 3.1 - Dataset Construction:* Assemble the final training dataset. The input features, $\mathbf{X}$, will consist of the patient covariate matrix (body weight, concomitant medication), and the target variables, $\mathbf{Y}$, will be

the corresponding vector of individual PK/PD parameters (e.g., CL_i, V_i) estimated in Stage 2.

  - *Task 3.2 - Quantum Model Implementation:* Implement the quantum kernel regression pipeline using Python with the PennyLane and scikit-learn libraries.
  - *Task 3.3 - Classical Benchmark Implementation:* To ensure rigorous comparison, implement two strong classical benchmarks: (1) a Support Vector Regressor with a tuned Radial Basis Function (RBF) kernel, and (2) a Gradient Boosted Trees model (e.g., XGBoost).

- **Model Training and Hyperparameter Tuning**
  - *Task 3.4 - Cross-Validation Strategy:* Implement a k-fold cross-validation scheme on the training set to guide hyperparameter tuning and prevent overfitting.
  - *Task 3.5 - Quantum Model Tuning:* Optimize the quantum model by experimenting with the structure of the feature map (e.g., number of layers, qubit connectivity) and tuning the hyperparameters of the classical SVR backend (e.g., regularization parameter C).
  - *Task 3.6 - Classical Model Tuning:* Perform extensive hyperparameter tuning for the classical SVR and XGBoost models using a grid search or Bayesian optimization approach.

- **Validation and Final Model Selection**
  - *Task 3.7 - Performance Evaluation:* Evaluate the final, tuned quantum and classical models on the held-out test set. The primary metric for comparison will be the Root Mean Squared Error (RMSE) between the predicted and true PK/PD parameters.

- **Simulation, Analysis, and Reporting**
  - *Task 3.9 - In Silico Trial Simulation:* Utilize the selected predictive model to generate personalized PK/PD parameters for large virtual patient populations under the conditions specified in the proposal questions (e.g., altered body weight distribution).
  - *Task 3.10 - Dosage Optimization:* For each scenario, simulate drug clearance outcomes across the required range of doses to identify the optimal dose that meets the target biomarker suppression threshold.

2.3.3 Stage 3 Preliminary Results

To have flexibility in testing and iteration, we first generated a synthetic dataset to represent N subjects before utilizing the given dataset. The data consisted of a feature matrix, $\mathbf{X} \in \mathbb{R}^{N\times 2}$, and a target vector, $\mathbf{y} \in \mathbb{R}^N$. The two features simulated patient covariates: Body Weight (x_1), a continuous variable sampled from a uniform distribution $U(N, 120)$, and Concomitant Medication (x_2), a binary variable from $\{0, 1\}$. The target variable, drug clearance (y), was generated as a non-linear function of these features with additive Gaussian noise, $y = f(x_1, x_2) + \mathcal{N}(0, 0.5^2)$.

The quantum kernel is defined by the squared overlap (fidelity) of their quantum state representations: $K(\mathbf{x}_i, \mathbf{x}_j) = |\langle\phi(\mathbf{x}_i)|\phi(\mathbf{x}_j)\rangle|^2$. The mapping from the classical feature vector to the quantum state, $|\phi(\mathbf{x})\rangle = U_{\phi(\mathbf{x})}|0\rangle^{\otimes n}$, is governed by a parameterized quantum circuit, or feature map, implemented in a $n = 2$ qubit system. The unitary operator $U_{\phi(\mathbf{x})}$ for this feature map is composed of two layers: (1) a feature encoding layer consisting of single-qubit rotations, $U_{rot}(\mathbf{x}) = \bigotimes_{k=1}^{n} R_X(x_k)$, where each feature value x_k parameterizes a rotation around the x-axis; and (2) an entangling layer, U_{ent}, consisting of a Controlled-NOT gate applied to the qubits. The unitary complete feature map is thus $U_{\phi(\mathbf{x})} = U_{ent} \cdot U_{rot}(\mathbf{x})$. The kernel value was calculated by a separate quantum circuit that applies the unitary $U_{\phi(\mathbf{x}_i)}U^{\dagger}_{\phi(\mathbf{x}_j)}$ and measures the probability of the resulting state collapsing to the initial $|0\rangle^{\otimes n}$ state.

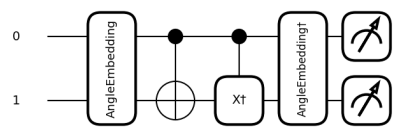


Figure 5. Simple example circuit used for mapping training data to a quantum kernel.

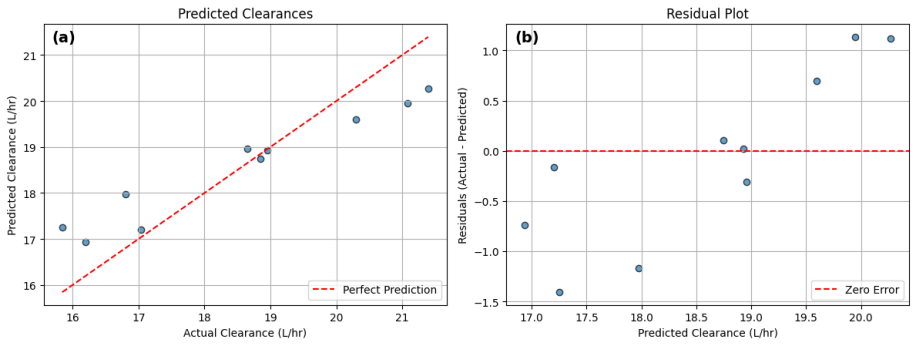


Figure 6. Accuracy of our simple prediction model using quantum kernel regression. This example has an $R^2 = 0.804$.

A classical Support Vector Regressor (SVR) was utilized as the learning algorithm. Currently we are using the SVR included in the Scikt-learn package [20], but more advanced methods are available in the future if necessary. The SVR was con-

figured with the `kernel='precomputed'` option to accept the Gram matrix derived from the quantum kernel function.

After training, the model was able to achieve a clearance prediction R^2 consistently above 0.65 with an average of 0.705 after 10 tests of randomized synthetic datasets. One such test is shown in Figure 6, where we show the distribution of error for our predictions. While lower than we would like, we believe future work will improve this drastically as we implement and iterate on more complex and thoughtful quantum circuit mappings that better

References

[1] C. Keeler, W. Munizzi, and J. Pollack, *Bounding entanglement entropy with clifford double cosets*, *Physical Review A* **112** (Aug., 2025).

[2] C. Keeler, W. Munizzi, and J. Pollack, *Entropic lens on stabilizer states*, *Phys. Rev. A* **106** (2022), no. 6 062418, [arXiv:2204.07593].

[3] W. Munizzi and H. J. Schnitzer, *Entropy cones and entanglement evolution for Dicke states*, *Phys. Rev. A* **109** (2024), no. 1 012405, [arXiv:2306.13146].

[4] C. Cao, G. Cheng, A. Hamma, L. Leone, W. Munizzi, and S. F. E. Oliviero, *Gravitational back-reaction is magical*, arXiv:2403.07056.

[5] T. Lubinski, J. J. Goings, K. Mayer, S. Johri, N. Reddy, A. Mehta, N. Bhatia, S. Rappaport, D. Mills, C. H. Baldwin, L. Zhao, A. Barbosa, S. Maity, and P. S. Mundada, *Quantum algorithm exploration using application-oriented performance benchmarks*, . arXiv:2402.08985.

[6] A. Mehta, S. E. Smart, J. Peetz, D. A. Mazziotti, and P. Narang, *Unitary dilation strategy towards efficient and exact simulation of non-unitary quantum evolutions*, . arXiv:2501.18697.

[7] E. Sibieude, A. Khandelwal, P. Girard, J. S. Hesthaven, and N. Terranova, *Population pharmacokinetic model selection assisted by machine learning*, *Journal of Pharmacokinetics and Pharmacodynamics* **49** (Apr., 2022) 257–270.

[8] Z. Huang, P. Denti, H. Mistry, and F. Kloprogge, *Machine learning and artificial intelligence in pk-pd modeling: Fad, friend, or foe?*, *Clinical Pharmacology & Therapeutics* **115** (Apr., 2024) 652–654.

[9] L. Zhang, P. B. Gilbert, E. Capparelli, and Y. Huang, *Simulation-based pharmacokinetics sampling design for evaluating correlates of prevention efficacy of passive hiv monoclonal antibody prophylaxis*, *Statistics in Biopharmaceutical Research* **14** (Oct., 2022) 611–625.

[10] G. Bender, J. Gosset, J. Florian, K. Tan, M. Field, S. Marshall, J. DeJongh, R. Bies, and M. Danhof, *Population pharmacokinetic model of the pregabalin-sildenafil interaction in rats: Application of simulation to preclinical pk-pd study design*, *Pharmaceutical Research* **26** (Oct., 2009) 2259–2269.

[11] T. Tatarinova, M. Neely, J. Bartroff, M. v. Guilder, W. Yamada, D. Bayard, R. Jelliffe, R. Leary, A. Chubatiuk, and A. Schumitzky, *Two general methods for population pharmacokinetic modeling: Non-parametric adaptive grid and non-parametric bayesian*, . arXiv:1303.6700.

[12] J. Grzegorzewski, J. Brandhorst, K. Green, D. Eleftheriadou, Y. Duport, F. Barthorscht, A. Köller, D. Y. J. Ke, S. De Angelis, and M. König, *Pk-db: pharmacokinetics database for individualized and stratified computational modeling*, *Nucleic Acids Research* **49** (Jan., 2021) D1358–D1364.

[13] Z. Liu and Z. Zhang, *Quantum-inspired hamiltonian monte carlo for bayesian sampling*, 2020.

[14] J. Christmann, P. Ivashkov, M. Chiurco, and G. Mazzola, *From quantum-enhanced to quantum-inspired monte carlo*, *Phys. Rev. A* **111** (Apr, 2025) 042615.

[15] S. Sun, J. Zhao, M. Gu, and S. Wang, *Variational hybrid monte carlo for efficient multi-modal data sampling*, *Entropy* **25** (2023), no. 4.

[16] F. Y. Bois, N.-H. Hsieh, W. Gao, W. A. Chiu, and B. Reisfeld, *Well-tempered mcmc simulations for population pharmacokinetic models*, *Journal of Pharmacokinetics and Pharmacodynamics* **47** (Dec., 2020) 543–559.

[17] N. J. Browning, F. A. Faber, and O. Anatole von Lilienfeld, *Gpu-accelerated approximate kernel method for quantum machine learning*, *The Journal of Chemical Physics* **157** (12, 2022) 214801.

[18] A. E. Paine, V. E. Elfving, and O. Kyriienko, *Quantum kernel methods for solving regression problems and differential equations*, *Phys. Rev. A* **107** (Mar, 2023) 032428.

[19] Y. M. Thant, T. Wakamiya, M. Nukunudompanich, K. Kameda, M. Ihara, and S. Manzhos, *Kernel regression methods for prediction of materials properties: Recent developments*, *Chemical Physics Reviews* **6** (02, 2025) 011306.

[20] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay, *Scikit-learn: Machine learning in Python*, *Journal of Machine Learning Research* **12** (2011) 2825–2830.