UNIVERSITY OF CALIFORNIA

Los Angeles

Using Particle Swarm Optimization to Find Efficient Designs for Mixed Effects Models with Sparse Grid and Predict Progression of Idiopathic Pulmonary Fibrosis using Baseline High Resolution Computed Tomography Scans with Random Forest

A dissertation submitted in partial satisfaction of the
requirements for the degree Doctor of Philosophy
in Biostatistics

by

Yu Shi

2019

ABSTRACT OF THE DISSERTATION

Using Particle Swarm Optimization to
Find Efficient Designs for Mixed Effects Models with Sparse Grid and
Predict Progression of Idiopathic Pulmonary Fibrosis using Baseline
High Resolution Computed Tomography Scans with Random Forest

by

Yu Shi
Doctor of Philosophy in Biostatistics
University of California, Los Angeles, 2019
Professor Weng Kee Wong, Co-Chair
Professor Grace Hyun Jung Kim, Co-Chair

There are many challenging optimization problems in the health sciences. Problems in health sciences are increasingly complex, and frequently the most advanced optimization techniques are required to tackle them. Researchers thus need various types of flexible optimization tools that are easily accessible and efficient. In this dissertation, we utilize a stochastic optimization technique called particle swarm optimization (PSO) and demonstrate its usefulness and flexibility using two applications in the biomedical field. For the first application, we propose a sparse grid hybridized PSO (SGPSO) algorithm to find different types of optimal or highly efficient designs for various longitudinal models. In particular, we consider non-linear mixed effects models useful in pharmacokinetic/pharmacodynamic studies and show SGPSO is a powerful tool for finding optimal or efficient designs that were previously thought to be intractable. For the second application, we propose a random forest hybridized quantum PSO algorithm for predicting disease progression of idiopathic pulmonary fibrosis (IPF) using quantitative information on high-resolution computed tomography (HRCT) imaging. IPF is a fatal type of lung disease with unpredictable functional progression at the time of diagnosis. We leverage single time point HRCT scans to predict the 6 months to 1 year follow-up status of IPF subjects. Results show that the two hybridized PSO approaches tackle important biomedical optimization problems effectively, and since the proposed methodology is not problem specific, there is potential for further applications to solve other biomedical optimization problems.

The dissertation of Yu Shi is approved.

Jonathan Goldin

Hua Zhou

Grace Hyun Jung Kim, Committee Co-Chair

Weng Kee Wong, Committee Co-Chair

University of California, Los Angeles

2019

To my family and friends

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

ACKNOWLEDGMENTS

First and foremost, I am deeply grateful for my advisors, Dr. Weng Kee Wong and Dr. Grace Kim. Nothing in this dissertation would have been possible without their support from the day I started my PhD program. They allowed me to do my research with autonomy and freedom, and guided me through the whole journey to make it enjoyable and fruitful. I am also indebted to Dr. Jonathan Goldin for taking tremendous time and efforts to provide the essential data set for Project II. I would like to thank Dr. Hua Zhou for serving in my committee. My appreciation also goes to Dr. Gang Li for serving as my mentor for the UCLA-CSST program before my PhD program, and helping me plan the coursework for the first two years of my graduate work. Special thanks to Dr. Minping Qian at Peking University who provided valuable training opportunities in my early career.

I would like to express my gratitude to the UCLA biostatistics community, and special thanks to Juhyun Kim for countless days and nights we spent together in the library. I have not only learned practice and profession of statistics, but also met good friends who helped me thrive in this journey. I am also thankful for my experiences at computer vision and imaging biomarkers laboratory. The excellent infrastructure, stimulating work environment, and experts in house are keys to the success of Project II. I want to acknowledge the support of France Mentré, Thu Thuy Nguyen, Jérémy Seurat, and Sebastian Ueckert for their help on developing the software for Project I.

Lastly, I wish to thank my family and friends. I am forever thankful for my friends who keep me happy and cheerful. Special thanks to Di, Xinke, Gao, and Haoli for being there with me during hard times. I also wish to thank my parents for their unconditional love and support throughout my entire life. Not a single moment they stop believing in me, and their unwavering trust has been my source of strength.

This research is partially sponsored by National Institute of General Medical Sciences under the Award Number R01GM107639, the National Heart, Lung, and Blood Institute of the National Institutes of Health under the Award Number R21HL123477-01A1 and R21HL140465-01A1, UCLA graduate summer research mentorship, and UCLA disserta-

tion year fellowship. Sections 1.2 and 3.5.4 were adapted from *Yu Shi, Zizhao Zhang, and Weng Kee Wong. "Particle swarm based algorithms for finding locally and pseudo-Bayesian D-optimal designs." Journal of Statistical Distributions and Applications, 6(1):3, Apr 2019.* Sections 1.3 and 4.1-4.4 were adapted from the *Yu Shi, Weng Kee Wong, Jonathan Goldin, Matthew Brown, and Grace Hyun Kim. "Prediction of Progression in Idiopathic Pulmonary Fibrosis using CT Scans at Baseline: A Quantum Particle Swarm Optimization - Random Forest Approach." Artificial Intelligence in Medicine. (Under review).* Sections 3.3, 3.4, 3.6.1 and 3.6.2 were adapted from *Yu Shi, Thu Thuy Nguyen, Jérémy Seurat, France Mentré, and Weng Kee Wong. "Particle swarm optimization for constructing optimal designs in pharmacometrics." (to be submitted).* Sections 3.2, 3.6.3, and 3.6.4 were adapted from *Yu Shi and Weng Kee Wong. "Sparse grid - particle swarm optimization for finding optimal designs of non-linear mixed effects models." (in preparation).* Section 4.5 was adapted from *Yu Shi, Jonathan Goldin, Weng Kee Wong, and Grace Kim. "Prediction of Progression using Quantum Particle Swarm Optimization at Single Time Point HRCT Scans in Subjects with IPF." (in preparation).*

VITA

2015	B.A. (Economics), Peking University.
2015-2019	Graduate Student Researcher, Radiological Sciences Department, UCLA.
2015-2018	Graduate Student Researcher, Biostatistics Department, UCLA.

PUBLICATIONS AND PRESENTATIONS

Yu Shi, Zizhao Zhang, and Weng Kee Wong. “Particle swarm based algorithms for finding locally and pseudo-Bayesian D-optimal designs.” Journal of Statistical Distributions and Applications, 6(1):3, Apr 2019.

Yu Shi, Weng Kee Wong, Jonathan Goldin, Matthew Brown, and Grace Hyun Kim. “Prediction of Progression in Idiopathic Pulmonary Fibrosis using CT Scans at Baseline: A Quantum Particle Swarm Optimization - Random Forest Approach.” Artificial Intelligence in Medicine. Under review.

Grace Kim, Stephen Weight, John Belperio, Yu Shi, and Jonathan Goldin. “Prediction of Idiopathic Pulmonary Fibrosis Progression Using Early Quantitative Changes on CT Imaging for a Short Term of Clinical 18-24 Month Follow-ups.” European Radiology. Under review.

Yu Shi, Thu Thuy Nguyen, Jérémy Seurat, France Mentré, and Weng Kee Wong. “Particle swarm optimization for constructing optimal designs in pharmacometrics.” (to be submitted)

Yu Shi and Weng Kee Wong. “Sparse grid - particle swarm optimization for finding optimal designs of non-linear mixed effects models.” (in preparation)

Yu Shi, Jonathan Goldin, Weng Kee Wong, and Grace Kim. “Prediction of Progression using Quantum Particle Swarm Optimization at Single Time Point HRCT Scans in Subjects with IPF.” (in preparation)

UCLA provisional patent application 2019-731-1. Grace Hyun Kim, Yu Shi, Jonathan Goldin, Weng Kee Wong. Early Prediction of Progression in Idiopathic Pulmonary Fibrosis using a Single Time Point HRCT Scan.

Yu Shi, Jonathan Goldin, Weng Kee Wong, Joshua Lai, Matthew Brown, Grace Kim. Validation of IPF Prediction Model Using Quantum Particle Swarm Optimization Hybridized Random Forest. Radiological Society of North America (RSNA) Annual Meeting. Chicago, USA, 2018.

Yu Shi, Jonathan Goldin, Weng Kee Wong, Joshua Lai, Matthew Brown, Grace Kim. HRCT texture feature selection and imaging pattern prediction of IPF using quantum particle swarm optimization. Radiological Society of North America (RSNA) Annual Meeting. Chicago, USA, 2017.

Yu Shi, Zizhao Zhang, Weng Kee Wong. Sparse grid hybridized PSO for finding pseudo-Bayesian optimal designs. Conference on Latest Advances in the Theory and Applications of Design and Analysis of Experiments, Banff, Alberta, Canada, 2017.

CHAPTER 1

Introduction

Optimization is widely involved in many public health problems. Researchers often attempt to solve problems that involve sophisticated objective functions without easily exploitable mathematical properties. Thus an optimization tool should be helpful to public health researchers. Ideally the tool is versatile and able to handle various objective functions and constraints. It also should be computationally efficient and easy to implement. In this work, we propose particle swarm optimization (PSO) as a generic optimizer to tackle hard optimization problems in health sciences, and demonstrate its usefulness and flexibility using two applications in the biomedical field.

Researchers use PSO for various public health related problems. [FLW09] used PSO to choose screening spots at public transportation stations during period of infectious disease outbreak so that population covered by screening can be optimized. [IP12] used PSO to optimize the scanning window to determine disease clusters. [WMN15] used PSO to optimize an environmental/economic dispatch problem to reduce emissions that cause air-borne contaminants due to electricity generation. [CM11] used PSO to allocate health infrastructure in a living area so that routes to infrastructures are optimized. [LD13] used PSO to optimize the architecture and weights of the artificial neural network to identify the factors related to common mental disorders. [SRJ13] used PSO to train a risk estimation technique to predict spatial PM10 concentrations at un-sampled locations using sampled and measured location information. [KSD15] used PSO to study geographical pattern of cardiac attack happened in public areas so that cardiac rehabilitation defibrillators can be installed accordingly. There are also many others not listed above.

In this dissertation, we use PSO hybridized with other techniques to tackle two difficult problems in biomedical sciences. The first project focuses on finding efficient exact designs and pseudo-Bayesian optimal approximate designs for models commonly used in biomedical studies, and in particular, we try to find efficient designs for non-linear mixed effects models (NLMEMs). Design issues are rarely discussed for NLMEMs and one major challenge is to handle multi-dimensional numerical integration nested in the optimization process. It is critical to set up efficient yet robust designs for this class of models, as experimental costs keep rising and there is a constant need to reduce the number of repeated measures from subjects, especially when there are neonates, infants and/or children [DPK11]. Efficient data collection designs are critically important in making meaningful statistical inferences. By using optimal design theory, we can help scientists produce the most reliable results that are more likely to translate into useful biomedical applications at minimum cost.

The second project aims to accurately predict the progression outcomes of idiopathic pulmonary fibrosis (IPF) disease subjects using single time point high-resolution computed tomography (HRCT) images. IPF is a progressive and ultimately fatal lung disease with heterogeneous progression status. From 2001-2011, among newly diagnosed IPF subjects with Medicare, the median survival time was 3.8 years [RCY14]. The prevalence estimate of IPF in the US ranges from 14.0 to 42.7 per 100,000 persons depending on the case definition used [RCE11]. The disease exhibits a highly heterogeneous natural history, and time to the disease progression is not predictable at the time of diagnosis: some subjects may experience episodes of acute respiratory worsening even though the disease was stable before [RCE11]. Therefore it is critical to distinguish subgroups of IPF subjects who are expected to progress from those who are expected to stay stable. The identification of subgroups helps clinicians decide whether to switch the current treatment, or refer the subject for a lung transplantation at an early stage. HRCT imaging has shown its potential in detection of lung disease pattern at an early stage [RSD16]. However, HRCT data are usually hard to analyze because they are heterogeneous, spatially correlated, and high dimensional. Using HRCT information, and in particular, using only single time point HRCT information, to predict IPF progression is both clinically important and technically challenging.

1.1 Aims and novelty

In this study, we hybridize PSO with other techniques to tackle these two problems, that both involve hard optimization process. Specifically, the first project hybridizes PSO with a sparse grid technique that finds exact and pseudo-Bayesian optimal approximate designs for various generalized linear models (GLMs) and non-linear mixed effects models. In the second project, we propose a random forest hybridized PSO algorithm to mine the rich information in HRCT images and build a predictive model for IPF subjects. Thus the dissertation work has two aims that both use PSO for the core optimization process:

1. Build a computational tool that finds efficient exact designs and pseudo-Bayesian optimal approximate designs for GLMs and NLMEMs efficiently, under different model specifications. Pseudo-Bayesian optimal designs are obtained from averaging the classical design criteria over the parameter space [RDM16]. The tool will be flexible and facilitate us to find more efficient designs more effectively than those recently reported in the literature [RMB02, ME06, WLE06, GJS09, NM14a, RUM16]. PSO is utilized because the closed-form solutions are unavailable and the numerical integration nested in the optimization process makes the objective function computationally hard to tackle.

2. Collect a labeled training data set on baseline HRCT images and develop a model that predicts the IPF disease progression at the follow-up visits. We implement a study design that makes it possible to collect a labeled training set at regions of interest (ROI) level to build a supervised learning model. We develop a feature selection and classification algorithm that selects a small subset of HRCT texture features and gives superior classification performance relative to other commonly used algorithms. We train and validate the algorithm at the ROI level and expand the algorithm to the whole lung level so that the algorithm can be more readily applicable in practice. An advanced version of PSO is utilized to deal with the feature selection process involved because the objective function is not analytically available.

The two projects are novel and results of the proposed work have potential impact in practice. For project I, we develop a novel method for finding efficient designs for non-linear mixed effects models for which the efficient designs have remained largely elusive. Finding efficient designs for such models will likely result in more efficient biomedical studies in practice. For project II, our study design makes it possible to collect labeled ROI information at baseline scans. The integrated predictive approach of radiological imaging texture feature selection and pattern prediction is new and has potentially superior classification performance to traditional filter methods and many wrapper methods. We show that it is possible to predict the IPF disease progression using only single time point HRCT images: clinicians should be able to make more informed and timely medical decisions on IPF treatment based on the prediction results.

The rest of the dissertation is organized as follows. Section 1.2 and 1.3 cover background knowledge and the motivations of the two projects. Chapter 2 covers the basics of PSO, including its variants and applications. Project I is discussed in chapter 3. Section 3.1 briefly reviews algorithms for finding optimal designs. Section 3.2 discusses the sparse grid technique used to tackle multi-dimensional integration process. Section 3.3 discusses non-linear mixed effects models and the difficulty in finding efficient designs for such models. Section 3.4 proposes the main methodology for Project I. We find pseudo-Bayesian optimal approximate designs for GLMs in section 3.5, and efficient exact designs for non-linear mixed effects models in section 3.6. Project II is discussed in Chapter 4. Section 4.1 is a brief literature review on the project background. Section 4.2 covers the implementation of study design and data acquisition that enabled a supervised learning approach. The methodology is proposed in section 4.3. The preliminary results at the ROI level are discussed in section 4.4. Section 4.5 discusses an expansion of the ROI level to the whole lung level in predicting progression. Chapter 5 concludes the dissertation, with some limitations and future work direction discussed.

1.2 Optimal designs

Designing efficient experiments is often the first step in a scientific study. To design an experiment and extract scientific conclusions from it, scientists often need to utilize optimal design theory to make decisions on data collection strategy. Optimal designs are a class of experimental designs that are optimal with respect to some statistical criteria. Biomedical researchers have applied optimal design theory and constructed various efficient studies in biomedicine, such as in dose-response studies and design of early phase clinical trials. For longitudinal models, there are some advance in this area, see for example, [OGG05, OTB02, PS16, JM16].

Finding optimal designs often involve complicated numerical computation, and designs that involve numerical integration are especially challenging. One of such problem is finding pseudo-Bayesian optimal designs. A pseudo-Bayesian approach accounts for model uncertainty by specifying prior distributions to model parameters [CL89], and pseudo-Bayesian optimal designs are usually more robust to model mis-specification [DPW09]. Another such problem is finding efficient designs for non-linear mixed effects models. The evaluation of Fisher information matrix for such models often involve multi-dimensional integration over the random effects, which makes it expensive to evaluate the quality of one design. For some complicated models, existing work usually avoids direct optimization and compares a small number of candidate designs. This approach is simple but is likely to miss the efficient design over the design space and therefore lose efficiency. In our work, we propose to use PSO to search the design space, and use a sparse grid technique to evaluate numerical integration nested in the optimization processes.

We consider both exact and approximate designs in this work. For a pre-fixed sample size n and a given criterion, an exact design specifies the number of observations to take at each support point and where these support points are. There is no general method to verify the optimality of exact designs. An alternative is to work with optimal approximate designs, which for a fixed sample size, determine the total number of support points (k), the locations of the design points and the proportions (p_i's) of observations to take at these points. In practice, the number of observations at the point x_i is np_i^* after rounding each np_i to the nearest integer np_i^*, $i = 1, \dots, k$ and subject to the constraint $np_1^* + np_2^* + \cdots + np_k^* = n$. In contrast to optimal exact designs, there is a theoretical and unified framework for finding optimal approximate designs, including established algorithms for finding many of them, see [KBO85].

The general framework of optimal designs for non-linear models is given as follows. Let $Y(x)$ be the response and let $\eta(x;\theta)$ be its nonlinear mean function. We assume that

$$Y(x) = \eta(x;\theta) + \epsilon, \quad \epsilon \sim N(0, \sigma^2), \quad x \in X,$$

and experimental subjects are independent. Following convention, the worth of a design ξ is gauged by its Fisher information matrix (FIM). The information matrix $M(\theta, \xi)$ is proportional to the negative of the expectation of the second derivatives of the log-likelihood function with respect to the model parameters in the mean function. It depends on the model, parameters in the mean function and the variance structure, and also on the model parameters when the model is nonlinear. The FIM plays a crucial role in the determination of optimal designs.

The optimality design criterion is usually formulated as a convex or concave function of the information matrix. The popular D-optimality criterion is formulated as the negative of the log determinant of the information matrix and it is a convex function [Fed72]. The D-optimal design minimizes the volume of the confidence ellipsoid of the model parameters when errors are independent and normally distributed. This means that the D-optimal design provides the most accurate inference for the model parameters. The D-optimal design for estimating all model parameters using θ_0 as nominal values is the design ξ_D that maximizes the determinant of information matrix

$$\xi_D = \underset{\xi \in \Xi}{\text{argmax}} \{\log(\det[M(\theta_0, \xi)])\}, \tag{1.1}$$

where Ξ is the set of all feasible designs. The larger the criterion values is, the better is the design. When prior information on the model parameter θ is available in the form of a density $\pi(\theta)$, a pseudo-Bayesian D-optimal design ξ_{BayesD} maximizes the same criterion after averaging out the model parameters with respect to the prior density. It is defined by

$$\xi_{BayesD} = \underset{\xi \in \Xi}{\text{argmax}} \int \{\log(\det[M(\theta, \xi)])\} \pi(\theta)\, d\theta, \tag{1.2}$$

and as before, the larger the pseudo-Bayesian D-optimality criterion values is, the better is the design. Clearly, when the prior density is degenerate, the resulting design becomes locally D-optimal. Both criteria are appropriate for estimating model parameters.

When the design criterion is convex in Ξ, an equivalence theorem is available to verify optimality of a design over all designs on the design space X. Such a theorem comes from directional derivative considerations of a convex functional and is discussed in design monographs like [Fed72, BW05]. For example, if m is the dimension of θ and δ_x is a design that takes all observations at x, it can be shown that the design ξ_D is locally D-optimal if and only if

$$tr\{M(\xi_D, \theta)^{-1} M(\delta_x, \theta)\} - m \leq 0, \quad for\ all\ x \in X,$$

with equality at the support points of ξ_D. The function on the left is sometimes called the sensitivity function. Different convex design criteria lead to different sensitivity functions but they all have a similar form. In practice, the optimality of a design is verified by plotting the sensitivity function against the design space and checking whether the equivalence theorem is satisfied. If it is not, the design is not D-optimal. Similarly, ξ_{BayesD} is a pseudo-Bayesian D-optimal with respect to the prior density $\pi(\theta)$ if and only if

$$\int_\theta tr\{M(\xi_{BayesD}, \theta)^{-1} M(\delta_x, \theta)\} \pi(\theta) d\theta - m \leq 0, \quad for\ all\ x \in X,$$

with equality at the support points of ξ_{BayesD}.

We compare designs using relative efficiency, which is commonly defined as the ratio of the two criteria values or a function of the ratio for better interpretation. When one of the designs been compared is the optimal design, relative efficiency becomes the design efficiency. For example, if the θ_0 is the vector of nominal value for the nonlinear model and the criterion is D-optimality, we compare two designs ξ_1 and ξ_2 via the m^{th} root of the ratio of the determinants of their information matrices:

$$\left\{\frac{|M(\xi_1, \theta_0)|}{|M(\xi_2, \theta_0)|}\right\}^{1/m}.$$

The relative efficiency ratio compares performance of the two designs for estimating the model parameters. If the above ratio 0.5 or 50% efficiency, this means that the design ξ_1 needs twice as many observations for it to do as well as the design ξ_2. When ξ_2 is the D-optimal design, the above ratio is simply the D-efficiency of the design ξ_1. A pseudo-Bayesian efficiency lower bound on the D-efficiency of a design ξ is defined as $e^{-s_m/m}$, where s_m is the maximum positive value of the sensitivity function across the design space [Paz86].

In this work we develop a hybridized algorithm that combines sparse grid and PSO to efficiently find optimal designs that involve numerical integration and optimization. We implements our algorithm and demonstrate its efficiencies for finding pseudo-Bayesian optimal design and efficient exact designs for non-linear mixed effects models. In particular, we show that our algorithm is able to find pseudo-Bayesian optimal approximate designs and their optimality can be verified by the equivalence theorem. For exact designs, there is no theoretical tool to confirm optimality of the design. In this case, we compare our designs with designs reported in the literature or designs currently adopted in clinical trials and show that our algorithm can find more efficient designs for the given setting or in a broader setup.

1.3 Idiopathic pulmonary fibrosis

Idiopathic pulmonary fibrosis (IPF) is a fatal lung disease characterized by an unpredictable progressive decline in lung function. Natural history of IPF is unknown and the prediction of disease progression at the time of diagnosis is notoriously difficult. Because of the heterogeneous natural history of IPF, scientists and clinicians have devoted many efforts to build predictive tools to diagnose and treat IPF subjects [RCE11, RRM18]. Some predictive clinical models have been developed but few incorporate HRCT imaging data. HRCT plays an important role in diagnosis and detection of IPF. HRCT provides a non-invasive, 3D perspective of lung morphology with disease distributions and can reflect regional anatomic changes over time. Studies have shown that HRCT features can be good representation of disease severity [KBC15] and they are considered useful and sensitive to predict survival in subjects with IPF [KBW16]. A predictive model for IPF progression at voxel-wise level has never been developed using only baseline HRCT scans. Mainly, there are two challenges: (a) obtaining a data set of features for region of interest (ROI) on baseline HRCT scans and their follow-up status; and (b) simultaneously selecting important features from high-dimensional space, and optimizing the prediction performance. We resolved the first challenge by implementing a study design and having an expert radiologist contour ROIs at baseline scans, depending on its progression status in follow-up visits. For the second challenge, we integrated the feature selection with prediction by developing an algorithm using a wrapper method that combines quantum particle swarm optimization to select a small number of features with random forest to classify early patterns of progression.

In this dissertation, we build a predictive model using only baseline (single time point) HRCT information. We seek to leverage the HRCT imaging data to build a predictive model that 1) select a small subset of HRCT features; 2) has high prediction accuracy; and 3) achieves balanced sensitivity-specificity. The main difficulty in selecting HRCT texture features is that these features are usually high-dimensional and are correlated in a complicated manner (see figure 1.1 for a visualization), and many commonly used methods such as forward/backward selection or penalized function can search only a fraction of feature space, but often not able to quantify important interactions between features [Tib96, FL01].

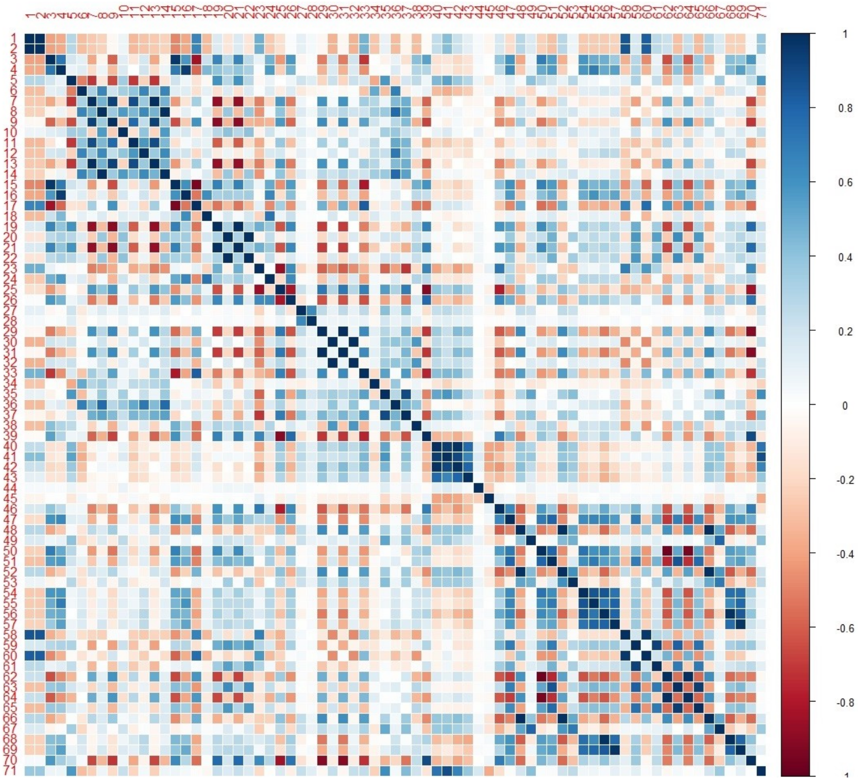


Figure 1.1: The extracted 71 texture features on X and Y axis and the color shades show the magnitude of their paired correlations. The features are correlated in a complicated manner.

Intuitively, the texture features distributions of progressed IPF subjects and stable IPF subjects seem to be different. Figure 1.2 shows the distributions of a few statistical HRCT texture features of progressed vs stable groups. An integrated model that leverages these differences can be useful in predicting the disease outcome.

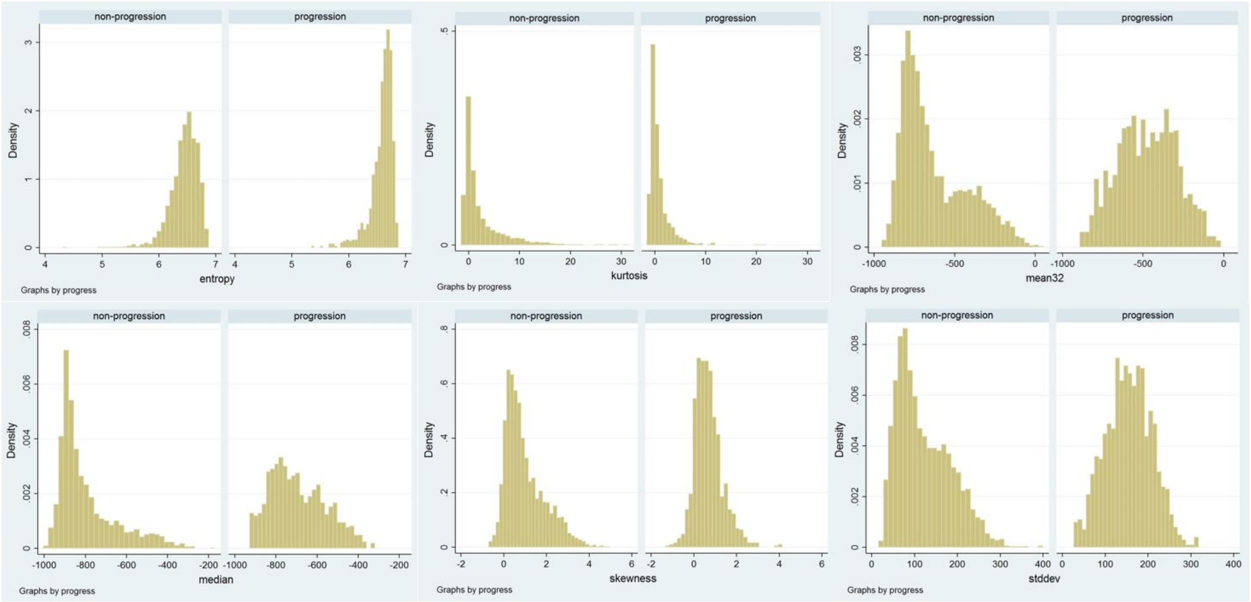


Figure 1.2: Histograms of descriptive statistics at population level. Metrics stratified by progression status: entropy (left upper), kurtosis (middle upper), mean32 (right upper), median (left lower), skewness (middle lower), standard deviation (right lower). The two groups have different distributions.

We first collect data at the ROI level labeled by an experienced radiologist for model training, and then seek to expand the model to predict at the whole lung level. The prediction in whole lung eliminates the bias of ROI contouring and makes the prediction more robust. The expansion could be automated in the prediction process with little radiologist input. The ROI level progression status is based on the examination of IPF subjects' baseline versus follow-up scans ("progression" or "non-progression"). The whole lung level outcome is defined as the quantitative lung fibrosis (QLF) score change between the baseline and follow-up images. QLF is a classifier-model-derived score and is a measure of IPF disease extent [KTC10]. Research has shown that quantitative scores that are summated for the whole lung correlates with PFT results, and is related to extent and severity of interstitial lung disease [TVT16]. We aim to extract one *baseline* metric for each subject that is associated with the *follow-up* QLF changes at the whole lung level.

CHAPTER 2

Particle swarm optimization algorithm

Research on optimization methods has a long history, and is still an area under much study nowadays. The mathematical optimization theories have been well developed for many algorithms. In particular, the theories for convex optimization are particularly well developed, and these methods usually assume convexity, differentiability or twice differentiability of the objective function or functions. Even then, there is no guarantee that a derivative-based method will converge to the global optimum when the objective function lacks good properties [Sho13]. When assumptions on the objective function is not tenable, metaheuristic algorithms are useful. Metaheuristic algorithms typically do not impose any assumption on the objective functions; in particular, they do not assume the functions to be convex or differentiable.

Metaheuristics mean “to discover by trial and error”. They have not made their way into mainstream statistics yet, but they are increasingly used in various disciplines, including in public health research because of their flexibility and versatility. Researchers in engineering and computer science have demonstrated their ability to solve different types of hard optimization problems that may be high-dimensional or the objective function is non-convex or non-differentiable. These algorithms typically involve a combination of randomness and local searches. The randomization process makes these algorithms non-deterministic even when the inputs are the same each time, but end up with frequently not too different results [Yan10]. There is no guarantee that the best solutions can be found as the mathematical properties of these algorithms remain unclear. The hope is that the algorithms find quality solutions to a tough optimization problem in a reasonable amount of time, and can also be validated by other metaheuristic algorithms.

This dissertation focuses on one of a relatively recent metaheuristic algorithms called Particle Swarm Optimization (PSO). PSO was first developed in 1995 by Kennedy and Eberhart [EK95]. Motivated by swarm intelligence, PSO simulates candidate designs using them as birds in a swarm looking for food on the ground. The swarm collectively acts and communicates to update where each bird believes where the food is (personal best) and flies toward it in the direction of the group best, which is where the flock believes the food lies after sharing information within the flock. The best candidate design gets updated at each iteration as the bird flies over time in search of a quality solution. PSO has been widely used in machine learning and data mining research in computer science and engineering. Research has shown that PSO has the same effectiveness (finding the true global optimal solution) as Genetic Algorithm (GA), which is another very popular metaheuristic algorithm, but with significantly better computational efficiency (less function evaluations) [HCD05]. GA was inspired by the principles of genetics and evolution, and mimics the reproduction behavior observed in biological populations [BGH89]. In general, the main advantages of PSO are its flexibility and simplicity of its operators [Eng06, BE06]. Below is the pseudo-code for PSO [Ken06]:

```
Begin
    Initialize particle position and velocity
While maximum iterations or minimum error criteria is not attained
Do
    For each particle
        Evaluate objective function
        Update particle personal best
    End
    Set particle with the best objective function value as the group best
    For each particle
        Update particle velocity: $v_{id}^{t+1} = wv_{id}^{t} + c_1\psi_1(p_{id}^{t} - x_{id}^{t}) + c_2\psi_2(p_{gd}^{t} - x_{id}^{t})$
        Update particle position: $x_{id}^{t+1} = x_{id}^{t} + v_{id}^{t+1}$
    End
End
```

In the pseudo-code, w is inertia weight, v_{id}^{t} and x_{id}^{t} are velocity and position of d^{th} dimension of i^{th} particle at iteration step t, c_1, c_2 are weight constants, ψ_1 and ψ_2 are random values from uniform$[0, 1]$ distribution, p_{id} is the personal best of d^{th} dimension of particle i (the best position particle i ever visited), p_{gd} is the d^{th} dimension of the group best of the swarm (the best position the group ever visited).

There are many applications of PSO in the field of engineering, computer science and public health. We briefly review some of its applications in public health to demonstrate its success in solving real-world optimization problems. We also briefly review the use of PSO on feature selection problems. Further, since its first introduction in 1995, there are now many variants of PSO with various enhancements. The enhancements are motivated by both the theoretical study of PSO movements and empirical experiments with many benchmark functions and real-world experience. We briefly summarize some of the enhancements, and choose one of the variants, named quantum-inspired particle swarm optimization, to tackle project II which leads to superior results.

2.1 Application of particle swarm optimization (PSO)

2.1.1 PSO for optimization in public health

Optimization techniques are increasingly used in public health research. Many problems can be formulated as an optimization problem, for example, we may want to develop a policy to maximize facility utilization or minimize paths to optimal health infrastructure. Examples can be found in all aspects of public health research, from experimental design [KGR07] and data collection [PPM14] to data analysis and decision-making [DB10]. Public health data is often large and heterogeneous, and the problems researchers want to solve often involve sophisticated objective functions that may not be convex, differentiable or continuous. Consequently many traditional and effective mathematical programming tools may not be applicable to solve the optimization problems in public health. A desirable optimization tool should be one that is capable of solving a wide range of problems, computationally efficient, and easy to implement so that practitioners can use easily without having to have special expertise in optimization algorithms.

Particle swarm optimization (PSO) is an ideal tool that can serve as a black box where researchers can easily input their objective functions and constraints in and the algorithm is able to generate optimal or near-optimal solutions. Researchers have used PSO to solve various optimization problems. [IP12] used PSO to optimize the scanning window to determine disease clusters, which is helpful to develop an early outbreak detection system for unusual public health events. The method has capabilities to find irregular shaped clusters, with higher likelihood ratio, in comparison with traditional approach. [WMN15] used PSO to optimize an environmental/economic dispatch (EED) problem in order to reduce emissions that cause air-borne contaminants due to electricity generation. The results show that the approach is effective in minimizing the overall electricity production cost over a given period with considerations on both fuel cost and emission factors. [FLW09] used PSO to choose screening spots at public transportation stations during period of an infectious disease outbreak so that an appropriate segment of the population can be optimally screened for further spread of the disease. [CM11] used PSO to allocate health infrastructure in a living area so that routes to infrastructures are optimized. [LWJ12] used PSO to solve a weighted multi-objective land-use zoning problem with constraints. PSO is also increasingly used in bioinformatics, which is an emerging interdisciplinary research area that holds great promise in the advancement of research and development in public health and related disciplines; examples include RNA secondary structure prediction, gene clustering, phylogenetic tree construction, energy minimization and protein modeling [AS15].

Like other metaheuristic algorithms, PSO has also been used with other methods to enhance its performance. For example, [LD13] used PSO to optimize the architecture and weights of the artificial neural network (ANN) in order to identify the factors related to common mental disorders, and showed that when PSO is appropriately hybridized with another algorithm, it has a higher classification accuracy compared to ANN alone. [ZLB10] used PSO to choose optimal parameters for kernel functions of support vector machine (SVM) model in order to identify influential determinants of effects of children's health and socioeconomic status on educational attainment, The approach is efficient in performing dimensionality reduction and identifying important features. Similarly, [SA14] used PSO as an optimization technique for parameters selection for least squares support vector machine in order to classify type II diabetes subjects and the results show the superiority of the hybrid approach in terms of classification accuracy and robustness. Likewise, [SRJ13] used PSO to train a risk estimation algorithm to predict spatial PM10 concentrations at un-sampled locations using sampled and measured location information and the result shows that the hybrid approach is more accurate for modeling uncertainty in the process prediction of PM10 data.

2.1.2 PSO for feature selection

In feature selection, there are broadly two types of algorithms: filter methods and wrapper methods. Filter methods assess the relevance of features by looking only at the intrinsic properties of the data [YL06, PLD05]. For example, step-wise regression adds or eliminates individual variables sequentially until there is no statistically significant change in model performance. The selected variables then serve as inputs to the classification algorithm. Filter methods separate feature selection from classification, and typically, feature selection is performed only once before the classification task.

For filter methods, there are different optimality criteria: commonly used are model-based Akaike information criterion (AIC) and Bayesian information criterion (BIC). AIC and BIC incorporate a measure of accuracy penalized by the number of features selected. Many greedy search algorithms such as forward inclusion, backward elimination, etc. suffer instability in the selection process [Bre96] and not effective for high-dimensional data. These algorithms only search for a fraction of all predictor combinations, as the possible subsets number $\binom{k}{q}$ increase exponentially with k, making it an NP-hard problem, and thus may only achieve a local optimum.

Model-based feature selection has different format of penalized terms. The L_0-penalized least squares takes the form:

$$\underset{\beta}{\operatorname{argmin}}\,||y - X\beta||_2^2 + n\lambda^2 I(|\beta_j| \neq 0),$$

where AIC sets $\lambda = \sigma\sqrt{2/n}$, BIC sets $\lambda = \sigma\sqrt{log(n)/n}$. For large p and small n problems, L_1-penalized least squares, which is often termed as regularized regression methods are commonly employed. The most popular least absolute shrinkage and selection operator (LASSO) [Tib96] takes the form

$$\underset{\beta}{\operatorname{argmin}}\,||y - X\beta||_2^2 + \lambda\sum_j^d |\beta_j|,$$

where users can tune λ. If the response is binary then a logistic link is employed

$$log\frac{P(y_i = 1)}{P(y_i = 0)} = X\beta.$$

Alternatively, a revised regularized regression method that uses L_0-penalized least squares term is smoothly clipped absolute deviation penalty (SCAD) [FL01], which takes the form:

$$\underset{\beta}{\operatorname{argmin}}\,||y - X\beta||_2^2 + \lambda\{I(X\beta \leq \lambda) + \frac{(a\lambda - \beta)_+}{(a-1)\lambda} I(X\beta > \lambda)\}.$$

SCAD penalty is a smooth transition from L_1 penalty to L_0 penalty. It has so-called "oracle" property that includes unbiasedness, sparsity and continuity.

The main advantage of filter techniques is that they are computationally simple and fast. However, filter methods ignore the interactions of the feature selection step and the classification step, which may result in a compromised classification performance [SIL07]. Different from filter methods, wrapper methods evaluate predictors holistically using procedures that add and/or remove predictors simultaneously, in order to find a combination that optimizes the performance [CYK08, MPD15]. Wrapper methods treat different feature subsets as inputs and the performance of the models as outputs [JKP94, LY05]. There are two components in a wrapper method: an optimizer and a classifier. The metrics from the classifier serve as the objective function for the optimizer to search the best feature subset. Compared to filter methods, wrapper methods are usually computationally expensive because many candidates of subsets have to be evaluated against the previous best subset [KJ97]. Wrapper methods also have a higher risk of over-fitting due to the lack of established criteria of a penalized function [SIL07]. On the other hand, wrapper methods are able to directly relate feature subsets to the classification performance. They include the interaction between feature subset search and model selection, and they are able to take into account feature dependencies [SIL07].

Evolutionary computation (EC) has received much attention from the feature selection community because of the good global optimization properties of many state-of-the-art EC-based feature selection techniques. Compared with traditional searching methods, EC techniques do not need domain knowledge and do not make any assumptions about the feature space, such as linear separability and differentiability [XZB16]. EC algorithms such as genetic algorithms and PSO are increasingly used for feature selection purposes [XZB16], mostly in a wrapper setting. EC searches the feature space efficiently, optimizing single or multi-objective functions of different classification performance metrics.

PSO is one of the most used EC in the feature selection task [XZB16]. Typically PSO is coupled with a classifier to form a wrapper. Examples of the classifiers include support vector machines [UMC11, AGJ07, ZH05, TSY05, ACC07, HD08]; K-nearest neighbors [XZB12, RSF12, ZGH15]; artificial neural network [AC02, VMF12, HH07]; etc. Some variants of PSO were also specifically proposed for the feature selection task [XZB14].

We propose a wrapper method that incorporates a quantum-inspired PSO (QPSO) as the optimizer and uses a random forest as a classifier to build a novel wrapper for HRCT texture feature selection and IPF prediction. We performed a comparative study and show that QPSO yields superior performance.

2.2 Algorithm properties and enhancements

PSO has been shown in the literature to have advantages over many other metaheuristic algorithms. As an example, [ANA15] shows that PSO outperforms genetic algorithm, ant colony, artificial bee colony, cuckoo search, and other swarm intelligence algorithm on 30 benchmark functions. A very detailed, comprehensive review of PSO and its properties, modifications, and hybridization is provided in [BM17]. In this section, we only briefly discuss some of the algorithm properties and enhancements.

We call the PSO version proposed in [SE99] as the standard PSO. The standard PSO has two theoretical limitations: convergence and transformation invariance [BM14]. For convergence, it is proven that the PSO is only locally convergent for one-dimensional but not higher dimensional problems [SW15]. Transformation invariance means that the algorithm performance is independent of how the coordinate axes are placed on the search space. Analysis show that standard PSO is rotation variant [WKG07]. In practice, however, people concern mostly about the quality of the solution found, and how fast the optimization algorithm is able to find the solution. Most algorithm comparisons are based on these two criteria.

Numerous variants have been proposed since the inception of PSO in 1995. In the year of 2014 alone there are more than 2000 publications on PSO [BM17]. Many studies of the original PSO inspired the modification of the algorithm and led to different variants that improves different aspects of algorithm performance. We briefly summarizes some research work on the topology, parameter selection (including inertia weights, coefficients c_1 and c_2), and the velocity and position update rule. We work with standard PSO for project I because it is computationally efficient and we could obtain optimal or highly efficient designs. We choose to work with one of the variants called quantum-inspired particle swarm optimization in project II to obtain superior optimization performance.

1. Topology. This may be viewed as strategies to determine neighboring particles that contribute to the velocity update rule. Examples include global best topology (the origi-

nal PSO), ring topology, wheel topology, pyramid typology, etc. [CE14] showed that the topology does not affect the convergence boundaries, but it might affect the speed of convergence/divergence. [Eng13] claims that the performance of particles depends largely on the problem at hand and general claims about topologies are not correct.

One variant which considered hierarchical structure used a tree structure to rank the particles and update the velocity based on each particle's parent node [JM05]. Another variant, "scale-free fully informed particle swarm optimization" [ZY11], uses the analogue of weak network of social network and categorized particles into active and inactive groups. The inactive particles are not updated until they become activated by connecting to one of the existing active particles at each iteration. Another variant uses an analogue of "six degree of separation" in sociology [GZ13]. Particles update different dimensions using different network of particles. Empirical comparison on limited number of benchmark functions show superior performance of such variants over the original PSO.

An extensive comparison based on 60 benchmark functions between the global best and local best topologies showed similar results with a slight favor toward the global best topology; it also emphasizes that the best topology depends on the problems at hand [Eng13]. In our work, we use the global best topology and show that it performs well on our problems at hand in general.

2. Coefficients. The selection of coefficients is critically important in particles' behavior, because they affect their exploration and exploitation abilities, theoretical local convergence properties, and empirical performance. These coefficients include the inertia weight and acceleration coefficients. Inertia weight, originally introduced in [SE99], controls the influence of the previous velocity on the updated velocity. Acceleration coefficients are c_1 and c_2 in the velocity update rule. Some convergence analyses have shown that certain parameter settings enable convergence to a point in probability [OM99, CK02, Tre03, JLY07], and such parameters greatly affect the convergence speed [BE06]. These convergence analyses are usually simplified by ignoring the randomness that comes from the coefficients ψ_1 and ψ_2. Moreover, these parameters affect the oscillation patterns of particles (smooth or zigzag) [Tre03]. Empirically, many studies found that the best values for the inertia weight is

somewhere between 0.4 and 1.1 and the best values for the acceleration coefficients falls in 1.5 to 3, for most experiments [Cle99, SE98a, ES00]. When the acceleration coefficients are fixed, larger values of the inertia weight result in a better exploration and smaller values of the inertia weight result in a better exploitation. [SE98a] proposed a linear decreasing inertia weight from 0.9 to 0.4 and showed that such a strategy has good performance. Similar adaptive coefficients were also proposed in [ZZL09, NES11, LY12].

3. Velocity and position update rule. Inspired by the study on the trajectory of the original PSO particles, there are many variants of PSO that change the velocity and position update rule. For basic PSO and many of its variants, the particles follow trajectories and continuously aim for the local optima. Sometimes, this can result in premature convergence. One variant that addresses this premature convergence is the competitive swarm optimizer (CSO) algorithm [CJ15]. CSO proposes to remove the strong influence of personal and global best in the velocity update formulas by pairing up particles and let the "loser" (particle with inferior objective function value) learn from the "winner" (particle with superior objective function value). Another variant that shifts the paradigm of the original PSO is a quantum-inspired particle swarm optimization (QPSO). Inspired by basic particles in quantum physics, QPSO let particles move based on an uncertainty principle: particles appear in different locations with probabilities, and do not follow trajectories. In QPSO, the velocity term is removed from the algorithm and thus simplifies the algorithm and also increases the chances to escape from local optima at the same time. When the tunning parameters are properly selected, local convergence is guaranteed. In project II, we work with a binary encoded QPSO to take advantage of its local convergence theoretical properties and its empirical superior performance over standard PSO using several benchmark functions [SFX04, SXF07, SLW11, ZSX13]. Section 2.2.1 provides details of QPSO and [SLW11] gives a comprehensive review of the algorithm.

There are additional ways to enhance the PSO performance. For example, [ARC08] borrowed the ideas of having mutation and crossover operations from genetic algorithms and incorporate them into the PSO algorithm. In project II, we include such operations in the quantum particle swarm optimization algorithm for enhanced performance.

2.2.1 An enhanced variant: quantum-inspired PSO (QPSO)

Inspired by basic particle movements in the quantum mechanics framework, quantum-inspired PSO (QPSO) is a global optimization algorithm [SFX04] with superior searching capabilities compared to other EC algorithms [SLW11]. It is different from traditional PSO algorithms in that particles have no trajectory, instead, the particles appear in positions with probabilities. See [SLW11] for a comprehensive summary of the characteristics and properties of the QPSO algorithm.

The QPSO algorithm is suitable for project II because it is hard to exploit any mathematical properties of the objective function; in fact, the objective function couldn't be written down analytically. More importantly, we use QPSO due to its superior capability in searching high-dimensional space and its successful applications in real-world problems including those in the imaging field [LJS15, JJ15].

To prevent QPSO from premature convergence, we also enhanced the algorithm by using probabilistic cross-over operations and random mutation operations, as cross-over and mutation operators have shown to improve the PSO performance in feature selection [NXA17]. We coded each particle in QPSO using a binary coding scheme. As an example, suppose a feature space has 5 features and a particle is encoded as (1, 0, 0, 1, 1). This means that the 1^{st}, 4^{th}, and 5^{th} features are included and the 2^{nd} and 3^{rd} are excluded in selection.

A binary QPSO follows the steps as [ZSX13]

Algorithm 1 Binary QPSO Algorithm

```
Input: Pre-specified parameters of the algorithm: number of particles, maximum number
    of iterations $max_{iter}$, particle dimension $d$.
Output: the global best solution when the stopping criteria are met. Here we use the
    maximum number of iterations as the stopping criterion.
    Initialization : Algorithm initialization
for $i \leq max_{iter}$: do
    Determine the mean best (mbest) for each particle:
        $m_d^{best} = \begin{cases} 1, & \frac{1}{p}\sum_i^p x_{i,d}^{pbest} > 0.5 \\ 0, & \frac{1}{p}\sum_i^p x_{i,d}^{pbest} < 0.5 \\ 1 \; with \; prob = \frac{1}{2}, & \frac{1}{p}\sum_i^p x_{i,d}^{pbest} = 0.5 \end{cases}$
    and select one of the offspring randomly.
    For each dimension, flip the particle with mutation probability defined as
        $min(\frac{\beta \times d_H(x_i, x^{mbest}) \times ln\frac{1}{u}}{p}, 1)$
    where $d_H$ is the counts of bits different in the two strings, u is a random variable
    from unif(0,1) and $\beta$ is a tuning parameter set to decrease from 1.4 to 0.4 over the
    iterations.
    Evaluate.
end for
return global best solution at the last iteration.
```

In project I, we largely work with the standardized PSO, with a global topology and a linearly decreasing inertia weight. This is because the standard PSO works well for most of problems at hand and is computationally efficient, obtaining superior results than many other commonly used metaheuristic algorithms. Scientists and practitioners also have easy access to PSO code. In project II, we work with quantum particle swarm optimization to achieve superior prediction results [SLW11]. Many parameters we work with are consistent with the recommendations in the literature. For certain parameters such as the maximum number of iterations and swarm size, their choices are based on the available computational budget. In the literature, many recommendations on whether one algorithm outperforms another is based on testing the algorithms using benchmark test functions. In our case, we prefer to choose proper configurations empirically based on our actual problems at hand. We do not claim that we use the best configuration for each problem, since there are many variations and possible enhancements and running experiments are subject to budgetary constraints. Further, PSO might not be the best algorithm for every problem we tackled here. As the famous "no free lunch theorem" [WM97] suggests, there is no optimization algorithm that universally works better than all the other algorithms for all problems. The "best" algorithm should be discussed on a case-by-case basis. When we find optimal approximate designs, we have theoretical tools to confirm whether PSO finds the optimum and if not, we are able to provide an assessment on how far the design is from the optimum without knowing the latter, however, for finding efficient exact designs and feature selection problems, there is no such theoretical underpinning and we show that PSO finds quality solutions which are empirically better than many other commonly used approaches.

CHAPTER 3

Project I: finding efficient designs for generalized linear models and non-linear mixed effects models using sparse grid hybridized PSO

3.1 Background

Longitudinal models are frequently used in biomedical studies. The outcomes are measurements for the same experimental unit over a period of time or under a series of experimental conditions, and the measurement process is repeated for all units. While there are some analytic results available on constructing efficient designs for linear models, the widely used generalized longitudinal models resembles the nature of response curve better but lacks analytic results.

This chapter focuses on design issues for longitudinal models, especially nonlinear mixed effects models (NLMEMs). NLMEMs are frequently used in model-based drug development to analyze pharmacokinetic/pharmacodynamic (PK/PD) data [LKH07]. Different methods and software have been developed for maximum likelihood estimation of population parameters or function of the population parameters. For example, the median value of the parameters in the studied population and their inter- or intra-subject variabilities and effects of the covariates are frequently of interest [MU12, MU13, MU14]. NLMEMs can compensate for the lack of individual information by borrowing information from the whole population. In particular, they allow for precise parameter estimation even with sparse designs, where few samples are collected from each subject. Such designs are particularly useful when the experiment involves infants or children. NLMEMs are therefore increasingly used for the analysis of longitudinal clinical studies, for both continuous and discrete outcomes (binary, count or time-to-event).

Before modelling, it is crucial to choose an appropriate design to obtain good precision of parameter estimates. A design in NLMEMs, called a population design, is composed of the number of elementary designs it has, and the number of subjects in each of these elementary designs. In this setting, the term “elementary design” describes a group of subjects all with identical design characteristics, i.e. each subject in the same elementary design has the same number of sampling times and at the same time points or dose levels. To evaluate designs, two approaches have been proposed. The first approach is based on clinical trial simulation (CTS), which can be very time-consuming when the number of possible time points is large. For instance, if the total possible number of time points is 5 and we assumed a simplified case when only equal number of subjects at each time point is allowed, there are 5+10+10+5+1=31 elementary designs to compare. If we relax the equal number of subjects at each sampling time restriction, this number depends on the number of subjects and becomes large quickly. This greatly limits the number of population designs that can be properly evaluated in practice.

Alternatively, the expected Fisher Information Matrix (FIM) can be used [ADT09, FL14, PP13], as its inverse provides a lower bound of the variance-covariance matrix of any unbiased estimated parameters, according to the Cramer-Rao inequality. However, the FIMs of NLMEMs have no analytic form and their computation, which requires multi-dimensional integration, can be challenging. An approach that uses the first order (FO) linearization of the model around the expectation of random effects or around a guess value of the fixed effects has been widely used [MMB97, MS10] and implemented in several software programs [NBO14]. Although this method is generally efficient [BRM09, HV05, NBM12], FO has limitations when it is applied to NLMEMs with discrete outcomes [Riv16].

Alternative methods have been developed to compute the FIM for both continuous and discrete NLMEMs, such as Laplace approximations. Another method is to use Monte Carlo (MC) integration combined with Adaptive Gaussian quadrature (AGQ). This approach has been shown to be viable when designing studies based on NLMEMs with moderate number of random effects [UM17]. A recent extension of this approach uses quasi-random Monte Carlo (QRMC) instead of MC and results in huge reduction in the computational effort [UM15]. Another approach would be to use Markov chain Monte Carlo (MCMC) to replace AGQ [Riv16].

Experiments show that AGQ and CTS agrees quite well whereas Laplace approximation has quite large discrepancies. Empirically, FO works well for continuous models but not well for discrete models [RUM16]. This might be attributed to the fact that the discrete response surface could not be accurately approximated with only first-order terms in the Taylor expansion. MCMC only agrees well when the sampled observation is large enough, and the numerical results do not stabilize until thousands of observation samples [Riv16]. [Riv16] reported that evaluating just one design using MCMC takes over 134.8 minutes for a longitudinal continuous model, 149.9 minutes for a logistic model and 242.6 minutes for a Poisson model. In contrast, FO takes 0.1 minute and AGQ are also generally much faster than MCMC. These results suggest that for NLMEMs with several random effects, the most effective and accurate approach is FO for continuous models or AGQ for both continuous and discrete models. In summary, there are two reasons we do not use Monte Carlo based sampling techniques: (a) they are too time-consuming for the iterative optimization that requires the evaluation of hundreds of candidate designs; and (b) the random effects are usually normally distributed so we can directly sample them, and an indirect sampling technique such as MCMC is usually not needed.

Once the FIM is evaluated, optimality criteria based on FIM can be computed and optimized to produce efficient designs. In this work we focus on the D-criterion, which maximizes the determinant of the FIM after it is normalized by the number of parameters to be estimated. Several algorithms have been developed for this purpose [ADT09]. Conventional algorithms for finding optimal experimental designs include Dykstra's method [Dyk71], DETMAX [Mit74] and modified Fedorov exchange methods ([OGG05], [Fed72]) and more recently, multiplicative algorithms ([DPZ08], [Yu10]) and coordinate exchange algorithm [MN95, GJS09]. A recent review of such algorithms is [MWY15] where they alerted that such algorithms for finding optimal experimental designs can be limiting. For example, some assume the design criterion is differentiable or require that the design space be discretized. Some are narrowly focused for finding a specific optimal design for a particular model. These methods are usually not flexible enough for practical use.

Finding efficient designs for NLMEMs can be very challenging and frequently, one approach in the literature is to simply compare a small number of candidate designs and select the best among this finite set of designs [HW08b]. A justification for this approach is that the finite set of designs are the only implementable designs approved by the clinicians or physicians. Thus, one selects the best among the few designs using a statistical-based criterion for more precise inference. An alternative approach to find an efficient design is by discretizing the search space and selecting the design points. However, this method is feasible only when the design region is small; otherwise, the grid size on the design space can be extremely large when we have a high-dimensional model [NLS18]. A more advanced approach "population Fisher information matrix" (PFIM) is described at www.pfim.biostat.fr [DLL18]. Two algorithms are used in the optimization process and they are the Simplex [NM65] and Fedorov [Ret07] algorithms. The Simplex algorithm is used to find efficient designs over a continuous design space and the Fedorov algorithm is used when we have discrete design space. Each of these algorithms uses a different optimization strategy. However, for complicated models defined on a continuous design space, there is still a lack of efficient yet easy to use algorithm that finds quality designs. PSO is found to be a powerful approach, particularly when handling complicated design criteria [CCW15, KW17, PCW16, WCH15]. In this work, we show that PSO combined with FIM evaluation techniques makes it possible to thoroughly explore a continuous design space and find highly efficient designs that provides valuable insights on efficient designs for NLMEMs in pharmacometrics.

A general non-linear longitudinal model for studying repeated responses is

$$Y_i(t) = \eta(t;\theta) + \epsilon, \quad \epsilon \sim N(0, \sigma^2),$$

where Y_i is the response variable, i=1,...n, η is a mean response function, t is a series of sampling time points, dose levels, or other experimental conditions, and θ is the model parameters vector. Under the pseudo-Bayesian design framework, an n-run exact design ξ=$(t_1, ..., t_n)$ has Fisher information $M(\theta;\xi)$ for θ proportional to

$$M(\theta;\xi) \propto \sum_{i=1}^{n} (\frac{\partial \eta(t_i,\theta)}{\partial \theta})(\frac{\partial \eta(t_i,\theta)}{\partial \theta^T}).$$

For an approximate design $\xi = \{(t_1, p_1), ..., (t_K, p_K)\}$ with K distinct support points and $0 < p_i < 1$ with $\sum p_i = 1$, the Fisher information for θ is approximated by

$$M(\theta;\xi) \propto \sum_{i=1}^{K} p_i (\frac{\partial \eta(t_i,\theta)}{\partial \theta})(\frac{\partial \eta(t_i,\theta)}{\partial \theta^T}).$$

where p_i are weights we put at each support point.

The numerical computation required for finding a pseudo-Bayesian D-optimal design includes integration and optimization. The design is

$$\xi_{BayesD} = \underbrace{\underset{\xi}{\text{argmax}} \underbrace{\int \{\log(\det[M(\theta,\xi)])\}\pi(\theta)\, d\theta}_{\text{Integration}}}_{\text{Optimization}}, \tag{3.1}$$

where the integrals are often multi-dimensional and corresponds to the dimension of θ. Under this framework, we consider several types of models assuming observations are all independent: a logistic model and a compartmental model inspired by [GJS09]; a generalized linear model (GLM) with Gamma distributed outcome, inspired by [DW15]; and a series of exponential models with real-world applications in HIV studies.

We also apply our proposed algorithm to find efficient exact designs for NLMEMs. Such efficient designs are rarely reported in the literature because they are very challenging to find due to the nonlinearity of the regression function in the random parameters. In particular, a closed-form likelihood function of the observations does not exist unless the model is very simple [MS10]. We denote the n_i-vector of continuous observations obtained with the elementary design ξ_i for individual i by y_i $(i = 1, \ldots, N)$. Let f be a known nonlinear function with unknown parameters and represents the mean response. The vector of unknown parameters for subject i is expressed as a function g comprising fixed effects and random effects. The parameters for the fixed effects are μ and covariate effects β. The random effects are b_i and covariates z_i in the following model:

$$y_i = f(g(\mu, b_i, z_i, \beta), \xi_i) + \varepsilon_i,$$

where ε_i is a vector of errors. It is assumed that $b_i \sim N(0, \Omega)$ where Ω is the variance-covariance matrix of the random effects, and $\varepsilon_i \sim N(0, \Sigma)$ where Σ is a diagonal variance-covariance matrix of the errors. The variance terms in Σ can be a function of the parameters or depend on the structural model f. We let Ψ denote the vector composed of all unknown parameters to be estimated in μ, β, Ω and Σ, and let λ denote the vector composed of random effects parameters Ω and the error term Σ.

For discrete data, we model the probability for all observations $j = 1, \ldots, n_i$ from an individual i as follows:

$$p(y_i, \psi | b_i) = \prod_{j=1}^{n_i} l(y_{ij}, g(\mu, b_i, z_i, \beta), \xi_i),$$

where l is a known link function describing the probability model of observing the value y_{ij} with the elementary design ξ_i for subject i. The individual parameters vector is expressed as a function g of fixed effects μ, random effects b_i, the covariates z_i for individual i, and the covariate effects β. It is assumed that $b_i \sim N(0, \Omega)$ where Ω is the variance-covariance matrix of the random effects composed of unknown parameters. Let Ψ denote the vector composed of all unknown parameters to be estimated in μ, β and Ω.

We assume individuals are independent, and the population FIM, $M_{PF}(\Psi, \Xi)$, for a population design Ξ, is defined as the sum of the N information matrices $M_F(\Psi, \xi_i)$ computed for each individual,

$$M_{PF}(\Psi, \Xi) = \sum_{i=1}^{N} M_F(\Psi, \xi_i). \tag{3.2}$$

The FIM for individual i is defined by

$$M_F(\Psi, \xi_i) = E_{y_i}\left(\frac{\partial \log L(\Psi, y_i)}{\partial \Psi}\frac{\partial \log L(\Psi, y_i)}{\partial \Psi^T}\right), \tag{3.3}$$

where $\log L(\Psi, y_i)$ is the log-likelihood of the vector of observations from individual i for parameter vector Ψ. This likelihood function is

$$L(\Psi, y_i) = \int_{b_i} p(y_i|b_i, \Psi)p(b_i|\Psi)\, db_i, \tag{3.4}$$

where $p(y_i|b_i, \Psi)$ is the probability density function (p.d.f.) of y_i given the random effects b_i, and $p(b_i|\Psi)$ is the p.d.f. of the random effects.

Optimizing the design criterion is generally computationally expensive and in the current literature, the criterion is simply used to select the best design among a few selected designs after simulating data from a predictive distribution and examining the distribution of the posterior loss from the candidate designs [NM14a, RUM16]. In practice, the candidate designs come from experts' opinion or are based on feasibility requirements. In what is to follow, we show that it is possible to hybridize PSO with sparse grid technique to effectively find efficient designs.

We apply our algorithm to find efficient designs for a variety of NLMEMs. The first application is inspired by a study of a monoclonal antibody indicated in the treatment of wet age related macular degeneration (wet-AMD) [BUF18]. We used the previously developed mixed effects model for a continuous outcome and optimize both the measurement times and the different doses for the subjects. The second application is a randomized two arm clinical trial with repeated binary outcomes, and modelled by a mixed logistic model [LUM16, SNMon]. We also consider a Poisson model with repeated counts outcomes for each subject at different dose levels [NKH09, OA11]. The last application is a HIV clinical trial using an exponential model [WD99]. As explained earlier, finding efficient designs for these models are very difficult problems and we show next that the proposed algorithm can provide a viable option to solve the design problems.

3.2 Sparse grid

Numerical integration is an inevitable process for computing pseudo-Bayesian optimal design criterion values and calculating information matrices for NLMEMs. This is clear from equation (3.1) and equation (3.3). There are different numerical integration techniques in the design literature, and many are based on various Monte-Carlo types of techniques. A plain Monte Carlo may take 100,000 samples [NKH09], and an MCMC may need 10,000 iterations [HW08b, Wak96] to obtain a stable result. Another example is Hamiltonian Monte Carlo, which is similar to MCMC [RUM16]. Hamiltonian Monte Carlo reduces the correlation between successive samples and converges to high-dimensional target distributions much more quickly than simpler methods such as Metropolis-Hastings or Gibbs sampling [HG14]. Such methods are time consuming and not efficient for finding optimal designs. [Kuo03] used 71 benchmark functions and compared performance of the Gaussian quadrature, Monte Carlo and adaptive methods and found that Gaussian quadrature provides the most accurate results and has short CPU times, relative to those from the adaptive methods and plain MC methods. This is not surprising because the number of function evaluations required to obtain an accurate estimate is usually impractically large. Further, the iterative optimization process for finding the efficient design needs to approximate the FIM often hundreds of times, and it is critically important that we have efficient yet accurate approximation methods. Gaussian quadrature based algorithms are often much more efficient and in particular, adaptive Gaussian quadrature (AGQ) has shown to be useful in finding efficient designs for NLMEMs [UM17]. In the following section, we show that using sparse grid to extend the 1-dimensional Gaussian quadrature to multi-dimensional can further save computational efforts.

3.2.1 Numerical integration

A fast algorithm to approximate integrals is a Gaussian quadrature. We first discuss the rationale of 1-dimensional Gaussian quadrature and extend the method to multi-dimensional using a sparse grids technique.

In general, numerical integration approximates a definite integral with a weighted sum of function values evaluated at the nodes x_i, i.e.,

$$\int f(x)w(x)dx \approx \sum_{i}^{n} w_i f(x_i). \tag{3.5}$$

Gaussian quadrature is a way to select a small number of nodes and weights to obtain a relatively accurate approximation. The rationale of Gaussian quadrature is to use polynomial functions to approximate $f(x)$. Integrals for polynomial functions are analytically available and computationally very efficient. Theoretically, polynomial approximations can be justified by the Weierstrass approximation theorem. The theorem states that every continuous function defined on a closed interval [a, b] can be uniformly approximated as closely as desired by a polynomial function. Furthermore, due to Taylor's expansion, any k-times differentiable function can be approximated by means of a polynomial of degree k, suggesting that it is possible to integrate any function well when we use a relatively small number of systematic draws.

As an illustrative example, suppose we want to approximate the following integral

$$\int_{-\infty}^{+\infty} f(x)e^{-x^2}dx$$

by replacing the weighting function e^{-x^2} by carefully selected weights w_i's. A quadrature aims to obtain n nodes (ζ_1, ..., ζ_n) and n weights (w_1, ..., w_n) so that the approximation is exact if $f(x)$ is a polynomial of order $2n-1$. For example, to obtain exact answer for $f(x) = 1$ we set:

$$\int_{-1}^{1} 1e^{-x^2}dx = \sum_{i=1}^{n} w_i 1$$

To obtain exact answer for $f(x) = x$ we set:

$$\int_{-1}^{1} xe^{-x^2}dx = \sum_{i=1}^{n} w_i \zeta_i$$

etc. In general, to obtain exact answer for $f(x) = x^j$, j=0, ..., $2n-1$:

$$\int_{-\infty}^{+\infty} x^j e^{-x^2} dx = \sum_{i=1}^{n} w_i \zeta_i^j,\ j = 0, 1, ..., 2n-1$$

Solving the $2n$ equations for the $2n$ unknowns provides us the required nodes and weights. This is the technique we used when we have a normal prior distribution. We note that the format is not exactly consistent with the normal probability density function, so we apply a change of variables trick for $y \sim N(\mu, \sigma^2)$ as:

$$\int_{-\infty}^{+\infty} f(y) \frac{1}{\sigma\sqrt{2\pi}} exp(-\frac{(y-\mu)^2}{2\sigma^2})dy = \sum_{i=1}^{n} \frac{1}{\sqrt{\pi}} w_i f(\sqrt{2}\sigma\zeta_i + \mu)$$

A simple scaling and translation makes the nodes widely applicable to different normal priors.

For many commonly used $w(x)$ functions in equation (3.5), efficient quadrature rules (nodes and weights) are readily available. A different quadrature rule is required for each form of the weight function. Common quadrature rules include Gauss-Legendre quadrature for uniform distribution weight functions, Gauss-Hermite quadrature for normal distribution weight functions, Gauss-Stieltjes-Wigert quadrature for log-normal distribution weight functions, Gauss-Jacobi quadrature for Beta distribution weight functions, Gauss-Laguerre quadrature for Gamma distribution weight functions, etc. Users specify types of distributions and parameters of the distributions (mean, variance, etc.) based on their problems at hand. Many software provide such quadratures, for example, there are R packages "statmod", "gaussquad", "cubature", a MATLAB function numeric::quadrature, etc.

3.2.2 Sparse grid integration

Sparse grid is a technique to extend the 1-dimensional quadrature to multi-dimensions. In the AGQ paradigm, the multi-dimensional quadrature is extended from a 1-dimensional quadrature using a tensor product. A tensor product of two vectors is the pair of every possible combinations of the components. For example, if a 1-dimensional quadrature has the nodes equally at [-1, 0, 1], then an extended 2-dimensional quadrature has the nodes equally at [-1, 0, 1] $\otimes$ [-1, 0, 1] = [(-1, -1), (-1, 0), (-1, 1), (0, -1), (0, 0), (0, 1), (1, -1), (1, 0), (1, 1)]. Using tensor product rule, we need 3^2=9 nodes for a 2-dimensional integration, 3^5=243 nodes for a 5-dimensional integration and 3^{10}=59049 nodes for a 10-dimensional integration. The exponential increase in the number of nodes required is often referred to as "curse of dimensionality".

To mitigate this problem when dealing with moderate to high dimensional integration problems, we use a sparse grid (SG) method to limit the number of nodes. The fundamental idea behind quadrature is to use polynomial functions to approximate the integrand function. The main idea of SG is to limit the total order of polynomials used for approximation. Total order is the sum of polynomial order of each dimension. For instance, a 2-dimensional polynomial that extended to second order $(1, x, x^2) \otimes (1, y, y^2)$ includes 9 terms if extended by tensor product, but when we limit the total order of polynomials to 2, the expansion will only include terms that has a sum of order less or equal to 2, i.e. 1, x, y, x^2, xy, y^2 but not terms like x^2y, xy^2 (which has a total order of 3) and x^2y^2 (which has a total order of 4). As a result, SG uses far fewer nodes than tensor product when limiting the total order to a relatively low level. Adopting SG methodology reduces the number of function evaluations for integrals and substantially saves computational efforts.

Formally, we work with a quadrature that follows the format in equation (3.5). One of the properties of a quadrature rule is whether or not the points are nested. Nestedness means that some or all points from a lower level are re-used for higher levels, implying that fewer points are required overall for higher level approximations. In this dissertation, we mostly work with nested Gaussian quadrature as basis abscissa [GK96].

We define the 1-dimensional quadrature approximation of the integral as

$$V[f] = \sum_{k}^{n} f(x_k) w(x_k).$$

For multivariate integrals, the most commonly used approach is a tensor product rule. Tensor product multi-dimension rule states that for a D-dimensional integral, the quadrature is the product of each dimension's quadrature:

$$I[f] = V^{(1)} \otimes V^{(2)} \otimes ...V^{(D)}[f]$$

where $V^{(d)}, d = 1, ...D$ are d-th dimensional quadrature. Tensor product nodes grows exponentially with dimensions. Sparse grid sampling, on the other hand, only employs the lower order of quadratures and is thus able to produce a sparser sampling grid. The procedure for building a set of multi-dimensional nodes follows the below recipe [GG98]:

1. Consider 1-dimensional quadrature formulas for function f at accuracy level k:

$$V_k = \sum_{i=1}^{n} w_{ki} f(x_{ki})$$

2. Define the difference in quadrature formula by

$$\Delta_k[f] = (V_k - V_{k-1})[f]$$

with $V_0[f] = 0$.

3. D-dimensional function f for accuracy level (pre-determined) $l \in \mathbb{N}$

$$I[f] = \sum_{k_1+...+k_D \leq l+D-1} (\Delta_{k_1} \otimes ... \otimes \Delta_{k_D})f.$$

Alternatively, an equivalent polynomial expression of the formula of sparse grid is

$$\sum_{q=\kappa-D}^{\kappa-1} \sum_{\mathbf{i} \in \mathbb{N}_q^D} \sum_{x_1 \in \mathbb{X}_{i_1}} ... \sum_{x_D \in \mathbb{X}_{i_D}} f(x_1, ..., x_D)(-1)^{\kappa-1-q} \binom{D-1}{\kappa-1-q} \prod_{d=1}^{D} w_{i_d}(x_d), \tag{3.6}$$

where D is the dimension, κ is the accuracy level sought (i.e. total orders limitation), $\mathbf{i} = [i_1, ..., i_D]$, q is a nonnegative integer and $\mathbb{N}_q^D = \{\mathbf{i} \in \mathbb{N}^D : \sum_{d=1}^{D} i_d = D + q\}$. We take a 2-dimensional SG as an example to demonstrate the construction of such grids. Let D=2, κ=3, $1 \leq q \leq 2$. When $q = 1$, we have $\mathbb{N}_1^2 = \{[1, 2], [2, 1]\}$, $(-1)^{3-1-1} \binom{2-1}{3-1-1} = -1$; Similarly, when $q = 2$, we have $\mathbb{N}_2^2 = \{[1,3],[2,2],[3,1]\}$, $(-1)^{3-1-2}\binom{2-1}{3-1-2}$=1. The construction is thus $-\mathbb{X}_1 \otimes \mathbb{X}_2 - \mathbb{X}_2 \otimes \mathbb{X}_1 + \mathbb{X}_1 \otimes \mathbb{X}_3 + \mathbb{X}_2 \otimes \mathbb{X}_2 + \mathbb{X}_3 \otimes \mathbb{X}_1$ and figure 3.1 shows the SG has much fewer nodes compared to the nodes extended from the tensor product rule. Since there are pluses and minuses in equation (3.6), the sparse grid rule does not always preserve convexity, even if the quadrature rule is convex.

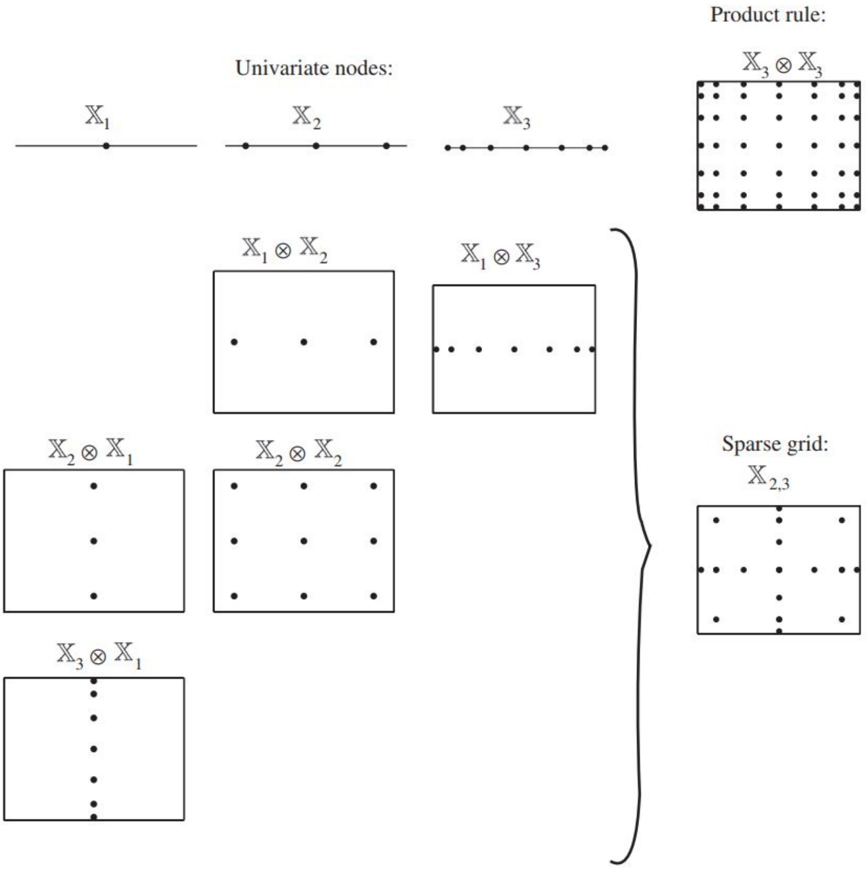

Product rule:
$\mathbb{X}_3 \otimes \mathbb{X}_3$
Univariate nodes:
$\mathbb{X}_1$
$\mathbb{X}_2$
$\mathbb{X}_3$
$\mathbb{X}_1 \otimes \mathbb{X}_2$
$\mathbb{X}_1 \otimes \mathbb{X}_3$
Sparse grid:
$\mathbb{X}_{2,3}$
$\mathbb{X}_2 \otimes \mathbb{X}_1$
$\mathbb{X}_2 \otimes \mathbb{X}_2$
$\mathbb{X}_3 \otimes \mathbb{X}_1$

Figure 3.1: An example of a 2-dimensional grid set constructed by sparse grids [HW08a].

The number of nodes grows polynomially with higher dimensions rather than exponentially as tensor product rule [BG04]. Empirically, [HW08a] showed that the computational costs for SG to achieve a negligible approximation error are considerably lower than with simulation techniques like pseudo-random Monte Carlo and quasi-random Monte Carlo. For N 1-dimensional nodes and a D-dimensional problem, the complexity of SG is $O(N(logN)^{D-1})$ instead of tensor product's $O(N^D)$, with error bound increased from $O(N^{-2})$

to $O(N^{-2}(logN)^{D-1})$. For proofs, see [Plu14]. Many properties of sparse grid depends on the quadrature used to build the multi-dimensional quadrature. Different quadrature rules have different error bounds [GG98].

We next demonstrate our proposed methods that approximate FIMs accurately are also able to evaluate the FIMs for models with more random effects parameters and able to search the design space more thoroughly for an efficient design.

3.3 Non-linear mixed effects models

3.3.0.1 FIM computation by FO

There is no analytic expression for the likelihood of NLMEMs and techniques such as linearization is often used to approximate the FIM [MMB97, MS10]. When linearizing the model around a nominal value of the fixed effects, the variance of the observations does not depend on the fixed effects, which leads to a block-diagonal expression of the FIM [MS10]. If an elementary design ξ is used for the model, a first-order approximation ignores the higher order terms in Taylor expansion of the mean response function:

$$\begin{aligned} Y_i(\xi) = f(\xi;\beta,b_i) + \epsilon &= f(\xi;\beta,0) + \frac{\partial f^T(\xi;\beta,0)}{\partial b_i} b_i + ... \\ &\approx f(\xi;\beta,0) + \frac{\partial f^T(\xi;\beta,0)}{\partial b_i} b_i, \end{aligned}$$

and the resulting FIM is approximated by a block-diagnal matrix of the form:

$$M_F(\Psi,\xi) = \begin{bmatrix} \frac{\partial f^T(\xi;\beta,0)}{\partial \beta} V^{-1} \frac{\partial f(\xi;\beta,0)}{\partial \beta} & 0 \\ 0 & \frac{1}{2}F \end{bmatrix}$$

where $V = var(y_i) = \frac{\partial f^T(\xi;\beta,0)}{\partial b_i} \Omega \frac{\partial f(\xi;\beta,0)}{\partial b_i} + \sigma^2 I_n$ and jk^{th} element of F is $F_{jk} = tr(V^{-1}\frac{\partial V}{\partial \lambda_j} V^{-1} \frac{\partial V}{\partial \lambda_k})$. The off-diagonal elements are zeros because V does not depend on the fixed effects. We recall that λ is the vector of random effects parameters and error terms.

For details, see [DLL18].

[NBO14] shows that this expression of the FIM in general performs well for models with continuous outcomes, in that they provide results close to those obtained by clinical trial simulations. In section 3.6.1, we will use this approach, implemented in PFIM program v4.0, to evaluate the FIM with continuous outcomes. However, it empirically has relatively large discrepancy when the NLMEMs have discrete outcomes [Riv16], which might be attributed to the fact that the non-linearity in such models could not be approximated closely using linear expansions.

3.3.0.2 FIM computation by QRMC-AGQ

[NM14b, UM17, Riv16] showed that adaptive Gaussian quadratures (AGQ) are more accurate than first-order approximation and Laplace approximation for analyzing our models with discrete outcomes across different models and scenarios. For example, to evaluate the FIM in equation (3.3), one computes two integrals in equation (3.4): one over the distribution of the random effects b_i and one over the observations y_i. Here we evaluate the former by AGQ as proposed in [UM17] and the latter by quasi-random MC as proposed in [UM15], Specifically, in equation (3.3), let

$$h(y, \Psi) = \frac{\partial log(L(y, \Psi))}{\partial \Psi} \frac{\partial log(L(y, \Psi))}{\partial \Psi^T} = \frac{\frac{\partial L(y, \Psi)}{\partial \Psi} \frac{\partial L(y, \Psi)}{\partial \Psi^T}}{L(y, \Psi)^2}. \tag{3.7}$$

Since the FIM can be expressed as $M_F(\Psi, \xi) = \int h(y, \Psi) p(y, \Psi) dy$, this outer integral can be approximated using quasi-random sequences with R observation samples $(y_1, \ldots, y_R)$:

$$M_F(\Psi, \xi) = \frac{1}{R} \sum_{r=1}^{R} h(y_r, \Psi),$$

where R is a user-specified value.

To approximate $h(y, \Psi)$, different approaches can be used and AGQ is one of them [NM14a]. Gaussian quadrature uses a weighted sum of function evaluations at pre-determined "nodes" to approximate the integral. To approximate the likelihood $L(y, \Psi)$ in $h(y, \Psi)$, we sample k nodes s_1, ...s_k and corresponding weights w_1,...,w_k from a Gauss-Hermite quadrature. The nodes are then centered and scaled according to the areas of high density to improve the approximation. The approximated likelihood in equation (3.7) is then computed as

$$L(y_m, \Psi) \approx \sum_{j=1}^{k} w_k p(y_m|s_k, \Psi),$$

where y_m is a sampled observation and s_k, w_k, k are AGQ nodes and weights. Since normal distribution p.d.f. can be easily computed, the approximation is computationally efficient. More details can be found in [UM15, UM17]. In section 3.6.2, we use this approach, implemented in R by [UM15],to evaluate the FIM with binary outcomes.

3.3.0.3 FIM computation by QRMC-SG

A further improvement is to use sparse grid to obtain multi-dimensional nodes for integration. Similar to AGQ, to approximate the likelihood $L(y, \Psi)$ in $h(y, \Psi)$, we sample k nodes x_1, ..., x_k and corresponding weights w_1,...,w_k from a Gauss-Hermite quadrature. Then we use the sparse grid rule to extend the 1-dimensional quadrature to the required dimension. The multi-dimensional nodes are then centered and scaled according to the areas of high density to improve the approximation. The approximated likelihood in equation (3.7) is

$$L(y_m, \Psi) \approx \sum_{j=1}^{k} w_k p(y_m|x_k, \Psi),$$

where y_m is a sampled observation and s_k, w_k, k are SG nodes and weights. Since it employs a sparser scheme, the number of required function evaluations is smaller than that of AGQ and is thus more efficient. With the improved methods to approximate FIMs, we gain computational efficiency that allows us to evaluate FIMs containing more random effects parameters for more candidate designs. Consequently, our proposed method can search the design space more thoroughly. The iterative optimization process for finding efficient designs needs to approximate the FIM often hundreds of times, and it is critically important that we have efficient yet accurate approximation methods.

3.4 Proposed methodology

We develop an algorithm for finding optimal designs for longitudinal models using sparse grid for the numerical integration and PSO for optimizing criterion values (objective functions). The SG method could be replaced by other numerical integration techniques. We demonstrate below that our proposed methods is fast and computational results are reasonably accurate for finding optimal designs. The flowchart of the proposed sparse grid - particle swarm optimization (SGPSO) algorithm is as follows:

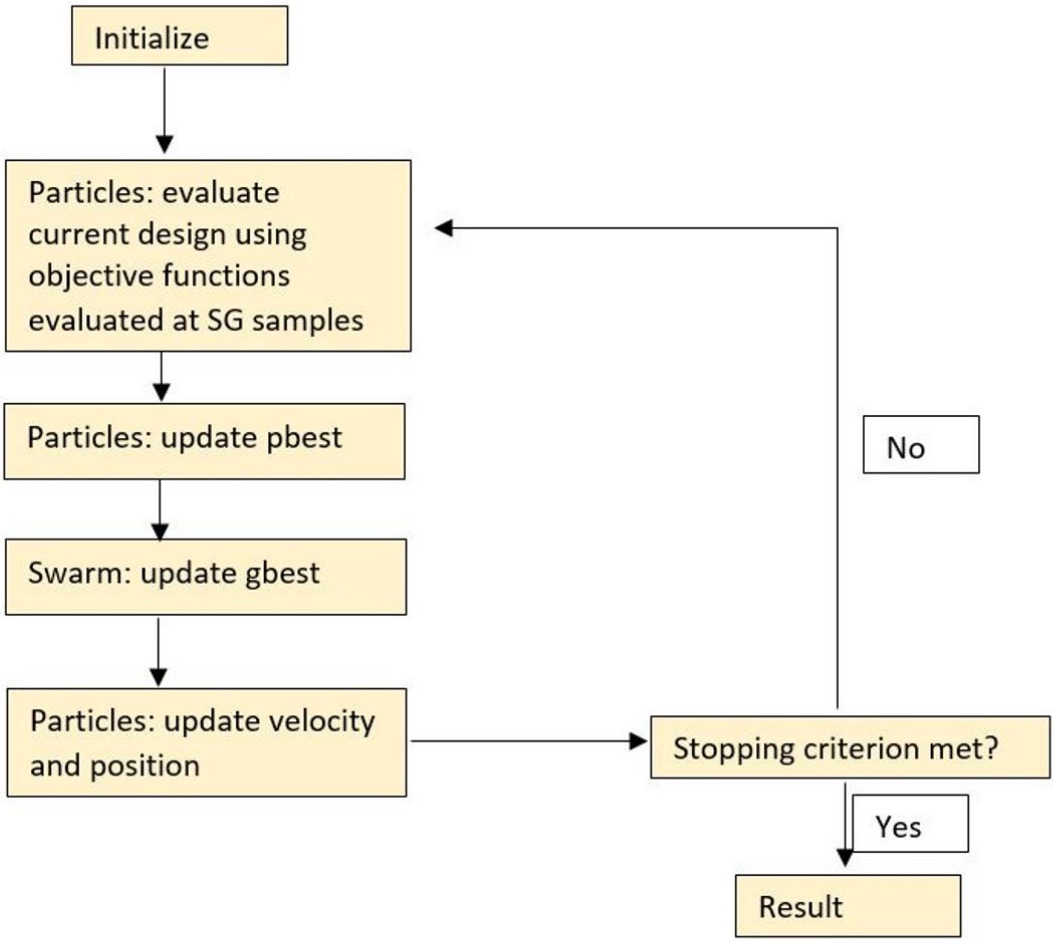

Initialize
Particles: evaluate current design using objective functions evaluated at SG samples
Particles: update pbest
Swarm: update gbest
Particles: update velocity and position
Stopping criterion met?
No
Yes
Result

Figure 3.2: The flowchart of the SGPSO algorithm.

We choose to work with PSO but not conventional optimization algorithm because: (1) sparse grid does not preserve convexity, and many conventional algorithms assume that the objective function is convex or has nice mathematical properties. PSO does not use mathematical properties of the objective function and is thus applicable in our case. and (2) PSO has superior computational efficiency and empirically works well for finding optimal or highly efficient designs for our problems.

We denote an n-run design by $\xi = \{t_1, ..., t_n\} \in X$ where n is user-specified and each t_i is a design point representing a sampling time point or a dose level to observe the outcome. We initiate particles in an n dimensional *design* space to determine the optimal values of $t_1, ..., t_n$. We apply sparse grid sampling on the random effects *parameter* space. For approximate designs with k support points, we denote by $\xi = \{(t_1, p_1), ..., (t_K, p_k)\} \in R^{2k}$ subject to $p_1 + p_2 + ... + p_k = 1$ and $p_i \geq 0,\ i = 1, ..., k$. We initiate particles in a $2k - 1$ dimensional *design* space (we subtract one because of the weights sum up to one constraint). Then we apply sparse grid sampling on the *parameter* space with uncertainty and its dimension is the same as the number of unknown parameters in the model. To fix ideas, we use D-optimality and take the normalized determinant of information matrix as the objective function. These values then serve as fitness values when evaluating the performance of a selected design. PSO then searches the *design* space for an optimizer. The iterations stop when some pre-determined algorithm stopping criteria are met. Some typical stopping rules are the maximum number of iterations allowed and relative improvement in the fitness values from iteration to iteration, which should be very small and specified by the user.

It is often difficult to determine the number of support points of pseudo-Bayesian optimal designs because they usually have more support points than the number of parameters in the model. SGPSO finds the number of support points heuristically. One can put arbitrary number of support points into the algorithm (usually more than number of parameters in the model) and the algorithm can find the correct number of support points, by either collapsing some points to zero weights or dividing the weights appropriately.

3.4.1 R programs

The SGPSO methodology is coded in R. We used version 3.3.0 "Supposedly Educational". The program requires installation of Rtools (https://cran.r-project.org/bin/windows/Rtools/). Packages required include R6, pracma, compiler, statmod, Matrix, SparseGrid, and ppso. We compare performance of our SGPSO with other metaheuristic algorithms and they include Quantum PSO (QPSO), genetic algorithm (GA) and differential evolution (DE). We used the package "GA" for the GA and DE algorithms, manually coded QPSO and used several benchmark functions to compare their performances.

These algorithms have varying numbers of tunning parameters and users have to select appropriate values for them. The parameters for PSO are generally more flexible and can be adjusted based on the problem at hand, and the most important parameters for computational efficiency are the stopping criteria and the maximum number of iterations. Users could test each problem using a small number of iterations to get a sense of the number of iterations that is computationally feasible. The accuracy level of sparse grid could also be adjusted to fit the computational needs. In general, the higher this value is, the more accurate is the integral approximation but more computationally intensive.

Users need to code their models as R functions, and specify the parameters that need to be integrated. Users will also need to pre-specify the design parameters (number of sampling times, dose levels, number of subjects, etc.), although for the optimal approximate designs setting, users could assign a relatively large number of support points and the program could find the correct number of support points by collapsing the weights. The interested reader can write to the author and request a copy of the program codes.

3.5 Bayesian exact and approximate designs for generalized linear models

We use a variety of model and design space configurations to test if our proposed algorithm works well under the pseudo-Bayesian framework. In the following examples, we show our algorithm finds optimal or highly efficient designs for generalized linear models (GLMs). In this section, we use Monte Carlo-PSO (MCPSO) and SGPSO for finding optimal designs and compare their performances. In the examples to follow, we seek pseudo-Bayesian exact designs and optimal approximate designs. They maximize the criterion value in equation (1.2) in a continuous design space. Throughout this section, we use 40 particles for PSO and set the maximum number of iterations to be 1000. The inertia weight from 0.9 to 0.4 and $c_1 = c_2 = 2$ [SE98b].

3.5.1 A logistic model

One of the commonly used non-linear model is a logistic models where the outcomes are binary and independent. As an example, we consider a five-parameter additive logistic model on the design interval X=[-1,1]. We let the probability of a response y at the dose level x be

$$E(y) = p(x, \theta) = \frac{e^{\beta_0+\beta_1 x_1+\ldots\beta_4 x_4}}{1 + e^{\beta_0+\beta_1 x_1+\ldots\beta_4 x_4}}$$

and we want to estimate all the five parameters as accurately as possible using a D-optimal design. Following [WLE06], we assign independent uniform priors for the parameters as follows: $\beta_0 \sim unif[-3, 3]$, $\beta_1 \sim unif[4, 10]$, $\beta_2 \sim unif[5, 11]$, $\beta_3 \sim unif[-6, 0]$, and $\beta_4 \sim unif[-2.5, 3.5]$.

The maximum likelihood estimator of the vector of parameters has an asymptotic variance-covariance matrix that is the inverse of the Fisher information matrix

$$M(\theta, \xi) = X^T W X$$

where $W = diag\{w(x_j)\}$ and $w(x_j) = p_j(1-p_j)$ are the weights for the logistic link function.

We compare the SGPSO results with those in [GJS09, WLE06] using 1000 Monte Carlo samples. We set the SG accuracy level at 4 and for this 5-dimensional integral, we use 151 nodes in total. One way to assess the performance of a GLM design under model uncertainty is through comparison with locally optimal designs across a range of model assumptions with different simulated parameters. Following [WLE06], we calculate local efficiencies by randomly sampling 100 sets of parameters in the parameter space to perform the procedure. Table 3.1 summarizes the results.

Table 3.1: Comparison of performance for finding pseudo-Bayesian exact designs using SGPSO, MCPSO, and the methods due to Gotwalt et al. (2009) and Woods et al. (2006). Asterisk marked values were reported in Gotwalt et al. (2009) and local efficiencies are D-efficiencies of locally exact design averaged over simulations.

	Woods (2006)	Gotwalt (2009)	MCPSO	SGPSO
criterion value (to minimize)		4.043233	4.319407	3.884441
CPU time (s)		<10	1579.01	462.83
local efficiency (median)	42.3%*	44.8%*	46.58%	51.73%
local efficiency (sd)		8.90%	8.79%	10.09%

It is clear that SGPSO approach finds the most efficient design, in terms of the mean and median local efficiencies under simulated parameter sets. Although SGPSO is much slower than the approach proposed in [GJS09], it is a general optimizer and is versatile for many more scenarios. Compared to the similar but more general algorithm MCPSO, SGPSO only used 29% of the computational time required by MCPSO.

3.5.2 A compartmental model

Compartmental models are widely used in pharmaceutical research to ascertain how the drug distributes itself after administration into the body by one of several ways. Design questions may pertain how best to sample blood or urine from the subject to ascertain properties of the drug, such as its average time inside a specific organ, the maximum dose of the drug in the organ and the time it takes to reach the maximum dose in the target organ. Such models are also used in non-clinical research. For example in veterinary science, [Fre84] used a simple compartmental model to study the effect of Theophylline on horses in a veterinary science study. In biomedical studies, [DBJ99] used a compartmental model to predict biological levels of tetrahydrofuran under different exposure scenarios. Other common use of such models are in toxicokinetic experiments [BBU92]. To fix ideas, in what is to follow, we consider a 3-parameter compartmental model commonly used to understand characteristics after a drug is administered and we wish to estimate specific features of the drug.

Let us revisit the design proposed in [Fre84], where each of the six horses received 15 mg/kg of theophylline and a decision was made to collect 18 measurements of the theophylline concentration over a user-specified time period. Let $\theta^T = (\theta_1, \theta_2, \theta_3)$ be the three parameters in the compartmental model. The mean concentration of the drug at time $t > 0$ (minute) after administration in the targeted compartment (organ) is

$$\eta(t, \theta) = \theta_3\{exp(-\theta_2 t) - exp(-\theta_1 t)\},$$

after accounting for the inflow and outflow of the drug in the compartment.

Pseudo-Bayesian exact designs were also found for estimating the model parameters in the above problem [GJS09, ACH93]. We compare their results with ours and for this reason, we use the same prior distributions they had for the parameters: $\theta_1 \sim$ unif(0.01884, 0.09884), $\theta_2 \sim$ unif(0.298, 8.298), and assume they are independent with 21.8 as the nominal value of θ_3. The priors on θ_1 and θ_2 reflect parameter values actually estimated from experimental data in [But79]. For θ_3, it is easy to show that the design efficiency does not depend on the value of this parameter.

We apply our approach to derive a 18-run pseudo-Bayesian D-optimal exact design for this model. A total of 1000 MC samples were used, and the SG accuracy level was set at 6 (33 nodes in total). Table 3.2 displays designs found by SGPSO and MCPSO, along with results reported in [GJS09, ACH93], using the same experimental settings. We compared the three designs in terms of their criterion values and local efficiencies under simulated parameters. Local efficiencies are D-efficiencies of locally exact design averaged over simulations. All designs are compared and their properties are summarized in table 3.3 after rounding the support points to the fourth decimal places. Atkinson et al. (1993) did not provide details on how they computed the average log determinant criterion. Gotwalt et al. (2009) found the design by integration over just a "small set of points" in the parameter space. We compare designs' local efficiencies based on 100 parameter vectors sampled from the parameter space.

Table 3.2: 18-run pseudo-Bayesian exact designs found by four methods: SGPSO, MCPSO, the methods due to Gotwalt et al. (2009), and Atkinson et al. (1993) for the compartmental model.

Time	Atkinson et al. (1993)	Gotwalt et al. (2009)	SGPSO	MCPSO
1	0.2034	0.2030	0.2031	0.2028
2	0.2034	0.2030	0.2031	0.2028
3	0.2034	0.2030	0.2031	0.2028
4	0.2034	0.2030	0.2031	0.2028
5	0.2034	0.2030	0.2031	0.2028
6	1.1967	0.9717	1.2122	1.1655
7	1.1967	0.9717	1.2122	1.1655
8	1.1967	0.9717	1.2122	1.1655
9	1.1967	0.9717	1.2122	1.1655
10	2.8323	3.3439	3.2812	3.0310
11	2.8323	3.3439	3.2812	3.0310
12	7.8229	3.3439	7.2340	6.3436
13	20.1899	20.1917	20.2754	20.4242
14	20.1899	20.1917	20.2754	20.4242
15	20.1899	20.1917	20.2754	20.4242
16	20.1899	20.1917	20.2754	20.4242
17	20.1899	20.1917	20.2754	20.4242
18	20.1899	20.1917	20.2754	20.4242

The distributions of support points roughly follow a similar pattern, with more support points near the boundary of the design space and fewer points in the middle. The differences after the first decimal places seem minimal: the experiments in practice may not have such high precision.

It is clear for the compartmental model from table 3.3 that although our approach samples a relatively large amount from the sample space, the design formed by our approach has higher criterion value, and better local efficiencies. Our approach does not make any assumptions, so the method is applicable to other optimal design problems.

Table 3.3: Performance of the four methods for finding a 18-run Bayesian exact design for the compartmental model.

Approach	Atkinson (1993)	Gotwalt (2009)	SGPSO	MCPSO
# support points	5	4	5	5
criterion value	7.1059	7.0951	7.1115	7.1141
CPU time	not reported	not reported	47.82	860.54
local efficiency (mean)	82.82%	82.73%	82.58%	82.74%
local efficiency (median)	85.19%	84.84%	84.81%	85.28%
local efficiency (sd)	9.52%	8.70%	8.79%	8.90%

3.5.3 A generalized linear model with Gamma distributed response

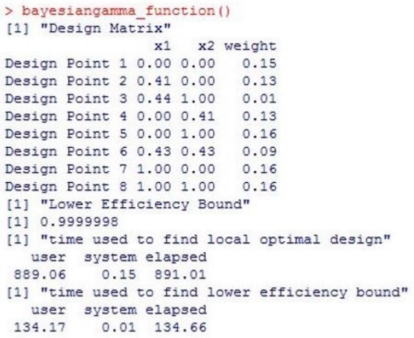
To further demonstrate the flexibility of our algorithm, we apply it to find optimal approximate designs for various models in this and the next sections. Equivalence theorems and sensitivity plots are then used to confirm optimality of the designs found by our algorithm. First, we consider a model with six parameters and two predictors here with a Gamma distributed response y defined on the design space $X \in [0,1]^2$ and the mean function is given by

$$g(\mu) = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \beta_3 x_1^2 + \beta_4 x_2^2 + \beta_5 x_1 x_2$$

where $g(\mu) = \dfrac{1}{\mu}$ is the selected link function. We want to find locally optimal approximate designs and pseudo-Bayesian optimal approximate designs for estimating all parameters in this model. The region for parameter values is $\Theta = [\beta_0^L, \beta_0^U] \times [\beta_1^L, \beta_1^U] \times [\beta_2^L, \beta_2^U] \times [\beta_3^L, \beta_3^U] \times [\beta_4^L, \beta_4^U] \times [\beta_5^L, \beta_5^U]$. All the parameters are assumed to be normally distributed. To fix ideas, we assume each parameter to be between 0 and 1. [DPZ13] provides a derivation of the information matrix of this model.

Finding locally and pseudo-Bayesian optimal approximate designs for this model is proven to be a hard problem [DW15]. We resort to PSO, use the default values for the tunning parameters and set the maximum number of iterations to be 10000 and the number of particles to be 40. The SG accuracy level is set at 4, and for this 6-dimensional integral, we use 257 nodes in total. These choices are arrived at after several initial experiments and they seem to work well for finding relatively efficient designs; users can also set different values for the tuning parameters on their own.

We seek a 8 support points design for a pseudo-Bayesian optimal approximate design. An experiment with more support points is possible, where PSO automatically split the weights among the design points. This phenomena is also reported in [QCW14]. Figure 3.3 shows the optimal design found, along with pseudo-Bayesian efficiency lower bound, and CPU time. We recall that a pseudo-Bayesian efficiency lower bound on the D-efficiency of a design ξ is defined as $e^{-s_m/m}$, where s_m is the maximum positive value of the sensitivity function across the design space. Figure 3.4 displays the sensitivity function plot to confirm the optimality of the design. This figure not only enables us to check if the equivalence theorem is satisfied at all points over the design space, but also confirms the number of locations of the support points of the optimal design. We observe that on average, generating a pseudo-Bayesian optimal design takes about 891 seconds (about 15 minutes) and confirming the locally optimal design by the equivalence theorem and calculating a efficiency lower bound take about 135 seconds (about 2 minutes). The algorithm produces a 8-point design with an efficiency lower bound of 99.9%; so it is optimal for all practical purposes.



```
> bayesiangamma_function()
[1] "Design Matrix"
                  x1   x2 weight
Design Point 1 0.00 0.00   0.15
Design Point 2 0.41 0.00   0.13
Design Point 3 0.44 1.00   0.01
Design Point 4 0.00 0.41   0.13
Design Point 5 0.00 1.00   0.16
Design Point 6 0.43 0.43   0.09
Design Point 7 1.00 0.00   0.16
Design Point 8 1.00 1.00   0.16
[1] "Lower Efficiency Bound"
[1] 0.9999998
[1] "time used to find local optimal design"
   user  system elapsed
 889.06    0.15  891.01
[1] "time used to find lower efficiency bound"
   user  system elapsed
 134.17    0.01  134.66
```

Figure 3.3: Results found by SG-PSO for the Gamma distributed generalized linear model model.

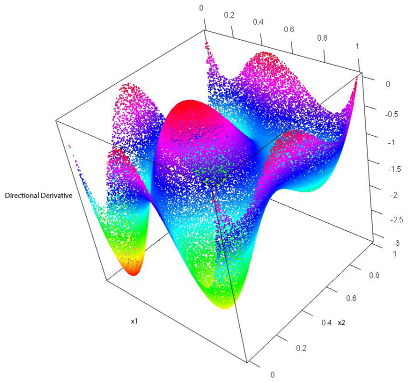


Figure 3.4: Sensitivity function plot of the pseudo-Bayesian optimal approximate design with Gamma distributed responses

We note that, for the configuration we used for this example, the design points x_1 and x_2 are inside the [0, 1] design interval. We note that our algorithm works fine when we allow the design space for each factor to be [0, u] and u >1. If we allow negative values in the design space, the algorithm becomes less robust.

3.5.4 An exponential model for HIV studies

In HIV studies, exponential models are frequently used to characterize viral load changes with time after administration of a potent inhibitor of HIV-1 protease in vivo [PNM96]. Derived from a series of ordinary differential equations that describes the virus change in different compartments, such models are good representations of longitudinal HIV dynamics. The important parameters in such models include virus clearance rate and infected cell life span [WD99].

We adopt a simplified framework developed in [WD99] which models the the total virus copies or its log-transformation for the i-th subject at time t_j as:

$$Y_{t_j} = \sum_r P_r e^{-\delta_r t_j} + \epsilon_{tj},$$

where P_r represents the initial viral production rate, δ_r represents the exponential decay rate of virus in the compartment and $\epsilon_{tj} \sim N(0, \sigma^2)$ are normally distributed isotropic residual. For the exponential model under fairly unrealistic conditions, some analytic locally optimal designs and pseudo-Bayesian optimal designs are available [HC03, DN97], but they may not be flexible for practical use.

To study the drug effect and understand the longitudinal viral dynamics, a schedule to sample plasma to measure the HIV-1 RNA copies must be set up in advance. In this subsection, we show that SGPSO can directly find optimal designs for exponential regression models, such as those summarized in [HC03]:

$$Y_j = P_1 e^{-\delta t_j} + \epsilon_j \tag{3.8}$$

$$Y_j = P_0 + P_1 e^{-\delta t_j} + \epsilon_j \tag{3.9}$$

$$Y_j = log(P_0 + P_1 e^{-\delta t_j}) + \epsilon_j \tag{3.10}$$

where Y_j is the viral load at time t_j, $\epsilon_j \overset{iid}{\sim} N(0, \sigma^2)$, P_0, P_1, $\delta > 0$ are unknown parameters, and $t_j \in [t_{min}, t_{max}] \subseteq [0, +\infty)$ are the sampling times to be determined. We follow [HC03] and use the following independent priors: P_0, P_1 $\sim$ unif(0.5, 1.5), and $\delta \sim$ unif(0.9, 1.1). Additionally we have a different specification for model 3.10 with the same prior distributions for P_0 and P_1 but with $\delta \sim$ unif(0, 0.2) on $t \in$ [0, 60]. We call model (3.10) with the former specification model 3.10-1 and the latter model 3.10-2, and compare properties of the resulting pseudo-Bayesian optimal designs under different specifications. The priors can be quite flexible and they do not have to be independent. Specifically, model 3.10 describes the trajectory of plasma HIV RNA level under antiviral treatment [WD99]; model 3.8 and 3.9 are special cases when P_0 is treated as a nuisance parameter. [HC03] reported pseudo-Bayesian optimal design for model 3.8 when $t \in$ [0, 2] and SGPSO is able to replicate the result. The above three models are simplified exponential regression models with two to three parameters, and one can easily extend them to more complicated models with more parameters based on the disease stages. The computation should be similar.

We compare performances of SG and MC for the numerical integration process. We used 1000 MC samples and set the SG accuracy level at 3. The PSO parameters were set similarly to other examples except that the maximum iteration number was set to 1000. Table 3.4 reports pseudo-Bayesian optimal designs for these models and the CPU time required by SGPSO to generate them.

Table 3.4: SGPSO-generated optimal designs for exponential models.

Model	#Parameters	Design	CPU time(s)
		(time, weight)	SG/MC
3.8	2	((0, 1/2), (1.01, 1/2))	2.31/80.39
3.9	3 (1 nuisance)	((0, 1/3), (1, 1/3), (t_{max}, 1/3))	4.75/94.25
3.10-1	3	((0, 1/3), (1.27, 1/3), (t_{max}, 1/3))	6.99/113.23
3.10-2	3	((0, 0.32), (10.17, 0.28), (28.30, 0.10), (60, 0.30))	6.55/142.53

The sensitivity function plots of the four SGPSO-generated designs are shown below and they all confirm that the designs are pseudo-Bayesian optimal.

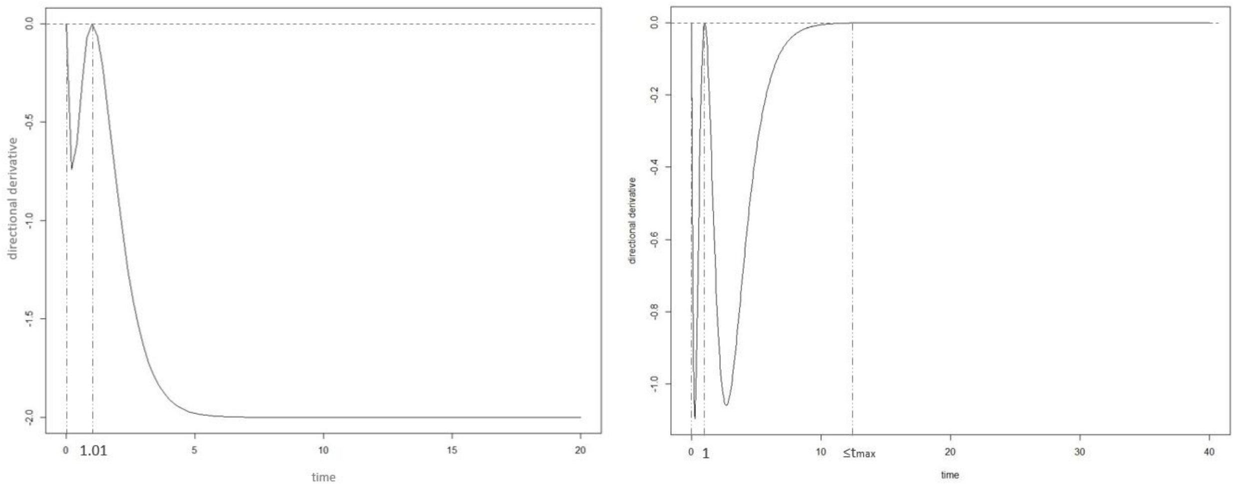


Figure 3.5: Sensitivity function plots of the SGPSO-generated designs for the exponential regression model 3.8 (left) and model 3.9 (right).

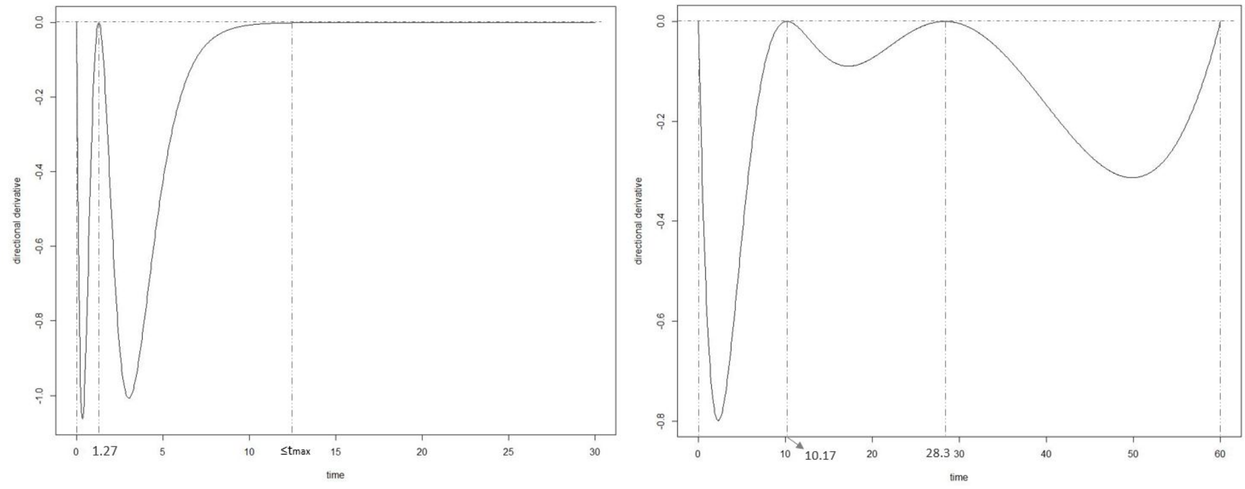


Figure 3.6: Sensitivity function plots of the SGPSO-generated designs for exponential regression model 3.10 under specification 3.10-1 (left) and specification 3.10-2 (right).

The sensitivity function plots of the four PSO-generated designs are shown in figure 3.5 and 3.6 and they all confirm that the designs are pseudo-Bayesian optimal. The plots suggest that the design remains optimal if $t = 14$ instead of t_{max}. We note that changing prior specifications considerably changes the optimal design. For example, model 3.10-2 and model 3.10-1 have different prior specifications on two parameters and we observe that the locations of the support points and the number of support points of the pseudo-Bayesian optimal designs also change. The optimal design for model 3.10-2 requires 4 support points compared to 3 support points for model 3.10-1. Moreover, the optimal design support points for model 3.10-2 are more dispersed, and include middle and end time points, whereas the optimal design support points for model 3.10-1 are concentrated in the earlier time. As is shown in figure 3.7, the differences in designs might be attribute to the fact that larger δ values in equation (3.10) flatten out responses quickly, and design points drawn at earlier time points are critical in providing insights on the exponential decay rate. In contrast, small δ values mean the response curve decay smoothly, and later time points are still on the decreasing curve so that more information can be obtained by allocating design points sparsely.

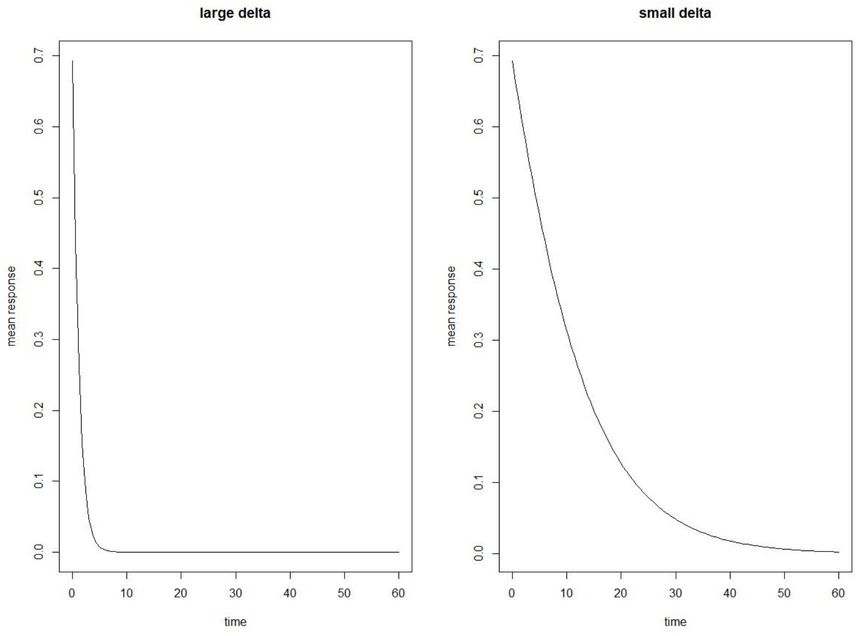


Figure 3.7: Mean response curves of model 3.10 with δ fixed at 1 (left) and 0.1 (right) from mean values of the parameters.

3.6 Exact designs for non-linear mixed effects models

In the following section, we seek to find exact designs for NLMEMs. In practice, the number of repeated measure scientists or practitioners could obtain from the experimental subjects is highly limited, and we use a population approach to find sparse designs that could be useful in real-world settings.

When evaluating optimal exact designs, another criterion value is often used. The optimal design satisfies

$$\xi_D = \underset{\xi \in \Xi^*}{\operatorname{argmax}} \{ (det[M(\theta_0, \xi)])^{1/m} \},$$

where Ξ^* is the set of all designs on the design space, and m is the number of parameters in the model. This normalized FIM determinant is the D-criterion value and the larger this value is, the better is the design. In this section, we will use this normalized FIM as the criterion.

3.6.1 An Emax model with continuous outcomes

This example is inspired by the clinical development of a monoclonal antibody indicated in the treatment of wet age related macular degeneration (wet-AMD) [RBH06]. The treatment reduces vessel leakiness and improves visual acuity (VA) by neutralizing the vascular endothelial growth factor (VEGF) in the retina. The trial duration was 24 months with several VA measurements per subject randomized to receive one of into different dose groups.

We intend to use our proposed algorithm to optimize the VA measurement times, doses levels and subject allocations to these dose levels for a similar but hypothetical study, which has a published model for disease progression and anti-VEGF treatment effect [BUF18].

In wet-AMD, disease progression and treatment effect are assessed using a VA test based on the early treatment diabetic retinopathy study (ETDRS) chart which contains 14 lines (70 letters). The ETDRS chart is used to measure the number of letters successfully read by the subject at a given visit. We use the following model to describe the time course of

VA in wet-AMD subjects, with or without anti-VEGF treatment:

$$f(d,t,\theta) = VA_0 + (1 - e^{-k\,t}) * \Big(\frac{E_{max}\,d}{ED_{50} + d} - \beta\,VA_0\Big). \tag{3.11}$$

The vector of model parameters θ has 5 elements: $\theta^T = (VA_0, k, \beta, E_{max}, ED_{50})$. VA_0 is the visual acuity at baseline and in the absence of treatment. VA exponentially decreases over time t in untreated subjects, reaching an asymptote $VA_0 \times (1 - \beta)$ at a rate k. Based on k, one can derive the average time to steady state $(5 \times \frac{log(2)}{k})$ which equals approximately 24 months and corresponds to the end of the trial. In line with the literature, the model mimics a mixture of both disease modifying and symptomatic effects. The dose-response relationship is assumed to be an Emax function, where E_{max} represents the maximum number of letters that an individual can gain and ED_{50} is the dose at which 50% of the maximal effect is achieved. For a very high dose and at a steady state, the predicted visual acuity is $VA_0 \times (1 - \beta) + E_{max}$.

Inter-individual variability is assumed for parameters VA_0, k, β, and E_{max}, with various variances. We assume that the E_{max} parameter has a normal distribution, namely, $E_{max}{=}\mu_{E_{max}} + b_{E_{max}}$ where $b_{E_{max}} \sim \mathcal{N}(0, \omega^2_{E_{max}})$. We assume the remaining vector of parameters (VA_0, k, β) follows log-normal distributions, namely, $(VA_0, k, \beta) = (\mu_{VA_0} \exp(b_{VA_0})$, $\mu_k \exp(b_k), \mu_\beta \exp(b_\beta))$ where $(b_{VA_0}, b_k, b_\beta) \sim \mathcal{N}(0, diag(\ \omega^2_{VA_0}, \omega^2_k,\ \omega^2_\beta))$. Table 3.5 displays nominal values of the model parameters.

Table 3.5: Parameter settings of the model (3.11) (μ is the fixed effects, ω^2 represents the variance of each of the variance components, and σ^2 is the variance of the errors).

Parameter	**distribution**	μ	ω^2	σ^2
VA_0 $(Letter)$	log-normal	55	0.07	-
k (Day^{-1})	log-normal	0.005	0.5	-
β	log-normal	0.2	1.0	-
E_{max} $(Letter)$	normal	30	150	-
ED_{50} (μg)	log-normal	150	-	-
σ^2 $(Letter^2)$	-	-	-	28

Figure 3.8 displays the predicted VA time course curves from 0 to 24 months for the four dose groups (0, 50, 100, 500 μg) used in simulations in [BUF18].

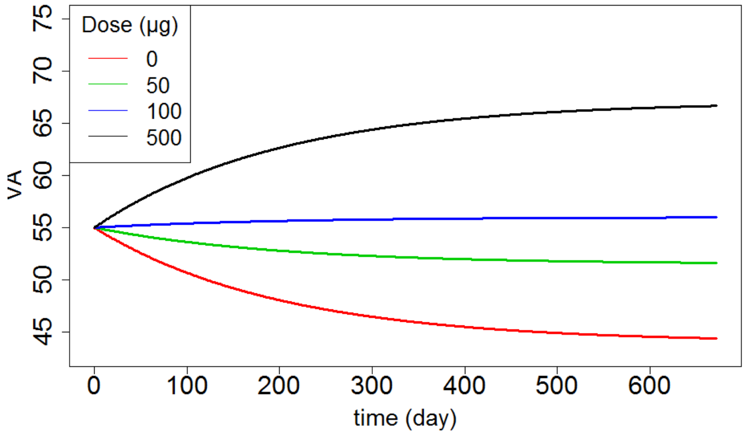


Figure 3.8: The time course of the visual acuity (VA, in *Letter*) predicted by model (3.11) for the four dose groups: 0, 50, 100 and 500 μg.

In the problem at hand, there are 300 subjects to be randomized to different dose groups or levels between 0 to 500 μg. The reference (non-optimized) design has 4 doses (0, 150, 300, 500 μg) and 4 measurement times were taken from each of the dose groups at 0, 224, 420, 672 days.

We now apply PSO to optimize the sampling time points and dose levels under various assumptions in four scenarios. For the "Fix patient allocation" scenario, we fix the subject allocation the same as the reference design, and find dose levels (4 levels) and sampling times (16 time points) that maximizes the D-criterion value. Similarly, for the "Fix sampling times" scenario, we fix the sampling time points the same as the reference design, and optimize the four dose levels and subject allocations in the four groups. For the "Fix doses" scenario, we fix the four dose levels and optimize the four time points in each group and the subject allocation in each group. For the "All flexible" scenario, we search for the optimal four dose levels, sampling time points and subject allocation. In each case, the constraint is that the total cohort is 300 patients for the study.

The population FIM is evaluated using first-order approximation of the model with the R program PFIM 4.0 [DLL18]. Following convention, we set PSO parameters for this example as follows: 1000 maximum number of iterations, 40 particles, inertia linearly decreasing from 0.9 to 0.4 and $c_1 = c_2 = 2$ [SE99].

Table 3.6 summarizes results obtained by PSO for the scenarios. The D-criterion values for the four PSO-generated designs are reported. The four PSO-generated designs and their D-efficiencies relative to the reference design are reported in the parentheses under the D-criterion value. Table 3.6 also shows that the FO computation is highly efficient and CPU times for 1000 iterations are all under 25 seconds for the four scenarios.

Table 3.6: PSO-generated designs for model (3.11) under four scenarios. When appropriate, the optimized doses (μg), or the sampling times (days), or the number of subjects per group are shown in the middle columns. The criterion values and relative efficiencies of the designs are displayed in the second to the last column. The CPU time required to generate each design in the last column is the averaged time in seconds for the 10 runs.

	Doses	Sampling times	# subjects per group	D-criterion (RE)	CPU time
Reference design	0	(0, 224, 420, 672)	75	15.451	
	150	(0, 224, 420, 672)	75		
	300	(0, 224, 420, 672)	75		
	500	(0, 224, 420, 672)	75		
Fix patient allocation	0	(0, 180.86, 180.86, 672)	75	19.525 (126%)	17.1
	69.15	c(0, 0, 672, 672)	75		
	500	(0, 170.36, 170.36, 672)	75+75		
Fix sampling times	0	(0, 224, 420, 672)	110	17.721 (115%)	18.1
	77.8	(0, 224, 420, 672)	57		
	500	(0, 224, 420, 672)	133		
Fix doses	0	(0, 179.62, 179.62, 672)	116	19.028 (123%)	18.0
	150	(0, 0, 672, 672)	58		
	500	(0, 170.73, 170.73, 672)	126		
All flexible	0	(0,178.90, 179.36, 672)	105	19.714 (128%)	23.1
	74.8	(0, 0, 672, 672)	47		
	75	(0, 0, 672, 672)	14		
	500	(0, 171.08, 171.54, 672)	134		

Compared to the reference design, the designs we generated have higher D-criterion values and their D-efficiencies relative to the reference design range from 114.7% to 127.6%. Not surprisingly, the "All flexible" scenario has the highest efficiency. The results show that optimizing the sampling time points has the greatest impact on the D-criterion value and the “Fix sampling times” scenario has the least efficiency gain. The optimized sampling time points tend to concentrate at the end points and earlier time points compared to the reference design before the response curve becomes flat. We observe that the PSO-generated designs have one fewer dose levels for the “Fix patient allocation”, “Fix sampling times” and “Fix doses” scenarios. There are two dose levels for the “All flexible” scenario that are fairly close (74.8 and 75). The designs with fewer dose levels might save costs and reduce the number of subjects exposed to high and potentially unsafe dose levels.

3.6.2 A logistic model

3.6.2.1 A 2-parameter example

This example is a one-year study with repeated binary outcomes (responder vs non-responder) in two balanced treatment groups, inspired from [LUM16, SNMon]. The response probability over time is described by a mixed logistic model, including a covariate for the treatment effect. We aim to determine the location of a limited number of measurement times between 0 and 12 months.

Specifically, we have binary responses ($y = 0$ or 1) recorded over time ($t = 0$ to 12 months) in two different treatment groups. The response probability at time t is given by

$$p(y=1|b) = \frac{\exp\Big(\theta_1 + \theta_2(1+\beta\times 1_T)t\Big)}{1+\exp\Big(\theta_1 + \theta_2(1+\beta\times 1_T)t\Big)}, \tag{3.12}$$

where 1_T is the treatment group indicator variable (with $1_T = 0$ if placebo group and $1_T = 1$ if treated group) and β is the treatment effect. We assume the parameters θ_1 and θ_2 follow a normal distribution: $(\theta_1, \theta_2) = (\mu_{\theta_1}, \mu_{\theta_2}) + (b_{\theta_1}, b_{\theta_2})$ where $(b_{\theta_1}, b_{\theta_2}) \sim \mathcal{N}(0, diag(\omega^2_{\theta_1}, \omega^2_{\theta_2}))$. The nominal values of the model parameters are given in table 3.7.

Table 3.7: Parameter settings of the model (3.12) (μ is the fixed effects, ω^2 is the variance of the random effects and β is the treatment effect).

Parameter	**distribution**	μ	ω^2	β
θ_1	normal	-2	0.49	-
θ_2	normal	0.09	0.03	-
β	-	-	-	5

Figure 3.9 displays the model predicted curves from 0 to 12 months for the two groups.

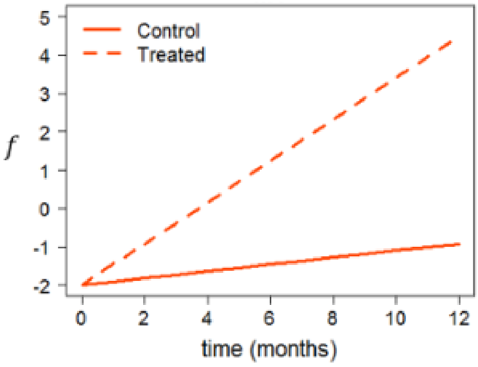


Figure 3.9: Mean of the logit of the response probability (f) over 12 months predicted by the model (3.12). The corresponding parameter values are in table 3.7. The solid line represents the control group and the dashed line represents the treated group.

We next apply PSO to optimize measurement times $\xi = \{t_1, t_2, t_3, t_4\}$ from 0 to 12 months for subjects in both treatment arms based on the D-optimality. We assume $N = 100$ subjects equally distributed in the two treatment groups. The end points t_1 and t_4 can either be fixed at 0 and 12 months (*i.e.* at the start and the end of the study) and only optimize the middle two points t_2 and t_3, or optimize all four time points at the same time. We then consider two situations: one uses the same elementary designs for both arms and the other uses different elementary designs for the two arms. In total, we want to apply our algorithms to generate designs for model (3.12) for four scenarios. To this end, we first apply the QRMC-AGQ method, implemented in R by [UM15], to evaluate the FIM for the repeated binary responses. Several settings were evaluated and the following settings were chosen:

the number of QRMC samples were fixed to 1000 and the number of AGQ nodes to 7.

We set PSO parameters for this example as follows: 100 maximum number of iterations, 40 particles, inertia linearly decreasing from 0.9 to 0.4 and $c_1 = c_2 = 2$. The optimization for this task is more time consuming than that in application 1 because of the numerical integration process that application 1 avoided by the FO method. We used 100 iterations for the computational feasibility consideration and depending on the scenario, the 100 iterations can take up to 2.2 hours.

Table 3.8: PSO-generated designs for model (3.12) for the four scenarios. The sampling times are in months and the CPU times are in seconds.

		Group	Sampling times	D-criterion (RE)	CPU time
Reference		Treated	(0, 4, 8, 12)	36.673	
		Placebo	(0, 4, 8, 12)		
Same elementary designs for 2 groups	Fix end points	Treated	(0, 1.5, 6.2, 12)	40.889 (111%)	6542.5
		Placebo	(0, 1.5, 6.2, 12)		
	Flexible end points	Treated	(0, 1.51, 6.17, 11.98)	40.980 (112%)	7253.2
		Placebo	(0, 1.51, 6.17, 11.98)		
Different elementary designs for 2 groups	Fix end points	Treated	(0, 1.5, 2.8, 12)	43.243 (118%)	6859.7
		Placebo	(0, 4.2, 9.1, 12)		
	Flexible end points	Treated	(0, 1.4, 2.0, 7.6)	44.542 (121%)	7942.1
		Placebo	(0, 3.9, 9.7, 11.6)		

Table 3.8 contains the results and they show that maximizing 2 time points (fix 0 and 12) or 4 time points often give similar design. The PSO-generated designs have D-efficiencies relative to the reference design that range from 111% to 122%. Allowing different elementary designs for the treatment and the placebo groups results in higher efficiency, and the most flexible scenario that has the most design aspects to optimize gives the highest efficiency. The best results found under this scenario suggests that the sampling times should be set

up differently for the treated and the placebo groups. A key conclusion is that the treated group should have more earlier time points.

3.6.2.2 A higher dimensional example: gains of using sparse grid

This example is a more complicated version of application 2 with binary outcomes. The model now has four random effects parameters and we show that it is possible to use PSO coupled with sparse grid to quickly find more efficient designs than the reference design for moderately high dimensional models under different scenarios.

Our model describes binary responses ($y = 0$ or 1) recorded over time ($t = 0$ to 12 months) for two treatment groups. The response probability at time t is

$$p(y = 1|b) = \frac{\exp\Big(\theta_1 + \theta_2(1 - \beta 1_T)\exp(\theta_3 t - 1)\Big)}{1 + \exp\Big(\theta_1 + \theta_2(1 - \beta 1_T)\exp(\theta_3 t - 1)\Big)}, \tag{3.13}$$

where 1_T is a treatment group indicator variable (with $1_T = 0$ for the placebo group and $1_T = 1$ for the treated group) and β is a treatment effect covariate. The parameters θ_1, θ_2, β and θ_3 follow normal distributions: $(\theta_1, \theta_2, \beta, \theta_3) = (\mu_{\theta_1}, \mu_{\theta_2}, \mu_\beta, \mu_{\theta_3}) + (b_{\theta_1}, b_{\theta_2}, b_\beta, b_{\theta_3})$ where $(b_{\theta_1}, b_{\theta_2}, b_\beta, b_{\theta_3}) \sim \mathcal{N}(0, diag(\omega^2_{\theta_1}, \omega^2_{\theta_2}, \omega^2_\beta, \omega^2_{\theta_3}))$, and table 3.9 displays the nominal values of the model parameters.

Table 3.9: Parameter settings of model (3.13) where μ are the fixed effects and ω^2 represents the variance of each of the variance components.

Parameter	**distribution**	μ	ω^2
θ_1	normal	-1	0.5
θ_2	normal	4	4
β	normal	0.4	0.11
θ_3	normal	0.33	0.11

Figure 3.10 shows the predicted curves from 0 to 12 months for the two treatment groups.

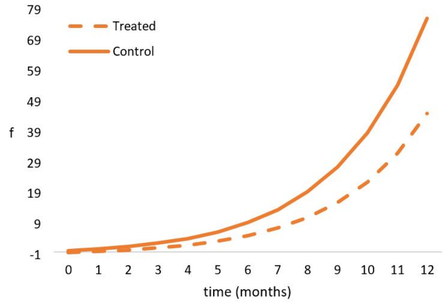


Figure 3.10: Mean of the logit of the response probability (f) over 12 months predicted by model (3.13) using parameter values in table 3.9. The solid line represents the control group and the dashed line represents the treated group.

PSO is used to optimize measurement times from 0 to 12 months for subjects in both treatment arms based on the D-optimality. The PSO parameters for this example are: 50 maximum number of iterations, 40 particles, inertia linearly decreasing from 0.9 to 0.4 and $c_1 = c_2 = 2$. The optimization for this task is much more time consuming than that in applications 1 and 2 because of the higher-dimensional numerical integration process and more complicated model structure. We used 50 iterations for computational feasibility consideration.

Under the same design settings as in the previous example, we applied the QRMC-SG method, implemented in R, to evaluate the FIM for the repeated binary responses. Several settings were evaluated and the following settings were chosen: the number of QRMC samples were fixed to 500 and the number of 1-dimensional SG nodes to 3. We also compare QRMC-SG to QRMC-AGQ to show the benefit of SG. The number of 1-dimensional AGQ nodes is also fixed to 3. In total, AGQ requires 3^4=81 grid points while SG requires only 33 grid points. The choice of these parameters considered both approximation accuracy and computational budget. Increasing 1-dimensional nodes number from 2 to 3 provides significant accuracy gain while from 3 to 4 provides not much improvement.

Table 3.10 displays the SGPSO results and CPU times required by SGPSO and AGQ. We observe from the table that (i) the D-efficiency of PSO-generated designs relative to the reference design range from 140% to 191%, (ii) optimizing all 8 time points has almost twice efficiency as the reference design, (iii) fixing the end time points (0 and 12 months) limits the overall efficiency, (iv) allowing different elementary designs for treatment and placebo groups results in higher efficiency, and (v) the CPU time is generally tolerable, ranging from 45 to 51 minutes, depending on the complexity of the problem.

Table 3.10: PSO-generated designs for model (3.13) for four scenarios. CPU time is in seconds.

		Group	Sampling times	D-criterion (RE)	CPU time (SG/AGQ)
Reference		Treated	(0, 4, 8, 12)	16.555	
		Placebo	(0, 4, 8, 12)		
Same elementary designs for 2 groups	Fix end points	Treated	(0, 2.4, 5.6, 12)	23.120 (140%)	2714/5011
		Placebo	(0, 2.4, 5.6, 12)		
	Flexible end points	Treated	(0, 0.2, 2.3, 9.0)	29.658 (179%)	2735/6433
		Placebo	(0, 0.2, 2.3, 9.0)		
Different elementary designs for 2 groups	Fix end points	Treated	(0, 3.1, 10.1, 12)	26.495 (160%)	2758/6447
		Placebo	(0, 0.8, 4.3, 12)		
	Flexible end points	Treated	(0.7, 3.1, 4.3, 9.0)	31.640 (191%)	3083/6489
		Placebo	(0, 2.4, 3.5, 8.6)		

If AGQ were to be used, even the simplest scenario takes 1.4 hours, almost twice as much time but not better results. This is not surprising, since AGQ requires as many as 145% more nodes to be evaluated compared to SG. However, we note that PSO/AGQ and PSO/SG find designs for the four scenarios have similar D-efficiencies relative to the reference design and they are 137.7%, 171.7%, 173.9% and 211.3% respectively. The computational gain by PSO/SG in time is critically important when we need to evaluate more candidate designs, tackle models with more random effects parameters, and allows us to find efficient designs for more complicated models, which are increasingly more common in practice.

3.6.2.3 A comprehensive comparison of using adaptive Gaussian quadrature vs. sparse grid on logistic models

Here, we conduct a more comprehensive comparison of utilities of adaptive Gaussian quadrature (AGQ) versus sparse grid (SG) for our models and problems. To this end, we consider a series of longitudinal logistic NLMEMS with different numbers of random effects and use two techniques to integrate out the random effects. This section only compares AGQ and SG to show the computational gains of using SG over AGQ for moderate dimensional problems, and does not involve the optimization processes. Below are the models we consider and listed in increasing order of their complexity.

The simplest model with 2-dimensional random effects is:

$$logit\ E(y) = \theta_1 + \theta_2(1 - 0.4 \times 1_T)t$$

where $(\theta_1,\ \theta_2) = \mu + b$, the fixed effects are μ=(-1, 4) and the vector of random effects b follows a joint normal distribution $N(\mathbf{0}, diag(0.5, 4))$.

A 3-dimensional random effects is:

$$logit\ E(y) = \theta_1 + \theta_2(1 - \beta 1_T)t$$

where $(\theta_1,\ \theta_2,\ \beta) = \mu + b$, the fixed effects are μ=(-1, 4, 0.4), and the vector of random effects b follows a joint normal distribution $N(\mathbf{0}, diag(0.5, 4, 0.01))$.

Next, a model with 4-dimensional random effects is:

$$logit\ E(y) = \theta_1 + \theta_2(1 - \beta 1_T)exp(\theta_3 t - 1)$$

where $(\theta_1,\ \theta_2,\ \beta,\ \theta_3) = \mu + b$, the fixed effects are μ = (-1, 4, 0.4, 0.33), and the vector of random effects b follows a joint normal distribution $N(\mathbf{0}, diag(0.5, 4, 0.01, 0.01))$.

The most complicated case we consider is a model with 5-dimensional random effects:

$$logit\ E(y) = \theta_1 + \theta_2(1 - \beta_0 - \beta_1 1_T)[exp(\theta_3 t) - 1]$$

where (θ_1, θ_2, β_0, β_1, θ_3) $= \mu + b$, the fixed effects are $\mu =$ (-1, 4, 0, 0.4, 0.33), and the vector of random effects b follows a joint normal distribution $N(\mathbf{0}, diag(0.5, 4, 0.01, 0.01, 0.01))$.

We compare designs found by the two techniques based on nodes of different accuracy levels, the corresponding the CPU times and the criterion values. The outer integration with respect to the observations are integrated out by QRMC. For the purpose of comparison of AGQ and SG. We used 500 QRMC samples and used the same 1-dimensional Gaussian quadrature. The reference design used for comparison allocates 50 patients to each of the treated and placebo groups and the sampling times are at month 0, 4, 8, and 12.

Figure 3.11 shows the number of AGQ and SG node points at different accuracy levels for the 2-5 dimensional models. The number of node points directly relates to the number of objective function evaluations. More node points require more evaluations. When the dimension is 2, AGQ is more efficient than SG; when the dimension is 3, SG is slightly more efficient than AGQ; and when the dimensions are 4 and 5, it is clear that SG requires much fewer node points than AGQ. It is clear that AGQ suffers "curse of dimension problem" while SG mitigates this problem to some extent. Table 3.11 summarizes the number of points required by AGQ and SG for different dimensions.

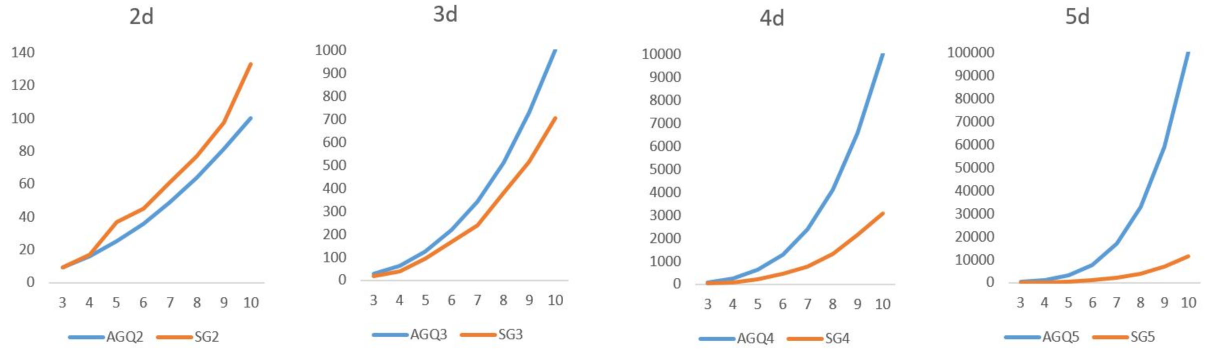


Figure 3.11: Numbers of 2-5 dimensional node points at different accuracy levels

Table 3.11: Numbers of 2-5 dimensional node points required by AGQ and SG at different accuracy levels.

accuracy level	AGQ2	SG2	AGQ3	SG3	AGQ4	SG4	AGQ5	SG5
3	9	9	27	19	81	33	243	51
4	16	17	64	39	256	81	1024	151
5	25	37	125	93	625	201	3125	401
6	36	45	216	165	1296	441	7776	993
7	49	61	343	237	2401	761	16807	2033
8	64	77	512	381	4096	1305	32768	3793
9	81	97	729	513	6561	2129	59049	6913
10	100	133	1000	703	10000	3065	100000	11323

The number of node points affects CPU times, but not in a linear way because of other computer system related factors. Nevertheless, we clearly observe the same trend in the CPU time comparison as we observed in number of node points comparison. Figure 3.12 shows that the CPU times required by AGQ and SG at different accuracy levels for evaluating the 2-5 dimensional models. We observe that for the 2-dimensional case, AGQ outperforms SG; then SG starts to outperform AGQ for the 3-dimensional case, and the the trend becomes pronounced for the 4- and 5-dimensional model. In particular, we note that for the 5-dimensional case, SG saves up to 87% of CPU time. Table 3.12 summarizes the CPU time for 2-5 dimensional models using AGQ and SG at various accuracy levels.

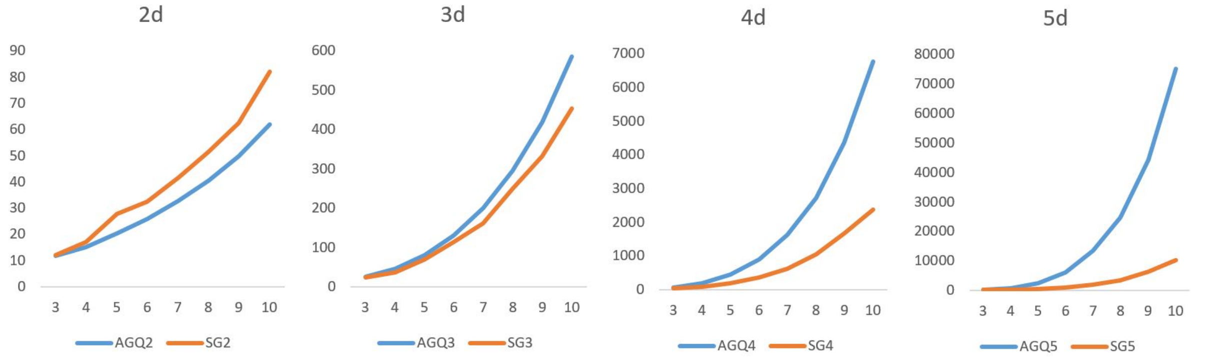


Figure 3.12: CPU times (seconds) required by AGQ and SG for evaluating the 2-5 dimensional models at different accuracy levels.

Table 3.12: CPU times (seconds) required by AGQ and SG for evaluating the 2-5 dimensional models at different accuracy levels.

accuracy level	AGQ2	SG2	AGQ3	SG3	AGQ4	SG4	AGQ5	SG5
3	11.79	12.12	24.84	22.96	68.15	41.9	200.1	68.66
4	15.14	16.98	44.87	37.06	185.4	81.72	789.7	162.47
5	20.26	27.85	80.96	69.9	436.61	178.6	2446.45	390.17
6	25.89	32.57	130.83	113.81	891.44	362.87	6139.61	928.72
7	32.62	41.62	200.53	161.03	1627.55	618.89	13317.84	1860.11
8	40.42	51.56	296.1	250.1	2720.44	1050.75	24644.29	3365.9
9	49.77	62.41	418.03	332.45	4373.07	1657.87	44401.22	6195.67
10	61.83	82.18	585.39	452.97	6760.44	2374.22	75047.77	10111.11

We also compare the criterion values approximated by the two techniques. Ideally, we want the two techniques to have the same or very similar values. The criterion here is the normalized Fisher information matrix determinant. We observe that AGQ and SG agrees well, especially at higher accuracy levels. Figure 3.13 shows the criterion values obtained by the two techniques for the 2-5 dimensional models at different accuracy levels. At accuracy level 5 or 6, the difference becomes indistinguishable. Both methods have some discrepancy at lower accuracy but become negligible at higher accuracy levels. As the model becomes more complicated, higher accuracy levels are needed as in the 5-dimensional model. Table 3.13 summarizes the criterion values. There is no magnitude difference across models at different accuracy levels, and it is clear that SG gives the same or very close evaluation, while saving computational efforts considerably.

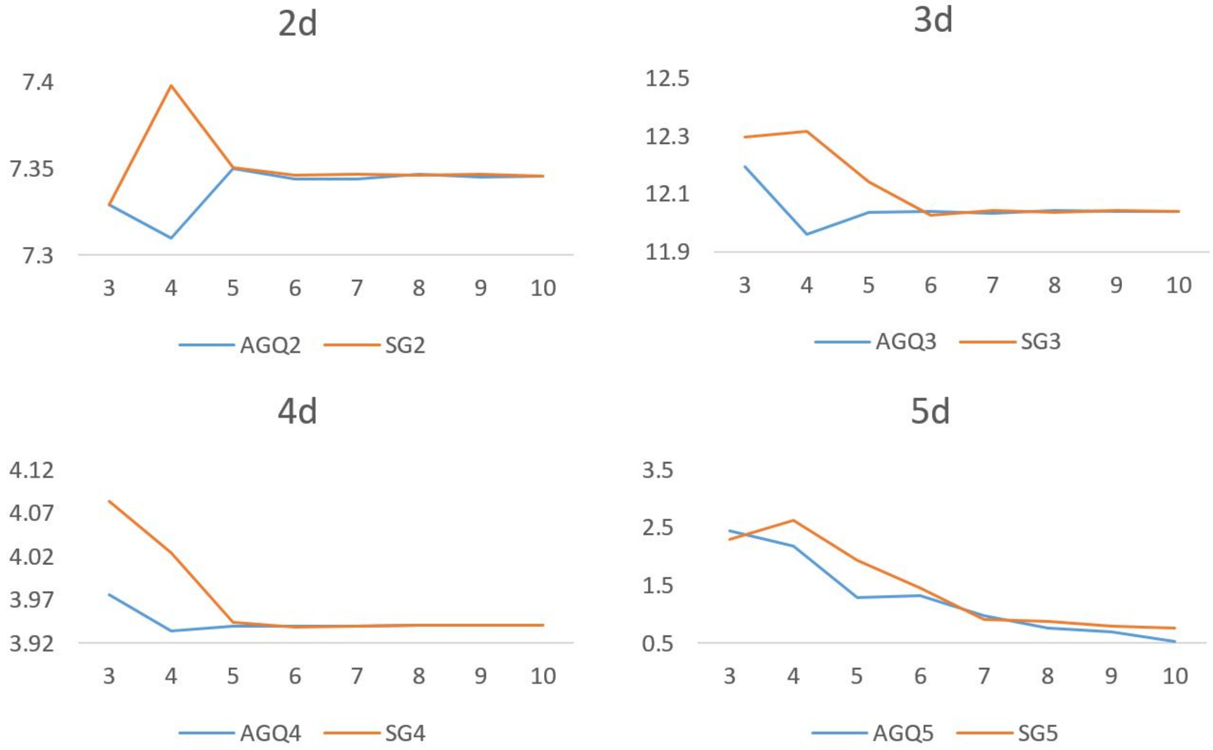


Figure 3.13: Criterion values of 2-5 dimensional models at different accuracy levels

Table 3.13: Criterion values of 2-5 dimensional models at different accuracy levels.

accuracy level	AGQ2	SG2	AGQ3	SG3	AGQ4	SG4	AGQ5	SG5
3	7.329	7.329	12.194	12.298	3.976	4.084	2.451	2.295
4	7.310	7.398	11.960	12.316	3.935	4.024	2.179	2.625
5	7.350	7.350	12.037	12.142	3.940	3.944	1.284	1.938
6	7.344	7.346	12.039	12.027	3.940	3.939	1.325	1.457
7	7.344	7.347	12.033	12.043	3.940	3.940	0.984	0.904
8	7.347	7.346	12.041	12.036	3.940	3.941	0.758	0.877
9	7.345	7.347	12.038	12.042	3.940	3.940	0.695	0.802
10	7.346	7.346	12.039	12.041	3.940	3.940	0.524	0.761

We conclude this section by discussing on the proper accuracy level to use to approximate FIMs for NLMEMs with different number of random effects. Sparse grids have a great advantage when the dimension is moderately high. A 2-dimensional model doesn't benefit much from using SG, but a higher dimensional model using SG is computationally much more efficient than using AGQ. We emphasize that for the types of problems and models we have, our work shows for moderate dimensional NLMEMs, SG is a highly accurate, and computationally efficient algorithm, compared to AGQ.

3.6.2.4 A comparison of optimization algorithms for finding efficient designs of a 4-parameter logistic NLMEM

In this subsection, we compare PSO with three other competing metaheuristic optimization algorithms and show that PSO is computationally efficient and finds comparable or better designs. The algorithms we considered include genetic algorithm (GA) [BGH89], differential evolution (DE) [SP97], and a variant of PSO called quantum-inspired PSO (QPSO) [SFX04]. We discussed GA and QPSO in Chapter 2. DE adopts similar evolution strategy as GA and creates new candidate solutions based on the weighted difference between two randomly selected population members added to a third population member [SP97]. We use these algorithms coupled with SG to find efficient designs for the 4-parameter logistic NLMEM in section 3.6.2.2. We consider the most flexible scenario, which allows different elementary designs for the treated and placebo groups, and we optimize all four time points for both groups.

The parameters of GA are set as follows: the population size is 50, the number of generations is 50, elitism is 2, cross-over probability is 0.8, and mutation probability is 0.1. The parameters of DE are set as follows: the population size is 80, the number of generations is 50, step size is 0.8, and cross-over probability is 0.5. The parameters of QPSO are set as follows: the alpha is linearly decreasing from 1.4 to 0.4, the population size is 40, the maximum number of iterations is 50.

Table 3.14 summarizes the results of using PSO, QPSO, GA, and DE when we hybridize them with SG for finding D-optimal exact designs. It is clear that PSO finds designs with better D-criterion values than GA and DE, and its value is comparable with the one found by QPSO. However, PSO is computationally more efficient than QPSO; even though we note from the table that the designs found by the three algorithms are not the same. Another advantage of using PSO is that scientists and practitioners can easily find codes for PSO but not QPSO. We thus conclude that PSO is efficient and is good for practical use.

Table 3.14: A comparison of using different optimization algorithms for finding efficient designs for the 4-parameter logistic NLMEM.

Optimizer	Group	sampling times	D-criterion	Relative efficiency
PSO	Treated	(0.7, 3.1, 4.3, 9.0)	31.64	100.0%
	Placebo	(0, 2.4, 3.5, 8.6)		
QPSO	Treated	(0.8, 2.0, 8.4, 9.1)	31.87	100.7%
	Placebo	(0.7, 1.3, 2.0, 7.4)		
GA	Treated	(1.9, 3.0, 4.1, 8.7)	27.92	88.2%
	Placebo	(1.8, 3.7, 5.0, 8.5)		
DE	Treated	(1.0, 2.0, 2.1, 8.3)	30.90	97.7%
	Placebo	(1.1, 1.8, 3.4, 7.9)		

3.6.3 A Poisson model with count outcomes

Count data often arise in biomedical studies. For instance, in a toxicity study, the outcome could be the number of organisms or cells that survive when exposed to different doses of toxicants. There is limited work to address design issues for such models and they usually only have one or two explanatory variables in the model [RWL09, WMS06, Qiu14]. Inspired by a previous work [UM17], we consider a Poisson NLMEM with various dose levels. To fix ideas, we consider designs that use the same elementary designs for each subject and all responses are non-negative integer values.

Similar to the previous section, we first compare AGQ and SG performances for evaluating designs to show the advantages of SG at higher dimensions. We then work with a 4-parameter Poisson NLMEM to show that SGPSO can find highly efficient designs in a reasonable amount of time.

3.6.3.1 Finding efficient designs for a 4-parameter Poisson NLMEM

The response probability at dose level $D = D_i$ is given by

$$p(y_i = k|b) = \frac{\lambda exp(\lambda)}{k!} \quad with \ \log(\lambda) = \theta_1\big(1 - \theta_4 - \frac{\theta_2 D_i}{D_i + \theta_3}\big) \tag{3.14}$$

where θ_1, θ_2, θ_3 follow a log-normal distribution: $(\theta_1, \theta_2, \theta_3) = (\mu_1, \mu_2, \mu_3)exp(b_1, b_2, b_3)$ where $(b_1, b_2, b_3) \sim N(\mathbf{0}, diag(\omega_1^2, \omega_2^2, \omega_3^2))$, and θ_4 follows a normal distribution: $\theta_4 = \mu_4 + b_4$ where $b_4 \sim N(0, \omega_4^2)$. Table 3.15 provides the nominal values of model parameters.

Table 3.15: Parameter settings of model (3.14) where μ are the fixed effects and ω^2 represents the variance of each of the variance components.

Parameter	**distribution**	μ	ω^2
θ_1	log-normal	1.0	0.09
θ_2	log-normal	0.5	0.09
θ_3	log-normal	1	0.01
θ_4	normal	0	0.01

We assume 20 subjects in total, each contributes 90 observations at different dose levels. The dose levels are to be chosen from 0 to 1. The reference (non-optimized) design has 3 doses (0, 0.4, 0.7) and 30 observations at each level [Riv16, UM17].

We evaluate the population FIM using QRMC-AGQ and QRMC-SG. PSO is used to optimize the dose levels and observation allocations based on D-optimality. We consider two scenarios. For "Fix allocation" scenario, we fix the observation allocation the same as the reference design, and find 3 dose levels that maximize the D-criterion value. For the "Flexible" scenario, we additionally search for the three dose levels. We set PSO parameters for this example as follows: 100 maximum number of iterations, 40 particles, inertia linearly decreasing from 0.9 to 0.4 and $c_1 = c_2 = 2$ [SE99]. We use 1000 QRMC samples and use the same 1-dimensional 3-points Gaussian quadrature for AGQ and SG. The maximum number of iterations considered the computational budget.

Table 3.16 reports the results for the two scenarios, along with the D-criterion value for each design. Relative efficiencies for the four PSO-generated designs compared to the reference design are reported in the parentheses under D-criterion values.

Table 3.16: PSO-generated designs for model 3.14 under two scenarios. AGQ and SG were used for comparison. CPU time is in seconds.

	Dose levels	Observations allocations	D-criterion (RE)	CPU time
Reference Design	0	30	57.691	N/A
	0.4	30		
	0.7	30		
Fix allocation - AGQPSO	0	30	73.098 (135.4%)	27164.47
	0.25	30		
	0.88	30		
Fix allocation - SGPSO	0	30	75.694 (131.2%)	16203.58
	0.25	30		
	0.94	30		
Flexible - AGQPSO	0	19	76.699 (132.9%)	27455.37
	0.21	29		
	0.78	42		
Flexible - SGPSO	0	19	76.699 (132.9%)	16259.36
	0.21	29		
	0.78	42		

We observe from table 3.16 that the we found much more efficient designs than the reference design: our deisngs have relative D-efficiencies ranging from 131% to 135%. SGPSO approach finds equivalent or better designs than AGQPSO; in terms of CPU time, the former can require less than 60% of CPU time than AGQPSO used.

3.6.3.2 A comparison of using adaptive Gaussian quadrature and sparse grid on Poisson NLMEMs

This subsection compares the performances of using adaptive Gaussian quadrature (AGQ) and sparse grid (SG) for Poisson NLMEMEs. We use a series of Poisson regression models with different numbers of random effects via AGQ and SG. Similar to section 3.6.2.3, in this section we only compare AGQ and SG to show the computational gains of using SG over AGQ on moderate dimensional problems, and does not involve the optimization processes.

The responses recorded over time are non-negative integer values (counts). The simplest 2-dimensional NLMEM models the response probability at dose level $D = D_i$ by:

$$p(y_i = k|b) = \frac{\lambda exp(\lambda)}{k!} \;\; with \;\; \log(\lambda) = \theta_1\big(1 - \frac{D_i}{D_i + \theta_2}\big)$$

where θ_1, θ_2 follow a log-normal distribution: $(\theta_1, \theta_2) = (\mu_1, \mu_2)exp(b_1, b_2)$. We assume (μ_1, μ_2)=(1, 0.5) and $(b_1, b_2) \sim N(\mathbf{0}$, diag(0.09, 0.09)).

A more complicated 3-dimensional Poisson NLMEM is:

$$p(y_i = k|b) = \frac{\lambda exp(\lambda)}{k!} \;\; with \;\; \log(\lambda) = \theta_1\big(1 - \frac{\theta_3 D_i}{D_i + \theta_2}\big)$$

where θ_1, θ_2, θ_3 follow a log-normal distribution: $(\theta_1, \theta_2, \theta_3) = (\mu_1, \mu_2, \mu_3)exp(b_1, b_2, b_3)$. We assume (μ_1, μ_2, μ_3)=(1, 0.5, 0.1) and $b_1, b_2, b_3 \sim N(\mathbf{0}$, diag(0.09, 0.09, 0.01)).

The most complicated case we consider is a 4-dimensional Poisson NLMEM:

$$p(y_i = k|b) = \frac{\lambda exp(\lambda)}{k!} \;\; with \;\; \log(\lambda) = \theta_1\big(1 - \theta_4 - \frac{\theta_3 D_i}{D_i + \theta_2}\big)$$

where θ_1, θ_2, θ_3 follow a log-normal distribution: $(\theta_1, \theta_2, \theta_3) = (\mu_1, \mu_2, \mu_3)exp(b_1, b_2, b_3)$. We let (μ_1, μ_2, μ_3)=(1, 0.5, 0.1), $(b_1, b_2, b_3) \sim N(\mathbf{0}$, diag(0.09, 0.09, 0.01)). We assume θ_4 follows a normal distribution: $\theta_4 = \mu_4 + b_4$ where μ_4=0 and $b_4 \sim N(0, 0.01)$.

As before, we compare the two techniques by their number of node points at different accuracy levels, CPU times, and criterion values. The outer integration with respect to the observations are integrated out by QRMC. For the purpose of comparison AGQ and SQ, we fixed all other conditions, including using 1000 QRMC samples and the same 1-dimensional 3-points Gaussian quadrature. We assume 20 subjects in total, each contributes 90 observations at different dose levels between 0 (0%) to 1 (100%). The reference design has 20 patients with 30 observations at each level of the three doses at 0, 0.4, 0.7.

Table 3.11 shows the numbers of AGQ and SG node points at different accuracy levels. Again, we observe a similar and clear trend in the CPU times as in section 3.6.2.3. Figure 3.14 shows the CPU times required by AGQ and SG at different accuracy levels for 2-4 dimensional models. For the 2-dimensional case, AGQ outperforms SG; then SG starts to outperform AGQ for the 3-dimensional case, and the the trend becomes pronounced in 4-dimensional models, with saving in CPU time up to 67% for the 4-dimensional SG. Table 3.17 summarizes the CPU times for 2-4 dimensional models using AGQ and SG at various accuracy levels.

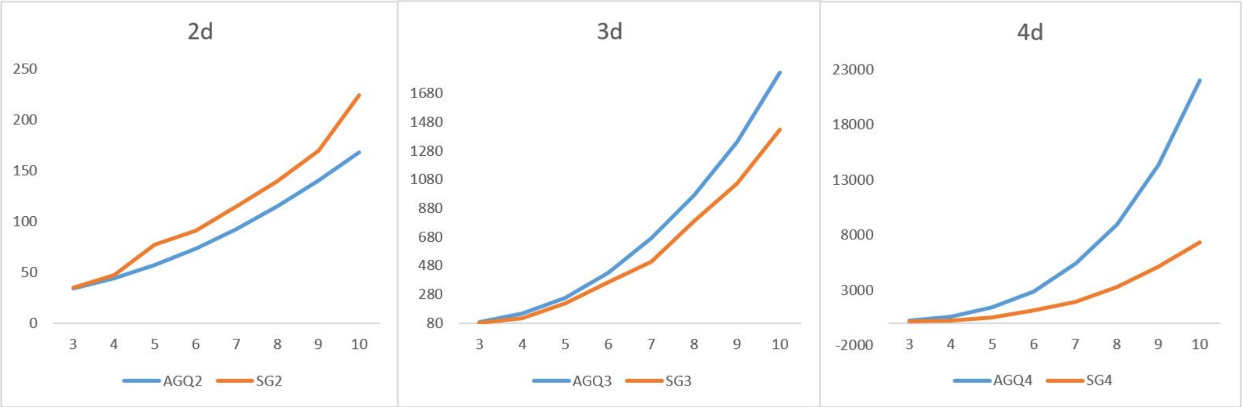


Figure 3.14: CPU times (seconds) required by the AGQ and SG for evaluating 2-4 dimensional models at different accuracy levels

Table 3.17: CPU times (seconds) required by the AGQ and SG for evaluating 2-4 dimensional models at different accuracy levels

accuracy level	AGQ2	SG2	AGQ3	SG3	AGQ4	SG4
3	34.39	35.06	88.71	81.83	228.25	132.86
4	44.14	47.58	147.38	114.45	609.84	255.33
5	57.32	77.6	258.03	220.14	1405.89	552.57
6	73.77	90.98	432.1	363.89	2873.53	1136.06
7	92.99	114.9	667.89	506.42	5429.69	1910.34
8	115.11	139.41	967.62	788.23	8986.94	3260.36
9	140.54	169.53	1338.39	1050.98	14347.99	5160.50
10	168.38	224.5	1824.84	1425.42	22019.89	7334.63

We also compare the criterion values approximated by using the two techniques. The criterion here is the normalized Fisher information matrix determinant. Results from the AGQ and SG techniques agree well, especially at higher accuracy levels. Figure 3.15 shows the criterion values for the generated designs for the 2-4 dimensional models at different accuracy levels. At higher accuracy levels, the difference becomes indistinguishable. Both methods have some discrepancy at lower accuracy levels, and becomes quite small at accuracy level 4. As the model becomes more complicated, higher accuracy levels are needed, as evident from the 4-dimensional model. Table 3.18 summarizes the criterion values. There is no magnitude difference across models at different accuracy levels, and it is clear that SG gives results similar to those from AGQ, while saving computational efforts considerably. In summary, we reach a similar conclusion as we had in section 3.6.2.3: SG is a highly accurate and computationally efficient algorithm compared to AGQ.

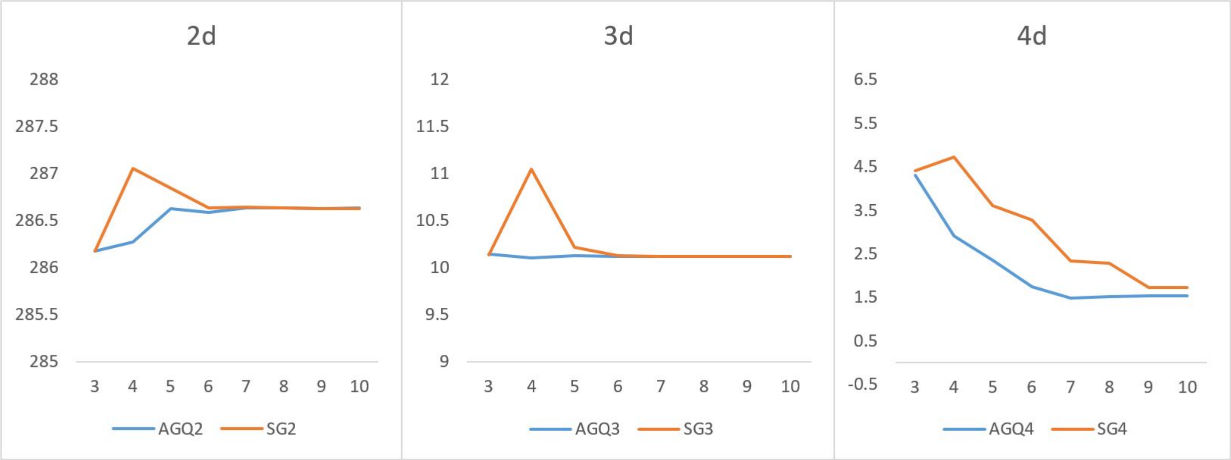


Figure 3.15: Criterion values of the generated designs for the 2-4 dimensional models at different accuracy levels.

Table 3.18: Criterion values of the generated designs for the 2-4 dimensional models at different accuracy levels.

accuracy level	AGQ2	SG2	AGQ3	SG3	AGQ4	SG4
3	286.174	286.174	10.148	10.140	4.307	4.406
4	286.269	287.058	10.108	11.050	2.922	4.724
5	286.628	286.845	10.129	10.215	2.350	3.609
6	286.588	286.636	10.123	10.127	1.741	3.286
7	286.635	286.645	10.124	10.124	1.491	2.342
8	286.637	286.633	10.124	10.124	1.517	2.284
9	286.630	286.631	10.124	10.123	1.533	1.723
10	286.634	286.630	10.12	10.125	1.532	1.732

3.6.3.3 A comparison of optimization algorithms for finding efficient designs of a 4-parameter Poisson NLMEM

In this section, we compare PSO with three other competing metaheuristic optimization algorithms and show that PSO is computationally efficient and finds comparable or better designs. Similar to section 3.6.2.4, the algorithm we considered include genetic algorithm (GA), differential evolution (DE), and quantum-inspired PSO (QPSO). We use these algorithms coupled with SG to find efficient designs for the 4-parameter Poisson NLMEM in section 3.6.3.1. We consider the most flexible scenario, which optimizes dose levels and patient allocations.

The parameters of GA are set as follows: the population size is 50, the number of generations is 100, elitism is 2, cross-over probability is 0.8, and mutation probability is 0.1. The parameters of DE are set as follows: the population size is 80, the number of generations is 100, step size is 0.8, and cross-over probability is 0.5. The parameters of QPSO are set as follows: the alpha is linearly decreasing from 1.4 to 0.4, the population size is 40, the maximum number of iterations is 100.

Table 3.19 summarizes the results of using PSO, QPSO, GA, and DE hybridized with SG for finding efficient exact designs. It is clear that PSO finds designs with better D-criterion values than all other three algorithms. We reach the same conclusion that PSO is efficient and is good for practical use.

Table 3.19: A comparison of using different optimization algorithms for finding efficient designs of the 4-parameter Poisson NLMEM.

	Dose levels	Patient allocations	D-criterion (relative efficiency)
PSO	0	19	76.70 (100%)
	0.21	29	
	0.78	42	
QPSO	0	46	65.08 (84.9%)
	0.26	29	
	0.97	15	
GA	0.02	29	69.89 (91.1%)
	0.31	30	
	0.88	31	
DE	0.01	7	70.24 (91.6%)
	0.22	47	
	0.97	36	

3.6.4 Exponential models with real-world applications

In this section, we use a real-world application to show the usefulness of SGPSO in finding efficient designs for NLMEMs. In HIV trials, blood samples are drawn at a few time points. We focus on a NLMEM and call it simply an exponential model. It models the response curve as follows [WD99]

$$y_{ij} = log(V_{ij}) = log(e^{\theta_i - \delta_i t_j} + e^{\gamma_i - \lambda_i t_j}) + e_{ij}.$$

Let $\beta_i = (\theta_i, \gamma_i, \delta_i, \lambda_i)$ and assume we have a homogeneity error term $e_{ij}|\beta_i \sim N(0, \sigma^2)$. The outcome variable y_{ij} is the log-transformation of the total viral load measurement for the i-th subject at j-th time point, $i = 1, ..., n$ and $j = 1, ...m$. The vector of parameters (θ_i, γ_i) represents the initial viral production rate, and the parameters (δ_i, λ_i) represent the exponential decay rate of virus in the corresponding compartment. We let $\beta_i = \beta + b_i$ where the fixed effects are $\beta = (\theta, \gamma, \delta, \lambda)$ and we assume the random effects $b_i = (b_1, b_2, b_3, b_4)$ each has a normal distribution; $b_i \sim N(\mathbf{0}, D)$. We use nominal values estimated from viral load data in the AIDS clinical trial group (ACTG) 315 protocol data of 46 subjects and they are $\hat{\beta} = (12.142, 7.624, 0.442, 0.032)^T$, $\hat{\sigma} = 0.267$, and $\hat{D} = diag(1.397^2, 1.545^2, 0.137^2, 0.015^2)$ [WD99].

We seek efficient designs for different clinical trial protocols when we have three different schedules:

1) ACTG Protocol 5055 protocol (Schedule 1): this was a phase 1/2, randomized, open-label, 24-week comparative study of the pharmacokinetic (PK), tolerability, safety, and ARV effects of combined antiviral drugs in HIV-1-infected subjects failing their first protease inhibitor (PI)-containing regimen combination therapy [AWH04]. The design space, i.e. the time span allowed for drawing blood samples, is from day 0 to day 168 (24 weeks). The reference design implemented in practice took observations at 0, 7, 14, 28, 56, 84, 112, 140, 168 days after trial entry.

2) ACTG Protocol 315 (Schedule 2): HIV infected subjects were treated with potent antiviral drugs (ritonavir, 3TC, and AZT) and Plasma HIV-1 RNA was repeatedly quantified. The design space is from day 2 to day 84 as the data is only available after day 2 and viral load is only detectable before day 84. The reference design, which was implemented in practice, took observations at 2, 7, 12, 13, 21, 27, 55, 84 days after entry into the trial [WD99].

3) The standard sampling schedule (Schedule 3): this is a sampling schedule currently used in many AIDS clinical trials. The study design reduces the visits for individual subjects from 9 times in Schedule 1 to 6 visits. The reference design took observations during a 24 weeks period at 0, 28, 56, 84, 112, 168 days [HW08b].

We set the PSO parameters as follows: 100 maximum number of iterations, 40 particles, inertia linearly decreasing from 0.9 to 0.4 and $c_1 = c_2 = 2$. The SG uses 81 nodes in total to evaluate the FIMs. The efficient designs generated for the three protocols are summarized in table 3.20.

Table 3.20: SGPSO-generated designs for the exponential model for different protocols in the HIV studies, and their D-criterion values and CPU times.

Schedule	Reference Design	Sampling times	D-criterion	CPU time
			(RE)	
1	0, 7, 14, 28, 56, 84, 112, 140, 168	0, 8.6, 14.3, 18.9, 30.3, 157.9, 164.5, 167.8, 168	174.55 (276%)	1983
2	2, 7, 12, 13, 21, 27, 55, 84	2, 2.3, 15.8, 16.2, 34.1, 37.7, 65.1, 83.0	93.48 (138%)	1847
3	0, 28, 56, 84, 112, 168	0, 29.3, 29.4, 32.3, 168, 168	142.28 (242%)	1688

We observe from the table 3.20 that compared to the reference designs adopted in the protocols, the SGPSO algorithm generates more efficient designs within a reasonable amount of time. Specifically, the SGPSO generated designs have efficiencies ranging from 138% to 276%, suggesting that the designs implemented in practice may not be the most efficient ones. The SGPSO algorithm is very flexible and can accommodate different model specifications, including various design spaces, number of design points and various priors for the random effects. It is therefore a very useful tool and the SGPSO-generated designs can provide very good compelling alternatives for practical applications.

In summary, our proposed SGPSO algorithm is fast and efficient for finding pseudo-Bayesian optimal designs for GLMs in section 3.5, and efficient exact designs for non-linear mixed effects models in section 3.6. We showed that the algorithm works quite well for finding pseudo-Bayesian optimal approximate designs. It also seems to work well for finding more quality exact designs than the the reference designs, for complicated and moderately large dimensional models that were previously deemed intractable. SGPSO is able to fully exploit the design space and find designs more efficient than the implemented designs and in a relatively short time. It has great potential for further applications in practice without requiring a lot of mathematics. We conclude our Project I with a note that there is no absolute guarantee that our proposed algorithm works well for finding all types of optimal designs. In fact, it should be noted that there is no guarantee that our algorithm finds the optimal exact design because there is no theoretical ways to confirm optimality. We recall the so-called "no free lunch theorem" [WM97], which states that each random search technique has its own advantages on a certain set of problems over other methods and there is not a universally best algorithm for all problems. We also note that the sparse grid technique tends to work better with normal priors. We plan to apply the SGPSO algorithm to construct optimal designs for other design criteria for non-linear mixed effects models and other models in the future.

CHAPTER 4

Project II: single time point prediction of idiopathic pulmonary fibrosis progression using random forest hybridized QPSO

Idiopathic Pulmonary Fibrosis (IPF) is a chronic irreversible and ultimately fatal interstitial lung disease (ILD) with unknown etiology. It is characterized by an unpredictable progressive decline in lung function and typically affects people in the age group of 50-70 years. The disease exhibits a highly heterogeneous natural history, and the disease progression is unpredictable at the time of diagnosis: some subjects may experience episodes of acute respiratory worsening despite being previously stable [RCE11]. It is critically important to distinguish subgroups of IPF subjects who are likely to progress from those who are likely to remain stable. The identification of a candidates who is likely to progress or to be stable helps clinicians to make a decision of continuing or switching a treatment, or to refer for a lung transplantation at an early stage.

High-resolution Computed Tomography (HRCT) plays an important role in the diagnosis of IPF [RCE11]. Studies have shown that HRCT features are useful and sensitive in predicting progression in IPF subjects based on the patterns changes of usual interstitial pneumonia (UIP) between two scans [KSB01, Tib96]. Research shows that UIP patterns on HRCT are associated with high mortality and disease progression in subjects with IPF [Tib96]. Changes in quantitative HRCT measures provide a sensitive indication of disease progression and response to treatment [KTG16, KBW17, RSD16]. Features extracted from HRCT images are usually high-dimensional, which pose a challenge for image recognition systems because redundant or non-informative features sometimes reduce classification accuracy. For this reason we need a model using artificial intelligence, and feature selection is one of important steps for robust model to stratify the groups of subjects who are likely to progress or not.

Several methods have been proposed to select features and build classification models in the medical imaging field. Regularization methods, such as least absolute shrinkage and selection operator (LASSO) [FCT03] and smoothly clipped absolute deviation (SCAD) [FL01], are handy and popular methods in the field [MTS14, MRR13]. The field has increasingly used the advanced techniques [BBM15], such as random forest [CS13], support vector machine (SVM) [FCD00, MZA09, Mot15], neural network (NNET) [US18], etc. However, most of studies either has no feature selection step, or separate the feature selection from classification steps, which fail to select the optimized feature subset that leads to the best classification performance.

An appropriate approach to integrate feature selection and classification is a wrapper method, which directly uses the classification performance of a given model to assess selected subsets of features [XZB16]. Efficient search strategies are critical for a wrapper method to identify the best feature subset. Evolutionary computation (EC) is particularly useful in our case because the objective function in our problem does not have an analytical form, and the common optimization techniques which require leveraging mathematical properties of the objective function cannot be used. Evolutionary computation is a problem-solving paradigm that simulates the powerful process of natural evolution, such as reproduction, competition, mutation, and selection [BFM97]. Evolution is essentially an optimization process [May88], and through iterations the evolution can discover highly precise functional solutions to particular problems. EC has been used in a wide spectrum of areas, and [BFM97] broadly categorize the applications into areas including planning, design, simulation and identification, control, and classification. The obvious success of EC application on many problems makes EC attractive in practice, although the "no free lunch theorem" clearly states that there is no universally superior algorithm on all problems [WM97].

In this work, we propose to use a type of EC algorithm called quantum-inspired particle swarm optimization (QPSO), coupled with a random forest algorithm (RF) as a wrapper method to build a prediction model that has high accuracy and a good balance between sensitivity and specificity. Inspired by wave functions in quantum physics, QPSO has enhanced searching ability and improved optimization results over many other commonly used EC algorithms [SFX04, SLW11, LJS15, FDZ12, LMW18]; empirically, it is superior based on comparative experiments using benchmark test functions [XSX08]. It has been applied in the imaging field in recent years [LJS15, JJ15] and shows promising potential in dealing with high-dimensional imaging data.

To our best knowledge, this work is the first ROI-based computer-aided-diagnostic (CAD) model that can be applied to a baseline HRCT scan for predicting progression at 6 months to 1 year follow-up. The methodological contributions of this work include two aspects: (a) a study design of collecting a data set with ground truth for prediction via visual registration by a radiologist; and (b) the development of an objective metric and an algorithm that simultaneously achieves high prediction accuracy, balanced sensitivity and specificity with a parsimonious feature subset, using a relatively small number of subjects. Further, we apply the classifier trained at the ROI level to the voxels extracted from whole lungs. We hypothesize that a whole lung level metric derived from a baseline scan is associated with progression in a follow-up scan.

4.1 Background

Because of the heterogeneous natural history of IPF, a multidisciplinary team of pulmonologists, radiologists, and pathologists has devoted to build a guidance of diagnostic models of IPF for subjects with interstitial lung disease [RCE11]. According to the guidelines, a HRCT scan is required for diagnosis of IPF. Quantitative image analyses (QIA) using texture-based features from HRCT scans have been utilized intensively in pulmonary related diseases [UHS99, KGK05, KLG08]. For example, QIA are used for robust classification of interstitial lung disease patterns. Scores from QIA can be a good representation of IPF disease severity [KBC15]. The models that leverage HRCT quantitative imaging data usually require measurements of changes from baseline to follow up [KSB01]. However, not many subjects have a starndard clinical follow-up with utilization of HRCT scans unless they have experienced shortness of breath or suspicion of progression. Given that HRCT scans are not utilized for monitoring purposes but for confirmation of progression, and subjects with IPF with the longitudinal HRCTs tend to have short median survivals. Considering the clinical suspicions of longitudinal HRCT scans, it would be desirable to develop a prediction model for the IPF progression using only a baseline or single time point HRCT scan.

Texture features extracted from images can be considered as a mathematical characterization of images. They reflect granular spatial information quantitatively. They describe the grey levels of voxels along different orientations and spatial relationships of voxels within a local neighborhood. We extract features using a grid sampling procedure described in [KLG08]. Grids composed of 4-by-4 voxel squares were placed contiguously. From each grid, a voxel was selected. The grid sampling was used to assure complete coverage of regions of interest. We extract 71 imaging features from each sampled voxel. In one of classification modeling problems, a critical step is to carefully select a small number of features for prediction. This selected subset of features can substantially reduce the processing time, and give robust and superior results to using the full set of features [MHS95].

While deep learning has gained popularity in the medical imaging field recently, it suffers from overfitting and underfitting problems when the training samples are small [GLO16]. Deep learning framework is not practical to our design of data collection. The goal of this study is to use supervised voxel-wise level for predicting the progression. Deep learning approach can provide a subject-level of predicting progression or not, but not necessary at the voxel-level. Alternatively, a promising approach is a wrapper method with a machine learning algorithm in the task of texture feature selection and classification. In the following sections, we show that this approach is feasible and can provide a method of a supervised learning at voxel-level of predicting a likelihood of progression.

4.2 Study design and data acquisition

For the pilot study of prediction at the ROI level, a total of 50 anonymized HRCT images of subjects with IPF from a multi-center clinical trial were selected. At least two consecutive HRCT scans were available for each subject for the purpose of model building. We collected a data set with well-characterized HRCT scans from subjects of IPF and the dates of baseline scan ranged from 2011 to 2015. Anonymized HRCT images of these subjects were archived at the UCLA Computer Vision and Imaging Biomarkers Laboratory. The use of anonymous image data was approved by local institutional review board. The population had a mean age of 65.9 years and a mean forced vital capacity (FVC) percentage predicted value of 69.2%.

We collected a baseline scan and a 7.3 month (SE ± 1.6 months) follow-up HRCT scan from each subject with IPF. The first task is for a radiologist to visually mark ground truth of the regions of interest (ROI) at baseline scans to collect the textural information. Three steps of radiologist's assessment were: (1) a thoracic radiologist identified the ROI in chest CT as reflecting progression or non-progression by reviewing the two paired longitudinal HRCT scans; (2) contoured the ROI at baseline scan (before the changes occurred), the ROIs were contoured to avoid airways and blood vessels; and (3) labeled the ROI type as progression or non-progression. The visual registration was used to matches the baseline and follow-up in the anatomical correspondence of ROIs. This step was performed by a single radiologist (more than 20 years of experience).

Table 4.1 provides the baseline characteristics of the subject population by ROI status, namely, whether or not a subject had at least one progressed ROI (progression vs. no progression). Demographic profiles and the percent predicted FVC between the two groups are comparable (66.7% vs 68.1%). Quantitative lung fibrosis (QLF) is a classifier-model-derived score and is a measurement of disease extent [KBC15]. The means of QLF scores were also similar between two groups who had progressed ROI and those who had no progressed ROI (14.2% vs 13.4%). This suggests that the progression at the ROI level provides insights beyond disease extent and visual patterns of IPF, and is promising for detecting early signs of disease progression. Within each contoured ROI, a square of 4-by-4 grid sampling was implemented to generate voxel instances, and texture features were calculated based on these local neighborhoods of voxels. See [Har79, SHB14] for texture feature computation details.

Table 4.1: Baseline characteristics of the IPF cohort.

	Visual assessment	
	Subjects with $\geq$ 1 progressed ROI	Subjects with no progressed ROI
# Subjects	33	17
# ROIs	166	84
# Progression ROIs	72	0
# Non-progression ROIs	94	84
Age (mean ± SE), yrs	66.7 ± 7.7	68.1 ± 7.1
% Female	17	23
Baseline % predicted FVC (mean ± SE), %	68.7 ± 3.3	69.8 ± 2.8
Follow up % predicted FVC (mean ± SE), %	63.2 ± 3.0	67.8 ± 2.4
Baseline QLF (mean ± SE), %	14.2 ± 1.4	13.4 ± 1.4
Follow up QLF (mean ± SE), %	16.1 ± 1.8	15.3 ± 1.7

Importantly, the radiologist labeled a ROI on the baseline scan (before the changes occurred) as "expected to be progressed" if the ROI got worse in the follow-up scan, or "expected to be non-progression" if the ROI got better or stayed stable on the follow-up scan. Figure 4.1 depicts this data acquisition procedure. We say that our model is *predictive* because texture features were only obtained from baseline scans when we built the classification model. The model is evaluated and compared with the ground truth obtained by a thoracic radiologist assessment. The procedure of data acquisition enables a supervised learning approach since we collected HRCT images from subjects with follow-up scans. The ultimate application of this algorithm is to predict a subject's status for those who does not necessarily have follow-up scans.

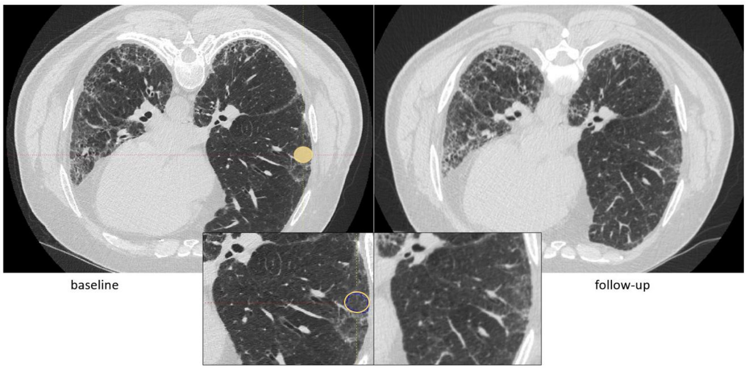


Figure 4.1: The baseline and follow-up HRCT scans of an IPF subject. A radiologist compared the two ROIs and determined the progression status based on the imaging pattern change.

4.3 Proposed methodology

There are two components in a wrapper method: an optimizer and a classifier. The classification metrics returned from the classifier serve as objective function(s) for optimization. We use a QPSO algorithm as the classifier; details of QPSO has been provided in section 2.2.1.We use a random forest as the optimizer.

4.3.1 The objective function

In this work, we want to build a prediction model that has a good balance between sensitivity and specificity. The previous IPF studies in machine learning have often focused on improving sensitivities, as it is clinically meaningful to detect progression, when there were no effective therapeutic treatments [MPB12, RDC17]. With effective anti-fibrotic IPF therapeutic treatments now being available [KBC14], there is an increasing need to understand the early signs of improvement or stabilization (not only progression), which led to the need of optimizing specificity. As such, we developed an objective metric to be maximized as the minimum of sensitivity and specificity. This metric is beneficial for a balanced classification. We compare the proposed algorithm to other wrappers and non-wrapper methods in terms of accuracy, sensitivity, specificity, and number of selected features.

Empirically, directly optimizing accuracy of classification would produce a skewed result with high specificity and low sensitivity, since non-progression is the majority class and specificity is the main driver of accuracy in an unbalanced data. To build a more balanced classifier, we need to consider a composite metric that includes both sensitivity and specificity. In computer science, a commonly used metric is F_1 score, which is a harmonic mean of precision and recall. One can think of F_1 score as a smoothed minimum of precision and recall [Lia05]. However, F_1 score only compromises mildly when we have unbalanced classification results.

In this work, we propose a new objective metric to maximize the minimum of the sensitivity and specificity, produced by the classification algorithm. To our best knowledge, there is only one similar metric in the 2012 PhysioNet/Computing in Cardiology Challenge [SMS12], which sought to maximize the minimum of positive predictive value (also known as precision) and sensitivity (also known as recall). Their choice stimulated competitors to optimize the Precision-Recall (PR) curve of classifiers, whereas our metric optimize the usual receiver operating curve (ROC). The classification results on our data set suggest that this metric is beneficial for a balanced sensitivity and specificity. Specifically, the objective is

$$\underset{\mathbf{F}_s}{\operatorname{argmax}}\ min(sensitivity,\ specificity)$$

where $\mathbf{F}_s$ is any feature subset, and both sensitivity and specificity are classification metrics on the test set returned by the classification algorithm built upon $\mathbf{F}_s$.

The solution to this optimization problem does not have an analytical form, which means that it is hard to derive mathematical properties, such as convexity or differentiability of the objective function. Consequently, we resort to heuristic algorithms such as QPSO, and show that it can be particularly useful in this case.

4.3.2 The classifier: random forest

We build the classification model based on selected features using the random forest algorithm [Bre01]. Research has shown that tree-ensemble-based machine learning techniques such as random forest are effective for solving real world problems [DBP12, LWV14, ZSW17]. Random forest has also been widely used in the medical imaging field and is highly successful in many classification tasks [CS13, HBL18]. For high-dimensional data, random forest is easily applicable in distributed systems. A random forest classifier is particularly suitable for our problem because: 1) it is computationally fast and reasonably easy to train for high-dimensional data; 2) it minimizes storage, and this is especially important in our future work when we want to scale our algorithm to expand to whole lung level as is discussed in section 5; and 3) it tends to provide feature diversity, which is particularly important when dealing with heterogeneous data [Die00].

The parameters we used in the random forest classifier were set as follows: the maximum depth of each tree is 20, the number of trees is 100, the proportions of subsample from the data set is 0.5, and the proportion of features randomly selected at each node of each tree is 0.5. The guidelines for selecting parameters such as tree depth, forest size and sampling density remain an open question [CS13]. Since we have 71 features in total, a depth of 20 uses up to 30% features in individual trees and is sufficient in our case. We also observe that when the depth further grows, RF tends to over-fit, compromising the classification results; the memory required also increases exponentially as the trees grow deeper. On the other hand, a shallow tree will produce low-confidence, high entropy posteriors. The choice of the number of trees considered the computational budget of running the experiments and empirical experiences of many other studies [KKK13, Zha15, YP13]. In section 4.5.1, a 5-fold cross validation is used to select the best tree depth as 33; this depth has been used in the whole lung expansion study.

The pseudo code is as below:

After the maximum number of trees to grow, we obtain the final decision tree **H**, which

```
Algorithm 2 Random Forest Algorithm
Input: a data set $\mathbf{S}$, a feature set $\mathbf{F}$, and a selected number of trees in forest $\mathbf{B}$.
Output: the final decision tree $\mathbf{H}$, which can then be used to classify the test set.
  Initialization : Start with an empty forest $\mathbf{H}$.
  for $i \leq \mathbf{B}$: do
      Bootstrap a subsample $S^{(i)}$ from $\mathbf{S}$.
      At each node, select a small subset of $\mathbf{F}$ split on the best feature in this feature subset.
      Save the tree $h_i$ learned from this feature subset.
      Add $h_i$ to $\mathbf{H}$.
  end for
  return $\mathbf{H}$
```

can then be used to classify the test set.

4.3.3 The re-sampler: SMOTE

In our data set, we observe pattern imbalance, which is a common problem in machine learning. Class imbalance can have significant impact on classification performance, especially for minority classes. Re-sampling is a routinely used pre-processing technique in statistics [FHT01] to obtain balanced pattern frequencies and thus promising to achieve more balanced classification. Various re-sampling techniques have been developed in pulmonary imaging studies. Two general re-sampling approaches are up-sampling and down-sampling. Up-sampling is any technique that simulates or imputes additional data points to improve balance across classes, and down-sampling refers to any technique that reduces the number of samples to improve the balance across classes. Research has shown that a combination of over-sampling the minority class and under-sampling the majority class can achieve better classification performance than using either alone, so we apply a synthetic minority over-sampling technique (SMOTE) [CBH02] to carry out the re-sampling task. In SMOTE, the over-sampling approach for minority class is to over-sample by creating "synthetic" examples rather than by over-sampling with replacement. The majority class is under-sampled by randomly removing samples from the majority class population until the minority class becomes some specified percentage of the majority class. The procedure follows as:

In this work, we apply SMOTE technique on the training set and let the test set untouched

Algorithm 3 SMOTE Algorithm

```
Input: a data set $\mathbf{S}$ and a feature set $\mathbf{F}$.
Output: a resampled data set $\mathbf{S}_1$.
    Create Synthetic minority samples:
    Take the difference between a feature vector and one of its $k$ nearest neighbors ($k$ is set to 5);
    Multiply this difference by a random uniform[0,1] scalar;
    Add the difference to the feature value of the original feature vector.
    Under-sampling the majority samples:
    Randomly remove until we have a class balance.
    return The resampled data set.
```

to be more consistent with the nature of the data. We compare approaches with and without SMOTE to illustrate its usefulness.

4.3.4 Random forest hybridized QPSO algorithm

We propose QPSO-RF as an integrated novel algorithm that performs HRCT feature selection and imaging pattern prediction effectively in this section. First in the resampling step, we use SMOTE to resample the training set. We then use QPSO as the optimizer to search the feature subsets (Step: 1-6), and random forests are built upon selected subsets (Step: 7-12) which produces the evaluation metrics. The iterative process of QPSO-RF searches the feature space in all the particle best (pbest) and returns the global best (gbest) at the last iteration as the best feature subset that gives the maximized objective function.

The parameters in our QPSO algorithm are the same as those recommended in [ZSX13]: the number of particles is 40, the dimension of each particle is 71 (as we have 71 features), the maximum number of iterations is 1000 and each iteration involves 5-dimensional mutation at any one time (See the steps 1-5 in Algorithm 4). The choices of the number of particles and the maximum number of iterations follow the convention, which were used and suggested in many PSO literatures for their good performance on the benchmark functions [Sug99, RC11]. A binary encoded QPSO determines the probability of flips using a function in step 6 in Algorithm 4, which is inspired by a wave function in quantum mechanics [SFX04]. We note that the solution may not be optimal or unique, as in practice we use only a limited number of iterations, and there might be several feature subsets that produce the same or similar fitness values, which is a typical problem in feature selection.

Figure 4.2 is the procedure flowchart of the random forest hybridized QPSO algorithm, which describes the steps of the algorithm. QPSO-RF uses the QPSO to select features from a resampled data set to build a random forest classifier, and uses the objective function to guide the algorithm to find a best feature subset.

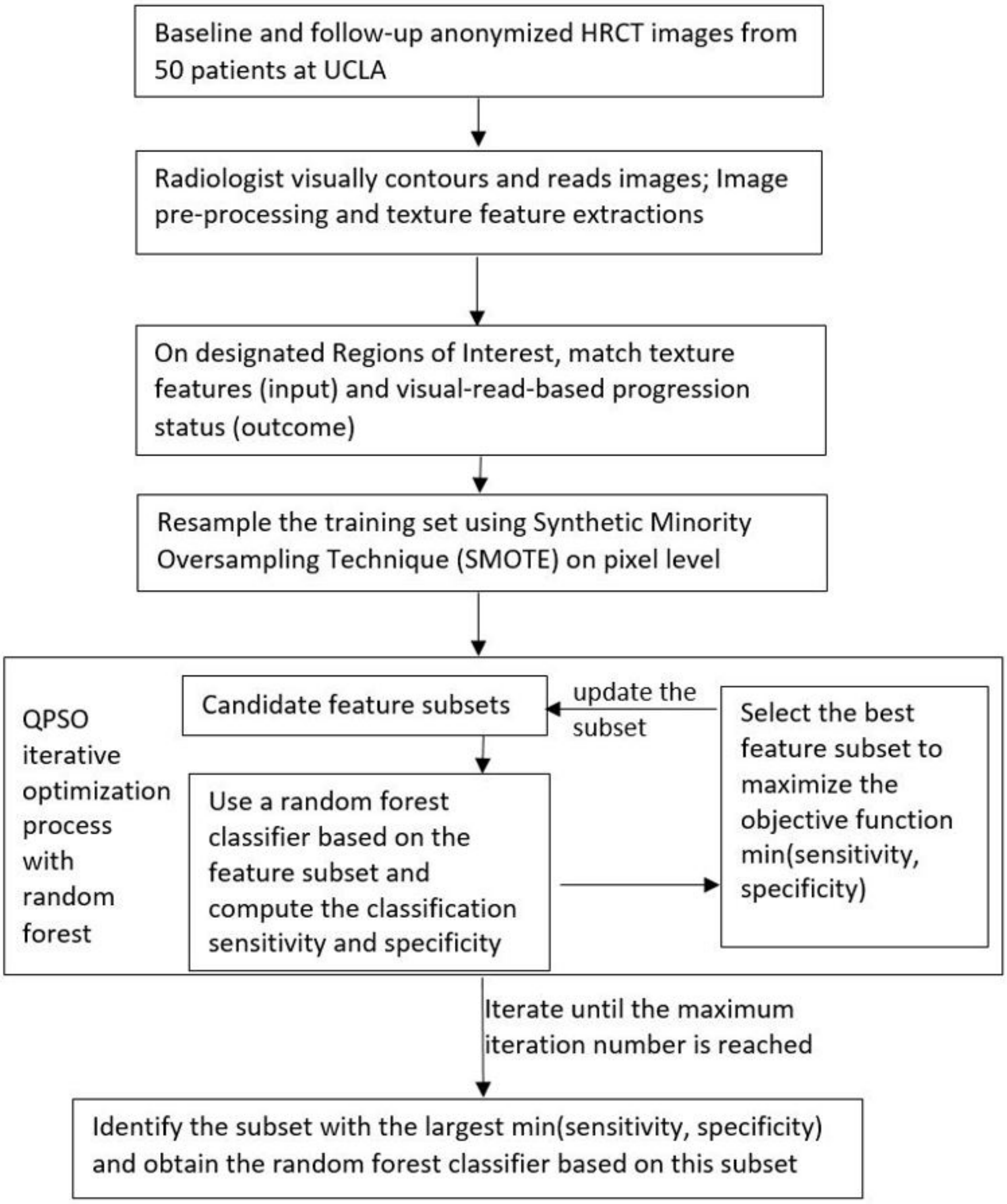

Baseline and follow-up anonymized HRCT images from 50 patients at UCLA
Radiologist visually contours and reads images; Image pre-processing and texture feature extractions
On designated Regions of Interest, match texture features (input) and visual-read-based progression status (outcome)
Resample the training set using Synthetic Minority Oversampling Technique (SMOTE) on pixel level
QPSO iterative optimization process with random forest
Candidate feature subsets
update the subset
Select the best feature subset to maximize the objective function min(sensitivity, specificity)
Use a random forest classifier based on the feature subset and compute the classification sensitivity and specificity
Iterate until the maximum iteration number is reached
Identify the subset with the largest min(sensitivity, specificity) and obtain the random forest classifier based on this subset

Figure 4.2: Flowchart of random forest hybridized QPSO algorithm for IPF prediction

Algorithm 4 QPSO-RF Algorithm

```
Input: a data set $\mathbf{S}$, a feature set $\mathbf{F}$, pre-fixed parameters including number of particles $p$,
    dimension of the particles $d$, maximum iteration number $max_{iter}$, and a decision tree $\mathbf{H}$
    as a learning algorithm. Set QPSO parameters as: $max_{iter} = 1000$, $p = 40$, and $d = 71$.
    Additionally, set RF parameters as: the number of trees $\mathbf{B} = 100$, the maximum depth
    of each tree = 20 and the proportions of subsample from the data set = 0.5.
Output: a feature subset $\mathbf{F}_s$ that gives the best fitness value.
    Resampling: use SMOTE algorithm to resample the training set.
    Initialization of QPSO: randomize binary bits for particles, particles' personal best pbest
    and swarm's global best gbest. Twelve steps are the procedure of QPSO-RF, where RF
    is nested into each iteration of QPSO.
for $iter \leq max_{iter}$: do
    for $i \leq p$: do
        Perform mutation operation on each particle.
        Determine the mean best (mbest) for each particle at dimension $d$:

            $m_d^{best} = \begin{cases} 1, & \frac{1}{p}\sum_i^p x_{i,d}^{pbest} > 0.5 \\ 0, & \frac{1}{p}\sum_i^p x_{i,d}^{pbest} < 0.5 \\ 1\ with\ prob = \frac{1}{2}, & \frac{1}{p}\sum_i^p x_{i,d}^{pbest} = 0.5 \end{cases}$

        Select one of the offspring randomly.
        For each dimension, flip the particle with mutation probability defined as

            $min(\frac{\beta \times d_H(x_i, x^{mbest}) \times ln\frac{1}{u}}{p}, 1)$

        where $d_H$ is the counts of bits different in the two strings, $u \sim$ unif(0,1) and $\beta$ is a
        tuning parameter set to decrease from 1.4 to 0.4 over the iterations. The decreasing
        values of $\beta$ allow more localized search as the swarm evolves.
        On the resampled training set, evaluate each particle using random forest:
        Start with an empty forest $\mathbf{H}$.
        for $i \leq \mathbf{B}$: do
            Bootstrap a subsample from $\mathbf{S}$.
            At each node, split on the best feature represented by the particle. Save the tree
            $h_i$ learned from this feature subset.
            Add $h_i$ to $\mathbf{H}$.
        end for
        Obtain minimum of sensitivity and specificity from H. Each particle is evaluated
        using this metric; update pbest as the best feature subset one particle visited so far
        and gbest as the best feature subset of the swarm visited so far.
    end for
end for
return gbest vector that represent the best feature subset, which has maximum
    min(sensitivity, specificity).
```

The implementation of the algorithm is primarily done in R. The procedure for doing the prediction and result visualization are fully automated and has been tested locally on a Windows platform. The SMOTE re-sampling is carried out by “unbalanced” package. The realization of the random forest classifier is by “XGBoost” package. The use of XGBoost allows potential scalable future applications and convenience when merging the algorithm into the CVIB pipeline. Some pre-processing and simple descriptive statistical analysis are done in Stata.

4.3.5 Statistical analysis plan

We implemented statistical procedures to test if the differences of prediction accuracy between the proposed QPSO-RF and other comparator methods were statistically significant. We used conditional logistic regression models for analysis, which is a standard method for analyzing matched case-control data [HLS13]. The outcome is whether the ROI prediction is accurate (binary), and the covariate is the indicator of method. Since each subject may contribute multiple ROIs, we used a clustered sandwich estimator as the variance structure. The clustered sandwich estimator allows for intragroup correlation within ROIs that were from the same subjects, and maintains the independence across ROIs from different subjects [BH95]. The p-values for each model’s coefficient are reported in table 4.2 and 4.3.

We also addressed the multiple comparison using a Benjamini–Hochberg procedure controlling the false discover rate (FDR) at 0.05 significance level [BH95]. In total, we have 23 pairwise comparisons of the proposed algorithm to comparators trained on resampled or original training data set.

4.3.6 R programs

We use R to code the QPSO-RF method. The packages used include dplyr, xgboost, caret, sqldf, and e1071. We coded binary QPSO algorithm manually.

The texture features were generated from the in-house pipeline, and we omit the details here. The denoised texture features used for whole lung expansion is also generated from the pipeline.

There are several parameters that users can adjust. The most important parameter that affects the computational budget is the maximum number of iterations of QPSO. The hyper-parameters of random forest can be selected using cross-validation.

During the model training process, we input the texture feature data of the training set, the QPSO parameters, and an objective function to select the feature subset. The output is the feature subset selected as the group best at the last iteration and the random forest classifier built on the selected feature subset. After the model training, we apply the classifier trained to the texture feature data of the test set, at the ROI or whole lung level, and the output is the prediction results of each input voxel.

4.4 Preliminary results at regions of interest level

We split the data into a training set and a test set at the subject level. The training set had 26 subjects, and the test set had a different group of 24 subjects. The training set had 77 non-progression and 50 progression ROIs, adding up to 127 ROIs; at the voxel level the training set had 1172 non-progression and 582 progression voxels, adding up to 1754 voxels. The test set had 101 non-progression and 22 progression ROIs, adding up to 123 ROIs; at the voxel level the test set had 1605 non-progression and 336 progression voxels, adding up to 1941 voxels. In our data set, the progression and non-progression lung morphology outcomes had an unbalanced ratio. At the voxel level the ratio was 1:3 (918 progression: 2777 non-progression instances) and at the ROI level the ratio was 1:2.5 (72 progression vs 178 non-progression). For the training set we applied the re-sampling SMOTE technique as a class imbalance remedy; the post-SMOTE training set had 1746 progression and 1746 non-progression voxel instances and the model was trained on this re-sampled set. The test set was evaluated without any resampling modifications. Texture features were calculated at the voxel level and the radiologist references were ROI-based, we therefore performed a majority voting strategy to transform the classifier outcome from voxel to ROI level.

We compared the performance of our algorithm with other feature selection and classification techniques. Firstly we considered a range of wrapper methods that use different optimizers and classifiers. The classifiers used included support vector machine (SVM) and shallow neural network (NNET); classifiers using all features (without feature selection) were also included in the comparison. The optimizers used for comparison included the basic version PSO and the Genetic Algorithm (GA). The GA employs the "survival of the fittest" principle in its search process to select and generate individuals (in our case, feature subsets) that are adapted to their environment (objectives). The desirable traits tend to over-express over a number of generations (iterations), leading to better and better solutions. GA has also been frequently used in feature selection problems [YH98]. Research has shown that PSO has the same effectiveness as a Genetic Algorithm (GA) with significantly improved computational efficiency [HCD05].

We also compared the method to non-wrapper, model-based methods including LASSO and SCAD. LASSO and SCAD were used to perform feature selection by employing penalties. They have been widely adopted and can effectively enhance the prediction performance of the regression model [MTS14, MRR13].

The comparators used in this work are commonly used methods in the medical imaging field [BBM15] and have comparable computational efforts. The configurations of each comparator algorithm were set as follows. For LASSO and SCAD, we standardized the feature set and then used cross validation to choose the best penalty parameters for prediction. We then applied the cross-validated LASSO or SCAD model to the test set. For GA, we used 1000 iterations, 40 populations, cross over rate of 0.8 and mutation rate of 0.02, which was comparable to QPSO and PSO parameter settings. For SVM, we used a Gaussian radial basis function (RBF) kernel with a scaling factor 1 [MZA09, Mot15]. For NNET, we used a two-layer feedforward network with 10 hidden-layer neurons, a sigmoid transfer function in the hidden layer, and a softmax transfer function in the output layer [US18]. These parameters for the comparator optimizers and classifiers were pre-specified based on the computational feasibility consideration and best practices reported in the literature.

Despite our best intent to compare the proposed methods to other methods used in feature selection and classification tasks, it was hard to do an exhaustive comparison. Potentially, there could be other methods that are even more effective in our data set.

The QPSO-RF algorithm yielded a model with 19 texture features and achieved 81.8% sensitivity, 82.2% specificity and 82.1% accuracy at the ROI level on the test set in predicting progression at 6 months to 1 year follow-ups. Table 4.2 compares the results from different methods trained on resampled training sets and then applied on the test set. It is clear that QPSO-RF is superior compared to other wrapper methods and model-based methods, in a sense that QPSO-RF selects a smaller set of texture features than most of the other algorithms, has a higher and more balanced classification sensitivity and specificity, and has higher accuracy. Compared to LASSO, SCAD, other wrapper methods, and classification models without feature selection, QPSO-RF provides the only solution that both sensitivity and specificity achieves above 80%. It also achieves the highest classification accuracy with the smallest feature set. Using conditional logistic regressions and controlling the overall FDR at the significance 5% level, we observe that QPSO-RF accuracy is statistically significantly higher than all other methods except for the SVM without feature selection; the latter method is not ideal because it classifies all ROIs as non-progression. The QPSO-RF selected features include 2 summary statistical features, mean and mean32 (gray-level mean of 32 brightest voxels), and 17 gray-level co-occurrence matrices (GLCM) features. These features are important to understand the characteristics of the images and are good representations of the images.

In addition, a re-sampling technique is helpful to achieve better classification results. Table 4.3 compares results from same comparators trained without resampling. There is no method that achieves above 80% for both sensitivity and specificity. The resampled QPSO-RF achieves statistically significantly higher classification accuracy than six other methods in table 4.3. Generally, wrappers trained without resampling produce higher specificity, but much lower sensitivity and reduced overall accuracy compared with the wrappers trained with resampling. This is because that the data without resampling has under-representative progression class, and as a result, it's hard for the models to pick up the minority class. In particular, if QPSO-RF was applied without re-sampling, 30 features were selected and the sensitivity dropped to 72.7%, specificity dropped to 76.2%, with overall accuracy reduced to

75.6%. We note that compared to other algorithms in table 4.3, QPSO-RF without SMOTE still achieves one of the highest accuracy levels, balanced sensitivity and specificity, and select one of the smallest feature sets.

Empirically, random forest was to be the superior classifier in our data set. On the resampled data set, RF (without feature selection) and PSO-RF had the highest accuracy among all methods following QPSO-RF; on the data set without resampling, RF (without feature selection) had the highest classification accuracy, and QPSO-RF and PSO-RF also had higher accuracy than most other methods. These results suggest that RF is a superior classifier for this data set. Further, QPSO method was found to be the best optimizer for our data set, and it selected a parsimonious feature subset compared to PSO and GA optimizers. As a result, the resampled QPSO-RF was found to be clearly a superior method that achieved all of our goals of high accuracy, balanced sensitivity and specificity, with a smaller number of selected features.

Table 4.2: Classification results at the ROI level from different algorithms applied on the test set using SMOTE in the training set. Asterisk mark indicates significance after controlling the overall false discovery rate at 0.05 significance level.

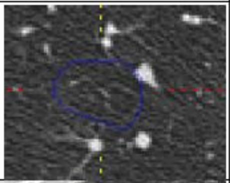
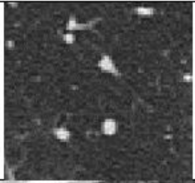
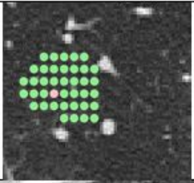
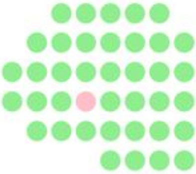
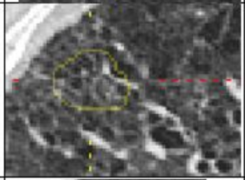
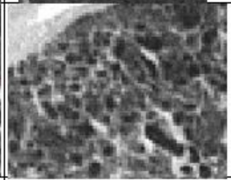
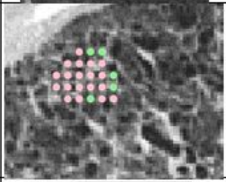
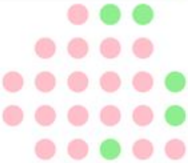
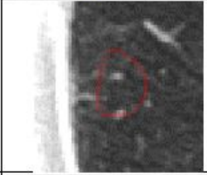
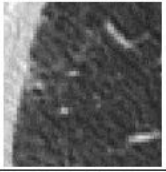
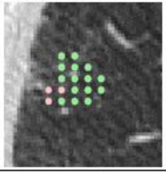
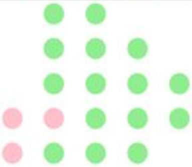
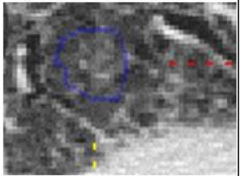
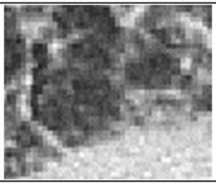
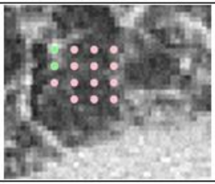
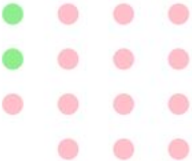
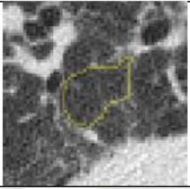
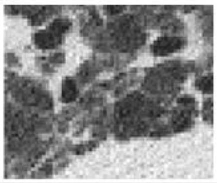
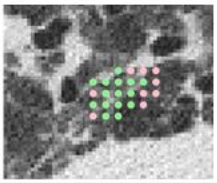
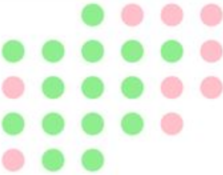
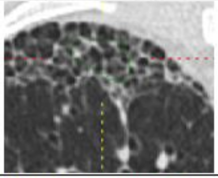
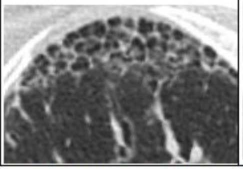
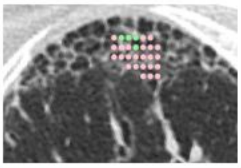
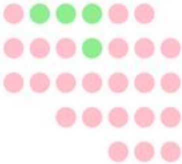
	Sensitivity	Specificity	Accuracy	P-value	#features
QPSO-RF	0.818	0.822	0.821		19
LASSO	0.864	0.554	0.61	<0.001 *	56
SCAD	0.818	0.564	0.61	<0.001 *	47
RF w/o feature selection	0.727	0.723	0.724	0.022 *	71
SVM w/o feature selection	0	1	0.82	1	71
NNET w/o feature selection	0.727	0.574	0.602	0.001 *	71
PSO-RF	0.864	0.663	0.699	0.004 *	29
PSO-SVM	0.864	0.653	0.691	0.017 *	39
PSO-NNET	0.818	0.594	0.634	0.001 *	39
GA-RF	0.909	0.584	0.642	0.001 *	38
GA-SVM	0.864	0.663	0.699	0.019 *	36
GA-NNET	0.909	0.604	0.659	0.001 *	37

Table 4.3: Classification results at the ROI level from different algorithms applied on the test set without SMOTE in the training set. Asterisk mark indicates significance after controlling the overall false discovery rate at the 0.05 significance level.

	Sensitivity	Specificity	Accuracy	P-value	# Features
QPSO-RF	0.727	0.762	0.756	0.159	30
LASSO	0.636	0.713	0.711	0.024 *	54
SCAD	0.682	0.772	0.756	0.108	10
RF w/o feature selection	0.682	0.802	0.78	0.367	71
SVM w/o feature selection	0.909	0.644	0.691	0.012 *	71
NNET w/o feature selection	0.727	0.733	0.732	0.016 *	71
PSO-RF	0.818	0.743	0.756	0.089	48
PSO-SVM	0.909	0.644	0.691	0.017 *	20
PSO-NNET	0.682	0.792	0.772	0.241	39
GA-RF	0.682	0.762	0.748	0.097	38
GA-SVM	0.864	0.693	0.724	0.032 *	31
GA-NNET	0.636	0.733	0.727	0.012 *	29

As we have coordinates of each voxel, we can visualize the voxel level classifications within each ROIs. Table 4.4 is a summary table that gives readers intuition about the algorithm classification. Four typical cases are provided in the table. Our algorithm correctly classified ROI patterns in case 1 and case 2 but fails in case 3 and 4. Case 1 is a non-progression ROI, and PSO-RF classified 97.4% voxels as non-progression, which transformed to a “non-progression” label to the ROI. Case 2 is a progression ROI with 83.3% voxels classified as progression, which transformed to a “progression” label to the ROI. Case 3 is a progression ROI with 62.5% voxels classified as non-progression, which transformed to a “non-progression” label to the ROI. Case 4 is a non-progression ROI with 85.7% voxels classified as progression, which transformed to a “progression” label to the ROI.

Table 4.4: Classification visualization, green dots=voxels QPSORF classified as non-progression, red dots=voxels PSORF classified as progression.

Case	Duration	Classfication type	ROI on CT-baseline	ROI on CT-followup	ROI on CT-followup w/ prediction	classification at voxel level
1	7.0 months	True negative: non-progression ROI classified as non-progression				
2	6.7 months	True positive: progression ROI classified as progression				
3	6.6 months	True negative: non-progression ROI classified as non-progression				
4	6.7 months	True positive: progression ROI classified as progression				
5	7.1 months	False negative: progression ROI classified as non-progression				
6	5.1 months	False positive: non-progression ROI classified as progression				
Legend 🔴 =voxel algorithm classified as progression; 🟢 =voxel algorithm classified as non-progression.						

We observe that the ROI-level classification relies on voxel numbers within ROI, and this may affect the classification outcome: if not a significant proportion of voxels are classified as "progression" then the progression ROI might be easily mis-classified, which makes the error tolerance rate low for ROIs with few voxels. Furthermore, as images were obtained from multi-center trials, the image quality may be inconsistent throughout ROIs and in particular, image brightness may affect the classification performance. To address this issue, we apply the classifier trained at the ROI level to the voxels extracted from whole lungs. The prediction in whole lung eliminates the bias of ROI contouring and makes the prediction robust. The expansion could be automated in the prediction process with little radiologist input. Research has shown that quantitative scores that are summated for the whole lung correlates with PFT results, and is related to extent and severity of interstitial lung disease [TVT16]. We hypothesize that a whole lung level metric derived from a baseline scan is associated with progression in a follow-up scan.

Further, we validated the algorithm with a larger data set of 172 IPF subjects. We collect anonymized longitudinal serial volumetric HRCT scans from IPF. Radiologists visually contoured regions of interest (ROI) and annotated lung morphology types into progression or non-progression, at the previous visits before the changes occurred. 191 texture features were extracted from the grid sampled voxels of baseline ROIs. Using the QPSO hybridized random forest algorithm, we calibrated the algorithm on a data set of 99 subjects (577 ROIs) using 5-fold cross validation and tested the algorithm on a separate test set of 73 subjects (414 ROIs). The algorithm yields a parsimonious model with 23 features (12% features selected) and achieves 70.8% sensitivity, 70.1% specificity and 70.4% accuracy at the ROI level on the cross-validation set, and 68.2% sensitivity, 65.4% specificity and 66.6% accuracy on the independent test set. Compared to other state-of-the-art algorithms, our approach selects a smaller feature subset, has higher prediction accuracy and achieves more balanced classification. As a conclusion, we validated the proposed QPSO hybridized random forest algorithm with a much larger set and showed that the algorithm stably achieves superior prediction performance.

In summary, our proposed algorithm works well for our data set in that it outperforms other methods by giving higher accuracy, higher and more balanced sensitivity and specificity, with a smaller number of selected features. The ROI-based prediction provides insights beyond disease extent and visual parenchymal patterns of IPF.

4.5 Prediction at the whole lung level

We consider an expansion of the current algorithm from the ROI level to the whole lung level, which is a typical practice for interstitial lung disease. The prediction in whole lung eliminates the bias of ROI contouring and makes the prediction robust. The expansion could be automated in the prediction process within a boundary of a segmented lung. Research has shown that quantitative scores that are summated for the whole lung correlates with PFT results, and is related to extent and severity of interstitial lung disease [TVT16].

To ensure that we extract robust texture features from images with different acquisition protocols, we apply a denoising technique [KLG08] on baseline HRCT images before extracting the texture features. Further, as a quality control, we removed subjects who 1) have post-contrast images in their baseline or follow-up visits; 2) have incomplete lung segmentation; or 3) have their two visits fall outside 5 - 13 months window. We extract features using a grid sampling procedure composed of 4mm-by-4mm voxel squares to assure complete coverage. From each grid, one voxel was selected. We computed 52 denoised texture features for each selected grid voxel.

We retrospectively collected anonymized longitudinal HRCT images of 193 IPF subjects from multiple clinical trials. Such images were collected for IPF diagnosis purposes. ROIs of 71 subjects were annotated and used to build the classifier, and 122 subjects were used to test the baseline metric generated by the classifier at the whole lung level. In total, the 71 subjects contributed 434 annotated ROIs, including 193 (44.5%) progression ROIs and 241 (55.5%) non-progression ROIs. Each subject contributed 3-16 ROIs. Among the 434 ROIs, 149 (34%) are from the upper lung, 185 (43%) are from the middle lung, and 100 (23%) are from the lower lung. 423 ROIs (97%) lie wholly or partially on the peripheral of the lung, which is consistent with the nature of the disease. This cohort has a mean age of 70.0 (SD ±7.5), 73% male/27% female, with the percent predictive FVC of 67.8% (SD ±12.3%). The baseline quantitative lung fibrosis (QLF) score is 15.4% (SD ±8.7). QLF is a classifier-model-derived score and is a measure of IPF disease extent [KBC15].

Using a prediction model from the feature selection and classification in the training set, we generate a summary metric called Single-scan Total Probability (STP) for each subject. In detail, there are 4 steps of obtaining a STP metric: (1) extract texture features selected by QPSO from denoised HRCT images; (2) run random forest classifier, which was developed using the radiologist's assessment as the ground truth; (3) record the number of progression voxels predicted by the classifier and total number of voxels per each slide; (4) sum the number of progression voxels and total number of voxels across slides and calculate the metric using the former divided by the latter. STP is a metric that only uses the information from baseline imaging (i.e. a single time point), and we showed that STP is statistically significantly associated with the 6-12 months follow-up changes of quantitative lung fibrosis (QLF) score. Figure 4.3 is flow chart of STP generation.

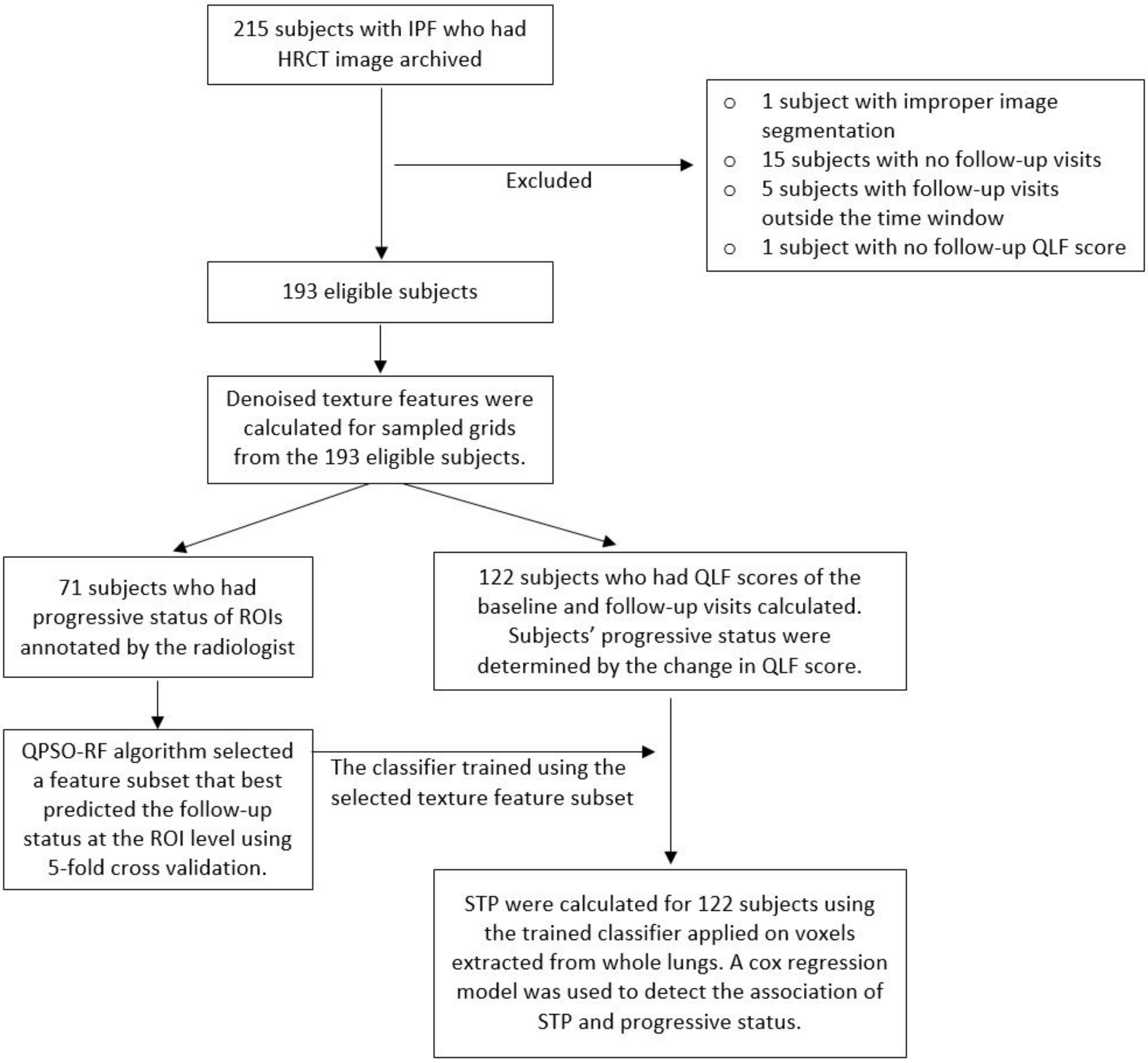

215 subjects with IPF who had HRCT image archived
Excluded
○ 1 subject with improper image segmentation
○ 15 subjects with no follow-up visits
○ 5 subjects with follow-up visits outside the time window
○ 1 subject with no follow-up QLF score
193 eligible subjects
Denoised texture features were calculated for sampled grids from the 193 eligible subjects.
71 subjects who had progressive status of ROIs annotated by the radiologist
122 subjects who had QLF scores of the baseline and follow-up visits calculated. Subjects' progressive status were determined by the change in QLF score.
QPSO-RF algorithm selected a feature subset that best predicted the follow-up status at the ROI level using 5-fold cross validation.
The classifier trained using the selected texture feature subset
STP were calculated for 122 subjects using the trained classifier applied on voxels extracted from whole lungs. A cox regression model was used to detect the association of STP and progressive status.

Figure 4.3: The flowchart of the whole lung expansion of the QPSO-RF algorithm.

There are different variations that we could add to the selected feature subset or to the classifier. Below we discuss three ways of generating such a summation metric: (1) using texture features alone, (2) adding axial distribution information, and (3) adding craniocaudal distribution information.

4.5.1 Robust classifier using textural information

As a first step, we consider building a classifier using only textural information. ROIs from 80% subjects were used as cross-validation training set to choose hyper-parameters for the random forest under each selected feature subset, and the random forest with the selected best parameter was then tested on the data for the rest 20% subjects to evaluate how well the selected feature subset is. The global best solution at the last iteration of QPSO was chosen as the best feature subset. QPSO-RF algorithm were run for 100 iterations, and at the last step, 27 features were selected for the best classifier. The parameters of random forest selected in the cross validation process were: the sub-sample rate of the data was 50%, the sub-sample rate of the variables was 100%, and the maximum tree depth was 33. The voxel level sensitivity is 82.1%, specificity is 65.1% and overall accuracy is 73.1%. After locking the model, we applied the classifier trained at the ROI level to the voxels extracted from whole lungs.

Below we examine two sample subjects' visualizations in detail. The first subject has an increased QLF of 23.5% from baseline to a 7-month follow-up, and has experienced more than 10% drop of FVC percent predicted value. Figure b in figure 4.4 is a visualization of this subject's upper lung change, and figure c in figure 4.4 is a level plot visualization of the same case on the same slide. We can generate such a level plot because we used a logistic link in our random forest classifier and has raw values for the predicted probabilities. We observe from figure 4.4 that peripheral voxels tend to be more progressive, which is consistent with the nature of the disease. We observe from figure b that our prediction model is able to pick up many progressive voxels in its exact locations.

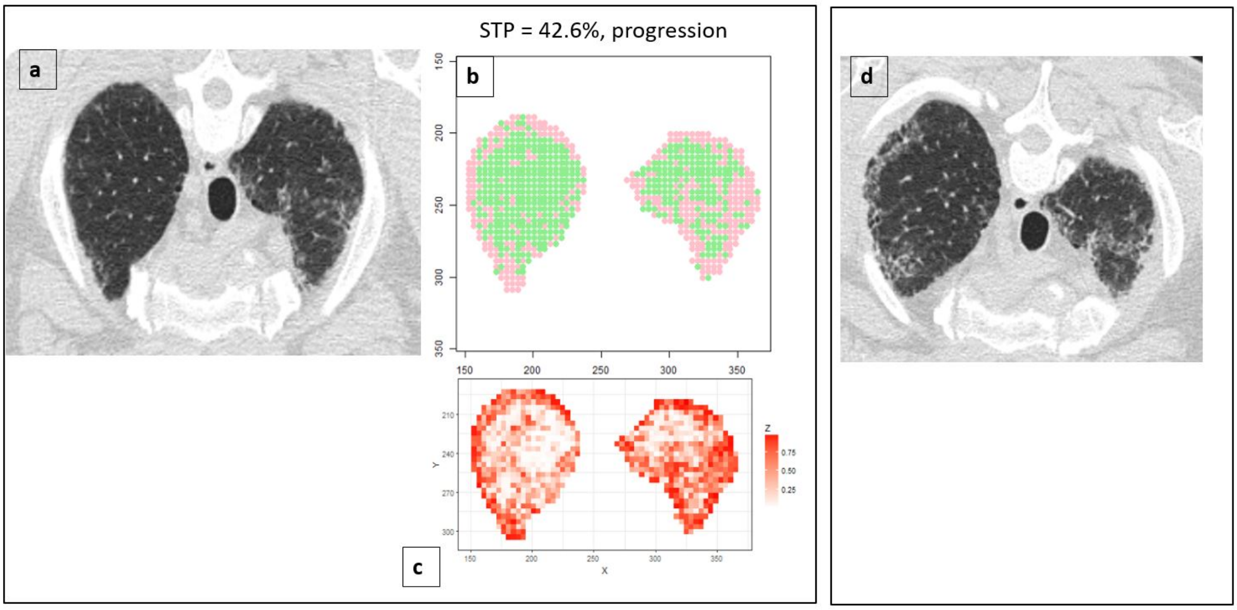


Figure 4.4: Generation of baseline Single-scan Total Probability (STP) for a sample subject. The subject has a QLF of 22% at baseline and 45.5% in a 7-month followup. Figure a. Baseline HRCT; b. Dichotomized classification of STP of figure a (green dots = stable voxels, red dots = progressive voxels, with 0.5 probability cutoff), the STP is 42.6% on this slide; c. STP result of figure a; d. 7-month follow-up HRCT showing progression.

Below is a sample slide showing our classifier is also able to predict the signs of stabilization. This subject has no QLF change in the follow-up scan, and has not experienced over 10% drop in FVC. Figure 4.5 shows that the majority of voxels were able to pick up the non-progression signals.

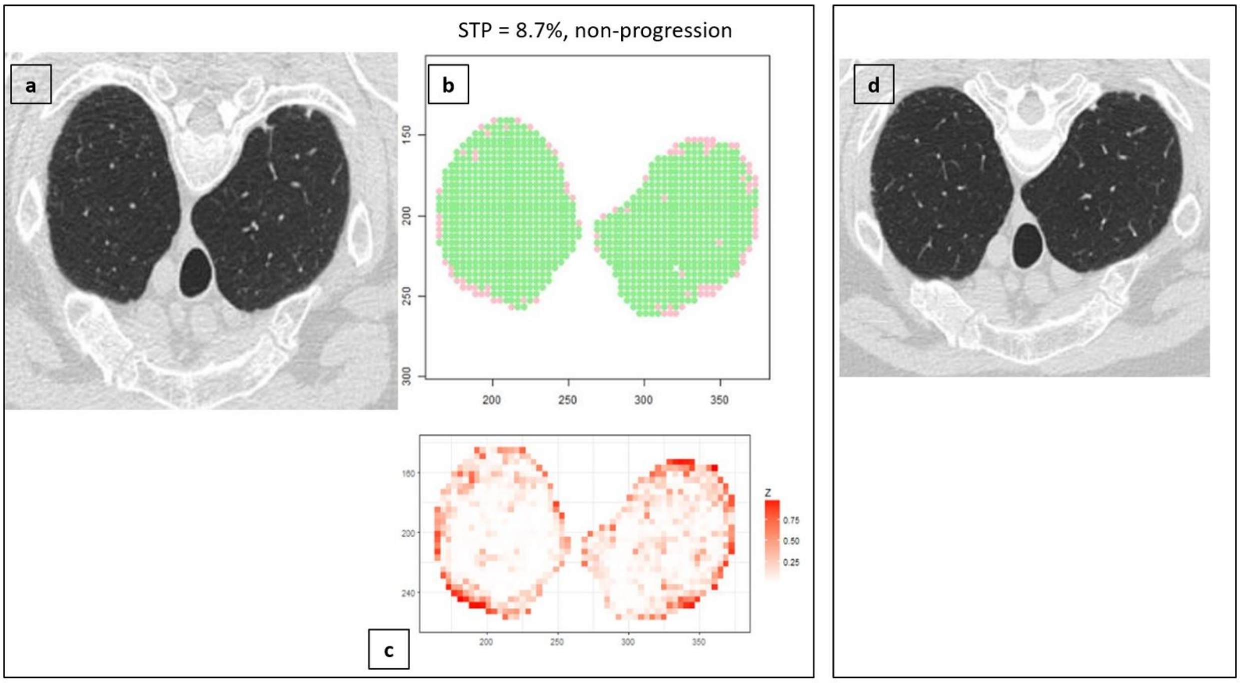


Figure 4.5: Generation of baseline Single-scan Total Probability (STP) for a sample subject. The subject has a QLF of 4% at baseline and 4% in a 12-month follow-up. Figure a. Baseline HRCT; b. Dichotomized classification of STP of figure a (green dots = stable voxels, red dots = progressive voxels, with 0.5 probability cutoff), the STP is 8.7% on this slide; c. STP result of figure a; d. 12-month follow-up HRCT showing progression.

We use a Cox regression model to determine whether the baseline metric STP is associated with the follow-up changes in population. We consider two types of outcomes, an imaging based outcome using quantitative lung fibrosis (QLF) score, and pulmonary function test based outcome using percentage predicted forced vital capacity (FVC). For the former, the event "progression" is defined by 4% increment or more of QLF score in the follow-up scan. The mean of imaging based progression free survival is 7.6 months (SE ± 0.2 months). The PFT based event was defined as 10% or more drop in percentage predicted FVC. PFT were typically performed as needed, so the duration might fall beyond the 5-13 months time window. The mean of PFT based progression free survival is 8.0 months (SE ± 0.4 months). The agreement between PFT and imaging based outcomes was poor with a kappa of 0.0823. This might be attributed to the fact that QLF based imaging changes are more sensitive in picking up early progressive signals, and may precede a future decrease of FVC [KWBon].

Table 4.5: The agreement between the imaging based outcomes and PFT based outcomes.

		PFT progression		Total
		0	1	
QLF progression	0	68	11	79
	1	31	6	37
Total		99	17	116

In the statistical analysis, we observe that STP was well correlated with the QLF changes in the follow-up scan. Higher STP is associated with higher risk of progression in a univariate analysis (hazard ratio=1.03; p = 0.027), and in a multivariate analysis after adjusting subjects' age and gender (hazard ratio=1.04; p = 0.041). We do not observe statistically significant trend using PFT based outcome, the percent predicted forced vital capacity (FVCpp), (hazard ratio=0.97, p=0.30 for univariate analysis; hazard ratio=1.00, p=0.99 for multivariate analysis adjusting for gender and age). Table 4.6 summarizes Cox regression results using imaging-based QLF change as the event outcome and FVC change as the event outcome.

Table 4.6: Summary of survival analysis results and baseline STP by the progression status in the follow-up.

			STP
Imaging outcome	Mean (± SE)	Δ QLF > 4	40.0 (± 1.8)
		Δ QLF ≤ 4	35.4 (± 1.2)
		Total	36.8 (± 1.0)
	Univariate hazard ratio (p-value)		1.03 (p=0.027*)
	Dichotomized #subjects (%)	> 40%	44 (36%)
		≤ 40%	78 (64%)
		logrank test	p=0.020*
	Multivariate hazard ratio (p-value)		1.04 (p=0.041*)
PFT outcome	Mean (± SE)	Δ FVCpp ≤ -10	36.7 (± 2.0)
		Δ FVCpp > -10	37.8 (± 1.1)
		Total	37.6 (± 1.0)
	Univariate hazard ratio (p-value)		0.97 (p=0.30)
	Dichotomized	logrank test	p=0.536
	Multivariate hazard ratio (p-value)		1.00 (p=0.99)

We observe that the STP is well correlated with imaging based progression. The STP is higher in progression group, is statistically significant in univariate and multivariate Cox regression results, and a log-rank test using STP cutoff at 40% show statistically significance. Using a STP cutoff at 40%, we plot the Kaplan-Meier curve using imaging based outcome as figure 4.6. It is clear that having a STP over 40% is a risk factor that leads to shorter progression free survival.

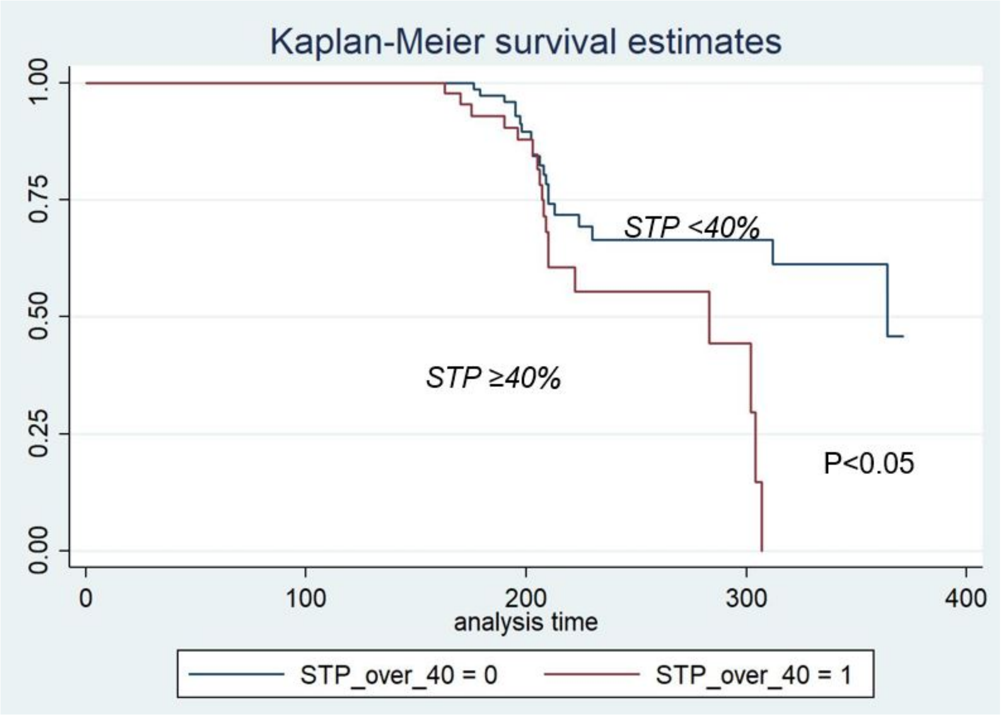


Figure 4.6: Kaplan-Meier curve for the STP cutoff at 40%. The event is defined by QLF score increase by 4% from baseline to follow-up.

We do not observe the same trend when using FVC percentage predicted values as outcomes. The STP is not statistically significant in either univariate or multivariate Cox regression results. When using STP cutoff at 40%, a log-rank test does not show statistically significance. We plot the Kaplan-Meier curve using FVC based outcome as figure 4.7. The curve does not separate well for subjects with and without a STP over 40%.

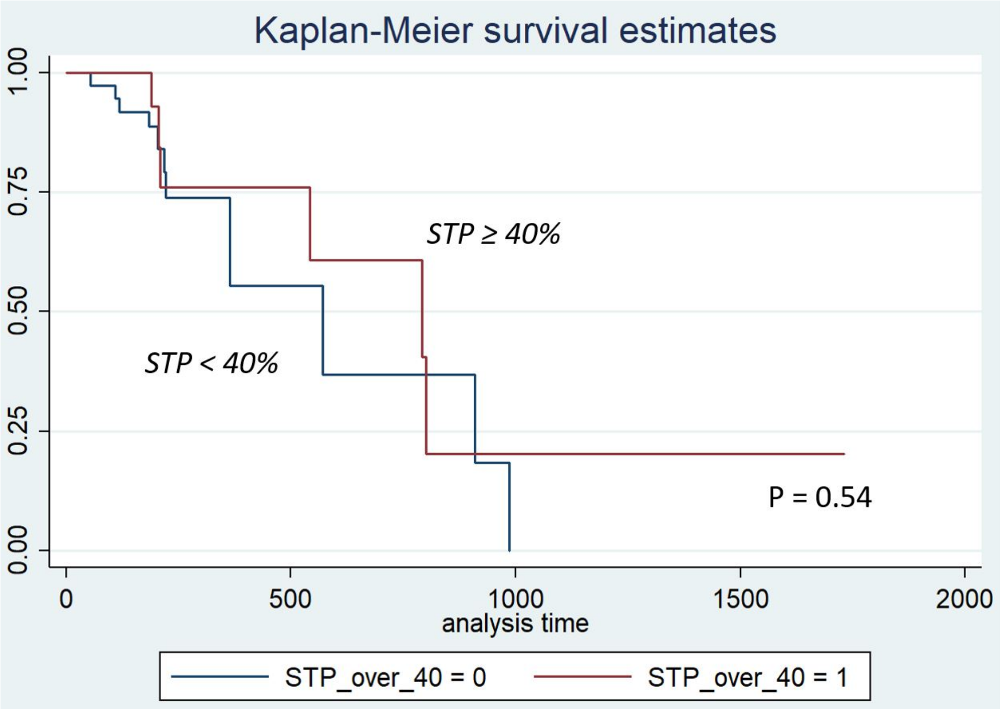


Figure 4.7: Kaplan-Meier curve for the STP cutoff at 40%. The event is defined by FVC percentage predicted values drop by over 10% from baseline to follow-up.

We further considered the refined predicted probabilities at subject level. Since we used a logistic link function, we have the original predicted values before dichotomizing the raw values into binary classes. We consider four ranges in the raw values: (i) $0 \leq P \leq 0.25$; (ii) $0.25 < P \leq 0.5$; (iii) $0.5 < P \leq 0.75$; (iv) $0.75 < P \leq 1$. For each subject, we sum up the number of voxels that fall in these four ranges, and calculate the percentages of voxels of the four ranges for each subject. By construction, the four percentages sum up to 1.

Figure 4.8 are scatter plots of the baseline QLF against the four probability ranges at the subject level. The 122 subjects in the test set were used in this analysis. We observe clearly that the range $0 \leq P \leq 0.25$ negatively correlates with the baseline QLF the most, with a R^2 of 0.46. This suggests that the low STP is correlated with a low baseline disease severity. On the other hand, the range $0.5 < P \leq 0.75$ positively correlates with the baseline QLF the most, with a R^2 of 0.59. This suggests that the high STP is correlated with a high baseline disease severity. This provides the connection of STP and baseline QLF.

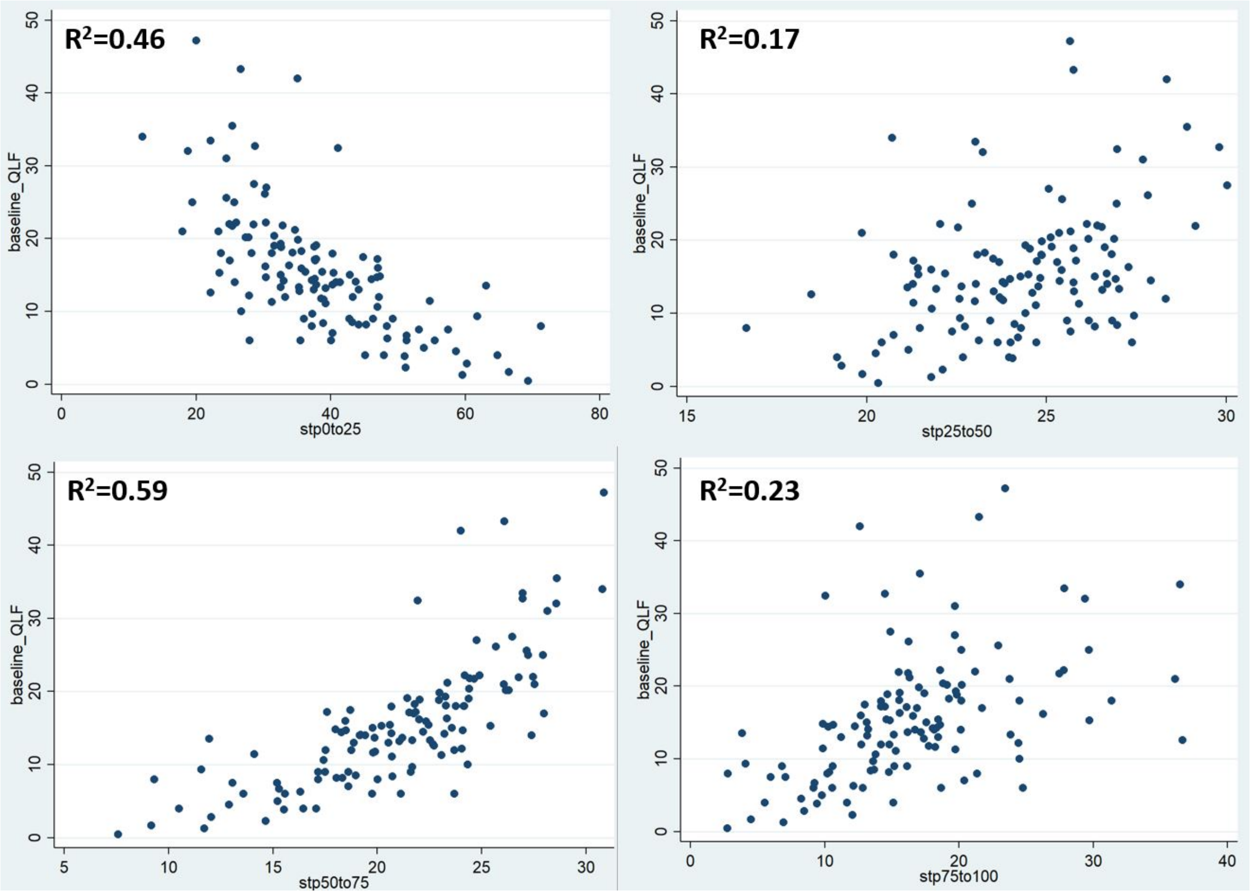


Figure 4.8: Baseline QLF against decomposed percentages of STP: [0, 25%], (25%, 50%], (50%, 75%], and (75, 100%] .

Overall, higher STPs are associated with higher baseline QLF. There is also a positive correlation trend in the QLF change and the STP. Figure 4.9 shows visualizations of baseline and delta QLF against STP.

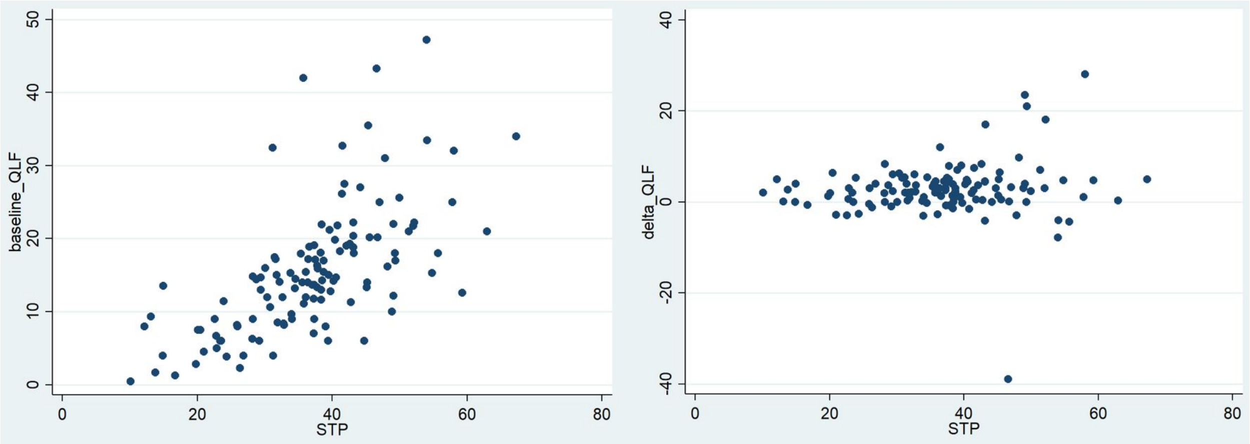


Figure 4.9: Baseline QLF against STP (left) and change in QLF between two visits against STP (right).

A statistical analysis using Cox regression models suggests the same trend we observe in figure 4.8: the range $0 \leq P \leq 0.25$ and $0.5 < P \leq 0.75$ seem to be the most relevant probability ranges in predicting the disease progression. Table 4.7 summarizes the Cox regression analysis results. The higher the percentage of $0 \leq P \leq 0.25$ in STP, the lower the hazard of getting progressed; the higher the percentage of $0.5 \leq P \leq 0.75$ in STP, the higher the hazard of getting progressed.

Table 4.7: Cox regression of decomposed percentages, using imaging outcomes. Reported results include hazard ratios (standard error) and p-values of univariate and multivariate cox regression analyses.

		$0\leq P \leq 0.25$	$0.25< P \leq 0.5$	$0.5< P \leq 0.75$	$0.75< P \leq 1$
Univariate	HR (SE)	0.97 (0.02)	1.04 (0.07)	1.08 (0.04)	1.04 (0.02)
	p-value	(0.049*)	(0.610)	(0.036*)	(0.083)
Multivariate	HR (SE)	0.97 (0.02)	1.03 (0.08)	1.08 (0.05)	1.05 (0.03)
(age, gender)	p-value	(0.078)	(0.718)	(0.078)	(0.076)

What's important about this prediction model is that it provides insights beyond the baseline disease severity. While the disease extent at baseline is an important factor in the disease natural history, this model seems to capture signals other than severity as well.

Specifically, we list two example cases to demonstrate this point. One example case has high disease severity at baseline (high QLF score at baseline), but is a non-progression case; this case has a baseline QLF of 42%, and a 6.4 months follow-up HRCT image has a QLF of 44%. The increase does not exceed 4% and is thus a non-progression case. The STP is 35.8%, suggesting progression is unlikely. Figure 4.10 is a visualization of a sample slide of this case.

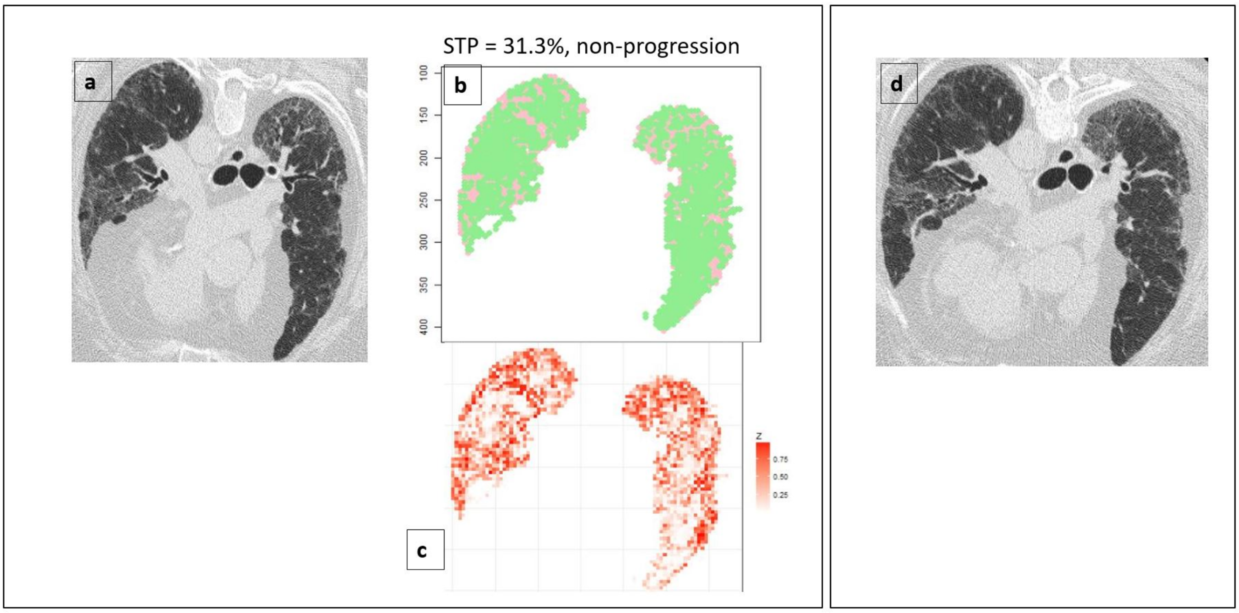


Figure 4.10: A sample case that experienced no progression but has high baseline QLF score. QLF at baseline is 42%, follow-up QLF is 44%, STP is 35.8%.

Another example case has low disease severity at baseline (low QLF score at baseline), but is a progression case; this case has a baseline QLF of 12.6%, and a 6.9 months follow-up HRCT image has a QLF of 17.3%. Since there is a more than 4% increase in QLF, this case is a progression case. The STP is 59.3%, suggesting progression is likely. Figure 4.11 is a visualization of a sample slide of this case.

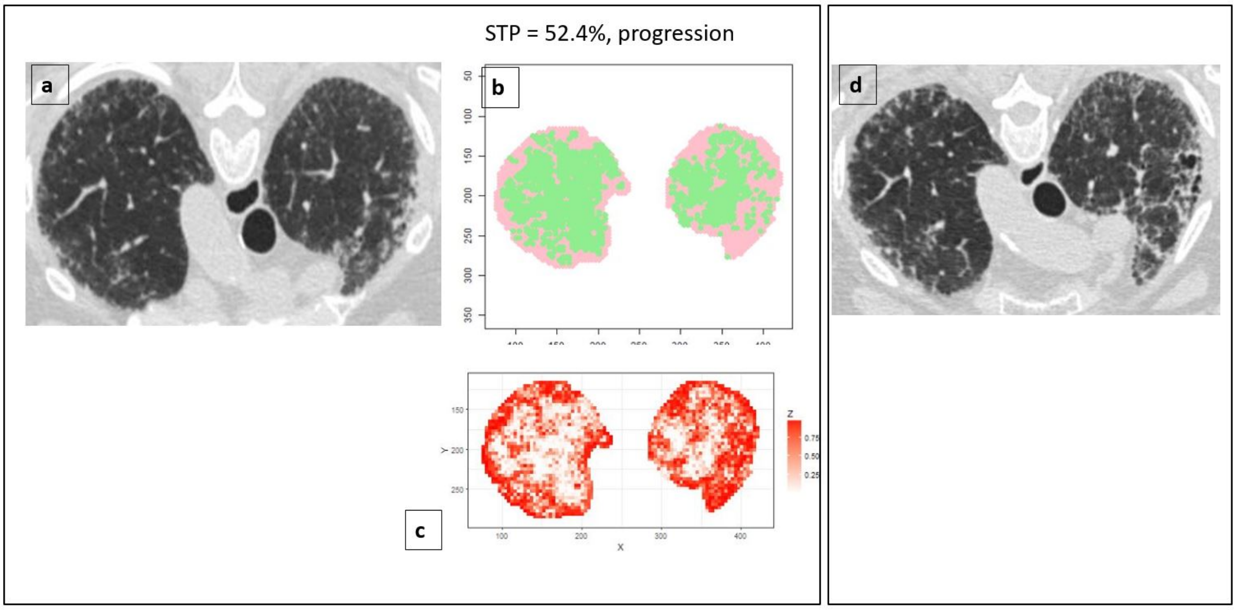
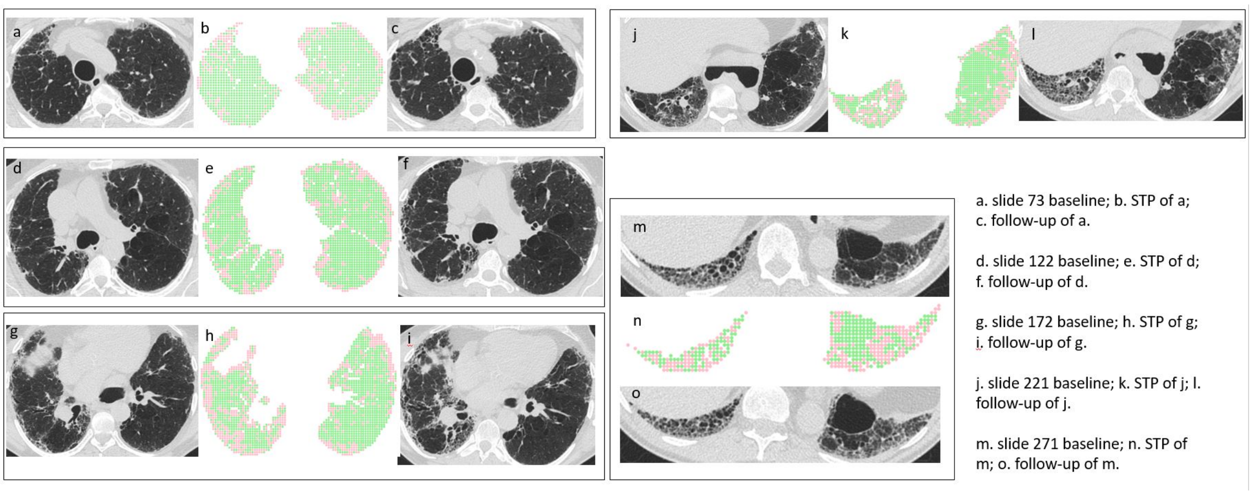


Figure 4.11: A sample case that experienced progression but has low baseline QLF score. QLF at baseline is 12.6%, follow-up QLF is 17.3%, STP is 59.3%.

We also show two cases that were mis-classified. The first example case has experienced progression but has a low whole lung STP of 29.4%. QLF at baseline is 13%, a 12-month follow-up QLF is 19%. The FVC percentage predicted value dropped from 75.2% at baseline to 60.7% at follow-up. Figure 4.13 is a visualization of 5 sample slides from different locations of the lung for this case. This subjects has 320 slides in total, and the selected slides are slide 73, 122, 172, 221 and 271. We observe that not enough voxels were classified as progression for this case, leading to a mis-classification. This is especially the case for the upper and middle lungs.



a. slide 73 baseline; b. STP of a; c. follow-up of a.

d. slide 122 baseline; e. STP of d; f. follow-up of d.

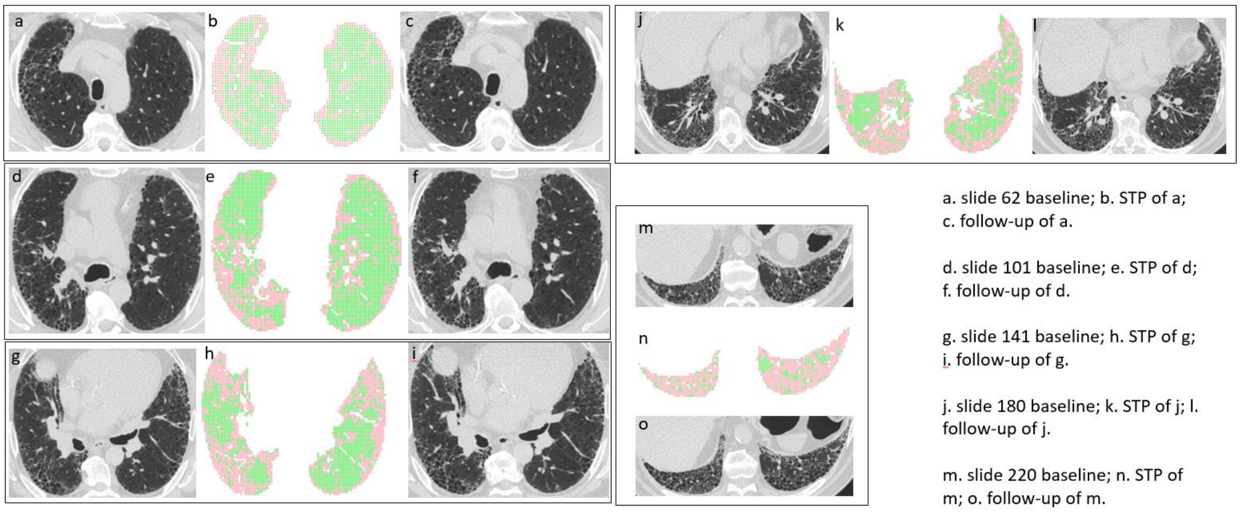
g. slide 172 baseline; h. STP of g; i. follow-up of g.

j. slide 221 baseline; k. STP of j; l. follow-up of j.

m. slide 271 baseline; n. STP of m; o. follow-up of m.

Figure 4.12: A sample case that experienced progression but has less than 40% STP. QLF at baseline is 13%, follow-up QLF is 19%, STP is 29.4%.

Another mis-classification example concerns a sample subject that did not experience progression but has a high whole lung level STP of 49.3%. The baseline QLF is 18%, and a 5-month follow-up had QLF of 18%. The FVC percentage predicted value increased from 70.3% to 74%. Figure 4.13 is a visualization of 5 sample slides from different locations of the lung for this case. This subjects has 259 slides in total, and the selected slides are slide 62, 101, 141, 180 and 220. We observe that STP was too sensitive and over-calls progression, leading to a mis-classification. This is especially the case for the lower lung.



a. slide 62 baseline; b. STP of a; c. follow-up of a.

d. slide 101 baseline; e. STP of d; f. follow-up of d.

g. slide 141 baseline; h. STP of g; i. follow-up of g.

j. slide 180 baseline; k. STP of j; l. follow-up of j.

m. slide 220 baseline; n. STP of m; o. follow-up of m.

Figure 4.13: A sample case that did not experience progression but has higher than 40% STP. QLF at baseline is 18%, follow-up QLF is 18%, STP is 49.3%.

In summary, the STP model is able to capture insights beyond the baseline disease severity because we implemented a study design that made it possible to provide a training sample that considered a variety of different scenarios of progressive patterns of ROIs. The non-progression ROIs in the training sample captures not only the normal lung patterns that stayed stable from baseline to follow-up, but also ground glass and fibrotic patterns that did not show worsening signs in the follow-up images. On the other hand, the progression ROIs in the training sample not only captures ROIs that got worse from gound glass or fibrotic patterns at baseline, but also those from normal at baseline and developed ground glass or fibrotic patterns at follow-up visits. Also, the even craniocaudal distribution of ROIs in the training sample also contributes to the stable prediction across different slides. The concern remains for the uneven axial distribution of ROIs, but this might be attributed to the nature of the disease.

In detail, table 4.8 summarizes the craniocaudal distributions of ROIs in the training sample. There are seven sub-types of ROIs in our training sample:
(i) expected better normal lung (eBNL) are regions in baseline that improve and become normal lung in follow-up. For example, a ROI that has ground glass in baseline and goes to

normal lung in follow-up.

(ii) expected better ground glass (eBGG) are regions in baseline that improve, and end up as ground glass in follow-up. For example, a ROI that has worse ground glass in baseline becomes improved ground glass in follow-up, OR lung fibrosis in baseline becomes ground glass in follow-up.

(iii) expected stable normal lung (eStNL) are normal lung in both follow-up and baseline.

(iv) expected stable ground glass (eStGG) are ground glass regions in baseline that are stable in the follow-up. Importantly, delineated region must contain voxels that are visually the same in follow-up.

(v) expected stable lung fibrosis (eStLF) are regions in baseline expected to be stable lung fibrosis in follow-up. Again, every voxel within region must contain voxels that are visually the same in follow-up.

(vi) expected progressive ground glass (ePrGG) are regions in baseline get worse, and end up as ground glass in the follow-up. For example, ground glass in baseline gets worse, and becomes worsened ground glass in follow-up, OR normal lung in baseline goes to ground glass in follow-up.

(vii) expected progressive lung fibrosis (ePrLF) are regions in baseline that get worse, and end up as lung fibrosis. For example, ground glass in baseline becomes lung fibrosis in follow-up, OR normal lung becomes lung fibrosis, OR lung fibrosis in baseline becomes even worse lung fibrosis in follow-up.

Table 4.8: The distribution of different ROI types in the training sample.

ROI type	Upper Lung	Middle Lung	Lower Lung	Total
eBNL	0	2	1	3
eBGG	3	1	5	9
eStNL	37	21	21	79
eStGG	31	39	19	89
eStLF	22	27	22	71
ePrGG	14	31	12	57
ePrLF	42	64	30	136
Total	149	185	100	434

4.5.2 Classifier with axial distribution information

In this section, we consider a classifier that takes advantage of axial distribution information. In each slides, we could classify voxels into two groups: voxels located within the 1cm radius of circle from the peripheral lesions, and voxels located outside of the 1cm radius of circle from the peripheral lesions. We create a variable named "outer2cm" and refer to peripheral voxels as "outer2cm=1" and other voxels as "outer2cm=0". Figure 4.14 is a visualization of the axial distribution in a sample slide.

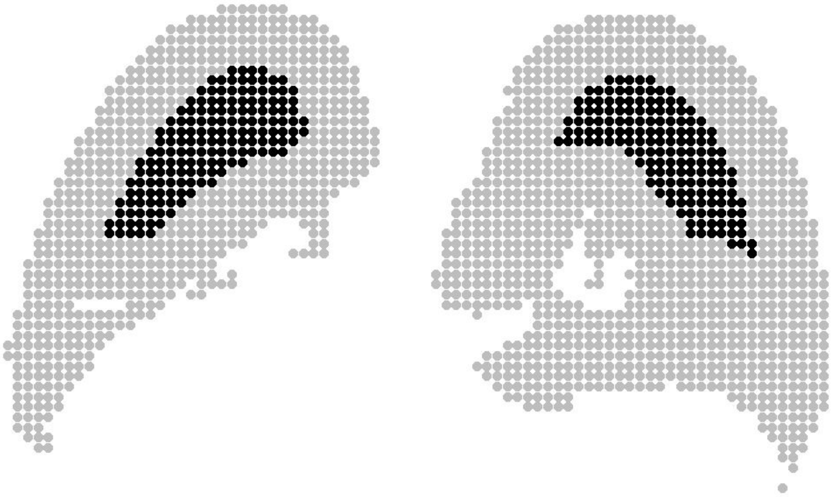


Figure 4.14: An example slide to show axial distribution information. Grey dots represent voxels located within the 1cm radius of circle from the peripheral lesions and black dots represent voxels located outside of the 1cm radius of circle from the peripheral lesions.

We combine the classifier we built using only textural information with the outer2cm variable. After we generate the prediction STPs, we re-weight the classification results using axial distribution information with three different weights combinations. The first re-weighting scheme, referred to as "RWT I", down-weights the peripheral voxels in the middle lung by 90% and voxels in the lower lung by 80%; namely, we multiply 90% on the voxels if they are in the middle lung and have "outer2cm"=1, and multiply 80% on the voxels if they are in the lower lung and have "outer2cm"=1. The second re-weighting scheme, referred to as "RWT II", down-weights the peripheral voxels in the middle lung by 60% and lower lung by 50%. The third re-weighting scheme, referred to as "RWT III", over-weights the peripheral voxels by multiplying 60% on voxels in the middle lung and have "outer2cm"=1, multiplying 40% on voxels in the middle lung and have "outer2cm"=0, and multiplying 50% to voxels in the lower lung. Table 4.9 summarizes the descriptive statistics and Cox regression outcomes of re-weighted STPs.

Table 4.9: Reweighted single-scan total probability summary table.

			RWT I	RWT II	RWT III
Imaging outcome	Mean (± SE)	Δ QLF>4	39.5 (± 1.8)	38.2 (± 1.8)	38.9 (± 1.8)
		ΔQLF≤4	34.9 (± 1.2)	33.4 (± 1.2)	34.2 (± 1.2)
		Total	36.3 (± 1.0)	34.9 (± 1.0)	35.7 (± 1.0)
	Univariate HR (p-value)		1.04 (p=0.02*)	1.04 (p=0.02*)	1.04 (p=0.02*)
	Dichotomized #subjects (%)	> 40%	41 (33%)	38 (31%)	38 (31%)
		≤ 40%	81 (67%)	84 (69%)	84 (69%)
		logrank test	p=0.04*	p=0.02*	p=0.02*
	Multivariate HR (p-value)		1.04 (p=0.04*)	1.04 (p=0.03*)	1.04 (p=0.03*)
PFT outcome	Mean (± SE)	ΔFVC≤-10	36.1 (± 2.0)	34.3 (± 2.0)	35.4 (± 2.0)
		ΔFVC>-10	37.3 (± 1.1)	35.9 (± 1.1)	36.6 (± 1.1)
		Total	37.1 (± 1.0)	35.7 (± 1.0)	36.4 (± 1.0)
	Univariate HR (p-value)		0.97 (p=0.30)	0.97 (p=0.32)	0.97 (p=0.30)
	Dichotomized	logrank test	p=0.57	p=0.57	p=0.57
	Multivariate HR (p-value)		1.00 (p=0.97)	1.00 (p=0.94)	1.00 (p=0.93)

We observe that all the conclusions in section 4.5.1 hold in the reweighted STP analysis. In the univariate analysis using ΔQLF ≥ 4 as the event, the hazard ratios are all beyond 1 and statistically significant. The multivariate analysis adjusting for age and gender also shows statistically significant hazard ratios beyond 1. We do not observe similar trends when using PFT-based events.

It seems that the weights do not have strong impact on the outcome. We thus conclude that the classifier using textural information is robust when incorporating axial distribution information.

4.5.3 Classifier with axial and craniocaudal distribution information

Since we are dealing with a series of slides, we could also leverage the information of craniocaudal distribution. Adding the z-axis distributions of the annotated ROIs into the feature selection process, we were able to obtain a feature subset that contains craniocaudal distribution information.

We standardize the craniocaudal distribution information for each subject using (slide number - first slide number)/(last slide number - first slide number). We re-selected a feature subset from 54 features (52 texture features for feature selection + craniocaudal variable + axial variable). QPSO-RF were run for 100 iterations, and at the last iteration, the best fit error rate was 25.0%, with 19 feature selected, including the axial and craniocaudal distribution variables.

While the Cox regression at population level showed significant association (hazard ratio=1.04, p=0.02 for univariate analysis; hazard ratio=1.07, p=0.02 for multivariate analysis adjusting for gender and age), the visual examination of the sample slides show that the overarching dominance of the craniocaudal distribution variable in the classifier leads to unstable prediction. We lose the generalizability at different locations in the lungs using this retrained classifier. Specifically, it over calls prediction in lower lungs, and under calls prediction in upper lungs. We thus conclude that the classification model is not robust when adding craniocaudal distribution information.

We conclude that the QPSO-random forest algorithm is stably effective in prediction when expanding to the whole lung level. A baseline STP metric from a model-driven percentage of progression is shown to be useful. This baseline metric can be explored further for predicting the clinical progression or the QLF changes in the follow-up HRCT scans.

CHAPTER 5

Discussion and conclusions

In this work, we proposed to use PSO as an efficient optimization algorithm coupled with different techniques for finding efficient designs and for predicting progression of IPF disease. The algorithm makes it possible to explore the design space or feature space more thoroughly. In particular, it finds optimal or highly efficient designs that were previously deemed to be intractable and also helps to generate a robust outcome from an algorithmic approach that predicts the IPF follow-up status of progression or not.

We note that there is no guarantee that PSO always outperforms other optimization algorithms. As the famous "no free lunch theorem" [WM97] suggests, there is no optimization algorithm that universally works better than all the other algorithms for all problems. The "best" algorithm should be discussed on a case-by-case basis. For project I, we note that there is no tool to confirm optimality of exact designs. Further, for high dimensional problems, we could only let PSO run so many iterations because of computational budget, and may not be able to find designs with the best efficiency. However, in our problems of finding optimal designs for biomedical studies, and the feature selection and classification problem in IPF prediction, PSO is able to provide high quality solutions.

Since PSO was proposed, there are many enhanced versions that provide various improvements for the basic PSO. Directions for future work include using one of these algorithms or a hybridized algorithm, that incorporates the best features of one or more algorithms into one and making the algorithm robust to a variety of mis-specifications in the model, prior distributions, and design criteria.

One potential issue in constructing high-dimensional or pseudo-Bayesian designs is that the computational time will increase when the model becomes large and the number of the design points increases. Parallel computing techniques can be applied to accelerate the computations. [HW12] proposed a GPU-accelerated PSO (GPSO) algorithm by using a thread pool model and implemented GPSO on a Graphic Processing Unit (GPU). The authors demonstrated that the proposed GPSO can significantly reduce the computational burden with satisfactory parallel efficiency. Likewise, [CHH13] proposed a discrete PSO approach, named LaPSO, to search for an optimal Latin hypercube design. The authors accelerated LaPSO by using GPU and showed that the GPU implementation can save computational time significantly for large optimization problems. We expect that the programs in this article can also be accelerated on parallel computers such as GPU. The parallelization will likely produce computing tools with faster response time and better user experience.

There are continuing challenging problems in our work. For project I, we do not claim we are able to find all types of optimal designs in a regression setup using the types of algorithms we proposed here. We point out a few of these problems: (i) we have experiences with metaheuristic algorithms that work well for some nominal values for a model but not for other nominal values; similarly, the same algorithm may not work well when the design space is changed, and especially when it is enlarged. These are likely scaling problems that we can address and understand later on. (ii) Confirming optimality of a generated design can be challenging because sometimes the verification is carried out graphically and it is difficult to appreciate interesting features in a high dimensional plot. An alternative is to find an efficiency lower bound, which amounts to solving another high-dimensional optimization problem to find the maximal value of the sensitivity function. This means that the metaheuristic algorithm has to be applied twice or find another more appropriate metaheuristic algorithm to solve this second optimization problem.

For project II, there are several limitations of the study. Firstly, we used a retrospectively collected data set and it remains a question whether the model can be applied to a prospective setting. Secondly, the training of the model is limited to the classical representation of small ROIs. Thirdly, the reference of gold standard was based on a single reader's visual assessment. This is mainly due to the time and budget constraints. Analytical and clinical validations are warranted in future to evaluate a predictive biomarker and to test the generalizability of algorithm in practice. Further, we only compared our algorithm to selected popular methods, with parameters reported as best practices in the literature. There are possibly better implementations of comparator methods, or other methods that are also suitable for our problem. Lastly, we do not observe a signal in the progression in FVC percentage predicted values, and this might be attributed to the study design to collect the PFT data. Currently, the PFT data was only collected when the subjects: (1) have a baseline and follow-up HRCT images; and (2) have PFT information close to the CT date. This study design did not collect the PFT information from subjects who dropped out at an early stage, and also missed PFT information that was collected in dates far from CT dates. Some lost information could not be recovered, and it is unclear whether the imaging based predictor could anticipate PFT changes.

In conclusion, our main contributions in this paper are 1) use PSO coupled with sparse grid to find efficient designs for non-linear mixed effects models; and 2) implement a novel study design, and use PSO coupled with random forest to build a model for IPF disease prediction using high dimensional CT images. These are challenging tasks with solutions largely unavailable before.

We close with several cautionary notes: (i) an optimal design should not be used religiously but should serve as a guide or benchmark. This is because the optimal design is found under a fixed set of assumptions that may not adequately reflect reality and so may not satisfy the needs of the practitioners. Different optimal designs under various settings should be compared carefully before the design is implemented. The guiding principle is that the implemented design should not stray too far from the optimum as measured by its efficiency relative to the optimum. (ii) The disease prediction model has not been validated prospectively and is only at the exploratory stage. While validating a biomarker takes years with prospective clinical evaluation, we hope that our model is a useful starting point for future model development.

[AC02] Dimitris K Agrafiotis and Walter Cedeno. "Feature selection for structure- activity correlation using binary particle swarms." *Journal of medicinal chemistry*, **45**(5):1098–1107, 2002.

[ACC07] Gabriel LFBG Azevedo, George DC Cavalcanti, and Edson CB Carvalho Filho. "An approach to feature selection for keystroke dynamics systems based on PSO and feature weighting." In *2007 IEEE Congress on Evolutionary Computation*, pp. 3577–3584. IEEE, 2007.

[ACH93] Anthony C Atkinson, Kathryn Chaloner, Agnes M Herzberg, and June Juritz. "Optimum experimental designs for properties of a compartmental model." *Biometrics*, pp. 325–337, 1993.

[ADT09] Aa Ca Atkinson, Aa Na Donev, and Re De Tobias. *Optimum Experimental Designs, with SAS.* Oxford Statistical Science. Oxford University Press, Oxford, 2009.

[AGJ07] Enrique Alba, Jose Garcia-Nieto, Laetitia Jourdan, and El-Ghazali Talbi. "Gene selection in cancer classification using PSO/SVM and GA/SVM hybrid algorithms." In *2007 IEEE Congress on Evolutionary Computation*, pp. 284–290. IEEE, 2007.

[ANA15] Mohd Nadhir Ab Wahab, Samia Nefti-Meziani, and Adham Atyabi. "A comprehensive review of swarm optimization algorithms." *PloS one*, **10**(5):e0122827, 2015.

[ARC08] M Senthil Arumugam, MVC Rao, and Aarthi Chandramohan. "A new and improved version of particle swarm optimization algorithm with global–local best parameters." *Knowledge and Information systems*, **16**(3):331–357, 2008.

[AS15] Shikha Agrawal and Sanjay Silakari. "A review on application of Particle Swarm Optimization in Bioinformatics." *Current bioinformatics*, **10**(4):401–413, 2015.

[AWH04] Edward P Acosta, Hulin Wu, Scott M Hammer, Song Yu, Daniel R Kuritzkes, Ann Walawander, Joseph J Eron, Carl J Fichtenbaum, Carla Pettinelli, Denise Neath, et al. "Comparison of two indinavir/ritonavir regimens in the treatment of HIV-infected individuals." *JAIDS Journal of Acquired Immune Deficiency Syndromes*, **37**(3):1358–1366, 2004.

[BBM15] Akram Belghith, Christopher Bowd, Felipe A Medeiros, Madhusudhanan Balasubramanian, Robert N Weinreb, and Linda M Zangwill. "Learning from healthy and stable eyes: a new approach for detection of glaucomatous progression." *Artificial intelligence in medicine*, **64**(2):105–115, 2015.

[BBU92] M Becka, HM Bolt, and W Urfer. “Statistical analysis of toxicokinetic data by nonlinear regression (example: inhalation pharmacokinetics of propylene).” *Archives of toxicology*, **66**(6):450–453, 1992.

[BE06] Frans Van den Bergh and Andries Petrus Engelbrecht. “A study of particle swarm optimization particle trajectories.” *Information sciences*, **176**(8):937–971, 2006.

[BFM97] Thomas Bäck, David B Fogel, and Zbigniew Michalewicz. *Handbook of evolutionary computation*. CRC Press, 1997.

[BG04] Hans-Joachim Bungartz and Michael Griebel. “Sparse grids.” *Acta numerica*, **13**:147–269, 2004.

[BGH89] Lashon B. Booker, David E. Goldberg, and John H. Holland. “Classifier systems and genetic algorithms.” *Artificial intelligence*, **40**(1-3):235–282, 1989.

[BH95] Yoav Benjamini and Yosef Hochberg. “Controlling the false discovery rate: a practical and powerful approach to multiple testing.” *Journal of the royal statistical society. Series B (Methodological)*, pp. 289–300, 1995.

[BM14] Mohammad Reza Bonyadi and Zbigniew Michalewicz. “A locally convergent rotationally invariant particle swarm optimization algorithm.” *Swarm intelligence*, **8**(3):159–198, 2014.

[BM17] Mohammad Reza Bonyadi and Zbigniew Michalewicz. “Particle swarm optimization for single objective continuous space problems: a review.”, 2017.

[Bre96] Leo Breiman. “Bagging predictors.” *Machine learning*, **24**(2):123–140, 1996.

[Bre01] Leo Breiman. “Random forests.” *Machine learning*, **45**(1):5–32, 2001.

[BRM09] Caroline Bazzoli, Sylvie Retout, and France Mentré. “Fisher information matrix for nonlinear mixed effects multiple response models: evaluation of the appropriateness of the first order linearization using a pharmacokinetic/pharmacodynamic model.” *Statistics in Medicine*, **28**(14):1940–1956, 2009.

[BUF18] Simon Buatois, Sebastian Ueckert, Nicolas Frey, Sylvie Retout, and France Mentré. “Comparison of Model Averaging and Model Selection in Dose Finding Trials Analyzed by Nonlinear Mixed Effect Models.” *The AAPS journal*, **20**(3):56, 2018.

[But79] Christopher Button. *Pharmacokinetics, bioavailability, and dosage regimens of digoxin in the horse and dog*. Button, 1979.

[BW05] Martijn PF Berger and Weng-Kee Wong. *Applied optimal designs*. John Wiley & Sons, 2005.

[CBH02] Nitesh V. Chawla, Kevin W. Bowyer, Lawrence O. Hall, and W. Philip Kegelmeyer. “SMOTE: synthetic minority over-sampling technique.” *Journal of artificial intelligence research*, **16**:321–357, 2002.

[CCW15] Ray Bing Chen, Shin Perng Chang, Weichung Wang, Heng-Chih Tung, and Weng Kee Wong. "Minimax optimal designs via particle swarm optimization methods." *Statistics and Computing*, **25**(5):975–988, 2015.

[CE14] Christopher W Cleghorn and Andries P Engelbrecht. "A generalized theoretical deterministic particle swarm model." *Swarm intelligence*, **8**(1):35–59, 2014.

[CHH13] Ray-Bing Chen, Dai-Ni Hsieh, Ying Hung, and Weichung Wang. "Optimizing Latin hypercube designs by particle swarm." *Statistics and computing*, **23**(5):663–676, 2013.

[CJ15] Ran Cheng and Yaochu Jin. "A competitive swarm optimizer for large scale optimization." *IEEE transactions on cybernetics*, **45**(2):191–204, 2015.

[CK02] Maurice Clerc and James Kennedy. "The particle swarm-explosion, stability, and convergence in a multidimensional complex space." *IEEE transactions on Evolutionary Computation*, **6**(1):58–73, 2002.

[CL89] Kathryn Chaloner and Kinley Larntz. "Optimal Bayesian design applied to logistic regression experiments." *Journal of Statistical Planning and Inference*, **21**(2):191–208, 1989.

[Cle99] Maurice Clerc. "The swarm and the queen: towards a deterministic and adaptive particle swarm optimization." In *Evolutionary Computation, 1999. CEC 99. Proceedings of the 1999 Congress on*, volume 3, pp. 1951–1957. IEEE, 1999.

[CM11] Ellen K Cromley and Sara L McLafferty. *GIS and public health.* Guilford Press, 2011.

[CS13] Antonio Criminisi and Jamie Shotton. *Decision forests for computer vision and medical image analysis.* Springer Science & Business Media, 2013.

[CYK08] Baek Hwan Cho, Hwanjo Yu, Kwang-Won Kim, Tae Hyun Kim, In Young Kim, and Sun I Kim. "Application of irregular and unbalanced data to predict diabetic nephropathy using visualization and feature selection methods." *Artificial intelligence in medicine*, **42**(1):37–53, 2008.

[DB10] Z Dzalilov and A Bagirov. "Cluster Analysis of A Tobacco Control Data Set." *International Journal of Lean Thinking*, **1**(2):40–45, 2010.

[DBJ99] PO Droz, M Berode, and JY Jang. "Biological monitoring of tetrahydrofuran: contribution of a physiologically based pharmacokinetic model." *American Industrial Hygiene Association Journal*, **60**(2):243–248, 1999.

[DBP12] Chesner Désir, Simon Bernard, Caroline Petitjean, and Laurent Heutte. "A random forest based approach for one class classification in medical imaging." In *International Workshop on Machine Learning in Medical Imaging*, pp. 250–257. Springer, 2012.

[Die00] Thomas G Dietterich. "An experimental comparison of three methods for constructing ensembles of decision trees: Bagging, boosting, and randomization." *Machine learning*, **40**(2):139–157, 2000.

[DLL18] Cyrielle Dumont, Giulia Lestini, Hervé Le Nagard, France Mentré, Emmanuelle Comets, Thu Thuy Nguyen, and PFIM Group. "PFIM 4.0, an extended R program for design evaluation and optimization in nonlinear mixed-effect models." *Computer Methods and Programs in Biomedicine*, **156**:217–229, 2018.

[DN97] Holger Dette and H-M Neugebauer. "Bayesian D-optimal designs for exponential regression models." *Journal of Statistical Planning and Inference*, **60**(2):331–349, 1997.

[DPK11] Roosmarijn FW De Cock, Chiara Piana, Elke HJ Krekels, Meindert Danhof, Karel Allegaert, and Catherijne AJ Knibbe. "The role of population PK–PD modelling in paediatric clinical research." *European journal of clinical pharmacology*, **67**(1):5–16, 2011.

[DPW09] Holger Dette, Andrey Pepelyshev, Weng Kee Wong, et al. "Optimal designs for dose-finding experiments in toxicity studies." *Bernoulli*, **15**(1):124–145, 2009.

[DPZ08] Holger Dette, Andrey Pepelyshev, and Anatoly Zhigljavsky. "Improving updating rules in multiplicative algorithms for computing D-optimal designs." *Computational Statistics & Data Analysis*, **53**(2):312–320, 2008.

[DPZ13] Holger Dette, Andrey Pepelyshev, Anatoly Zhigljavsky, et al. "Optimal design for linear models with correlated observations." *The Annals of Statistics*, **41**(1):143–176, 2013.

[DW15] Belmiro PM Duarte and Weng Kee Wong. "Finding Bayesian Optimal Designs for Nonlinear Models: A Semidefinite Programming-Based Approach." *International Statistical Review*, **83**(2):239–262, 2015.

[Dyk71] Otto Dykstra. "The Augmentation of Experimental Data to Maximize [X X]." *Technometrics*, **13**(3):682–688, 1971.

[EK95] Russell Eberhart and James Kennedy. "A new optimizer using particle swarm theory." In *Micro Machine and Human Science, 1995. MHS'95., Proceedings of the Sixth International Symposium on*, pp. 39–43. IEEE, 1995.

[Eng06] Andries P Engelbrecht. *Fundamentals of computational swarm intelligence*. John Wiley & Sons, 2006.

[Eng13] AP Engelbrecht. "Particle swarm optimization: Global best or local best?" In *2013 BRICS Congress on Computational Intelligence & 11th Brazilian Congress on Computational Intelligence (BRICS-CCI & CBIC)*, pp. 124–135. IEEE, 2013.

[ES00] Russ C Eberhart and Yuhui Shi. "Comparing inertia weights and constriction factors in particle swarm optimization." In *Evolutionary Computation, 2000. Proceedings of the 2000 Congress on*, volume 1, pp. 84–88. IEEE, 2000.

[FCD00] Terrence S Furey, Nello Cristianini, Nigel Duffy, David W Bednarski, Michel Schummer, and David Haussler. "Support vector machine classification and validation of cancer tissue samples using microarray expression data." *Bioinformatics*, **16**(10):906–914, 2000.

[FCT03] Kevin R Flaherty, Thomas V Colby, William D Travis, Galen B Toews, Jeanette Mumford, Susan Murray, Victor J Thannickal, Ella A Kazerooni, Barry H Gross, Joseph P Lynch III, et al. "Fibroblastic foci in usual interstitial pneumonia: idiopathic versus collagen vascular disease." *American journal of respiratory and critical care medicine*, **167**(10):1410–1415, 2003.

[FDZ12] Yangguang Fu, Mingyue Ding, and Chengping Zhou. "Phase angle-encoded and quantum-behaved particle swarm optimization applied to three-dimensional route planning for UAV." *IEEE transactions on systems, man and cybernetics, part A: systems and humans*, **42**(2):511–526, 2012.

[Fed72] Valerii Vadimovich Fedorov. *Theory of optimal experiments.* Elsevier, 1972.

[FHT01] Jerome Friedman, Trevor Hastie, and Robert Tibshirani. *The elements of statistical learning*, volume 1. Springer series in statistics Springer, Berlin, 2001.

[FL01] Jianqing Fan and Runze Li. "Variable selection via nonconcave penalized likelihood and its oracle properties." *Journal of the American statistical Association*, **96**(456):1348–1360, 2001.

[FL14] V. V. Fedorov and S. L. Leonov. *Optimal Design for Nonlinear Response Models.* Chapman and Hall/ CRC Press, Boca Raton, 2014.

[FLW09] Xiuju Fu, Sonja Lim, Lipo Wang, Gary Lee, Stefan Ma, Limsoon Wong, and Gaoxi Xiao. "Key node selection for containing infectious disease spread using particle swarm optimization." In *Swarm Intelligence Symposium, 2009. SIS'09. IEEE*, pp. 109–113. IEEE, 2009.

[Fre84] J Fresen. "Aspect of bioavailability studies." *Department of Mathematical Statistics, University of Capetown*, 1984.

[GG98] Thomas Gerstner and Michael Griebel. "Numerical integration using sparse grids." *Numerical algorithms*, **18**(3):209–232, 1998.

[GJS09] Christopher M Gotwalt, Bradley A Jones, and David M Steinberg. "Fast computation of designs robust to parameter uncertainty for nonlinear settings." *Technometrics*, **51**(1):88–95, 2009.

[GK96] Alan Genz and Bradley D Keister. "Fully symmetric interpolatory rules for multiple integrals over infinite regions with Gaussian weight." *Journal of Computational and Applied Mathematics*, **71**(2):299–309, 1996.

[GLO16] Yanming Guo, Yu Liu, Ard Oerlemans, Songyang Lao, Song Wu, and Michael S Lew. "Deep learning for visual understanding: A review." *Neurocomputing*, **187**:27–48, 2016.

[GZ13] Yue-jiao Gong and Jun Zhang. “Small-world particle swarm optimization with topology adaptation.” In *Proceedings of the 15th annual conference on Genetic and evolutionary computation*, pp. 25–32. ACM, 2013.

[Har79] Robert M Haralick. “Statistical and structural approaches to texture.” *Proceedings of the IEEE*, **67**(5):786–804, 1979.

[HBL18] Md Akter Hussain, Alauddin Bhuiyan, Chi D Luu, R Theodore Smith, Robyn H Guymer, Hiroshi Ishikawa, Joel S Schuman, and Kotagiri Ramamohanarao. “Classification of healthy and diseased retina using SD-OCT imaging and Random Forest algorithm.” *PloS one*, **13**(6):e0198281, 2018.

[HC03] Cong Han and Kathryn Chaloner. “D-and c-optimal designs for exponential regression models used in viral dynamics and other applications.” *Journal of Statistical Planning and Inference*, **115**(2):585–601, 2003.

[HCD05] Rania Hassan, Babak Cohanim, Olivier De Weck, and Gerhard Venter. “A comparison of particle swarm optimization and the genetic algorithm.” In *46th AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics and Materials Conference*, p. 1897, 2005.

[HD08] Cheng-Lung Huang and Jian-Fan Dun. “A distributed PSO–SVM hybrid system with feature selection and parameter optimization.” *Applied soft computing*, **8**(4):1381–1391, 2008.

[HG14] Matthew D Hoffman and Andrew Gelman. “The No-U-Turn sampler: adaptively setting path lengths in Hamiltonian Monte Carlo.” *Journal of Machine Learning Research*, **15**(1):1593–1623, 2014.

[HH07] Rui Huang and Mingyi He. “Feature selection using double parallel feedforward neural networks and particle swarm optimization.” In *2007 IEEE Congress on Evolutionary Computation*, pp. 692–696. IEEE, 2007.

[HLS13] David W Hosmer Jr, Stanley Lemeshow, and Rodney X Sturdivant. *Applied logistic regression*, volume 398. John Wiley & Sons, 2013.

[HV05] A Hooker and P Vicini. “Simultaneous population optimal design for pharmacokinetic-pharmacodynamic experiments.” *American Association of Pharmaceutical Scientists Journal*, **7**:759–785, 2005.

[HW08a] Florian Heiss and Viktor Winschel. “Likelihood approximation by numerical integration on sparse grids.” *journal of Econometrics*, **144**(1):62–80, 2008.

[HW08b] Yangxin Huang and Hulin Wu. “Bayesian experimental design for long-term longitudinal HIV dynamic studies.” *Journal of statistical planning and inference*, **138**(1):105–113, 2008.

[HW12] Yukai Hung and Weichung Wang. “Accelerating parallel particle swarm optimization via GPU.” *Optimization Methods and Software*, **27**(1):33–51, 2012.

[IP12] Hesam Izakian and Witold Pedrycz. "A new PSO-optimized geometry of spatial and spatio-temporal scan statistics for disease outbreak detection." *Swarm and Evolutionary Computation*, **4**:1–11, 2012.

[JJ15] Cong Jin and Shu Wei Jin. "Prediction approach of software fault-proneness based on hybrid artificial neural network and quantum particle swarm optimization." *Applied Soft Computing*, **35**:717–725, 2015.

[JKP94] George H John, Ron Kohavi, Karl Pfleger, et al. "Irrelevant features and the subset selection problem." In *Machine learning: proceedings of the eleventh international conference*, pp. 121–129, 1994.

[JLY07] Ming Jiang, Yupin P Luo, and Shiyuan Y Yang. "Particle swarm optimization-stochastic trajectory analysis and parameter selection." In *Swarm intelligence, Focus on ant and particle swarm optimization*. InTech, 2007.

[JM05] Stefan Janson and Martin Middendorf. "A hierarchical particle swarm optimizer and its adaptive variant." *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, **35**(6):1272–1282, 2005.

[JM16] Hao Ji and Hans-Georg Müller. "Optimal designs for longitudinal and functional data." *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 2016.

[KBC14] Talmadge E King Jr, Williamson Z Bradford, Socorro Castro-Bernardini, Elizabeth A Fagan, Ian Glaspole, Marilyn K Glassberg, Eduard Gorina, Peter M Hopkins, David Kardatzke, Lisa Lancaster, et al. "A phase 3 trial of pirfenidone in patients with idiopathic pulmonary fibrosis." *New England Journal of Medicine*, **370**(22):2083–2092, 2014.

[KBC15] Hyun J Kim, Matthew S Brown, Daniel Chong, David W Gjertson, Peiyun Lu, Hak J Kim, Heidi Coy, and Jonathan G Goldin. "Comparison of the quantitative CT imaging biomarkers of idiopathic pulmonary fibrosis at baseline and early change with an interval of 7 months." *Academic radiology*, **22**(1):70–80, 2015.

[KBO85] Jack Carl Kiefer, LD Brown, I Olkin, and J Sacks. *Jack Carl Kiefer Collected Papers: Design of Experiments*. Springer, 1985.

[KBW16] Hyun J Kim, Matthew Brown, Sam Weigt, John A Belperio, and Jonathan Goldin. "Prediction of IPF with the Early Changes in Quantitative Imaging Patterns Using High Resolution Computed Tomography." In *A103. IPF: MORE ABOUT THERAPY AND OUTCOMES*, pp. A2706–A2706. Am Thoracic Soc, 2016.

[KBW17] Grace Hyun J Kim, Matthew Brown, Sam Weigt, John A Belperio, Richard H Huynh, Yu Shi, and Jonathan G Goldin. "Prediction Of IPF Using Early Changes In Quantitative Imaging Patterns On High Resolution Computed Tomography." In *C39. BIOMARKERS AND PROGNOSTIC FACTORS IN ILD*, pp. A5432–A5432. Am Thoracic Soc, 2017.

[Ken06] James Kennedy. "Handbook of nature-inspired and innovative computing." *Swarm intelligence. Springer*, pp. 187–219, 2006.

[KGK05] Kwang Gi Kim, Jin Mo Goo, Jong Hyo Kim, Hyun Ju Lee, Byung Goo Min, Kyongtae T Bae, and Jung-Gi Im. "Computer-aided diagnosis of localized ground-glass opacity in the lung at CT: initial experience." *Radiology*, **237**(2):657–661, 2005.

[KGR07] Gary King, Emmanuela Gakidou, Nirmala Ravishankar, Ryan T Moore, Jason Lakin, Manett Vargas, Martha María Téllez-Rojo, Juan Eugenio Hernández Ávila, Mauricio Hernández Ávila, and Héctor Hernández Llamas. "A "politically robust" experimental design for public policy evaluation, with application to the Mexican universal health insurance program." *Journal of Policy Analysis and Management*, **26**(3):479–506, 2007.

[KJ97] Ron Kohavi and George H John. "Wrappers for feature subset selection." *Artificial intelligence*, **97**(1-2):273–324, 1997.

[KKK13] Cem Keskin, Furkan Kıraç, Yunus Emre Kara, and Lale Akarun. "Real time hand pose estimation using depth sensors." In *Consumer depth cameras for computer vision*, pp. 119–137. Springer, 2013.

[KLG08] Hyun J Kim, Gang Li, David Gjertson, Robert Elashoff, Sumit K Shah, Robert Ochs, Fah Vasunilashorn, Fereidoun Abtin, Matthew S Brown, and Jonathan G Goldin. "Classification of parenchymal abnormality in scleroderma lung using a novel approach to denoise images collected via a multicenter study." *Academic radiology*, **15**(8):1004–1016, 2008.

[KSB01] TALMADGE E King JR, Marvin I Schwarz, Kevin Brown, Janet A Tooze, Thomas V Colby, JAMES A Waldron JR, Andrew Flint, William Thurlbeck, and Reuben M Cherniack. "Idiopathic pulmonary fibrosis: relationship between histopathologic features and mortality." *American journal of respiratory and critical care medicine*, **164**(6):1025–1032, 2001.

[KSD15] Neda Kaffash-Charandabi, Abolghasem Sadeghi-Niaraki, and PARK Dong-Kyun. "Using a Combined Platform of Swarm Intelligence Algorithms and GIS to Provide Land Suitability Maps for Locating Cardiac Rehabilitation Defibrillators." *Iranian journal of public health*, **44**(8):1072, 2015.

[KTC10] HJ Kim, DP Tashkin, P Clements, G Li, MS Brown, R Elashoff, DW Gjertson, F Abtin, DA Lynch, DC Strollo, et al. "A computer-aided diagnosis system for quantitative scoring of extent of lung fibrosis in scleroderma patients." *Clinical and experimental rheumatology*, **28**(5 Suppl 62):S26, 2010.

[KTG16] Hyun J Kim, Donald P Tashkin, David W Gjertson, Matthew S Brown, Eric Kleerup, Semin Chong, John A Belperio, Michael D Roth, Fereidoun Abtin, Robert Elashoff, et al. "Transitions to different patterns of interstitial lung disease in scleroderma with and without treatment." *Annals of the rheumatic diseases*, **75**(7):1367–1371, 2016.

[Kuo03] Diego Kuonen et al. "Numerical integration in S-PLUS or R: A survey." *Journal of Statistical Software*, **8**(13):1–14, 2003.

[KW17] Seongho Kim and Weng Kee Wong. "Extended two-stage adaptive designs with three target responses for phase II clinical trials." *Statistical methods in medical research*, p. 0962280217709817, 2017.

[KWBon] Grace Hyun Kim, Stephen Weight, John Belperio, Yu Shi, and Jonathan Goldin. "Prediction of Idiopathic Pulmonary Fibrosis Progression Using Early Quantitative Changes on CT Imaging for a Short Term of Clinical 18-24 Month Follow-ups." in revision.

[LD13] Teresa B Ludermir and Wilson R De Oliveira. "Particle swarm optimization of MLP for the identification of factors related to common mental disorders." *Expert Systems with Applications*, **40**(11):4648–4652, 2013.

[Lia05] Percy Liang. *Semi-supervised learning for natural language*. PhD thesis, Massachusetts Institute of Technology, 2005.

[LJS15] Yangyang Li, Licheng Jiao, Ronghua Shang, and Rustam Stolkin. "Dynamic-context cooperative quantum-behaved particle swarm optimization based on multilevel thresholding applied to medical image segmentation." *Information Sciences*, **294**:408–422, 2015.

[LKH07] R. L. Lalonde, K. G. Kowalski, M. M. Hutmacher, W. Ewy, D. J. Nichols, P. A. Milligan, B. W. Corrigan, P. A. Lockwood, S. A. Marshall, L. J. Benincosa, T. G. Tensfeldt, K. Parivar, M. Amantea, P. Glue, H. Koide, and R. Miller. "Model-based drug development." *Clinical Pharmacology & Therapeutics*, **82**(1):21–32, Jul 2007.

[LMW18] Joshua Lukemire, Abhyuday Mandal, and Weng Kee Wong. "d-QPSO: A Quantum-Behaved Particle Swarm Technique for Finding D-Optimal Designs with Discrete and Continuous Factors and a Binary Response." *Technometrics*, pp. 1–27, 2018.

[LUM16] Giulia Lestini, Sebastian Ueckert, and France Mentré. "Robust design in model based analysis of longitudinal clinical data. Presented at PODE meeting, June 20, Uppsala, Sweden." 2016.

[LWJ12] Yaolin Liu, Hua Wang, Yingli Ji, Zhongqiu Liu, and Xiang Zhao. "Land use zoning at the county level based on a multi-objective particle swarm optimization algorithm: A case study from Yicheng, China." *International journal of environmental research and public health*, **9**(8):2801–2826, 2012.

[LWV14] AV Lebedev, Eric Westman, GJP Van Westen, MG Kramberger, Arvid Lundervold, Dag Aarsland, H Soininen, I Kłoszewska, P Mecocci, M Tsolaki, et al. “Random Forest ensembles for detection and prediction of Alzheimer’s disease with a good between-cohort robustness.” *NeuroImage: Clinical*, **6**:115–125, 2014.

[LY05] Huan Liu and Lei Yu. “Toward integrating feature selection algorithms for classification and clustering.” *IEEE Transactions on knowledge and data engineering*, **17**(4):491–502, 2005.

[LY12] Min-Shyang Leu and Ming-Feng Yeh. “Grey particle swarm optimization.” *Applied Soft Computing*, **12**(9):2985–2996, 2012.

[May88] Ernst Mayr. *Toward a new philosophy of biology: Observations of an evolutionist.* Number 211. Harvard University Press, 1988.

[ME06] France Mentré and Sylvie Escolano. “Prediction discrepancies for the evaluation of nonlinear mixed-effects models.” *Journal of pharmacokinetics and pharmacodynamics*, **33**(3):345–367, 2006.

[MHS95] Michael F McNitt-Gray, HK Huang, and James W Sayre. “Feature selection in the pattern classification problem of digital chest radiograph segmentation.” *IEEE Transactions on Medical Imaging*, **14**(3):537–547, 1995.

[Mit74] Toby J Mitchell. “An algorithm for the construction of “D-optimal” experimental designs.” *Technometrics*, **16**(2):203–210, 1974.

[MMB97] F Mentré, A Mallet, and D Baccar. “Optimal design in random effect regression models.” *Biometrika*, **84**:429–442, 1997.

[MN95] Ruth K Meyer and Christopher J Nachtsheim. “The coordinate-exchange algorithm for constructing exact optimal experimental designs.” *Technometrics*, **37**(1):60–69, 1995.

[Mot15] Yuichi Motai. “Kernel association for classification and prediction: A survey.” *IEEE transactions on neural networks and learning systems*, **26**(2):208–223, 2015.

[MPB12] Marco Mura, Maria A Porretta, Elena Bargagli, Gianluigi Sergiacomi, Maurizio Zompatori, Nicola Sverzellati, Amedeo Taglieri, Fabrizio Mezzasalma, Paola Rottoli, Cesare Saltini, et al. “Predicting survival in newly diagnosed idiopathic pulmonary fibrosis: a 3-year prospective study.” *European Respiratory Journal*, **40**(1):101–109, 2012.

[MPD15] Hongmei Mi, Caroline Petitjean, Bernard Dubray, Pierre Vera, and Su Ruan. “Robust feature selection to predict tumor treatment outcome.” *Artificial intelligence in medicine*, **64**(3):195–204, 2015.

[MRR13] Abolfazl Mehranian, Hamidreza Saligheh Rad, Arman Rahmim, Mohammad Reza Ay, and Habib Zaidi. “Smoothly clipped absolute deviation (SCAD) regularization for compressed sensing MRI using an augmented Lagrangian scheme.” *Magnetic resonance imaging*, **31**(8):1399–1411, 2013.

[MS10] T Mielke and R Schwabe. "Some considerations on the Fisher information in nonlinear mixed effects models." In *Proceedings of the 9th International Workshop in Model-Oriented Design and Analysis*, Bertinoro, Italy, 2010.

[MTS14] Benson Mwangi, Tian Siva Tian, and Jair C Soares. "A review of feature reduction techniques in neuroimaging." *Neuroinformatics*, **12**(2):229–244, 2014.

[MU12] D. R. Mould and R. N. Upton. "Basic concepts in population modeling, simulation, and model-based drug development." *CPT: Pharmacometrics & Systems Pharmacology*, **1**:e6, 2012.

[MU13] D. R. Mould and R. N. Upton. "Basic concepts in population modeling, simulation, and model-based drug development - Part 2: Introduction to Pharmacokinetic Modeling Methods." *CPT: Pharmacometrics & Systems Pharmacology*, **2**:e38, 2013.

[MU14] D. R. Mould and R. N. Upton. "Basic concepts in population modeling, simulation, and model-based drug development - Part 2: Introduction to Pharmacodynamic Modeling Methods." *CPT: Pharmacometrics & Systems Pharmacology*, **3**:e88, 2014.

[MWY15] Abhyuday Mandal, Weng Kee Wong, and Yaming Yu. "Algorithmic searches for optimal designs." *Handbook of Design and Analysis of Experiments*, pp. 755–783, 2015.

[MZA09] Ilias Maglogiannis, Elias Zafiropoulos, and Ioannis Anagnostopoulos. "An intelligent system for automated breast cancer diagnosis and prognosis using SVM based classifiers." *Applied intelligence*, **30**(1):24–36, 2009.

[NBM12] Thu Thuy Nguyen, Caroline Bazzoli, and France Mentré. "Design evaluation and optimisation in crossover pharmacokinetic studies analysed by nonlinear mixed effects models." *Statistics in Medicine*, **31**(11-12):1043–1058, 2012.

[NBO14] J Nyberg, C Bazzoli, K Ogungbenro, A Aliev, S Leonov, S Duffull, AC Hooker, and F Mentré. "Methods and software tools for design evaluation for population pharmacokinetics-pharmacodynamics studies." *British Journal of Clinical Pharmacology*, **79**:6–17, 2014.

[NES11] Ahmad Nickabadi, Mohammad Mehdi Ebadzadeh, and Reza Safabakhsh. "A novel particle swarm optimization algorithm with adaptive inertia weight." *Applied Soft Computing*, **11**(4):3658–3670, 2011.

[NKH09] Joakim Nyberg, Mats O Karlsson, and Andrew C Hooker. "Population optimal experimental design for discrete type data." *PAGE (Population Approach Group Europe) 18, Abstr*, **1468**, 2009.

[NLS18] Thu Thuy Nguyen, Florence Loingeville, Jérémy Seurat, Marie Karelle Riviere, and France Mentré. "Using Hamiltonian Monte-Carlo to design longitudinal studies with discrete outcomes accounting for parameter and model uncertainties." 2018.

[NM65] J.A. Nelder and R. Mead. "A Simplex method for function minimization." *Computer Journal*, **1**:308–313, 1965.

[NM14a] Thu Thuy Nguyen and France Mentré. "Evaluation of the Fisher information matrix in nonlinear mixed effect models using adaptive Gaussian quadrature." *Computational Statistics & Data Analysis*, **80**:57–69, 2014.

[NM14b] Thu Thuy Nguyen and France Mentré. "Evaluation of the Fisher information matrix in nonlinear mixed effect models using adaptive Gaussian quadrature." *Computational Statistics & Data Analysis*, **80**:57–69, 2014.

[NXA17] Hoai Bach Nguyen, Bing Xue, Peter Andreae, and Mengjie Zhang. "Particle Swarm Optimisation with genetic operators for feature selection." In *Evolutionary Computation (CEC), 2017 IEEE Congress on*, pp. 286–293. IEEE, 2017.

[OA11] Kayode Ogungbenro and Leon Aarons. "Population Fisher information matrix and optimal design of discrete data responses in population pharmacodynamic experiments." *Journal of pharmacokinetics and pharmacodynamics*, **38**(4):449–469, 2011.

[OGG05] Kayode Ogungbenro, Gordon Graham, Ivelina Gueorguieva, and Leon Aarons. "The use of a modified Fedorov exchange algorithm to optimise sampling times for population pharmacokinetic experiments." *Computer methods and programs in biomedicine*, **80**(2):115–125, 2005.

[OM99] Ender Ozcan and Chilukuri K Mohan. "Particle swarm optimization: surfing the waves." In *Evolutionary Computation, 1999. CEC 99. Proceedings of the 1999 Congress on*, volume 3, pp. 1939–1944. IEEE, 1999.

[OTB02] Mario JNM Ouwens, Prans ES Tan, and Martijn PF Berger. "Maximin D-optimal designs for longitudinal mixed effects models." *Biometrics*, **58**(4):735–741, 2002.

[Paz86] Andrej Pázman. *Foundations of optimum experimental design*, volume 14. Springer, 1986.

[PCW16] Frederick Kin Hing Phoa, Ray Bing Chen, Weichung Wang, and Weng Kee Wong. "Optimizing two-level supersaturated designs using swarm intelligence techniques." *Technometrics*, **58**(1):43–49, 2016.

[PLD05] Hanchuan Peng, Fuhui Long, and Chris Ding. "Feature selection based on mutual information criteria of max-dependency, max-relevance, and min-redundancy." *IEEE Transactions on pattern analysis and machine intelligence*, **27**(8):1226–1238, 2005.

[Plu14] Matthew Plumlee. "Fast prediction of deterministic functions using sparse grid experimental designs." *Journal of the American Statistical Association*, **109**(508):1581–1591, 2014.

[PNM96] Alan S Perelson, Avidan U Neumann, Martin Markowitz, Johm M Leonard, and David D Ho. “HIV-1 dynamics in vivo: virion clearance rate, infected cell lifespan, and viral generation time.” *Science*, **271**(5255):1582, 1996.

[PP13] Luc Pronzato and Andrej Pázman. *Design of Experiments in Nonlinear Models*. Springer, New York, 2013.

[PPM14] Susan N Partington, Vasil Papakroni, and Tim Menzies. “Optimizing data collection for public health decisions: a data mining approach.” *BMC public health*, **14**(1):593, 2014.

[PS16] Maryna Prus and Rainer Schwabe. “Interpolation and Extrapolation in Random Coefficient Regression Models: Optimal Design for Prediction.” In *mODa 11-Advances in Model-Oriented Design and Analysis*, pp. 209–216. Springer, 2016.

[QCW14] Jiaheng Qiu, Ray-Bing Chen, Weichung Wang, and Weng Kee Wong. “Using animal instincts to design efficient biomedical studies via particle swarm optimization.” *Swarm and evolutionary computation*, **18**:1–10, 2014.

[Qiu14] Jiaheng Qiu. *Finding Optimal Experimental Designs for Models in Biomedical Studies via Particle Swarm Optimization*. PhD thesis, UCLA, 2014.

[RBH06] Philip J. Rosenfeld, David M. Brown, Jeffrey S. Heier, David S. Boyer, Peter K. Kaiser, Carol Y. Chung, and Robert Y. Kim. “Ranibizumab for Neovascular Age-Related Macular Degeneration.” *New England Journal of Medicine*, **355**(14):1419–1431, October 2006.

[RC11] Antonio Bolufé Röhler and Stephen Chen. “An analysis of sub-swarms in multi-swarm systems.” In *Australasian Joint Conference on Artificial Intelligence*, pp. 271–280. Springer, 2011.

[RCE11] Ganesh Raghu, Harold R Collard, Jim J Egan, Fernando J Martinez, Juergen Behr, Kevin K Brown, Thomas V Colby, Jean-François Cordier, Kevin R Flaherty, Joseph A Lasky, et al. “An official ATS/ERS/JRS/ALAT statement: idiopathic pulmonary fibrosis: evidence-based guidelines for diagnosis and management.” *American journal of respiratory and critical care medicine*, **183**(6):788–824, 2011.

[RCY14] Ganesh Raghu, Shih Yin Chen, Wei Shi Yeh, Brad Maroni, Qian Li, Yuan Chi Lee, and Harold R Collard. “Idiopathic pulmonary fibrosis in US Medicare beneficiaries aged 65 years and older: incidence, prevalence, and survival, 2001–11.” *The Lancet Respiratory Medicine*, **2**(7):566–572, 2014.

[RDC17] Hasti Robbie, Cécile Daccord, Felix Chua, and Anand Devaraj. “Evaluating disease severity in idiopathic pulmonary fibrosis.” *European Respiratory Review*, **26**(145):170051, 2017.

[RDM16] Elizabeth G Ryan, Christopher C Drovandi, James M McGree, and Anthony N Pettitt. “A review of modern computational algorithms for Bayesian optimal design.” *International Statistical Review*, **84**(1):128–154, 2016.

[Ret07] Samson A et al Retout S, Comets E. “Design in nonlinear mixed effects models: optimization using the Fedorov-Wynn algorithm and power of the Wald test for binary covariates.” **26**:5162–5179, 2007.

[Riv16] Mentré F. Riviere M K, Ueckert S. “An MCMC method for the evaluation of the Fisher information matrix for non-linear mixed effect models.” **17**:737–50, 2016.

[RMB02] Sylvie Retout, France Mentré, and Rene Bruno. “Fisher information matrix for non-linear mixed-effects models: evaluation and application for optimal design of enoxaparin population pharmacokinetics.” *Statistics in medicine*, **21**(18):2623–2639, 2002.

[RRM18] Ganesh Raghu, Martine Remy-Jardin, Jeffrey L Myers, Luca Richeldi, Christopher J Ryerson, David J Lederer, Juergen Behr, Vincent Cottin, Sonye K Danoff, Ferran Morell, et al. “Diagnosis of idiopathic pulmonary fibrosis. An official ATS/ERS/JRS/ALAT clinical practice guideline.” *American journal of respiratory and critical care medicine*, **198**(5):e44–e68, 2018.

[RSD16] Ganesh Raghu, Mary Beth Scholand, João De Andrade, Lisa Lancaster, Yolanda Mageto, Jonathan Goldin, Kevin K Brown, Kevin R Flaherty, Mark Wencel, Jack Wanger, et al. “FG-3019 anti-connective tissue growth factor monoclonal antibody: results of an open-label clinical trial in IPF.” *European Respiratory Journal*, pp. ERJ–01030, 2016.

[RSF12] Caio César Oba Ramos, Andre Nunes de Souza, Alexandre Xavier Falcao, and João Paulo Papa. “New insights on nontechnical losses characterization through evolutionary-based feature selection.” *IEEE Transactions on Power Delivery*, **27**(1):140–146, 2012.

[RUM16] Marie Karelle Riviere, Sebastian Ueckert, and France Mentré. “An MCMC method for the evaluation of the Fisher information matrix for non-linear mixed effect models.” *Biostatistics*, **17**(4):737–750, 2016.

[RWL09] Kenneth G Russell, David C Woods, SM Lewis, and JA Eccleston. “D-optimal designs for Poisson regression models.” *Statistica Sinica*, pp. 721–730, 2009.

[SA14] Omar S Soliman and Eman AboElhamd. “Classification of Diabetes Mellitus using Modified Particle Swarm Optimization and Least Squares Support Vector Machine.” *arXiv preprint arXiv:1405.0549*, 2014.

[SE98a] Yuhui Shi and Russell Eberhart. “A modified particle swarm optimizer.” In *Evolutionary Computation Proceedings, 1998. IEEE World Congress on Computational Intelligence., The 1998 IEEE International Conference on*, pp. 69–73. IEEE, 1998.

[SE98b] Yuhui Shi and Russell C Eberhart. “Parameter selection in particle swarm optimization.” In *International conference on evolutionary programming*, pp. 591–600. Springer, 1998.

[SE99] Yuhui Shi and Russell C Eberhart. "Empirical study of particle swarm optimization." In *Evolutionary computation, 1999. CEC 99. Proceedings of the 1999 congress on*, volume 3, pp. 1945–1950. IEEE, 1999.

[SFX04] Jun Sun, Bin Feng, and Wenbo Xu. "Particle swarm optimization with particles having quantum behavior." In *Evolutionary Computation, 2004. CEC2004. Congress on*, volume 1, pp. 325–331. IEEE, 2004.

[SHB14] Milan Sonka, Vaclav Hlavac, and Roger Boyle. *Image processing, analysis, and machine vision.* Cengage Learning, 2014.

[Sho13] Naum Zuselevich Shor. *Nondifferentiable optimization and polynomial problems*, volume 24. Springer Science & Business Media, 2013.

[SIL07] Yvan Saeys, Iñaki Inza, and Pedro Larrañaga. "A review of feature selection techniques in bioinformatics." *bioinformatics*, **23**(19):2507–2517, 2007.

[SLW11] Jun Sun, Choi-Hong Lai, and Xiao-Jun Wu. *Particle swarm optimisation: classical and quantum perspectives.* CRC Press, 2011.

[SMS12] Ikaro Silva, George Moody, Daniel J Scott, Leo A Celi, and Roger G Mark. "Predicting in-hospital mortality of icu patients: The physionet/computing in cardiology challenge 2012." *Computing in cardiology*, **39**:245, 2012.

[SNMon] J. Seurat, T. T. Nguyen, and F. Mentré. "Robust designs accounting for model uncertainty in longitudinal studies with binary outcomes." *Statistical Methods in Medical Research*, In revision.

[SP97] Rainer Storn and Kenneth Price. "Differential evolution–a simple and efficient heuristic for global optimization over continuous spaces." *Journal of global optimization*, **11**(4):341–359, 1997.

[SRJ13] Jeetendra Bahadur Singh, Vijay Sena Reddy, Soumya Jana, and Swades De. "Assessment of health risk due to PM10 using fuzzy linear membership kriging with particle swarm optimization." In *EnviroInfo*, pp. 887–894, 2013.

[Sug99] Ponnuthurai N Suganthan. "Particle swarm optimiser with neighbourhood operator." In *Evolutionary Computation, 1999. CEC 99. Proceedings of the 1999 Congress on*, volume 3, pp. 1958–1962. IEEE, 1999.

[SW15] Manuel Schmitt and Rolf Wanka. "Particle swarm optimization almost surely finds local optima." *Theoretical Computer Science*, **561**:57–72, 2015.

[SXF07] Jun Sun, Wenbo Xu, Wei Fang, and Zhilei Chai. "Quantum-behaved particle swarm optimization with binary encoding." *Adaptive and Natural Computing Algorithms*, pp. 376–385, 2007.

[Tib96] Robert Tibshirani. "Regression shrinkage and selection via the lasso." *Journal of the Royal Statistical Society. Series B (Methodological)*, pp. 267–288, 1996.

[Tre03] Ioan Cristian Trelea. “The particle swarm optimization algorithm: convergence analysis and parameter selection.” *Information processing letters*, **85**(6):317–325, 2003.

[TSY05] E Ke Tang, Ponnuthurai N Suganthan, and Xin Yao. “Feature selection for microarray data using least squares svm and particle swarm optimization.” In *2005 IEEE Symposium on Computational Intelligence in Bioinformatics and Computational Biology*, pp. 1–8. IEEE, 2005.

[TVT16] Donald P Tashkin, Elizabeth R Volkmann, Chi-Hong Tseng, Hyun J Kim, Jonathan Goldin, Philip Clements, Daniel Furst, Dinesh Khanna, Eric Kleerup, Michael D Roth, et al. “Relationship between quantitative radiographic assessments of interstitial lung disease and physiological and clinical features of systemic sclerosis.” *Annals of the rheumatic diseases*, **75**(2):374–381, 2016.

[UHS99] Renuka Uppaluri, Eric A Hoffman, Milan Sonka, Patrick G Hartley, Gary W Hunninghake, and Geoffrey McLennan. “Computer recognition of regional lung disease patterns.” *American journal of respiratory and critical care medicine*, **160**(2):648–654, 1999.

[UM15] Sebastian Ueckert and France Mentré. “Computation of the Fisher information matrix for discrete nonlinear mixed effect models. 8th International Conference on Computational and Methodological Statistics, December 12, London, UK.” 2015.

[UM17] Sebastian Ueckert and France Mentré. “A new method for evaluation of the Fisher information matrix for discrete mixed effect models using Monte Carlo sampling and adaptive Gaussian quadrature.” *Computational Statistics & Data Analysis*, **111**:203–219, 2017.

[UMC11] Alper Unler, Alper Murat, and Ratna Babu Chinnam. “mr2PSO: A maximum relevance minimum redundancy feature selection method based on swarm intelligence for support vector machine classification.” *Information Sciences*, **181**(20):4625–4641, 2011.

[US18] Johanna Uthoff and Jessica C Sieren. “Information theory optimization based feature selection in breast mammography lesion classification.” In *Biomedical Imaging (ISBI 2018), 2018 IEEE 15th International Symposium on*, pp. 817–821. IEEE, 2018.

[VMF12] Susana M Vieira, Luis F Mendonça, Gonçalo J Farinha, and João MC Sousa. “Metaheuristics for feature selection: application to sepsis outcome prediction.” In *2012 IEEE Congress on Evolutionary Computation*, pp. 1–8. IEEE, 2012.

[Wak96] Jon Wakefield. “The Bayesian analysis of population pharmacokinetic models.” *Journal of the American Statistical Association*, **91**(433):62–75, 1996.

[WCH15] Weng Kee Wong, Ray Bing Chen, Chien Chih Huang, and Weichung Wang. “A modified particle swarm optimization technique for finding optimal designs for mixture models.” *PloS one*, **10**(6):e0124720, 2015.

[WD99] Hulin Wu and A Adam Ding. “Population HIV-1 Dynamics In Vivo: Applicable Models and Inferential Tools for Virological Data from AIDS Clinical Trials.” *Biometrics*, **55**(2):410–418, 1999.

[WKG07] Daniel N Wilke, Schalk Kok, and Albert A Groenwold. “Comparison of linear and classical velocity update rules in particle swarm optimization: Notes on scale and frame invariance.” *International journal for numerical methods in engineering*, **70**(8):985–1008, 2007.

[WLE06] DC Woods, SM Lewis, JA Eccleston, and KG Russell. “Designs for generalized linear models with several variables and model uncertainty.” *Technometrics*, **48**(2):284–292, 2006.

[WM97] David H Wolpert and William G Macready. “No free lunch theorems for optimization.” *IEEE transactions on evolutionary computation*, **1**(1):67–82, 1997.

[WMN15] Caisheng Wang, Carol J Miller, M Hashem Nehrir, John W Sheppard, and Shawn P McElmurry. “A load profile management integrated power dispatch using a Newton-like particle swarm optimization method.” *Sustainable Computing: Informatics and Systems*, **8**:8–17, 2015.

[WMS06] Yanping Wang, Raymond H Myers, Eric P Smith, and Keying Ye. “D-optimal designs for Poisson regression models.” *Journal of statistical planning and inference*, **136**(8):2831–2845, 2006.

[XSX08] Maolong Xi, Jun Sun, and Wenbo Xu. “An improved quantum-behaved particle swarm optimization algorithm with weighted mean best position.” *Applied Mathematics and Computation*, **205**(2):751–759, 2008.

[XZB12] Bing Xue, Mengjie Zhang, and Will N Browne. “New fitness functions in binary particle swarm optimisation for feature selection.” In *2012 IEEE Congress on Evolutionary Computation*, pp. 1–8. IEEE, 2012.

[XZB14] Bing Xue, Mengjie Zhang, and Will N Browne. “Particle swarm optimisation for feature selection in classification: Novel initialisation and updating mechanisms.” *Applied soft computing*, **18**:261–276, 2014.

[XZB16] Bing Xue, Mengjie Zhang, Will N Browne, and Xin Yao. “A survey on evolutionary computation approaches to feature selection.” *IEEE Transactions on Evolutionary Computation*, **20**(4):606–626, 2016.

[Yan10] Xin-She Yang. *Nature-inspired metaheuristic algorithms.* Luniver press, 2010.

[YH98] Jihoon Yang and Vasant Honavar. “Feature subset selection using a genetic algorithm.” In *Feature extraction, construction and selection*, pp. 117–136. Springer, 1998.

[YL06] Ming Yuan and Yi Lin. "Model selection and estimation in regression with grouped variables." *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, **68**(1):49–67, 2006.

[YP13] Heng Yang and Ioannis Patras. "Sieving regression forest votes for facial feature detection in the wild." In *Proceedings of the IEEE International Conference on Computer Vision*, pp. 1936–1943, 2013.

[Yu10] Yaming Yu. "Monotonic convergence of a general algorithm for computing optimal designs." *The Annals of Statistics*, pp. 1593–1606, 2010.

[ZGH15] Yong Zhang, Dunwei Gong, Ying Hu, and Wanqiu Zhang. "Feature selection algorithm based on bare bones particle swarm optimization." *Neurocomputing*, **148**:150–157, 2015.

[ZH05] Chunkai Zhang and Hong Hu. "Using PSO algorithm to evolve an optimum input subset for a SVM in time series forecasting." In *2005 IEEE international conference on systems, man and cybernetics*, volume 4, pp. 3793–3796. IEEE, 2005.

[Zha15] Owen Zhang. "Winning Data Science Competitions Some (hopefully) useful pointers.", 2015.

[ZLB10] Shang-Ming Zhou, Ronan A Lyons, Owen Bodger, Joanne C Demmler, and Mark D Atkinson. "SVM with entropy regularization and particle swarm optimization for identifying children's health and socioeconomic determinants of education attainments using linked datasets." In *Neural Networks (IJCNN), The 2010 International Joint Conference on*, pp. 1–8. IEEE, 2010.

[ZSW17] Ruijie Zhang, Jian Shen, Fushan Wei, Xiong Li, and Arun Kumar Sangaiah. "Medical image classification based on multi-scale non-negative sparse coding." *Artificial intelligence in medicine*, **83**:44–51, 2017.

[ZSX13] Jing Zhao, Jun Sun, and Wenbo Xu. "A binary quantum-behaved particle swarm optimization algorithm with cooperative approach." *International Journal of Computer Science*, **10**(1):112–118, 2013.

[ZY11] Chenggong Zhang and Zhang Yi. "Scale-free fully informed particle swarm optimization algorithm." *Information Sciences*, **181**(20):4550–4568, 2011.

[ZZL09] Zhi-Hui Zhan, Jun Zhang, Yun Li, and Henry Shu-Hung Chung. "Adaptive particle swarm optimization." *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, **39**(6):1362–1381, 2009.