Privacy-Preserving Computations of Predictive Medical Models with Minimax Approximation and Non-Adjacent Form

Jung Hee Cheon[1], Jinhyuck Jeong[1], Joohee Lee[1], and Keewoo Lee[1(✉)]

[1] Seoul National University (SNU), Seoul, Republic of Korea
{jhcheon,wlsyrlekd,skfro6360,activecondor}@snu.ac.kr

Abstract. In 2014, Bos et al. introduced a cloud service scenario to provide private predictive analyses on encrypted medical data, and gave a proof of concept implementation by utilizing homomorphic encryption (HE) scheme. In their implementation, they needed to approximate an analytic predictive model to a polynomial, using Taylor approximations. However, their approach could not reach a satisfactory compromise so that they just restricted the pool of data to guarantee suitable accuracy. In this paper, we suggest and implement a new efficient approach to provide the service using minimax approximation and Non-Adjacent Form (NAF) encoding. With our method, it is possible to remove the limitation of input range and reduce maximum errors, allowing faster analyses than the previous work. Moreover, we prove that the NAF encoding allows us to use more efficient parameters than the binary encoding used in the previous work or balaced base-B encoding. For comparison with the previous work, we present implementation results using HElib. Our implementation gives a prediction with 7-bit precision (of maximal error 0.0044) for having a heart attack, and makes the prediction in 0.5 seconds on a single laptop. We also implement the private healthcare service analyzing a Cox Proportional Hazard Model for the first time.
Keywords: Homomorphic Encryption, Healthcare, Predictive Analysis, Minimax Approximation, Non-Adjacent Form, Cloud Service

1 Introduction

The cloud computing paradigm provides promising scenarios for user-friendly healthcare services. Patient-to-Cloud healthcare scenario, which enables patients to self-check the hazards of having particular diseases, is one of the clear-eyed scenarios. To protect the crucial medical data, the scenario consists of the following procedures. At first, a user who needs predictive healthcare services feeds personal device with private health data such as age, sex, and ECG, where the device here is meant to be a smart device connected to networks via wireless protocols, e.g. smartphone and smartwatch. The device encrypts the inputs with the secret key stored in it, and sends them to a cloud server. After receiving the encrypted data from the device, The cloud server calculates an exposure risk of some disease with a predictive model on the encrypted data and sends the encrypted result back to the device. Then the device decrypts it with the secret key, providing an output to the user on its screen.

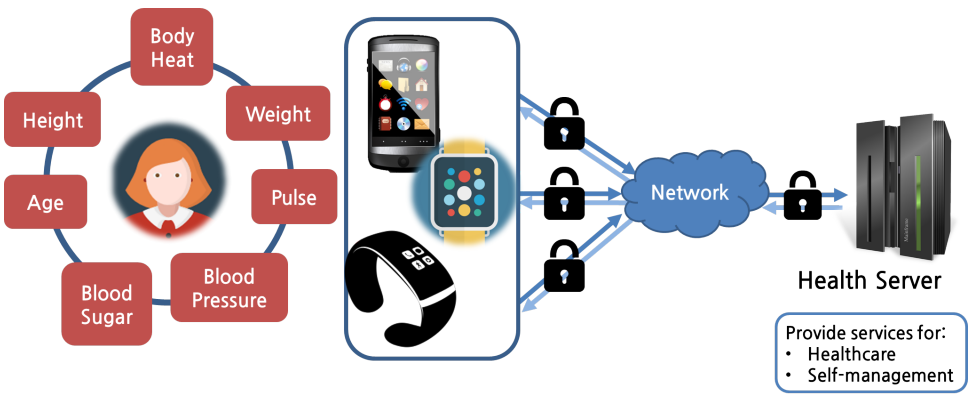

Fig. 1: Patient-to-Cloud Scenario

In this scenario, standard encryption schemes have a restriction: it is impossible to perform outsourced computational tasks on the encrypted data. To perform computations for the encrypted data, the data must be decrypted first and some information might be leaked to adversaries. For this obstacle, homomorphic encryption (HE) can be a solution allowing computations on encrypted data without decryption process. Thus, a secure cloud service can be realized by using a secure HE scheme.

Recently, Bos et al. [BLN14] implemented the private healthcare services using a HE scheme. They implemented a privacy-preserving cloud service providing the likelihood to have a heart attack based on the predictive model called Logistic Regression Model (LRM) [Cox58,LRM]. They used a scale-invariant leveled HE scheme which is a practical variant of YASHE [BLLN13] and used Taylor approximations.[1] However, to guarantee the accuracy, the inputs are required to be close to the expanding point 0 as one can see in Section 3.1.

In this paper, we suggest and implement another approach to provide a private predictive analysis. We use an optimized polynomial approximation method, called minimax approximation. That is to say, the minimax approximation is the optimal approximation in the sense of supremum norm if the range of input values is a bounded interval. Moreover, we employ the Non-Adjacent Form (NAF) encoding and present a proof that this gives us more efficient parameters theoretically than the binary encoding in [BLN14] or the balanced-base B encoding used in [DGBL$^+$15,CSVW]. As a result, our method allows removing the limitation of input range, reducing maximum errors on overall input range and providing faster analyses. In our implementation, we use the open library called HElib [HS13,HS14] based on the leveled HE scheme suggested by Brakerski et al. [BGV12]. Our implementation results of private predictive analysis compared with the previous result are summarized in the Table 1. Homomorphic evaluation of desired prediction for having a heart attack based on LRM would take 0.5 seconds which is about 50 times faster than the previous result ($>$ 30 seconds) using YASHE and degree 7 polynomial. We also put the service based on the Cox Proportional Hazard Model (CPHM) into practice and it permits us to analyze various diseases such as diabetes. For CPHM, it takes 2.2 seconds to analyze the risk of general cardiovascular disease.

Predictive Model	Logistic Model			Cox Model
Approach	BLN14	Our		Our
HE Scheme	YASHE	BGV&HElib		BGV&HElib
Encoding Method	Binary	NAF		NAF
Approximation Method	Taylor	Minimax		Minimax
Degree of Polynomial	7	7	5	7
Range of x	[-3.7,2.4]			[-3.6,5.7]
Range for Maximum Error to be 0.01	[-2.1,2.1]	[-3.7,2.4]		[-3.6,5.7]
Maximum Error	1.163	0.0010	0.0044	0.0095
Server Time (s)	>30	1.8	0.5	2.2
Client Time (s)		1.2	0.5	1.1

Table 1: Summary of our work, where the security parameter $\lambda = 80$.[2]

Organization. In Section 2, first we introduce two predictive models, LRM and CPHM, that we mainly considered. Then we explain how we approximate the models to polynomials in Section 3. In Section 4, we present our methods to evaluate the approximation with encrypted input values, using HE scheme. We also provide our implementation results in the same section.

Notation Throughout this paper, we use the following notations.

[1] Since only integer operations (addition and multiplication) are provided by the HE scheme, they needed to approximate the model to a polynomial which can be computed only by addition and multiplication.

[2] In the case of [BLN14], it is probably not 80-bit secure against modern attacks [Alb17].

- We use P_n to denote the set of polynomials with real number coefficients of degree equal or less than n.
- We use $C[a, b]$ to denote the set of continuous functions on $[a, b]$.
- For function $f \in C[a, b]$,
$$\|f\| := max\{|f(x)| : x \in [a, b]\}.$$

2 Models for Predictive Analysis in Healthcare Services

Many mathematical models to perform predictive analysis in healthcare have been suggested and studied for several decades. For example, one can use a statistical technique called regression such as the logistic regression model or some survival model for some disease such as the Cox proportional hazard model [Cox92]. In this section, we bring two such predictive models into focus: the logistic regression model and the Cox proportional hazard model.

2.1 The Logistic Regression Model

The logistic regression model is used to assess severity of a patient or to predict whether a patient will have a given disease based on observed characteristics of the patient (e.g. age, sex, body mass index, results of various blood tests, etc.). For example, Boyd et al. developed the Trauma and Injury Severity Score (TRISS) method [BTC87], which is widely used to predict mortality in injured patients. Some works [BRD$^+$00,KEAS00] used this model to predict mortality in patients with peritonitis, and Blankstein et al. [BWA$^+$05] proposed a predictor of mortality after certain types of heart surgery. Moreover, Tabaei and Herman [TH02] provided a method to use the model for a prediction of incident diabetes. The logistic regression model has been also used to analyze cardiovascular diseases [TCK67,DVP$^+$08,DPMC13,BSJ$^+$05].

To demonstrate logistic regression analysis, previous work [BLN14] used the following model for men[3] [LRM] to predict the possibility to have a heart attack for an individual. We would also adopt this model for our predictive healthcare services. The model is precisely described as follows: for given six inputs consisting of observed characteristics of a patient, age (a), systolic blood pressure (sys), diastolic blood pressure (dia), cholesterol level ($chol$), height (ht, inches), and weight (wt, pounds), the model provides the likelihood in an interval $[0, 1]$ to have a heart attack by calculating

$$L(\mathbf{x}) = \frac{e^{\mathbf{x}}}{e^{\mathbf{x}} + 1},$$

where $\mathbf{x}$ is the sum of the variables weighted by the logistic regression coefficients as

$$\mathbf{x} = 0.072 \cdot a + 0.013 \cdot sys - 0.029 \cdot dia + 0.008 \cdot chol - 0.053 \cdot ht + 0.021 \cdot wt.$$

We note that the range of $\mathbf{x}$ in the regression data is the interval [-3.755,2.403] [LRM].

2.2 The Cox Proportional Hazard Model

The Cox proportional hazard model suggested by Cox [Cox72] is a well-known procedure for analyzing the time-to-event curve. This model has been the most widely used model over many years in medical research because of its applicability to a wide variety of types of clinical studies [CO84]. For an application, it provides a general methodology for the statistical analysis of relationship between the survival of a patient and several explanatory variables such as age, gender, weight, height, etc. For example, [AYDA$^+$14] estimated the association between treatments and the survival times of breast cancer patients using the Cox model. Moreover, [TS14] also used this model for analyzing the tuberculosis, which is a chronic infectious disease and mainly caused by mycobacterium tuberculosis.

[3] measured 200 male patients, over an observation period which remains unspecific.

D'Agostino et al. [DVP+08] provided the following models analyzing the risk of general cardiovascular disease (CVD), where the population of interest consists of individuals 30-74 years old and without CVD at the baseline examination. Precisely, the model is described as follows. This model assesses the 10-year risk of general CVD. There are six predictive variables: age (A), cholesterol level ($Chol$), HDL cholesterol level (HDL), systolic blood pressure (SBP), smoker ($S = 1$) or not ($S = 0$), and having diabetes ($D = 1$) or not ($D = 0$). The model is given by

$$C(\mathbf{x}) = 1 - 0.95012^{\exp(\mathbf{x})} \quad \text{for women}^4$$

where $\mathbf{x}$ is computed from the variables as

$$\begin{aligned}\mathbf{x} = {} & 2.32888 \cdot \log(A) + 1.20904 \cdot \log(Chol) - 0.70833 \cdot \log(HDL) \\ & +2.76157 \cdot \log(SBP) + 0.52873 \cdot S + 0.69154 \cdot D - 26.1931.\end{aligned}$$

We set the range of the parameters manually as below. We followed the range of parameters which was used in the risk score calculator of Framingham Heart Study website [FHS].

- Age : 30−74
- Cholesterol : 100−405
- HDL : 10−100
- SBP : 90−200

The range of parameters give the range of $\mathbf{x}$ defined above, which is the interval [-3.6,5.7].

3 Polynomial Approximation of Analytic Functions

In our scenario, two models $L(\mathbf{x})$ and $C(\mathbf{x})$ will be approximated by polynomials, since our HE scheme only allows addition and multiplication of integers. To measure the reliability of outputs, errors will be computed in the sense of supremum norm.

In Section 3.1, we analyze the approach of [BLN14] which used the Taylor approximation. Moreover, we assert that using minimax approximation gives optimal error in Section 3.2.

3.1 Taylor Approximation in Previous Works

Bos et al. [BLN14] suggested using Taylor approximation for approximating predictive models to polynomials. For the logistic model $L(\mathbf{x})$ described in the section 2.1, Bos et al. claimed that truncating Taylor series at point 0 after degree 7 gives roughly 2 digits of accuracy. As they suggested, the approximation is very accurate near the expanding point 0. However, considering the range of input described in Section 2.1, error may become larger than the claimed accuracy at the end of the interval.

The graph in Fig. 2 plots logistic function and its Taylor approximation polynomial of degree 7 at point 0, along the range of input. In the graph, the approximation is very accurate near the point 0 but the error grows rapidly at the rear of the interval. Maximal errors are given at the two endpoints of the interval: 1.16 at the point -3.7 and 0.04 at the point 2.4. The errors are too large to be ignored, regarding that the result of the prediction model is a probability. To achieve 2 digits of accuracy, it is required to restrict the original interval to the interval [-2.1,2.1] which is quite smaller than the original one.

The Table 1 shows the maximum error in the interval [-3.7,2.4] between logistic function and Taylor approximation polynomials for various degrees.[5] As the table shows, increasing degree of approximation polynomial does not guarantee the error to decrease, in the sense of supremum norm. These problems occur because Taylor approximation is a local approximation rather than an approximation specialized for the intervals.

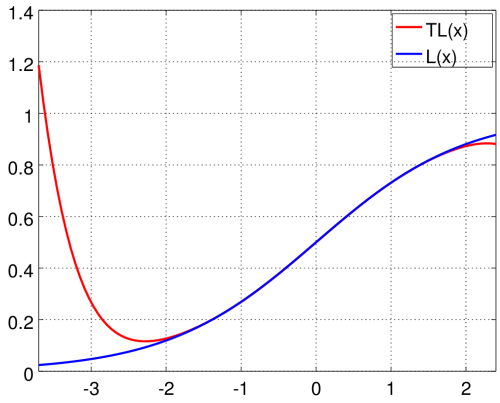

Fig. 2: Taylor Approximation of $L(\mathbf{x})$
Domain : [-3.7,2.4]

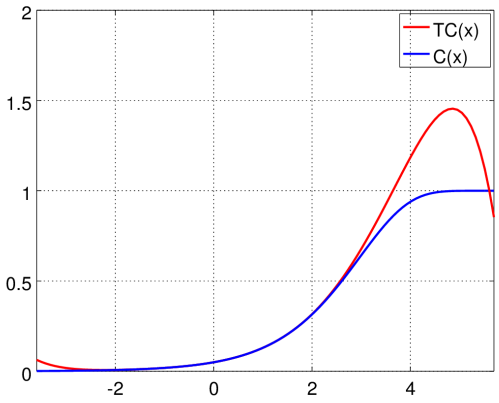

Fig. 3: Taylor Approximation of $C(\mathbf{x})$
Domain : [-3.6,5.7]

Degree of Taylor Polynomial	0	1	3	5	7	9	11	13
Maximum Error	0.476	0.449	0.606	0.839	1.163	1.613	2.237	3.103

Table 2: Taylor Approximation of Logistic Model

Degree of Taylor Polynomial	1	2	3	4	5	6	7	8	9
Expanding Point	4.0	0.2	-1.2	-2.5	-3.6	0.6	0.0	-0.6	-1.2
Maximum Error	0.3619	0.2707	0.3209	0.3497	0.4844	0.4910	0.4562	0.5227	0.6050

Table 3: Taylor Approximation of Cox Model

Similar problems occur in the case of Cox model. The graph in Fig. 3 plots $C(\mathbf{x})$ for women, and its Taylor approximation polynomial of degree 7 at point 0, which is the expanding point that gives minimal maximum error. The graph is drawn along the range of input described in Section 2.1. It can be seen that approximation is very accurate near the expanding point 0.3 but the error grows rapidly at the rear of the interval as in the case of $L(\mathbf{x})$.

The Table 2 shows the maximum error in the interval [-3.6, 5.7] between $C(\mathbf{x})$ and its Taylor approximation polynomials for various degrees. In addition, we give ideal expanding points for each degree. As the table shows, increasing degree of approximation polynomial does not guarantee the error to decrease. The maximum errors are at least larger than 0.2 which is 20%. Regarding that the result of the prediction model is a probability, the maximum errors are too large to be practical as the case of the logistic model.

3.2 Remez Therapy: Adopting Minimax Approximation

In this section, we introduce another polynomial approximation called minimax approximation and describe how it settles the problems of Taylor approximation.

Minimax Approximation and Remez Algorithm In this subsection, we present a brief explanation of minimax approximation and how to find it. For more details, see [Fra65].

Definition 1. *We say that $p \in P_n$ is an n-th minimax approximation of $f \in C[a,b]$ if*

$$\|f - p\| = inf\{\|f - q\| : q \in P_n\}.$$

[4] $C(\mathbf{x}) = 1 - 0.88936^{\exp(\mathbf{x}-23.9802)}$ for men

[5] We only give errors for odd degree polynomials, since in Taylor expansion of logistic function, constant and odd degree terms only appear. This is because the logistic function is a odd function up to a constant.

The name, minimax approximation, comes from the fact that it minimizes the maximum error over all $q \in P_n$. For the proof of its existence and uniqueness, see [Ach13]. Now, we consider a lemma, which is a key idea lying in the Remez algorithm [Rem34] to find the minimax approximation of a given polynomial. For the proof of this lemma, see [Ach13].

Definition 2. *A function $f \in C[a,b]$ is said to equioscillate on n points of $[a,b]$ if there exists n points $a \le x_1 < \cdots < x_n \le b$ such that*

$$|f(x_i)| = \|f\|, \quad i = 1, \cdots, n,$$

and

$$f(x_i) = -f(x_{i+1}), \quad i = 1, \cdots, n-1.$$

Lemma 1 ([Ach13]). *Let $f \in C[a,b]$ and $p \in P_n$. Then, p is an n-th minimax approximation for f on $[a,b]$ if and only if $(f-p)$ equioscillates on $n+2$ points of $[a,b]$.*

Now, we briefly describe the Remez algorithm. For given $(n+2)$ nodes, it repeats to interpolate given function with oscillating error and update nodes to make the difference between the maximum error and the oscillating error smaller. It is known that the Remez algorithm always terminates regardless of the initial choice of the set of nodes [NP51], and the rate of convergence is quadratic [Vei60].[6] However, it is recommended to use *Chebyshev nodes* as an initial choice for making the convergence faster. The Chebyshev nodes of degree n for the interval $[a,b]$ is defined by

$$\frac{1}{2}(a+b) + \frac{1}{2}(b-a)\cos\left(\frac{2k-1}{2n}\right), \ k = 1, \cdots, n.$$

The reason that the Chebyshev nodes are good for initial choice comes from the following lemma which implies the polynomial interpolated at Chebyshev nodes, called Chebyshev approximation, is a near-minimax approximation. For the proof and detailed discussion, see [Riv90].

Lemma 2 ([Riv90]). *Let $f \in C[a,b]$ and Mf and Cf be the n-th minimax approximation and the n-th Chebyshev approximation of f, repectively. Then, the following inequality holds.*

$$\|f - Cf\| < \left(2 + \frac{2}{\pi}\ln(n+1)\right)\|f - Mf\|.$$

How Remez Therapy Works The minimax approximation resolves problems of Taylor approximation mentioned in 3.1. The graph in Fig. 4 plots logistic function $L(x)$ and its 7-th minimax approximation $ML(x)$ for the interval [-3.8, 2.5]. The approximation is accurate throughout the whole interval and the error is much smaller than the Taylor approximation.

The Table 4 shows the maximum error between $L(x)$ and the minimax approximations for various degrees. The even degree coefficient in minimax approximation of $L(x)$ for symmetric interval is zero, since the logistic function is an odd function up to a constant. This makes reducing multiplications possible in the implementation. In spite that expanding the interval grows the maximum error, since the multiplication is expensive operation in our scheme it is worth to expand the interval to a symmetric interval. For this reason, We also give the errors of minimax polynomials for the interval [-3.8, 3.8] not only [-3.8, 2.5]. As degree of approximation polynomial increases, the error decreases and is small enough to be practical at not too high degree approximation.

The minimax approximation also settles problems of Taylor approximation for Cox model. The graph in Fig. 5 plots the function $C(\mathbf{x})$ and its minimax polynomial of degree 7 for the interval [-3.6, 5.7]. In the graph, the approximation is accurate throughout the whole interval

[6] Let us denote the maximum error between the function and the minimax approximation by e, and the oscillating error of kth iteration by e_k. The rate of convergence being quadratic means $|e - e_k| = O(|e - e_{k+1}|^2)$

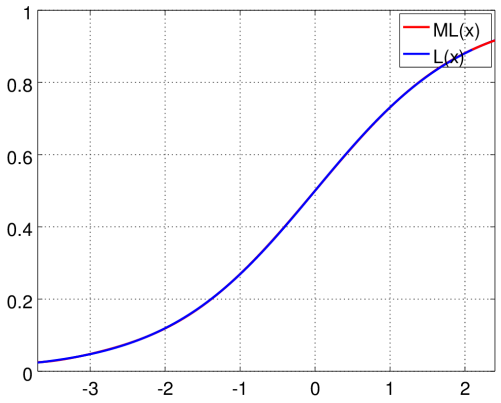

Fig. 4: Minimax Approximation of $L(\mathbf{x})$

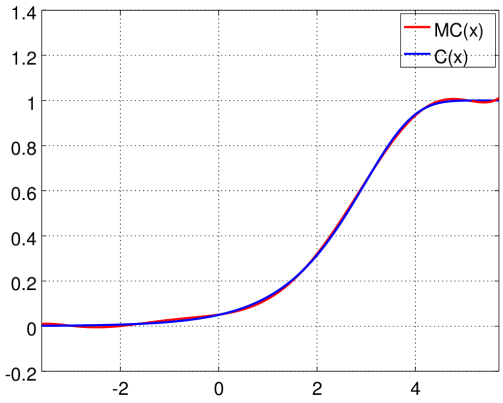

Fig. 5: Minimax Approximation of $C(\mathbf{x})$

Degree of Approximation Polynomial	3	5	7	9	11
Max Error of Minimax on [-3.8, 2.5]	0.0196	0.0039	0.0007	0.0001	0.0000
Max Error of Minimax on [-3.8, 3.8]	0.0198	0.0044	0.0010	0.0002	0.0000
Max Error of Taylor	0.606	0.839	1.163	1.613	2.237

Table 4: Maximum Errors of the Minimax and Taylor Approximations for LRM

Degree of Approximation Polynomial	3	4	5	6	7	8	9
Max Error of Minimax on [-3.6, 5.7]	0.1030	0.0387	0.0386	0.0227	0.0095	0.0091	0.0053
Max Error of Taylor	0.3209	0.3497	0.4844	0.4910	0.4562	0.5227	0.6050

Table 5: Maximum Errors of the Minimax and Taylor Approximations for CPHM

and the error is much smaller than the Taylor approximation. The Table 5 shows the maximum error in the interval [-3.6, 5.7] between the function $C(\mathbf{x})$ and the minimax polynomials for various degrees. It can be seen that, as degree of approximation polynomial increases, the error decreases and be small enough to be practical at not too high degree approximation.

The tables suggest that the maximum errors of minimax approximation are much smaller than the maximum errors of Taylor approximation, for same degree of approximation polynomial and also for any observed degree. This allows us to implement a disease prediction model with a low degree approximation, which will reduce the number of multiplications in the implementation and make the implementation faster as a result. We note that this was done without narrowing the interval as [BLN14] did.

4 Homomorphic Evaluation of Predictive Models

4.1 Practical Homomorphic Encryption

Homomorphic Encryption (HE) is a cryptographic primitive that enables homomorphic operations on encrypted data without decryption procedures. Since Gentry [Gen09a,Gen09b] proposed a blueprint of Fully Homomorphic Encryption(FHE), a plenty of work arose in this area [VDGHV10,CMNT11,CNT12,CCK$^+$13,CLT14,CKLY15]. In 2012, Brakerski, Gentry, and Vaikuntanathan [BGV12] suggested practical variant of leveled FHE scheme based on Ring Learning with Errors (RLWE) problem, which can evaluate L-level arithmetic circuits without bootstrapping. Assembling all of the techniques such as SIMD techniques for the ciphertext bits in [SV14] and bootstrappings in [HS15] to the scheme in [BGV12] with reduced error growths [Bra12], IBM researchers published a software library for HE, which is called HElib [HS13,HS14]. This library is well known to be efficient enough to serve the homomorphic evaluation of AES [GHS12] or fast fourier transformations [CSVW]. In our approach, we also

used HElib to evaluate the exposure risk of a disease securely with the predictive models in Section 2. We remark that we set our parameters not to run bootstrapping in the HElib, since it costs a lot.

We briefly explain the leveled homomorphic encryption scheme of depth L used in HElib here for self-containedness. We set the sequence of moduli for our homomorphic evaluation of depth L by choosing L small primes $p_0, p_1, \cdots, p_{L-1}$ and the t-th modulus in the scheme is defined by $q_t = \prod_{i=0}^{t} p_t$ for $0 \leq t \leq L-1$. We set the ring $\mathbb{Z}_q$ as $(-q/2, q/2) \cap \mathbb{Z}$. Let $\Phi_M(x)$ be a M-th cyclotomic polynomial of degree $\phi(M) = N$, $\mathbb{A}$ be a polynomial ring divided with an ideal generated by the cyclotomic polynomial, and $\mathbb{A}_q = \mathbb{A}/q\mathbb{A}$ for some integer q, i.e. $\mathbb{A}_q = \mathbb{Z}[x]/(\Phi_M(x), q)$.

- $\mathsf{KeyGen}()$: sample $s \in \mathbb{A}_2$ of low hamming weight, $a \leftarrow \mathbb{A}_{q_{L-1}}$ randomly, and e from a discrete Gaussian distribution in $\mathbb{A}_{q_{L-1}}$ with a standard deviation $\sigma > 0$. A public key would be $\mathsf{pk} = (a, b = a \cdot s + 2e)_{q_{L-1}} \in \mathbb{A}^2_{q_{L-1}}$ and a secret key is $\mathsf{sk} = s \in \mathbb{A}_{q_{L-1}}$.
- $\mathsf{Enc}_{\mathsf{pk}}(m \in \mathbb{A}_2)$: choose a small polynomial v with coefficients in $\{-1, 0, 1\}$ and sample Gaussian polynomials e_0, e_1 in the same distribution with that of KeyGen. Let $c_0 = b \cdot v + 2e_0 + m$ and $c_1 = a \cdot v + 2e_1$, where the calculations are held in $\mathbb{A}_{q_{L-1}}$. The ciphertext is $\mathsf{c} = \left(\left(c_0\ c_1\right), L-1, v\right)$ where v is a noise estimate so that it is polynomial of the value $\phi(m)$.
- $\mathsf{Dec}_{\mathsf{sk}}(\mathsf{c})$: For a ciphertext $\mathsf{c} = \left(\left(c_0\ c_1\right), t, v\right)$ at level t, setting $m' \leftarrow (c_0 - s \cdot c_1)_{q_t}$, output $m' \bmod 2$.
- $\mathsf{Add}(\mathsf{c},\mathsf{c}')$: For two ciphertexts $\mathsf{c} = \left(\left(c_0\ c_1\right), t, v\right)$ and $\mathsf{c}' = \left(\left(c'_0\ c'_1\right), t', v'\right)$ of plaintexts m and m' respectively, somehow matching the level of the ciphertexts, simply calculate

$$\mathsf{c}_{\mathsf{add}} = \left(\left(c_0 + c'_0 \quad c_1 + c'_1\right), t'', v + v'\right),$$

  for the new level t''.
- $\mathsf{Mult}(\mathsf{c},\mathsf{c}')$: given $\mathsf{c} = \left(\left(c_0\ c_1\right), t, v\right)$ and $\mathsf{c}' = \left(\left(c'_0\ c'_1\right), t', v'\right)$ for m and m' respectively, let $(d_0, d_1, d_2) \leftarrow (c_0 \cdot c'_0,\ c_1 \cdot c'_0 + c_0 \cdot c'_1,\ -c_1 \cdot c'_1)$. Managing the noise estimate with some techniques and matching the level of the ciphertexts, the ciphertext corresponding to the message $m \cdot m'$ is

$$\mathsf{c}_{\mathsf{mult}} = \mathsf{SwitchKey}((d_0, d_1, d_2),\ t'',\ v \cdot v')$$

  for the new level t'', where $\mathsf{SwitchKey}$ algorithm here basically switches the transformed ciphertext to be decrypted with an original secret key so that it can be decrypted correctly. In this $\mathsf{SwitchKey}$ algorithm, there is an usage of another modulus P which is aimed to boost up the modulus from q_t to $P \cdot q_t$ for time and space efficiency. In other words, the largest modulus used in this library is $P \cdot q_{L-1}$.

We omit all the important details like noise estimating and modulus switching techniques here and just look at how the basic functionality works, so for more details, we recommend to see the Appendix of [GHS09].

4.2 Encoding Strategy

Since the plaintext space of previous homomorphic encryption is a polynomial ring over $\mathbb{Z}_q$, we need encoding and decoding phases for practical use with real numbers in the real world. Proper encoding strategies are needed to guarantee correctness of the results and to not harm the performance of the scheme. In this section, we explain the encoding strategies used in our implementations. For explanation, we divide our encoding phase into two stages: encoding real numbers as integers and encoding integers as polynomials.

Encoding Real Numbers as Integers Encoding real numbers as integers can be done by the method in [BLN14] as following:

1. For each corresponding factors, give precision by rational numbers where denominators are power of 10.
2. Normalize them into integers by multiplying their denominators.
3. Operate homomorphic computations with scaled integers.
4. After decryption, divide the result with 10^n for appropriate n.

Note that, through this encoding technique, some errors might come up from Step 1 for the real value inputs. Therefore, we should take proper denominator for input values, which make the error of output sufficiently small. In our implementations, we take these parameters such that the error generated in the encoding phase is smaller than the error derived from polynomial approximation. For more details, see Section 4.3.

Encoding Integers as Polynomials: Previous Works and Our Approach Since the plaintext space of BGV scheme is a polynomial ring over $\mathbb{Z}_q$, we need to encode integers as polynomials. Choosing an adequate size of plaintext space would be important in this step because of the following two reasons: 1. After some additions and multiplications, coefficients of the polynomials might be reduced by the modulus q and we cannot decode the polynomial correctly. Hence, we need a sufficiently large modulus for correctness. 2. However, the performance of HE heavily relies on the size of the plaintexts, and the larger the modulus of plaintexts, the worse performance becomes. Moreover, some open source libraries may not support such a large modulus. Especially, HElib only supports modulus of long integers (i.e. up to 2^{32}).

One way to maintain small plaintext modulus is to use Chinese Remainder Theorem (CRT) to split the data into multiple smaller moduli. However, this procedure makes the source code more complicated, and since it requires different keys for different moduli, key management starts to disturb users. Another way is suggested in [DGBL$^+$15] and also studied in [CSVW], which utilizes balanced base-B encoding (Bal-B) to make a profit on the size of plaintexts with respect to those of the usual binary encoding as in [BLN14]. We describe their approach briefly, and then introduce a new method to achieve better results.

Definition 3. *For an odd integer B, the balanced base-B encoding of an integer n is $(n_\ell, \cdots, n_0)_{Bal\text{-}B}$, where $n = \sum_{i=0}^{\ell} n_i B^i$ with $n_i \in \{-\frac{B-1}{2}, \cdots, 0, \cdots, \frac{B-1}{2}\}$.*

Definition 4. *For non-negative integers d and e, define $c_{(d,e)}$ as $\|(1 + x + x^2 + \cdots + x^d)^e\|_\infty$*

Theorem 1 ([MR08]). *If either $e \neq 2$ or $d \in \{1, 2, 3\}$, it satisfies*

$$c_{(d,e)} < \sqrt{\frac{6}{\pi \cdot e \cdot d \cdot (d+2)}} \cdot (d+1)^e,$$

and the bound is tight in the sense that

$$\lim_{e\to\infty} \frac{\sqrt{e} \cdot c_{(d,e)}}{(d+1)^e} = \sqrt{\frac{6}{\pi \cdot e \cdot d \cdot (d+2)}}.$$

Definition 5. *Let L, D, A be nonnegative integers. For a given circuit of inputs in $[-L, L]$ which requires depth D for HE and allows A additions per depth, we define $\mathcal{B}_{\mathcal{E}}(L, D, A)$ by the greatest lower bound of modulus with respect to L, D, and A to guarantee correctness for the circuit, where $\mathcal{E}$ denotes the method used for encoding integers to polynomials.*

Theorem 2 ([CSVW]). *Assume there is a circuit we want to compute of depth D with allowed A additions per depth. If $\mathcal{E}$ is standard n-ary encoding or balanced base-B encoding, following equality holds.*

$$\mathcal{B}_{\mathcal{E}}(L, D, A) = c_{(d_{\mathcal{E}}, 2^D)} \cdot m_{\mathcal{E}}^{2^D} \cdot 2^{A(2^{D+1}-2)},$$

where $d_{\mathcal{E}}$ is the maximum number of digits of integers in $[-L, L]$ for $\mathcal{E}$ and $m_{\mathcal{E}}$ is the maximum value of the coefficient for $\mathcal{E}$.[7]

Theorem 2 states that the sufficient bound for correctness can be calculated by the formula, if we are using the standard n-ary encoding or the balanced base-B encoding. Moreover, together with Theorem 1, it can be shown that using balanced base-B encoding decreases the plaintext modulus to achieve correctness by double exponential factor of depth D, compared to the standard binary encoding. However, we still have a problem even if we use balanced base-3 encoding as [CSVW] did since the modulus has to be larger than 2^{32} to guarantee correctness for our models. To improve the result of [CSVW], we suggest using non-adjacent form (NAF) instead of balanced base-3 form.

Definition 6. *The Non-Adjacent Form (NAF) of a integer n is $(n_\ell, \cdots, n_0)_{NAF}$, where $n = \sum_{i=0}^{\ell} n_i 2^i$ with $n_j n_{j-1} = 0$ and $n_j \in \{-1, 0, 1\}$ for all j.*

For example, the NAF of 7 is $(1, 0, 0, -1)_{NAF}$. It is well-known that the NAF of an integer is unique. The following theorem suggests that using NAF is beneficial in general. Second equation says that using NAF instead of balanced base-3 encoding decreases the size of plaintext modulus to achieve correctness by double exponential factor with respect to the depth D. With NAF, we were able to use a plaintext modulus smaller than 2^{32} for the predictive models.

Theorem 3. *Under the same notations as Theorem 1, the followings hold.*

- $\mathcal{B}_{NAF}(L, D, A) = c_{(d_{NAF}, 2^D)} \cdot 2^{A(2^{D+1}-2)}$, $d_{NAF} = \left[\frac{[\log L]+1}{2}\right]$
- $\dfrac{\mathcal{B}_{Bal\text{-}3}(L, D, A))}{\mathcal{B}_{NAF}(L, D, A)} = O((\log 4/\log 3)^{2^D})$

Proof. See Appendix B.

Below, we present the pseudo-code of computing the NAF of an integer. We note that NAF of an integer can be obtained very efficiently.

Algorithm 1 Non-Adjacent Form

Input: n
Output: $m = (f_k, \cdots, f_0)_{NAF}$
Set i=0
while $n > 0$ **do**
1. **if** n is odd :
   $f_i \leftarrow 2 - (n \bmod 4)$
   $n \leftarrow n - f_i$
   else
   $f_i \leftarrow 0$
2. $n \leftarrow n/2$
3. $i \leftarrow i + 1$

return m

[7] The detailed formula can be found in [CSVW].

4.3 Parameter Selection

In this section, we describe the procedure for parameter selection to guarantee security and correctness. At the end of this section we provide the Table 6 consisting of actual parameters we used for implementations.

- Inputs: security parameter λ, predictive model (e.g. $L(\mathbf{x})$ or $C(\mathbf{x})$), and permissible maximum error with respect to the model.
- Output: L, q, M, P and q_t for $0 \leq t \leq L-1$.

1. Set the degree D of minimax approximation for a desired maximum error. For our work, it can be done by taking a glance at Table 4 and 5. For example, if we are concerning $L(\mathbf{x})$ and want to make maximum error be smaller than 0.01, we choose minimax approximation of degree 5. Note that the maximum error will become a bit larger than the error by polynomial approximation, since the error from the encoding process exists.
2. Calculate the suitable input precision R. We need to set the input precision to encode real numbers as integers as described in Section 4.2. The more precise the inputs become, the larger modulus should be. As a consequence, extravagant precisions unnecessarily slow down the performance of implementation. Thus we suggest using similar or a bit smaller maximum error precision for encoding relative to the error by polynomial approximation. For example, the maximum error for 5-th minimax approximation of $L(\mathbf{x})$ is 0.0044. To make the error by encoding to be less than 0.0044, we approximate real value inputs to rationals getting 2 digits of accuracy below the decimal point.
3. Set the proper plaintext modulus q which guarantees security and correctness. If one uses NAF for encoding, one can choose proper q by using bound from Theorem 3. However since the bound in Theorem 3 is for the general circuits with certain properties, it may be inefficient. Therefore, we recommend to analyze the circuit with help of the Corollary 1 in Appendix B. In other words, get a tighter upper bound of maximum coefficient of the results using the bound of the corollary. Let b be the bit size of the maximum coefficient of the results. We can use the smaller modulus q than 2^{b+1}, if the error generated from reducing by modulus is negligible relative to the error by polynomial approximation. For example, since an upper bound for absolute value of coefficients after computations is 2^{24}, we can use modulus $q = 33554467$ for 5-th minimax approximation of $L(\mathbf{x})$.
4. Get proper M with security parameter λ, HElib level L, and modulus q. To obtain the λ-bit security, we set the parameters in HElib so that our scheme is secure against the dual lattice attack by [Alb17] using the estimator [APS15][8]. After finding a proper M, we can use buildModChain function to set the rest of the parameters, q_i and P. For example, using the estimator we can find out that, for modulus $q = 33554467$, level $L = 13$, and security parameter $\lambda = 80$, it is sufficient to use $M = 13217$ for 5-th minimax approximation of $L(\mathbf{x})$.

4.4 Implementation Results

We give Table 7 so that one can see our performance at a glance and choose the parameters for similar applications. The time results are measured by the mean values of times to compute the wanted output for five independently measured input values. This implementation was performed on a laptop (Intel Core i5-3337U at 1.80GHz). Since the computations, in our scenario, are performed by cloud server with high performance, one can expect the time results to be much smaller.

[8] https://bitbucket.org/malb/lwe-estimator/src

	D	R	b	q	L	$\log_2 q_{L-1}$	$\log_2 P$	M
LRM	5	1	24	33554467	13	27	202	13217
	7	2	36	4294967291	17	370	259	17431
	9	3	45	4294967291	21	448	276	20191
	11	4	49	4294967291	25	539	323	23431
CPHM	4	2	20	1048583	8	194	134	9487
	5	2	26	67108879	12	278	192	13483
	6	2	31	2147483659	14	324	235	15943
	7	3	37	4294967291	16	370	259	17431
	8	3	41	4294967291	16	370	259	17431
	9	3	47	4294967291	21	448	276	20191

Table 6: Parameter settings with the security parameter $\lambda = 80$. Column D denotes the degree of minimax approximations, column R denotes the input precision, column b denotes the bit size of maximum coefficient of outputs, and the value $P \cdot q_{L-1}$ is the largest modulus used in the library.

	Logistic Model				Cox Model					
Degree of Approximation Polynomial	5	7	9	11	4	5	6	7	8	9
Maximum Error	0.0044	0.0010	0.0002	0.0000	0.0387	0.0386	0.0227	0.0095	0.0091	0.0053
Encoding & Encryption (ms)	463	1052	1308	1547	333	467	547	1035	1042	1240
Computation (ms)	479	1750	2777	4208	354	708	1099	2188	2630	4209
Decryption & Decoding (ms)	47	114	203	281	32	56	78	110	114	198

Table 7: Performance result

5 Conclusion

In this paper, we introduced the minimax approximation method and suggested it as an option for approximation polynomial of medical analyses with predictive models. This selection makes the analyses more efficient and accurate than the case one choose the Taylor approximation method as in [BLN14]. The previous work [BLN14] choose the Taylor approximation method and YASHE scheme as their option. On the other hand, we choose the minimax approximation method and HElib as our option. Additionally, we utilize the Non-Adjacent Form encoding method. As a result, we can evaluate the medical predictive models much faster than [BLN14] with smaller error as one can see in Table 1. Moreover, with minimax approximation, one can perform accurate analyses using Cox proportional hazard models which is impossible with Taylor approximation.

Acknowledgement This work was supported by Institute for Information & communications Technology Promotion (IITP) grant funded by the Korea government (MSIP) (No. B0717-16-0098). The authors would like to thank Yong Soo Song, Kyoohyung Han, and the anonymous reviewers for valuable comments and suggestions.

References

[Ach13] Naum I Achieser. *Theory of approximation*. Courier Corporation, 2013.

[Alb17] Martin R. Albrecht. On dual lattice attacks against small-secret LWE and parameter choices in HElib and SEAL. Cryptology ePrint Archive, Report 2017/047, 2017. `http://eprint.iacr.org/2017/047`.

[APS15] Martin R Albrecht, Rachel Player, and Sam Scott. On the concrete hardness of learning with errors. *Journal of Mathematical Cryptology*, 9(3):169–203, 2015.

[AYDA+14] Alireza Abadi, Parvin Yavari, Monireh Dehghani-Arani, Hamid Alavi-Majd, Erfan Ghasemi, Farzaneh Amanpour, and Chris Bajdik. Cox models survival analysis based on breast cancer treatments. *Iranian journal of cancer prevention*, 7(3):124, 2014.

[BGV12] Zvika Brakerski, Craig Gentry, and Vinod Vaikuntanathan. (leveled) fully homomorphic encryption without bootstrapping. In *Proceedings of the 3rd Innovations in Theoretical Computer Science Conference*, pages 309–325. ACM, 2012.

[BLLN13] Joppe W Bos, Kristin Lauter, Jake Loftus, and Michael Naehrig. Improved security for a ring-based fully homomorphic encryption scheme. In *IMA International Conference on Cryptography and Coding*, pages 45–64. Springer, 2013.

[BLN14] Joppe W Bos, Kristin Lauter, and Michael Naehrig. Private predictive analysis on encrypted medical data. *Journal of biomedical informatics*, 50:234–243, 2014.

[Bra12] Zvika Brakerski. Fully homomorphic encryption without modulus switching from classical gapsvp. In *Advances in Cryptology–CRYPTO 2012*, pages 868–886. Springer, 2012.

[BRD+00] Sebastiano Biondo, Emilio Ramos, Manuel Deiros, Juan Martı Ragué, Javier De Oca, Pablo Moreno, Leandre Farran, and Eduardo Jaurrieta. Prognostic factors for mortality in left colonic peritonitis: a new scoring system. *Journal of the American College of Surgeons*, 191(6):635–642, 2000.

[BSJ+05] SM Boekholdt, FM Sacks, JW Jukema, J Shepherd, DJ Freeman, AD McMahon, F Cambien, V Nicaud, GJ De Grooth, PJ Talmud, et al. Cholesteryl ester transfer protein taqib variant, high-density lipoprotein cholesterol levels, cardiovascular risk, and efficacy of pravastatin treatment individual patient meta-analysis of 13 677 subjects. *Circulation*, 111(3):278–287, 2005.

[BTC87] Carl R Boyd, Mary Ann Tolson, and Wayne S Copes. Evaluating trauma care: the triss method. *Journal of Trauma and Acute Care Surgery*, 27(4):370–378, 1987.

[BWA+05] Ron Blankstein, R Parker Ward, Morton Arnsdorf, Barbara Jones, You-Bei Lou, and Michael Pine. Female gender is an independent predictor of operative mortality after coronary artery bypass graft surgery contemporary analysis of 31 midwestern hospitals. *Circulation*, 112(9 suppl):I–323, 2005.

[CCK+13] Jung Hee Cheon, Jean-Sébastien Coron, Jinsu Kim, Moon Sung Lee, Tancrede Lepoint, Mehdi Tibouchi, and Aaram Yun. Batch fully homomorphic encryption over the integers. In *Annual International Conference on the Theory and Applications of Cryptographic Techniques*, pages 315–335. Springer, 2013.

[CKLY15] Jung Hee Cheon, Jinsu Kim, Moon Sung Lee, and Aaram Yun. CRT-based fully homomorphic encryption over the integers. *Information Sciences*, 310:149–162, 2015.

[CLT14] J.-S. Coron, T. Lepoint, and M. Tibouchi. Cryptanalysis of two candidate fixes of multilinear maps over the integers. *IACR Cryptology ePrint Archive*, 2014:975, 2014.

[CMNT11] Jean-Sébastien Coron, Avradip Mandal, David Naccache, and Mehdi Tibouchi. Fully homomorphic encryption over the integers with shorter public keys. In *Annual Cryptology Conference*, pages 487–504. Springer, 2011.

[CNT12] Jean-Sébastien Coron, David Naccache, and Mehdi Tibouchi. Public key compression and modulus switching for fully homomorphic encryption over the integers. In *Annual International Conference on the Theory and Applications of Cryptographic Techniques*, pages 446–464. Springer, 2012.

[CO84] David R Cox and David Oakes. *Analysis of survival data*, volume 21. CRC Press, 1984.

[Cox58] David R Cox. The regression analysis of binary sequences. *Journal of the Royal Statistical Society. Series B (Methodological)*, pages 215–242, 1958.

[Cox72] David R Cox. Regression models and life-tables. *Journal of the Royal Statistical Society, Series B. 34 (2): 187–220.*, 1972.

[Cox92] David R Cox. Regression models and life-tables. In *Breakthroughs in statistics*, pages 527–541. Springer, 1992.

[CSVW] Anamaria Costache, Nigel P Smart, Srinivas Vivek, and Adrian Waller. Fixed point arithmetic in SHE schemes. Technical report, Cryptology ePrint Archive, Report 2016/250, 2016. http://eprint.iacr. org/2016/250.

[DGBL+15] Nathan Dowlin, Ran Gilad-Bachrach, Kim Laine, Kristin Lauter, Michael Naehrig, and John Wernsing. Manual for using homomorphic encryption for bioinformatics. *Microsoft Research*, 2015. `http://research.microsoft.com/pubs/258435/ManualHEv2.pdf`.

[DPMC13] Ralph B D'Agostino, Michael J Pencina, Joseph M Massaro, and Sean Coady. Cardiovascular disease risk assessment: insights from Framingham. *Global heart*, 8(1):11–23, 2013.

[DVP+08] Ralph B D'Agostino, Ramachandran S Vasan, Michael J Pencina, Philip A Wolf, Mark Cobain, Joseph M Massaro, and William B Kannel. General cardiovascular risk profile for use in primary care the Framingham heart study. *Circulation*, 117(6):743–753, 2008.

[FHS] `http://www.framinghamheartstudy.org/risk-functions/cardiovascular-disease/10-year-risk.php`.

[Fra65] W Fraser. A survey of methods of computing minimax and near-minimax polynomial approximations for functions of a single independent variable. *Journal of the ACM (JACM)*, 12(3):295–314, 1965.

[Gen09a] Craig Gentry. *A fully homomorphic encryption scheme*. PhD thesis, Stanford University, 2009. `crypto.stanford.edu/craig`.

[Gen09b] Craig Gentry. Fully homomorphic encryption using ideal lattices. In *Proceedings of the 41st annual ACM symposium on Symposium on theory of computing-STOC'09*, pages 169–169. ACM Press, 2009.

[GHS09] Craig Gentry, Shai Halevi, and Nigel P Smart. Homomorphic evaluation of the aes circuit. Cryptology ePrint Archive, Report 2012/099, 2009. `https://eprint.iacr.org/2012/099`.

[GHS12] Craig Gentry, Shai Halevi, and Nigel P Smart. Homomorphic evaluation of the AES circuit. In *Advances in Cryptology–CRYPTO 2012*, pages 850–867. Springer, 2012.

[HS13] Shai Halevi and Victor Shoup. Design and implementation of a homomorphic-encryption library. *IBM Research (Manuscript)*, 2013.

[HS14] Shai Halevi and Victor Shoup. Algorithms in HElib. In *International Cryptology Conference*, pages 554–571. Springer, 2014.

[HS15] Shai Halevi and Victor Shoup. Bootstrapping for HElib. In *Annual International Conference on the Theory and Applications of Cryptographic Techniques*, pages 641–670. Springer, 2015.

[KEAS00] Murat Kologlu, Doruk Elker, Hasan Altun, and Iskender Sayek. Validation of MPI and PIA II in two different groups of patients with secondary peritonitis. *Hepato-gastroenterology*, 48(37):147–151, 2000.

[LRM] `http://www.claudiaflowers.net/rsch8140/logistic_regression_example.htm`.

[MR08] Lutz Mattner and Bero Roos. Maximal probabilities of convolution powers of discrete uniform distributions. *Statistics & Probability Letters*, 78(17):2992–2996, 2008.

[NP51] EP Novodvorskii and Ilia Sh Pinsker. The process of equating maxima. *Uspekhi Matematicheskikh Nauk*, 6(6):174–181, 1951.

[Rem34] Evgeny Y Remez. Sur le calcul effectif des polynomes d'approximation de tschebyscheff. *CR Acad. Sci. Paris*, 199:337–340, 1934.

[Riv90] Theodore-J Rivlin. Chebyshev polynomials. 1990.

[SV14] Nigel P Smart and Frederik Vercauteren. Fully homomorphic SIMD operations. *Designs, codes and cryptography*, 71(1):57–81, 2014.

[TCK67] Jeanne Truett, Jerome Cornfield, and William Kannel. A multivariate analysis of the risk of coronary heart disease in Framingham. *Journal of chronic diseases*, 20(7):511–524, 1967.

[TH02] Bahman P Tabaei and William H Herman. A multivariate logistic regression equation to screen for diabetes development and validation. *Diabetes Care*, 25(11):1999–2003, 2002.

[TS14] Kabtamu Tolosie and MK Sharma. Application of cox proportional hazards model in case of tuberculosis patients in selected addis ababa health centres, ethiopia. *Tuberculosis research and treatment*, 2014, 2014.

[VDGHV10] Marten Van Dijk, Craig Gentry, Shai Halevi, and Vinod Vaikuntanathan. Fully homomorphic encryption over the integers. In *Annual International Conference on the Theory and Applications of Cryptographic Techniques*, pages 24–43. Springer, 2010.

[Vei60] L Veidinger. On the numerical determination of the best approximations in the Chebyshev sense. *Numerische Mathematik*, 2(1):99–105, 1960.

A Approximation Polynomials

In this section, we list the approximation polynomials those have been used in this paper and the implementation.

A.1 Minimax Approximation for Logistic Model

Degree	0th term	1st term	3rd term	5th term	7th term	9th term	11th term
3	0.50000	0.21969	-0.0070164				
5	0.50000	0.24141	-0.013984	0.00042530			
7	0.50000	0.24771	-0.017996	0.0010405	-0.000026488		
9	0.50000	0.24941	-0.019789	0.0015352	-0.000076288	0.0000016561	
11	0.50000	0.24985	-0.020479	0.0018310	-0.00012735	0.0000054811	-0.00000010362

Table 8: Coefficients of Minimax Polynomials for Logistic Model in [-3.7,3.7]

A.2 Minimax Approximation for Cox Model

B Proof of Theorem 3

For $p \in \mathbb{Z}[x]$, we use $\|p\|_\infty$ to denote the maximum of absolute values of coefficients. We use $\mathbb{Z}_+[x]$ to denote the set of polynomials with coefficients of nonnegative integers. Let $p \in \mathbb{Z}_+[x]$

Degree	0th term	1st term	2nd term	3rd term	4th term	5th term	6th term	7th term	8th term	9th term
3	3.974e-2	1.409e-1	3.014e-2	-3.882e-3						
4	1.348e-2	6.502e-2	5.143e-2	3.997e-3	-1.738e-3					
5	1.344e-2	6.457e-2	5.164e-2	4.046e-3	-1.768e-3	3.060e-6				
6	3.266e-2	2.553e-2	4.380e-2	1.603e-2	-2.232e-3	-7.269e-4	9.144e-5			
7	5.096e-2	3.151e-2	2.118e-2	1.602e-2	1.968e-3	-1.088e-3	-1.068e-4	2.689e-5		
8	5.258e-2	3.225e-2	1.828e-2	1.639e-2	2.621e-3	-1.256e-3	-1.326e-4	3.846e-5	-9.284e-7	
9	5.511e-2	4.706e-2	1.048e-2	8.302e-3	6.069e-3	-2.761e-4	-6.022e-4	2.624e-5	1.860e-5	-1.754e-6

Table 9: Coefficients of Minimax Polynomials for Cox Model

be a polynomial of degree n defined by $p(x) = \sum_{i=0}^{n} p_i x^i$. Regarding $p_j = 0$ for all $j \geq n+1$, we define two vector representations of p as follows:

$$R_{std}(p) := (p_0, p_1, p_2, \cdots, p_i, \cdots) \quad \text{and} \quad R_{dec}(p) := (\tilde{p}_0, \tilde{p}_1, \tilde{p}_2, \cdots, \tilde{p}_i, \cdots),$$

where $\{\tilde{p}_i\}$ is the rearrangement of $\{p_i\}$ in decreasing order. $R_{dec}(p)$ is well-defined since p has only finite number of positive terms. For $p, q \in \mathbb{Z}_+[x]$, define a equivalence relation $\sim$ as following.

$$p \sim q \Leftrightarrow R_{dec}(p) = R_{dec}(q)$$

For any polynomial $p(x) = \sum_{i=0}^{n} p_i x^i$, we define $|p| \in \mathbb{Z}_+[x]$ by $|p|(x) = \sum_{i=0}^{n} |p_i| x^i$.

Definition 7 (Λ-shaped). *For $p \in \mathbb{Z}_+[x]$, we give some new definitions below.*

1. *p is Λ-shaped if $R_{std}(p) = (p_0, p_1, p_2, \cdots)$ satisfies the following condition.*
   - *(bisymmetricity) There exists $a \in \mathbb{Z} \cup (\mathbb{Z} + \frac{1}{2})$ such that $p_{\lfloor a+i+\frac{1}{2} \rfloor} = p_{\lceil a-i-\frac{1}{2} \rceil}$ for all $i \leq \lceil a - \frac{1}{2} \rceil$ and $p_i = 0$ for all $i > \lceil a - \frac{1}{2} \rceil$.*
   - *(one-peakness) If $p_i > p_{i+1}$ for some i, then $p_j \geq p_{j+1}$ for all $j \geq i$.*
2. *A polynomial p is potentially Λ-shaped if $p \sim q$ for some Λ-shaped q with nonzero constant term. In this case, we denote this q as $\hat{p}$.*

In other words, $p \in \mathbb{Z}_+[x]$ is Λ-shaped if $R_{std}(p)$ is bisymmetric after erasing some zeros at the end of the sequence and has at most one peak. We present a lemma which asserts that the set of Λ-shaped polynomials in $\mathbb{Z}_+[x]$ is closed for multiplication of polynomials as follows.

Lemma 3. *A finite product of Λ-shaped polynomials is Λ-shaped.*

Proof. It is enough to show for products of two Λ-shaped polynomials. For potentially Λ-shaped polynomials q and r, let $R_{std}^{sym}(\hat{q}) = (\hat{q}_0, \hat{q}_1, \hat{q}_2, \cdots, \hat{q}_n)$ and $R_{std}^{sym}(\hat{r}) = (\hat{r}_0, \hat{r}_1, \hat{r}_2, \cdots, \hat{r}_m)$ be bisymmetric sequences obtained by erasing some zeros at the end of $R_{std}(\hat{q})$ and $R_{std}(\hat{r})$ respectively. Then,

$$R_{std}(\hat{q} \cdot \hat{r}) = (\sum_{i+j=0} \hat{q}_i \hat{r}_j, \sum_{i+j=1} \hat{q}_i \hat{r}_j, \cdots, \sum_{i+j=n+m} \hat{q}_i \hat{r}_j, 0, \cdots).$$

The *bisymmetricity* holds since

$$\sum_{i+j=k} \hat{q}_i \hat{r}_j = \sum_{i+j=k} \hat{q}_{n-i} \hat{r}_{m-j} = \sum_{i+j=n+m-k} \hat{q}_i \hat{r}_j,$$

and the *one-peakness* comes from

$$\sum_{i+j=k} \hat{q}_i \hat{r}_j \leq \sum_{i+j=k} \hat{q}_{i+1} \hat{r}_j \leq \sum_{i+j=k+1} \hat{q}_i \hat{r}_j \text{ for all } k < \frac{n+m}{2}.$$

□

Definition 8. *Define a partial order $\preceq$ on $\mathbb{Z}_+[x]$ as following. For p and $q \in \mathbb{Z}_+[x]$, let $R_{dec}(p) = (p_0, p_1, p_2, \cdots)$ and $R_{dec}(q) = (q_0, q_1, q_2, \cdots)$.*

$$\begin{aligned} p \preceq q &\iff R_{dec}(q) \text{ majorizes } R_{dec}(p). \\ &\iff \sum_{i=0}^{\infty} p_i = \sum_{i=0}^{\infty} q_i \text{ and } \sum_{i=0}^{k} p_i \leq \sum_{i=0}^{k} q_i \text{ for all } k \in \mathbb{N}. \end{aligned}$$

Lemma 4. *If q and r are potentially Λ-shaped,*

$$p \preceq q \implies pr \preceq \hat{q}\hat{r}.$$

Sketch of Proof. Let $R_{std}(p) = (p_0, p_1, p_2, \cdots)$ and $R_{std}(q) = (q_0, q_1, q_2, \cdots)$. For $R_{std}^{sym}(\hat{q})$ and $R_{std}^{sym}(\hat{r})$, let us recycle the notations used in the proof of the Lemma 3. It is enough to show the following inequality holds for all $t \in \mathbb{N}$ and $K \subset \mathbb{N} \cup \{0\}$ with $|K| = t$, denoting K_{n+m} as $\mathbb{Z} \cap \left[\lceil \frac{n+m-t+1}{2} \rceil, \lfloor \frac{n+m+t}{2} \rfloor\right]$ of t elements.

$$\sum_{k \in K} \sum_{i+j=k} p_i r_j \leq \sum_{k \in K_{n+m}} \sum_{i+j=k} \hat{q}_i \hat{r}_j,$$

or equivalently,

$$\sum_{j=0}^{\infty} \left(r_j \sum_{i+j \in K} p_i \right) \leq \sum_{j=0}^{\infty} \left(\hat{r}_j \sum_{i+j \in K_{n+m}} \hat{q}_i \right).$$

Now the proof is completed by the fact that $\left(\sum_{i+j \in K_{n+m}} \hat{q}_i\right)$ majorizes $\left(\sum_{i+j \in K} p_i\right)$ as sequences with index j, which directly comes from the assumption $p \preceq q$. □

Theorem 4. *If p_i's are potentially Λ-shaped,*

$$\prod_{i=1}^{n} p_i \preceq \prod_{i=1}^{n} \hat{p}_i.$$

Proof. Suppose the theorem is true when $n = k - 1$. Then by Lemma 3 and Lemma 4,

$$\prod_{i=1}^{k} p_i = p_k \cdot \prod_{i=1}^{k-1} p_i \preceq \hat{p_k} \cdot \prod_{i=1}^{k-1} \hat{p_i} = \prod_{i=1}^{k} \hat{p_i}.$$

When $n = 1$, it is trivial. By mathematical induction, the theorem is proved. □

Corollary 1. *If p_i's are binary polynomials,*

$$\left\| \prod_{i=1}^{n} p_i \right\|_{\infty} \leq \left\| \prod_{i=1}^{n} \hat{p}_i \right\|_{\infty}.$$

Proof. Directly follows from Theorem 4 and the fact that every binary polynomial is potentially Λ-shaped. □

Theorem 5. *If a NAF polynomial p lies in P_n, the following inequality holds. Furthermore, the bound is sharp.*

$$\|p^e\|_{\infty} \leq c_{([\frac{n+1}{2}], e)}.$$

Proof. We have

$$\|p^e\|_{\infty} \leq \||p|^e\|_{\infty} \leq \|\hat{|p|}^e\|_{\infty} \leq c_{([\frac{n+1}{2}], e)},$$

where the first inequality follows from the triangle inequality and the second inequality comes from Corollary 1. The third inequality follows from the definition of NAF: the number of nonzero terms of NAF polynomial cannot exceed the half of the number of terms. For sharpness, consider the alternating NAF which make the equality holds: $(1010\cdots)_{NAF}$. □

Finally we obtain the first equation of Theorem 3 from Theorem 1 and 1. The second equation is also obtained from simple calculations combining Theorem 1 and the first equation.