Análisis estadístico seguro para ensayos clínicos

Alicia Quirós
Universidad de León
Departamento de Matemáticas
Campus de Vegazana, León, España
alicia.quiros@unileon.es

Diego Simón
Universidad de León
Departamento de Matemáticas
Campus de Vegazana, León, España
dsimog01@estudiantes.unileon.es

Adriana Suárez Corona
Universidad de León
Departamento de Matemáticas
Campus de Vegazana, León, España
asuac@unileon.es

***Resumen*—En este trabajo se plantea la comparación de proporción de eventos adversos para dos grupos de tratamiento, con un enfoque bayesiano, en el contexto de un ensayo clínico en el que participan varios hospitales. Con el objetivo de preservar la confidencialidad de los datos, utilizamos un esquema de cifrado homomórfico, que permite realizar cálculos sobre los datos cifrados. La viabilidad de este diseño se ilustra con el análisis de datos simulados a través de la implementación de un prototipo de sistema cliente-servidor, desarrollado en el lenguaje C++ y capaz de realizar operaciones homomórficas mediante la biblioteca Microsoft SEAL. La incorporación de la inferencia bayesiana presenta ventajas como el procesado de grandes conjuntos de datos, la reducción del tiempo global y la posibilidad de incorporar un diseño adaptativo en ensayos clínicos con compartición de datos segura. Se ha demostrado la viabilidad, mostrándose los tiempos de ejecución.**

***Index Terms*—Cifrado homomórfico, Cuaderno de recogida de datos electrónico, Inferencia bayesiana.**

Tipo de contribución: *Investigación en desarrollo*

I. Introducción

El tratamiento de datos sensibles hace imprescindible contar con mecanismos que garanticen su privacidad, de forma que sólo los usuarios autorizados puedan acceder a los mismos. Las herramientas criptográficas son esenciales en estos casos, pudiendo garantizar, en base a un modelo matemático, que se satisfacen las nociones de seguridad requeridas en cada situación [1]. Esto puede ser crucial cuando los datos obtenidos para poder realizar un diagnóstico médico deben compartirse con otros especialistas que no se encuentren físicamente cerca, cuando el promotor de un estudio requiere que varios investigadores pongan en común sus datos sin desvelar más información de la necesaria para hacer un análisis conjunto o cuando se desea compartir el historial médico de un paciente al ser transferido de un centro a otro.

Cuando el volumen de los datos se hace inmanejable, puede ser necesario el uso de arquitecturas de *cloud computing* que permitan externalizar cálculos sobre estos datos u ofrecer disponibilidad, de forma que cualquier cliente solamente tendrá que conectarse a él para efectuar alguna operación. En estos casos, es conveniente que los datos se cifren antes de enviarlos al servicio *cloud*, de forma que ese tercero no tenga acceso a los datos en claro, estos estén protegidos y se cumpla así con las normativas relativas a la protección de datos.

Para el almacenamiento y la compartición segura de los datos, es suficiente con utilizar esquemas de cifrado clásicos. Sin embargo, si es necesario realizar cálculos sobre los mismos, la herramienta más conveniente es un esquema de cifrado homomórfico, que permite realizar el cálculo sobre los datos cifrados, manteniendo la confidencialidad frente al servidor externo [2].

El contexto de aplicación de este trabajo son los ensayos clínicos en los que participan varios hospitales. De acuerdo con el Real Decreto 1090/2015, cuando se trate de investigación clínica sin ánimo comercial "la propiedad de los datos de la investigación pertenece al promotor" y tiene que acordar con el investigador el tratamiento de datos. El investigador, a su vez, "es el responsable de garantizar la veracidad de los datos" y de "garantizar la confidencialidad acerca de los sujetos del ensayo y la protección de datos de carácter personal".

En concreto, el tipo de ensayos clínicos que conforman el contexto de este trabajo tienen como objetivo principal comparar la proporción de eventos adversos entre dos grupos de tratamiento.

Para abordar el análisis del objetivo principal utilizaremos la inferencia bayesiana con el modelo beta-binomial [3]. La inferencia bayesiana resuelve el problema de comparación de proporciones incluso cuando no se observa ningún caso para alguno de los posibles resultados en algún grupo de tratamiento, gracias a que puede incorporar información experta o asumir ignorancia a priori sobre las proporciones. En este último caso, el resultado obtenido es equivalente al proporcionado por la inferencia frecuentista. La inferencia bayesiana, además, proporciona resultados en términos de probabilidad.

I-A. Contribuciones

Nuestra principal contribución es el diseño y la implementación de un caso de estudio que permite calcular las distribuciones a posteriori de las proporciones de cada grupo sobre datos cifrados con cifrado homomórfico.

En nuestro escenario, varios hospitales participan en un estudio clínico en el que el promotor dispone de un servicio basado en la nube como cuaderno de recogida de datos electrónico (eCRD). Gracias al cifrado homomórfico de los datos, los hospitales pueden compartir sus datos y el promotor puede externalizar el almacenamiento y el análisis de los datos a este servicio basado en la nube sin revelar, en ningún momento, el contenido de dichos datos.

II. Preliminares

II-A. Comparación de proporciones bayesiana

En el marco de la inferencia bayesiana, uno de los modelos más utilizados para abordar el problema de comparación de dos proporciones es el beta-binomial. Sean p_1 y p_2 las proporciones de pacientes que sufren un evento adverso para cada grupo de tratamiento.

A priori, asumimos que

$$p_i \sim \mathcal{B}(a_i, b_i) \text{ para } i = 1, 2,$$

es decir, que la distribución de ambas proporciones es beta de parámetros $\alpha = a_i > 0$ y $\beta = b_i > 0$, con a_i, b_i reales. En caso de haberlas, estos parámetros a_i y b_i pueden escogerse de forma que las distribuciones a priori correspondientes reflejen nuestras creencias sobre las proporciones, p_i, antes de observar los datos, D. Para asumir ignorancia a priori sobre las proporciones debemos escoger $a_i = b_i = 1$ para $i = 1, 2$, puesto que la distribución $\mathcal{B}(1, 1)$ es equivalente a una uniforme en el intervalo $(0, 1)$.

Si definimos las variables aleatorias, X_1 y X_2, como el número de pacientes que experimentan un evento adverso grave en cada grupo de tratamiento, podemos afirmar que

$$X_i \sim \text{Binomial}(n_i, p_i), \quad i = 1, 2,$$

donde n_i es el número de pacientes incluidos en el grupo experimental i, con $i = 1, 2$.

La teoría de inferencia bayesiana afirma que la distribución a posteriori de las proporciones p_1 y p_2 es

$$p_i|D \sim \mathcal{B}(a_i + x_i, b_i + n_i - x_i), \text{ con } i = 1, 2,$$

donde x_i es el número observado de pacientes que han experimentado un evento en el grupo i.

Este análisis puede hacerse de forma secuencial de forma que, al observar nuevos datos, $D^\star$, podemos tomar $p_i|D$ como distribución a priori y actualizar los parámetros de la distribución con $D^\star$, para obtener una nueva distribución a posteriori, $p_i|D, D^\star$.

Una vez que conocemos la distribución a posteriori, podemos proporcionar, para cada proporción, la estimación puntual; un intervalo de credibilidad al 95 % (a, b) tal que $P(a < p_i < b) = 0.95$; o la gráfica de la distribución a posteriori; además de la $P(p_2 - p_1 > 0|D)$, como evaluación del objetivo principal.

II-B. Cifrado homomórfico

Los esquemas de cifrado homomórfico permiten realizar operaciones sobre textos cifrados, de forma que al descifrar el resultado obtenido es el mismo que se obtendría al aplicar la operación a los datos sin cifrar. Es decir,

$$\text{Enc}(m_1) * \text{Enc}(m_2) = \text{Enc}(m_1 * m_2),$$

donde $*$ es la operación para la que el cifrado tiene la propiedad homomórfica.

Pueden distinguirse esquemas de cifrado *parcialmente homomórfico*, en los que se cumple la propiedad homomórfica para una sola operación, como el cifrado de Pallier [4] o RSA [5]; los esquemas *en cierto modo homomórficos*, que permiten realizar distintas operaciones sobre los datos cifrados, pero un número limitado de veces, como los propuestos por Sander, Young y Yung [6] o el propuesto por Boneh, Eu-Jin y Kobbi [7]; y los esquemas *completamente homomórficos* que permiten realizar sumas y multiplicaciones un número ilimitado de veces, como el esquema de Gentry [8]. Tras esta propuesta, se han presentado distintos esquemas de cifrado completamente homomórfico, y existen distintas bibliotecas escritas en C++, como HELib, que implementa el esquema BGV, de Brakerski, Gentry y Vaikuntanathan [9] o Microsoft SEAL [10], que implementa, tanto el esquema CKKS, propuesto por Cheon et al. [11], como el BFV (Brakerski/Fan-Vercauteren) [21].

En este caso usaremos el esquema CKKS puesto que permite trabajar con números tipo *float*. Como se explicó en la sección II-A, esto permite utilizar distribuciones a priori informativas, i.e. definidas por expertos.

III. Trabajos relacionados

Existe una biblioteca de R relacionada con el análisis estadístico seguro [12], aunque se ciñe a los tests frecuentistas y no paramétricos más utilizados y utiliza esquemas de computación multiparte, como muchos de los trabajos que cita. Otras dos bibliotecas "hermanas" de R implementan un sistema de cifrado homomórfico [13] y la adaptación de dos herramientas de *machine learning* [14]. En esta línea, se han realizado diversos estudios sobre clasificadores que preservan la privacidad [15], en particular el clasificador Naïve Bayes [16], [17], [18]. Estos estudios suelen utilizar esquemas de cifrado homomórfico o esquemas de computación multiparte que permiten el procesamiento confidencial de los datos. A diferencia de todas estas propuestas, en nuestro trabajo, el objetivo no es la clasificación, sino la inferencia bayesiana sobre datos cifrados con un esquema homomórfico.

IV. Caso de uso y resultados experimentales

Ilustraremos la aplicación de este trabajo con un diseño que asume que un promotor lleva a cabo un ensayo clínico con tres hospitales participantes. Estos comparten con el promotor del estudio sus datos de forma confidencial enviándolos al servicio *cloud*, de forma que se puedan realizar los cálculos necesarios sobre ellos sin poner en peligro su confidencialidad.

Utilizaremos un esquema de cifrado completamente homomórfico de forma que el servicio *cloud* realice los cálculos de los parámetros de las distribuciones a posteriori de las proporciones de cada grupo de tratamiento, que denominaremos:

$$\alpha_i = 1 + x_i \quad \beta_i = 1 + n_i - x_i,$$

para $i = 1, 2$, de una forma secuencial, a medida que vaya recibiendo los datos.

IV-A. Datos

El conjunto de datos usado en este trabajo es una simulación de datos basada en los resultados publicados del estudio *Synergy between PCI with Taxus and cardiac surgery* (SYNTAX) [19]. El objetivo del estudio fue comparar dos estrategias de tratamiento –la cirugía de revascularización miocárdica (CABG) y la angioplastia coronaria (PCI)– en pacientes con lesiones de 3 vasos, lesiones del tronco o de ambos. La comparación se realizó en términos de eventos adversos cardíacos y cerebrovasculares graves (MACCE) observados durante el primer año después de la intervención.

El conjunto de datos simulados contiene 1740 casos, 849 en el grupo CABG y 891 en PCI. La tabla I resume los datos simulados.

Aunque en el estudio SYNTAX participaron 85 hospitales, en los datos simulados hemos repartido aleatoriamente los datos en 3 subconjuntos.

Tabla I
NÚMERO Y PORCENTAJE DE CASOS CON ALGÚN EVENTO ADVERSO GRAVE (MACCE) EN LOS DATOS SIMULADOS, PARA CADA GRUPO DE TRATAMIENTO Y EN TODOS LOS DATOS.

	CABG	PCI	Total
	$n = 849$	$n = 891$	$n = 1740$
MACCE	105 (12.4 %)	159 (17.8 %)	264 (15.2 %)

IV-B. Prototipo

Se ha creado un prototipo siguiendo el modelo cliente-servidor. En el servicio *cloud* se ejecuta una aplicación servidora, que es la encargada de realizar las operaciones matemáticas con los datos previamente cifrados mediante un esquema homomórfico. Tanto cada uno de los hospitales como el promotor ejecutan en sus dispositivos la aplicación cliente, que muestra diferentes menús dependiendo de qué tipo de usuario inicie sesión (investigador o promotor). El servidor tendrá que validar las credenciales de cada uno de ellos en este paso, por medio de comprobación de usuario y contraseña.

El promotor será el único poseedor de la clave privada (*SK*) capaz de descifrar los datos enviados al servidor. Este tendrá la responsabilidad de enviar al servidor la correspondiente clave pública (*PK*) que se utilizará para cifrar los datos que forman parte del estudio. Además, podrá solicitar al servidor los resultados obtenidos a partir de los datos cifrado enviados por los hospitales que hayan participado en el estudio. Aparte, tendrá la posibilidad de manejar las altas y bajas de investigadores que forman parte del estudio.

El investigador responsable autorizado de cada uno de los hospitales participantes tendrá que indicar a la aplicación la ruta al fichero CSV en el que está almacenada la información a incluir. Dichos datos se cifran y se envían al servidor para que éste actualice los parámetros de las distribuciones beta.

El servidor dispone de una base de datos MySQL en la que almacena tanto la clave pública del promotor como los valores de los parámetros de las distribuciones beta calculados tras las diferentes interacciones con los hospitales participantes. Mediante el uso de la biblioteca CROW, se encarga de procesar y responder a las peticiones del promotor y los investigadores, realizadas a través del protocolo HTTP.

El prototipo se ha desarrollado en C++, empleando como base la biblioteca para cifrado totalmente homomórfico Microsoft SEAL, con el esquema CKKS.

La figura 1 muestra la arquitectura del sistema. Por un lado, se representa la aplicación cliente y, por otro, la aplicación servidora que se ejecuta en el servicio *cloud*.

Para realizar los cálculos de los parámetros de las distribuciones a posteriori de p_i, se realizan los siguientes pasos:

1. El promotor genera un par de claves (*PK*, *SK*) y envía *PK* al servidor para que la almacene en su base de datos.
2. El investigador responsable de un hospital selecciona el fichero CSV en el que se incluyen los datos a enviar. Estos se extraen como dos vectores de enteros en claro que después se cifran (dato a dato) con la *PK* del promotor, que le facilita el servidor. Los vectores cifrados se envían al servidor en formato *JSON* a través de un POST HTTP.
3. El servidor recibe el vector de datos cifrados y realiza los cálculos necesarios para actualizar los parámetros de las distribuciones beta que tiene almacenadas en su base de datos (en caso de ser el primer envío de datos, se parte de la distribución a priori).
4. Se repiten los pasos 2 y 3 tantas veces como lotes de datos en los que divida cada hospital su conjunto de datos.
5. El promotor solicita al servidor los valorescifrado de α_i y β_i de las distribuciones beta a posteriori asociadas a cada tipo de tratamiento (CABG y PCI) y los descifra.

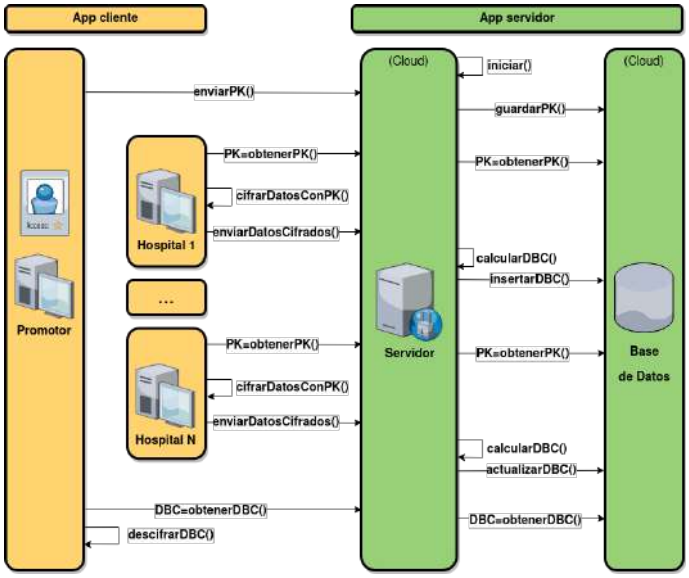

Figura 1. Diagrama de la arquitectura del sistema. En amarillo se representa la aplicación cliente y, en verde, la aplicación servidora que se ejecuta en el servicio *cloud*. Las flechas describen las interacciones entre los actores. Las siglas DBC se refieren a los parámetros de las distribuciones beta cifradas.

IV-C. Resultados

Todas las pruebas de rendimiento del prototipo se han llevado a cabo en un dispositivo con Ubuntu Desktop 21.04 como sistema operativo, equipado con 16GB de memoria RAM y CPU i7-8750H. Para la evaluación experimental del desempeño del prototipo hemos dividido el conjunto de datos en partes de diferente tamaño (100, 500, 1000 y todos los datos) con el objetivo de cuantificar el tiempo de ejecución. En concreto, se han medido los siguientes tiempos (en s):

- cifrado en la aplicación cliente (Enc),
- la conversión de datos cifrados a JSON en la aplicación cliente (Enc $\rightarrow$ JSON),
- la conversión de JSON a datos cifrados en la aplicación servidora (JSON $\rightarrow$ Enc),
- cálculos para actualizar α_i y β_i sobre datos cifrados (Comp).

Adicionalmente, se han calculado estos tiempos para cada uno de los tres hospitales. Todas las simulaciones se han repetido tres veces. El cálculo del tiempo total (para todos los datos) se ha calculado sumando el tiempo de los tres hospitales, puesto que el requerimiento de memoria del cifrado no permite el procesamiento de todos los datos. La tabla II muestra las medias y desviaciones típicas de los tiempos de ejecución.

Se ha comprobado que no hay error al realizar las operaciones sobre cifrados, en comparación con los mismos cálculos sobre datos en claro. La figura 2 muestra las distribuciones a

Tabla II
TIEMPOS DE EJECUCIÓN

	App Cliente		App Servidor	
n	Enc	Enc → JSON	JSON → Enc	Comp
100	3.2 ± 0.1	4.4 ± 0.1	5.0 ± 0.1	3.4 ± 0.0
500	18.3 ± 2.4	23.9 ± 3.5	25.0 ± 0.3	17.0 ± 0.5
1000	40.0 ± 7.1	56.8 ± 10.0	56.9 ± 4.5	42.9 ± 0.4
560 (h_1)	19.1 ± 1.0	25.8 ± 0.2	31.8 ± 2.9	20.6 ± 1.4
577 (h_2)	18.5 ± 0.1	26.4 ± 0.1	29.9 ± 0.7	20.9 ± 1.1
603 (h_3)	19.2 ± 0.1	27.4 ± 0.2	31.6 ± 1.4	22.3 ± 1.4
1740	56.8 ± 0.9	79.6 ± 0.3	93.3 ± 2.3	63.9 ± 3.0

Todos los tiempos se expresan en segundos y se describen mediante media ± sd, correspondientes a las 3 repeticiones de las simulaciones.

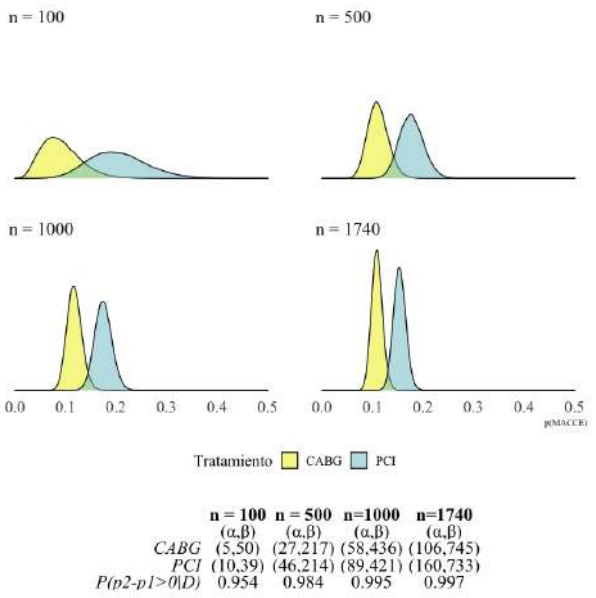

Figura 2. Distribución y parámetros de $p_i|D$ para cada grupo cuando se han observado 100, 500, 1000 y todos los datos.

posteriori de las proporciones, p_1 y p_2 y los parámetros de la mismas, para cada grupo cuando se han observado 100, 500, 1000 y todos los datos.

Una vez conocidos los parámetros de las distribuciones a posteriori, el promotor podría calcular la $P(p_2 - p_1 > 0|D)$ (ver tabla en la figura 2) a modo de evaluación de la hipótesis principal del estudio. A la vista de estos resultados, se alcanza una $P(p_2 - p_1 > 0|D) > 0.99$ con 1000 datos, por lo que se podría haber reducido el tamaño de la muestra de haberse propuesto un diseño adaptativo [20] para este estudio.

V. CONCLUSIONES Y TRABAJO FUTURO

En este trabajo presentamos un prototipo de eCRD en el que los hospitales participantes en un ensayo clínico comparten sus datos cifrados. Se elige un cifrado homomórfico para que se puedan calcular los parámetros de las distribuciones a posteriori correspondientes con los que el promotor realizará el análisis del objetivo principal del estudio.

Los resultados experimentales muestran que es factible el uso de un esquema de cifrado homomórfico para realizar estos cálculos con garantías de seguridad, y con tiempos de ejecución razonables tanto para el cliente como el servidor.

El enfoque bayesiano permite una actualización secuencial de las distribuciones a posteriori a medida que los investigadores disponen de nuevos datos, lo que permite el procesado de grandes conjuntos de datos, aminora el tiempo de ejecución global y, por otro lado, facilita la implementación de diseños adaptativos en los ensayos clínicos con compartición de datos segura.

Como trabajo futuro, sería interesante comparar la implementación con otros esquemas de cifrado homomórfico, como el esquema BFV que implementa SEAL, e incorporar otros modelos de análisis de datos.

AGRADECIMIENTOS

Este trabajo ha sido parcialmente financiado por: el proyecto MTM2017-83506-C2-2-P, financiado por el Ministerio de Economía y Competitividad; el proyecto PID2019-104790GB-I00, financiado por el Ministerio de Ciencia e Investigación; y las becas de colaboración del Ministerio de Educación y Formación Profesional.

V-A. Referencias

REFERENCIAS

[1] T. Baignères: “Provable security in cryptography”, pp. 28, 2017.
[2] J.W. Bos, K. Lauter, M. Naehrig: “Private predictive analysis on encrypted medical data”, en *Journal of Biomedical Informatics*, vol. 50, pp. 234-243, 2014.
[3] P.D. Hoff: “A First Course in Bayesian Statistical Methods”, Springer, 2009.
[4] P. Paillier: “Public-key cryptosystems based on composite degree residuosity classes”, in *EUROCRYPT, ser. LNCS*, vol. 1592. Springer, pp. 223–238, 1999.
[5] R.L. Rivest, A. Shamir, L. Adleman: “ A method for obtaining digital signatures and public-key cryptosystems”, en *Communications of the ACM*, num. 21, pp. 120–126, 1978.
[6] T. Sander, A. Young, M. Yung: “Non-interactive cryptocomputing for nc1”, en *40th Annual Symposium on Foundations of Computer Science*, pp 554–567, 1999.
[7] D. Boneh, E.-J. Goh, K. Nissim: “Evaluating 2-dnf formulas on ciphertexts”, en *Theory of Cryptography*, pp 325–341, 2005.
[8] C. Gentry: “A fully homomorphic encryption scheme”, PhD thesis, Stanford University, 2009.
[9] Z. Brakerski, C. Gentry, V. Vaikuntanathan: “(Leveled) Fully Homomorphic Encryption without Bootstrapping”, *ACM Trans. Comput. Theory*, vol. 6, n. 3, pp. 13:1-13:36, 2014.
[10] “Microsoft SEAL (release 3.0)”, http://sealcrypto.org, Oct. 2018, Microsoft Research, Redmond, WA.
[11] J.H. Cheon et al.: “Homomorphic encryption for arithmetic of approximate numbers”, in *ASIACRYPT, ser. LNCS*, vol. 10624. Springer, pp. 409–437, 2017.
[12] D. Bogdanov et al.: “Rmind: A Tool for Cryptographically Secure Statistical Analysis”, en *IEEE Trans. Dependable Secur. Comput.*, vol. 15, n. 3, pp. 481-495, 2018.
[13] L.J.M. Aslett, P.M. Esperança, C.C. Holmes: “A review of homomorphic encryption and software tools for encrypted statistical machine learning”, en *University of Oxford Technical report*, 2015.
[14] L.J.M. Aslett, P.M. Esperança, C.C. Holmes: “Encrypted statistical machine learning: new privacy preserving methods”, en *University of Oxford Technical report*, 2015.
[15] R. Bost et al.: “Machine Learning Classification over Encrypted Data”, en *NDSS*, 2015.
[16] S. Kim et al.: “Privacy-Preserving Naive Bayes Classification Using Fully Homomorphic Encryption”, en *ICONIP*, vol. 4, pp. 349-358, 2018.
[17] J. Chen et al.: “Non-interactive Privacy-Preserving Naïve Bayes Classifier Using Homomorphic Encryption”, en *SPNCE 2021, LNCS*, vol 423. Springer, 2022.
[18] Y. Yasumura, Y. Ishimaki, H. Yamana: “Secure Naïve Bayes Classification Protocol over Encrypted Data Using Fully Homomorphic Encryption”, en *iiWAS 2019*, pp. 45-54, 2019.
[19] P.W. Serruys et al.: “Percutaneous Coronary Intervention versus Coronary-Artery Bypass Grafting for Severe Coronary Artery Disease”, en *N Engl J Med*, vol. 360, pp. 961-972, 2009.
[20] S.M. Berry et al.: “Bayesian Adaptive Methods for Clinical Trials”, en *CRC press*, 2010.
[21] J. Fan, F. Vercauteren: “Somewhat Practical Fully Homomorphic Encryption”, en *Cryptology ePrint Archive, Report 2012/144*, 2012.