
Especificación de requerimientos de software

ParallelFischerScoring_binomial

Marzo de 2020

Ficha del documento

Fecha	Revisión	Autor	Verificado
16/03/2020	1	Nelson Enrique Vera Parra Andrés Ovidio Restrepo Rodríguez	Luis Eduardo Castillo Méndez

Documento validado por las partes en fecha:

Por la universidad
Universidad Distrital Francisco José de Caldas

Tabla de contenido

Especificación de requerimientos de software	1
1. Introducción	5
1.1. Propósito	5
1.2. Alcance	5
1.3. Personal Involucrado.....	5
1.4. Definiciones, acrónimos y abreviaturas	6
1.5. Referencias	6
1.6. Resumen.....	7
2. Descripción General	7
2.1. Perspectiva del producto	7
2.2. Funcionalidad del producto	8
2.3. Características de los usuarios	9
2.4. Restricciones	9
2.5. Suposiciones y dependencias.....	9
3. Requisitos específicos	10
3.1. Requisitos comunes de las interfaces	18
3.1.1. Interfaces de usuario.....	18
3.1.2. Interfaces de hardware	18
3.1.3. Interfaces de Software	18
3.2. Requerimientos funcionales	18
3.2.1. Requerimiento funcional 1.....	18
3.2.2. Requerimiento funcional 2.....	19
3.2.3. Requerimiento funcional 3.....	19
3.2.4. Requerimiento funcional 4.....	19
3.2.5. Requerimiento funcional 5.....	19
3.2.6. Requerimiento funcional 6.....	20
3.2.7. Requerimiento funcional 7.....	20
3.2.8. Requerimiento funcional 8.....	20
3.2.9. Requerimiento funcional 9.....	20
3.2.10. Requerimiento funcional 10.....	20
3.2.11. Requerimiento funcional 11.....	21

Especificación de requerimientos de Software

3.2.12.	Requerimiento funcional 12.....	21
3.2.13.	Requerimiento funcional 13.....	21
3.3.	Requerimientos no funcionales	21
3.3.1.	Requerimiento no funcional 1.....	21
3.3.2.	Requerimiento no funcional 2.....	22
3.3.3.	Requerimiento no funcional 3.....	22
3.3.4.	Requerimiento no funcional 4.....	22
3.3.5.	Requerimiento no funcional 5.....	22
3.3.6.	Requerimiento no funcional 6.....	23

1. Introducción

Este documento es una Especificación de Requisitos Software (ERS) para el software "ParallelFischerScoring_binomial". Esta especificación se ha estructurado basándose en las directrices dadas por el estándar IEEE Práctica Recomendada para Especificaciones de Requisitos Software ANSI/IEEE 830, 1998.

1.1. Propósito

El presente documento tiene como propósito definir las especificaciones funcionales y no funcionales para el desarrollo de una aplicación para la estimación de los parámetros de un modelo de regresión logística binaria con dos variables. Éste será utilizado por estudiantes y profesores afines al área.

1.2. Alcance

Esta especificación de requisitos está dirigida a integrantes de la comunidad académica/científica que requiera estimar los parámetros de modelos lineales generalizados y en particular los de un modelo de regresión logística mediante el algoritmo de Fisher Scoring.

1.3. Personal Involucrado

Nombre	Nelson Enrique Vera Parra
Rol	Analista y diseñador
Categoría Profesional	Ingeniería
Responsabilidad	Análisis de información y diseño
Información de contacto	neverap@udistrital.edu.co

Nombre	Luis Eduardo Castillo Méndez
Rol	Analista y diseñador
Categoría Profesional	Matemático
Responsabilidad	Análisis de información y diseño
Información de contacto	lecastillom@udistrital.edu.co

Especificación de requerimientos de Software

Nombre	Andrés Ovidio Restrepo Rodríguez
Rol	Analista, Diseñador y programador
Categoría Profesional	Ingeniería
Responsabilidad	Análisis de información, diseño y programación del software
Información de contacto	aorestrepo@correo.udistrital.edu.co

1.4. Definiciones, acrónimos y abreviaturas

Nombre	Descripción
Usuario	Persona que usará el software para ingresar los datos e iniciar el análisis.
Fisher Scoring	Algoritmo para estimación de parámetros.
Matriz	Arreglo bidimensional de datos del mismo tipo en conjunto
Matriz Diagonal	Matriz cuyos valores están ubicados en su diagonal, las demás posiciones no contienen valor
Norma	Es la raíz de la suma de los cuadrados de los valores dentro de una matriz
ERS	Especificación de Requisitos Software
RF	Requerimiento Funcional
RNF	Requerimiento No Funcional
Servidor	Programa informático que procesa datos realizando conexiones con peticiones de clientes.
Sistema	Sistema de Información para la estimación de parámetros de modelos logísticos.

1.5. Referencias

Título del Documento	Referencia
Standard IEEE 830 - 1998	IEEE

1.6. Resumen

Este documento consta de tres secciones. En la primera sección se realiza una introducción al mismo y se proporciona una visión general de la especificación de recursos del sistema.

En la segunda sección del documento se realiza una descripción general del sistema, con el fin de conocer las principales funciones que éste debe realizar, los datos asociados y los factores, restricciones, supuestos y dependencias que afectan al desarrollo.

Por último, la tercera sección del documento es aquella en la que se definen detalladamente los requerimientos que debe satisfacer el sistema

2. Descripción General

2.1. Perspectiva del producto

Los modelos lineales generalizados son una extensión flexible de una regresión lineal ordinaria, esto permite variables de respuesta que tienen modelos de distribución de errores distintos de una distribución normal. Así mismo, existen diferentes tipos de modelos lineales generalizados, entre alguno de estos modelos se destacan los modelos binomiales, modelos binomiales escalonados, modelo exponencial o el modelo Poisson.

Existen además diferentes métodos de estimación de modelos lineales generalizados, entre estos esta la estimación clásica, estimación Bayesiana y otras técnicas como Fisher Scoring. El algoritmo Fisher Scoring es por lo general usado para la estimación de los parámetros de modelos lineales generalizados y en particular en la estimación de parámetros de un modelo de regresión logística. El algoritmo de Fisher Scoring ha demostrado en diferentes pruebas tener una alta velocidad de convergencia con respecto a otros métodos de estimación. Para estudiar la capacidad de estimación y convergencia del algoritmo se simula un modelo de regresión logística binaria para dos variables de la forma $Y = 1 + 2X_1 + 3X_2$ estimando los parámetros (Betas), de tal forma que la variable dicotómica (Y) es explicada por medio de las dos variables (X1, X2). Se realizan pruebas utilizando CPU y GPU comparando el tiempo de respuesta para diferentes cantidades de datos multivariados simulados de las variables Y, X1 y X3 antes mencionados, simulados entre los 10.000 a los 30.000 datos.

2.2. Funcionalidad del producto

A continuación, se muestra la funcionalidad principal del producto ilustrado a través de un modelo funcional como se muestra en la Figura 1.

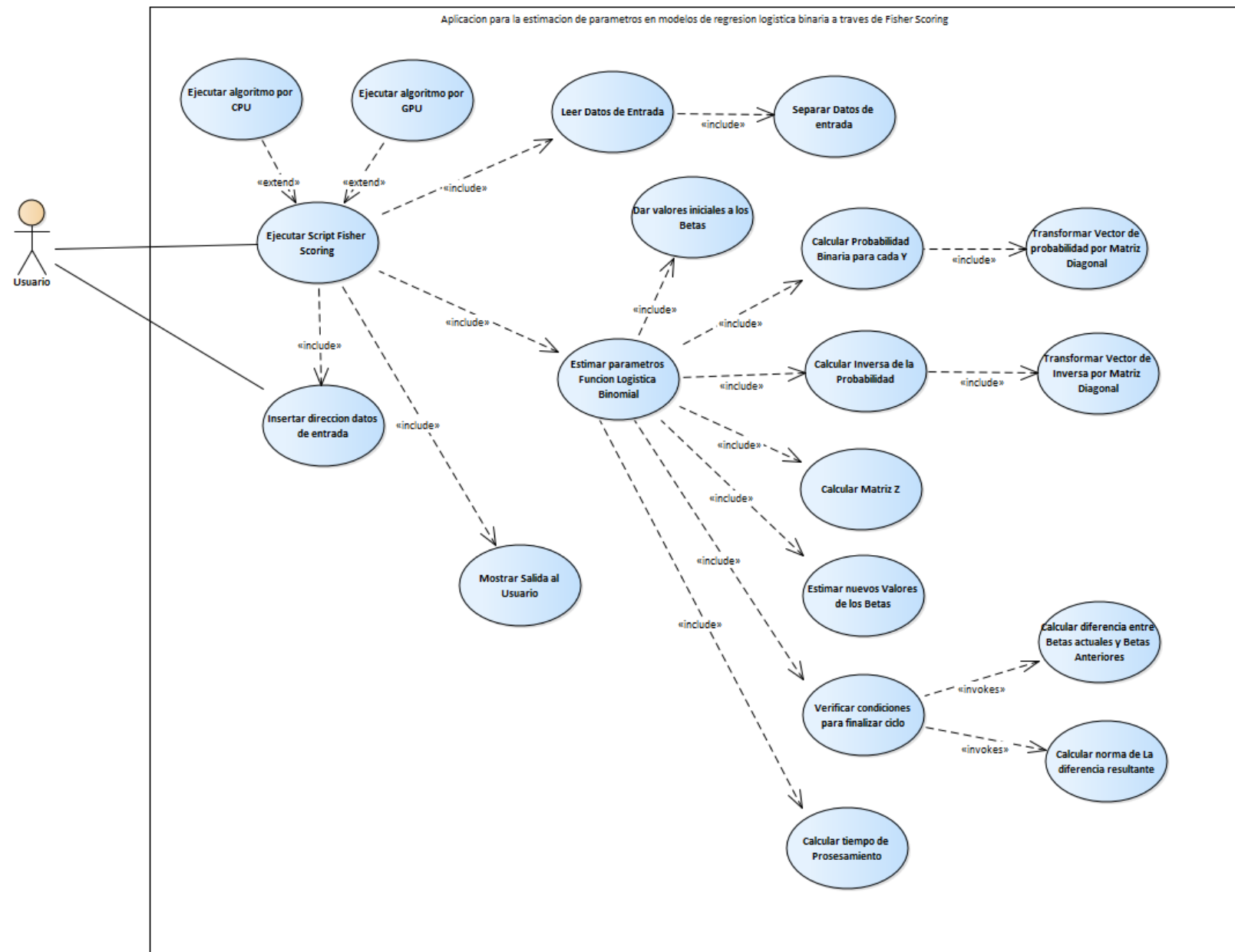


Figura 1. Modelo Funcional Producto

2.3. Características de los usuarios

Usuario	
Formación	Formación académica relacionada con estimación de parámetros en modelos lineales generalizados.
Actividades	Estimación de parámetros en modelos de regresión logística binaria.

2.4. Restricciones

- Interfaz para ser usada con terminal de Linux, Windows o Mac.
- Lenguajes y tecnologías en uso: Python, CUDA.
- El sistema deberá tener un diseño e implementación sencilla, independiente de la plataforma o del lenguaje de programación

2.5. Suposiciones y dependencias

- Se asume que los requisitos aquí descritos son estables.
- Los equipos en los que se vaya a ejecutar el sistema deben cumplir los requisitos antes indicados para garantizar una ejecución correcta de la misma.

3. Requisitos específicos

Requerimientos Funcionales

Identificación del requerimiento:	RF01
Nombre del requerimiento:	Leer Datos de Entrada
Descripción del requerimiento:	Una vez se ejecuta el archivo el sistema cargara los datos desde la dirección proporcionada por el usuario y los guardara en memoria para su posterior uso.
Características:	Los datos se transformarán en una estructura de datos Dataframe con el uso de la librería pandas, posteriormente se usarán como matrices Numpy.
Requerimiento funcional:	<ul style="list-style-type: none"> • RF02
Requerimiento NO funcional:	<ul style="list-style-type: none"> • RNF01 • RNF02 • RNF05 • RNF06
Prioridad de requerimiento:	Alta

Identificación del requerimiento:	RF02
Nombre del requerimiento:	Separar Datos de entrada
Descripción del requerimiento:	Con los datos transformados, se realiza una separación de la variable dicotómica (Y) con los datos independientes (r, x1, x2) de esta forma obtenemos dos arreglos.
Características:	La separación de datos se obtiene mediante indexación iloc dentro del Dataframe, de esta obtenemos dos nuevos Dataframe que posteriormente se convertirán en arreglos Numpy.
Requerimiento funcional:	<ul style="list-style-type: none"> • RF01
Requerimiento NO funcional:	<ul style="list-style-type: none"> • RNF01 • RNF02 • RNF05 • RNF06
Prioridad de requerimiento:	Alta

Especificación de requerimientos de Software

Identificación del requerimiento:	RF03
Nombre del requerimiento:	Estimar Parámetros Función Logística Binaria
Descripción del requerimiento:	Se realiza una estimación de parámetros en la función logística binaria a partir de los datos de entrada del usuario.
Características:	Para realizar la estimación de parámetros se deben seguir ciertos criterios y pasos que se explican detalladamente con los demás casos de uso del cual este se compone.
Requerimiento funcional:	<ul style="list-style-type: none"> • RF04 • RF05 • RF06 • RF07 • RF08 • RF09 • RF10 • RF11 • RF12 • RF13
Requerimiento NO funcional:	<ul style="list-style-type: none"> • RNF01 • RNF03 • RNF04 • RNF05 • RNF06
Prioridad de requerimiento:	Alto

Identificación del requerimiento:	RF04
Nombre del requerimiento:	Dar Valores Iniciales a los Betas
Descripción del requerimiento:	Se estiman unos valores iniciales para los parámetros, estos serán necesarios para una posterior comparación con las betas estimados más adelante.
Características:	El cálculo de los parámetros iniciales se debe realizar mediante la ecuación: $\beta = (X^t \cdot X)^{-1} \cdot X^t \cdot Y$
Requerimiento funcional:	<ul style="list-style-type: none"> • RF03 • RF05 • RF09 • RF012
Requerimiento NO funcional:	<ul style="list-style-type: none"> • RNF01 • RNF05 • RNF06
Prioridad de requerimiento:	Alta

Especificación de requerimientos de Software

Identificación del requerimiento:	RF05
Nombre del requerimiento:	Calcular Probabilidad para cada Y
Descripción del requerimiento:	Con las variables de entrada independientes (r, x1, x2) y las betas calculados se define la variable dicotómica Y a partir de una probabilidad binomial (entre 0 y 1), generando un vector resultante con cada probabilidad.
Características:	La variable de probabilidad es definida mediante la siguiente ecuación: $P \left[Y_i = \frac{1}{X_{i1}, \dots, X_{ip}} \right] = \frac{1}{1 + e^{(-\sum_{j=1}^p \beta_j X_{ij})}}$
Requerimiento funcional:	<ul style="list-style-type: none"> • RF03 • RF06 • RF07 • RF09
Requerimiento NO funcional:	<ul style="list-style-type: none"> • RNF01 • RNF05 • RNF06
Prioridad de requerimiento:	Alta

Identificación del requerimiento:	RF06
Nombre del requerimiento:	Transformar vector de probabilidad por matriz diagonal
Descripción del requerimiento:	Se toma el vector resultante del calculo de la probabilidad y se crea una matriz diagonal de tal forma que cada valor en el vector estará ubicado en la diagonal de la matriz cuadrada.
Características:	El vector que recibe tiene un tamaño de n posiciones, la matriz resultante tendrá un tamaño de nxn.
Requerimiento funcional:	<ul style="list-style-type: none"> • RF03 • RF05 • RF10
Requerimiento NO funcional:	<ul style="list-style-type: none"> • RNF01 • RNF05 • RNF06
Prioridad de requerimiento:	Alta

Identificación del requerimiento:	RF07
--	------

Especificación de requerimientos de Software

Nombre del requerimiento:	Calcular la inversa de la probabilidad
Descripción del requerimiento:	Se toma el vector de probabilidad y se saca la inversa del vector para ser usado mas adelante en el calculo de las nuevas betas.
Características:	El calculo de la inversa de un arreglo es el resultante del producto punto del vector con el mismo vector restado al escalar 1.
Requerimiento funcional:	<ul style="list-style-type: none"> • RF03 • RF05 • RF08
Requerimiento NO funcional:	<ul style="list-style-type: none"> • RNF01 • RNF05 • RNF06
Prioridad de requerimiento:	Alta

Identificación del requerimiento:	RF08
Nombre del requerimiento:	Transformar vector de inversa por matriz diagonal
Descripción del requerimiento:	Se toma el vector resultante del cálculo de la inversa de la probabilidad y se crea una matriz diagonal de tal forma que cada valor en el vector estará ubicado en la diagonal de la matriz cuadrada.
Características:	El vector que recibe tiene un tamaño de n posiciones, la matriz resultante tendrá un tamaño de nxn.
Requerimiento funcional:	<ul style="list-style-type: none"> • RF03 • RF07 • RF09
Requerimiento NO funcional:	<ul style="list-style-type: none"> • RNF01 • RNF05 • RNF06
Prioridad de requerimiento:	Alta

Identificación del requerimiento:	RF09
Nombre del requerimiento:	Calcular Matriz Z
Descripción del requerimiento:	Una vez calculadas las matrices diagonales se calcula la matriz Z que se usara para el posterior cálculo de las betas.
Características:	El calculo de la matriz Z se realiza mediante la siguiente ecuación: $Z = X \cdot \beta + W^{-1} \cdot (Y - \mu)$
Requerimiento funcional:	<ul style="list-style-type: none"> • RF03 • RF10

Especificación de requerimientos de Software

Requerimiento NO funcional:	<ul style="list-style-type: none"> • RNF01 • RNF05 • RNF06
Prioridad de requerimiento:	Alta

Identificación del requerimiento:	RF10
Nombre del requerimiento:	Estimar nuevos valores de las betas
Descripción del requerimiento:	Se estima los nuevos valores de las Betas o parámetros, que se utilizaran para verificar las condiciones para salir del ciclo, estas Betas deben aproximarse a 1, 2 y 3 respectivamente para la iteración final.
Características:	El cálculo de los parámetros Betas se debe realizar mediante la ecuación: $\beta^{(k+1)} = (X^t \cdot W^k \cdot X)^{-1} \cdot X^t \cdot W^k \cdot Y$
Requerimiento funcional:	<ul style="list-style-type: none"> • RF03 • RF11 • RF12 • RF13
Requerimiento NO funcional:	<ul style="list-style-type: none"> • RNF01 • RNF05 • RNF06
Prioridad de requerimiento:	Alta

Identificación del requerimiento:	RF11
Nombre del requerimiento:	Verificar condiciones para finalizar ciclo
Descripción del requerimiento:	El sistema verifica ciertas condiciones para seguir con el ciclo o salirse de él, en caso de continuar realiza los pasos anteriores remplazando las betas anteriores por los actuales, en caso de acabar muestra la información al usuario.
Características:	Las condiciones son haber cruzado 10 iteraciones o tener una norma menor a 0.00001.
Requerimiento funcional:	<ul style="list-style-type: none"> • RF03 • RF12 • RF13
Requerimiento NO funcional:	<ul style="list-style-type: none"> • RNF01 • RNF05 • RNF06
Prioridad de requerimiento:	Alta

Especificación de requerimientos de Software

Identificación del requerimiento:	RF12
Nombre del requerimiento:	Calcular diferencia entre betas actuales y betas anteriores
Descripción del requerimiento:	Se toman los parámetros Betas de la anterior iteración, o las Betas iniciales en caso de ser la primera iteración y se hace la diferencia con las betas actuales, para calcular posteriormente la norma del vector resultante.
Características:	La diferencia entre vectores se realiza fácilmente mediante la librería Numpy.
Requerimiento funcional:	<ul style="list-style-type: none"> • RF03 • RF11 • RF13
Requerimiento NO funcional:	<ul style="list-style-type: none"> • RNF01 • RNF05 • RNF06
Prioridad de requerimiento:	Alta

Identificación del requerimiento:	RF13
Nombre del requerimiento:	Calcular la norma de la diferencia resultante
Descripción del requerimiento:	Se toma el vector de la diferencia resultante entre las betas actuales y las betas anteriores y se calcula la norma para comprobar los criterios de salida de ciclo.
Características:	<p>La norma de un vector se calcula de la siguiente manera:</p> $\sqrt{X_1^2 + X_2^2 + \dots + X_i^2}$
Requerimiento funcional:	<ul style="list-style-type: none"> • RF03 • RF11
Requerimiento NO funcional:	<ul style="list-style-type: none"> • RNF01 • RNF05 • RNF06
Prioridad de requerimiento:	Alta

Requerimientos No Funcionales

Especificación de requerimientos de Software

Identificación del requerimiento:	RNF01
Nombre del requerimiento:	Ejecutar Script Fisher Scoring
Descripción del requerimiento:	El usuario entra en el servidor web y ejecuta el script de Python, el usuario puede ejecutar el archivo de ejecución con CPU o GPU.
Características:	Para ser ejecutar el programa es necesario que el usuario tenga internet para entrar al servidor o haber instalado correctamente todas las dependencias explicadas en el manual técnico.
Prioridad de requerimiento:	Alta

Identificación del requerimiento:	RNF02
Nombre del requerimiento:	Insertar Dirección Datos de entrada
Descripción del requerimiento:	Además de la selección del Script de ejecución el usuario deberá insertar la dirección del archivo que contiene los datos.
Características:	La dirección debe insertarse entre comillas y debe coincidir con un archivo csv existente en dicha ruta.
Prioridad de requerimiento:	Medio

Identificación del requerimiento:	RNF03
Nombre del requerimiento:	Calcular tiempo de procesamiento
Descripción del requerimiento:	Una vez se han importado las librerías correctamente el sistema calculara el tiempo en segundos en el cual el programa tarda en estimar los parámetros de la función logística binomial.
Características:	Este tiempo solo mide el procesamiento, cuando se corre el archivo por GPU la importación de las librerías suele demorarse más, pero este tiempo no es tomado en cuenta.
Prioridad de requerimiento:	Media

Especificación de requerimientos de Software

Identificación del requerimiento:	RNF04
Nombre del requerimiento:	Mostrar Salida al usuario
Descripción del requerimiento:	Cuando se han terminado de estimar los parámetros, el usuario podrá ver en pantalla una salida de datos que consta de 3 partes, la primera son el total de iteraciones que le tomo al programa para estimar los parámetros, la segunda son las Betas estimadas, y por ultimo el tiempo en segundos que le tomo al programa estimar dichos valores.
Características:	El numero de iteraciones consta de una variable int, las betas son mostradas a través de un arreglo de 3 posiciones, las betas deben tender a 1, 2 y 3 respectivamente, el tiempo es un float que representa los segundos de ejecución.
Prioridad de requerimiento:	Alta

Identificación del requerimiento:	RNF05
Nombre del requerimiento:	Ejecutar Algoritmo desde CPU
Descripción del requerimiento:	El usuario escoge el script de ejecución desde la CPU del ordenador, por lo que la ejecución se realiza a través del procesador.
Características:	La importación de librerías es leve ya que no es necesario recursos para traspaso de variables.
Prioridad de requerimiento:	Media

Identificación del requerimiento:	RNF06
Nombre del requerimiento:	Ejecutar Algoritmo desde GPU
Descripción del requerimiento:	El usuario escoge el script de ejecución desde la GPU del ordenador, por lo que la ejecución se realiza a través de la Tarjeta Gráfica.
Características:	La importación de librerías puede ser demorada ya que necesita recursos de traspaso de variables, también es necesario que CUDA reconozca la tarjeta gráfica para la ejecución.
Prioridad de requerimiento:	Media

3.1. Requisitos comunes de las interfaces

3.1.1. Interfaces de usuario

La interfaz del usuario consistirá en una terminal de Linux, Windows o MacOS con acceso a un conjunto de datos de prueba para la verificación de su funcionamiento.

3.1.2. Interfaces de hardware

Será necesario disponer de equipos de cómputo en perfecto estado para el servidor, con las siguientes características:

- Adaptadores de red.
- Procesador de mínimo 1.0GHz o superior con 4 núcleos.
- Memoria mínima de 16GB.
- Tarjeta gráfica NVIDIA Tesla K80.

3.1.3. Interfaces de Software

- Sistema Operativo: Windows, Linux.

3.2. Requerimientos funcionales

3.2.1. Requerimiento funcional 1

- **Leer Datos de Entrada:** Una vez se ejecuta el archivo el sistema cargara los datos desde la dirección proporcionada por el usuario y los guardara en memoria para su posterior uso.
 - ✓ Los datos se transformarán en una estructura de datos Dataframe con el uso de la librería pandas, posteriormente se usarán como matrices Numpy.

3.2.2. Requerimiento funcional 2

- **Separar Datos de entrada:** Con los datos transformados, se realiza una separación de la variable dicotómica (Y) con los datos independientes (r, x1, x2) de esta forma obtenemos dos arreglos.
 - ✓ La separación de datos se obtiene mediante indexación `iloc` dentro del Dataframe, de esta obtenemos dos nuevos Dataframe que posteriormente se convertirán en arreglos Numpy.

3.2.3. Requerimiento funcional 3

- **Estimar Parámetros Función Logística Binaria:** Se realiza una estimación de parámetros en la función logística binaria a partir de los datos de entrada del usuario.
 - ✓ Para realizar la estimación de parámetros se deben seguir ciertos criterios y pasos que se explican detalladamente con los demás casos de uso del cual este se compone.

3.2.4. Requerimiento funcional 4

- **Dar Valores Iniciales a los Betas:** Se estiman unos valores iniciales para los parámetros, estos serán necesarios para una posterior comparación con las betas estimados más adelante.
 - ✓ El cálculo de los parámetros iniciales se debe realizar mediante la ecuación:

$$\beta = (X^t \cdot X)^{-1} \cdot X^t \cdot Y$$

3.2.5. Requerimiento funcional 5

- **Calcular Probabilidad para cada Y:** Con las variables de entrada independientes (r, x1, x2) y las betas calculados se define la variable dicotómica Y a partir de una probabilidad binomial (entre 0 y 1), generando un vector resultante con cada probabilidad.
 - ✓ La variable de probabilidad es definida mediante la siguiente ecuación:

$$P \left[Y_i = \frac{1}{X_{i1}, \dots, X_{ip}} \right] = \frac{1}{1 + e^{(-\sum_{j=1}^p \beta_j X_{ij})}}$$

3.2.6. Requerimiento funcional 6

- **Transformar vector de probabilidad por matriz diagonal:** Se toma el vector resultante del cálculo de la probabilidad y se crea una matriz diagonal de tal forma que cada valor en el vector estará ubicado en la diagonal de la matriz cuadrada.
 - ✓ El vector que recibe tiene un tamaño de n posiciones, la matriz resultante tendrá un tamaño de nxn.

3.2.7. Requerimiento funcional 7

- **Calcular la inversa de la probabilidad:** Se toma el vector de probabilidad y se saca la inversa del vector para ser usado más adelante en el cálculo de las nuevas betas.
 - ✓ El cálculo de la inversa de un arreglo es el resultante del producto punto del vector con el mismo vector restado al escalar 1.

3.2.8. Requerimiento funcional 8

- **Transformar vector de inversa por matriz diagonal:** Se toma el vector resultante del cálculo de la inversa de la probabilidad y se crea una matriz diagonal de tal forma que cada valor en el vector estará ubicado en la diagonal de la matriz cuadrada.
 - ✓ El vector que recibe tiene un tamaño de n posiciones, la matriz resultante tendrá un tamaño de nxn.

3.2.9. Requerimiento funcional 9

- **Calcular Matriz Z:** Una vez calculadas las matrices diagonales se calcula la matriz Z que se usara para el posterior cálculo de las betas.
 - ✓ El cálculo de la matriz Z se realiza mediante la siguiente ecuación:

$$Z = X \cdot \beta + W^{-1} \cdot (Y - \mu)$$

3.2.10. Requerimiento funcional 10

- **Estimar nuevos valores de las betas:** Se estima los nuevos valores de las Betas o parámetros, que se utilizaran para verificar las condiciones para salir del ciclo, estas Betas deben aproximarse a 1, 2 y 3 respectivamente para la iteración final.

- ✓ El cálculo de los parámetros Betas se debe realizar mediante la ecuación:

$$\beta^{(k+1)} = (X^t \cdot W^k \cdot X)^{-1} \cdot X^t \cdot W^k \cdot Y.$$

3.2.11. Requerimiento funcional 11

- **Verificar condiciones para finalizar ciclo:** El sistema verifica ciertas condiciones para seguir con el ciclo o salirse de él, en caso de continuar realiza los pasos anteriores remplazando las betas anteriores por los actuales, en caso de acabar muestra la información al usuario.
 - ✓ Las condiciones son haber cruzado 10 iteraciones o tener una norma menor a 0.00001.

3.2.12. Requerimiento funcional 12

- **Calcular diferencia entre betas actuales y betas anteriores:** Se toman los parámetros Betas de la anterior iteración, o las Betas iniciales en caso de ser la primera iteración y se hace la diferencia con las betas actuales, para calcular posteriormente la norma del vector resultante.
 - ✓ La diferencia entre vectores se realiza fácilmente mediante la librería Numpy.

3.2.13. Requerimiento funcional 13

- **Calcular la norma de la diferencia resultante:** Se toma el vector de la diferencia resultante entre las betas actuales y las betas anteriores y se calcula la norma para comprobar los criterios de salida de ciclo.
 - ✓ La norma de un vector se calcula de la siguiente manera:

$$\sqrt{X_1^2 + X_2^2 + \dots + X_i^2}$$

3.3. Requerimientos no funcionales

3.3.1. Requerimiento no funcional 1

- **Ejecutar Script Fisher Scoring:** El usuario entra en el servidor web y ejecuta el script de Python, el usuario puede ejecutar el archivo de ejecución con CPU o GPU.

Especificación de requerimientos de Software

- ✓ Para Para ser ejecutar el programa es necesario que el usuario tenga internet para entrar al servidor o haber instalado correctamente todas las dependencias explicadas en el manual técnico.

3.3.2. Requerimiento no funcional 2

- **Insertar Dirección Datos de entrada:** Además de la selección del Script de ejecución el usuario deberá insertar la dirección del archivo que contiene los datos.
 - ✓ La dirección debe insertarse entre comillas y debe coincidir con un archivo csv existente en dicha ruta.

3.3.3. Requerimiento no funcional 3

- **Calcular tiempo de procesamiento:** Una vez se han importado las librerías correctamente el sistema calculara el tiempo en segundos en el cual el programa tarda en estimar los parámetros de la función logística binomial.
 - ✓ Este tiempo solo mide el procesamiento, cuando se corre el archivo por GPU la importación de las librerías suele demorarse más pero este tiempo no es tomado en cuenta.

3.3.4. Requerimiento no funcional 4

- **Mostrar Salida al usuario:** Cuando se han terminado de estimar los parámetros, el usuario podrá ver en pantalla una salida de datos que consta de 3 partes, la primera son el total, de iteraciones que le tomo al programa para estimar los parámetros, la segunda son las Betas estimadas, y por último el tiempo en segundos que le tomo al programa estimar dichos valores.
 - ✓ El número de iteraciones consta de una variable int, las betas son mostradas a través de un arreglo de 3 posiciones, las betas deben tender a 1, 2 y 3 respectivamente, el tiempo es un float que representa los segundos de ejecución.

3.3.5. Requerimiento no funcional 5

- **Ejecutar Algoritmo desde CPU:** El usuario escoge el script de ejecución desde la CPU del ordenador, por lo que la ejecución se realiza a través del procesador.
 - ✓ La importación de librerías es leve ya que no es necesario recursos para traspaso de variables.

3.3.6. Requerimiento no funcional 6

- **Ejecutar Algoritmo desde GPU:** El usuario escoge el script de ejecución desde la GPU del ordenador, por lo que la ejecución se realiza a través de la Tarjeta Gráfica.
 - ✓ La importación de librerías puede ser demorada ya que necesita recursos de traspaso de variables, también es necesario que CUDA reconozca la tarjeta gráfica para la ejecución.
 - ✓ acceso externo.