

Proyecto Final

Sistema de Recomendación

Bernardo García, Mauricio de Garay, Fernando Tapia

Taller de Desarrollo de Aplicaciones - Grupo A

César Arturo Angeles Ruiz

Universidad Iberoamericana

Ciudad de México, México

bernygr1999@gmail.com

mauriciodg00@gmail.com

ftapiaa53@gmail.com

I. ESPECIFICACIONES

El programa va a contar con la opción de administrador que va a consistir en agregar películas y usuarios. Por otra parte, también el programa se va a poder ingresar como usuario normal dando tu nombre, en el cual va a poder añadir a la base de datos las películas que vio, así como su calificación. De igual forma, el usuario normal tendrá una opción de ver recomendaciones, en la cual se le desplegará una lista de las $\%$ películas que el sistema de recomendaciones le asigna.

Para la elaboración del sistema de recomendación, el objetivo consiste en crear una nueva tabla en nuestra base de datos que nos permita ver el rating de cada usuario con respecto a las películas. Para esto, será simplemente hacer el producto punto entre features del usuario y de las películas.

Para obtener estos features, va a ser necesario implementar un método conocido como Stochastic Gradient Descent. Esto nos va a permitir crear una versión aleatoria de la tabla objetivo, de tal forma que la podamos comparar con la de entrenamiento, calcular el error y así ir mejorando la predicción del sistema. Recordemos que cada iteración de estos hace referencia a los EPOCHS, los cuales va a ser necesario contar. Una vez teniendo la tabla final, se va a poder usar para predecir las películas que le gusten a cada usuario.

- Contar con una base de datos. Una tabla destinada a usuarios, otra a películas y una como la combinación entre ellas para las recomendaciones.
- Generar un módulo que haga todo el procesamiento de datos (entrenar el modelo) basados en el modelo Stochastic Gradient Descent. Para esto será necesario ser óptimos en el código debido a la alta cantidad de procesamiento que se tiene que hacer (usar aritmética de apuntadores correctamente).
- Generar un primer menú que te permita ingresar como usuario.
- Entrar te va a permitir agregar películas y usuarios, así como desplegar todas las películas en pantalla. De igual manera, se te va a permitir generar los dos últimos puntos mencionados en este listado.
- Podrás agregar calificaciones, así como registrarte como usuario. Si agregas una calificación a una película que ya has calificado, se actualiza.
- Una vez ingresado, vas a poder agregar puntajes a las películas que quieras y ver tus recomendaciones de películas y de amigos.

- Estas recomendaciones se ordenan descendientemente en un porcentaje, con la película/amigo que tenga 100% siendo la más recomendada.
- Generar gráfico con ayuda de pipes para desplegar error vs Epocs.

II. Metodología

Para poder lograr todo lo requerido, necesitamos crear un diseño e implementaciones que permitan generalizar funciones, para así ahorrarnos código. Nuestra idea para fácilmente lograr todo esto, y permitir que nuestro proyecto sea fácilmente escalable, es utilizar bases de datos.

La manipulación de las bases de datos tiene que ser muy generalizada para que sea sencilla, es por eso que crearemos un módulo de bases de datos, que nos permite hacer 3 funciones principales:

- Iniciar la conexión a la base de datos.
- Leer cualquier query SELECT. Este se utilizará para todo: validaciones para login, registro, insertar y calificar; obtener datos para modelo predictivo; ver todas las películas; etc. Para lograr esto, debemos hacer una función **general** que nos retorne un doble apuntador de tipo caracter con los resultados del query, cada “fila” siendo un registro y cada columna del query estando separado por comas. Esto nos permitirá un manejo super general y fácil de los resultados con sscanf sin importar el query que hayamos hecho. Además, igual toma de argumento por referencia un entero “filas” que servirá para poder recorrer estos resultados sin importar el tamaño.
- Insertar cualquier registro a la base de datos, validando que se pueda insertar de manera adecuada.

Además de esto, necesitamos generar un modelo predictivo que, una vez obtenida las tablas de las bases de datos, pueda hacer una predicción de amigos o películas. Para generalizar esto, en ambos casos utilizaremos la técnica de stochastic gradient descent tras obtener un vector “aficiones”.

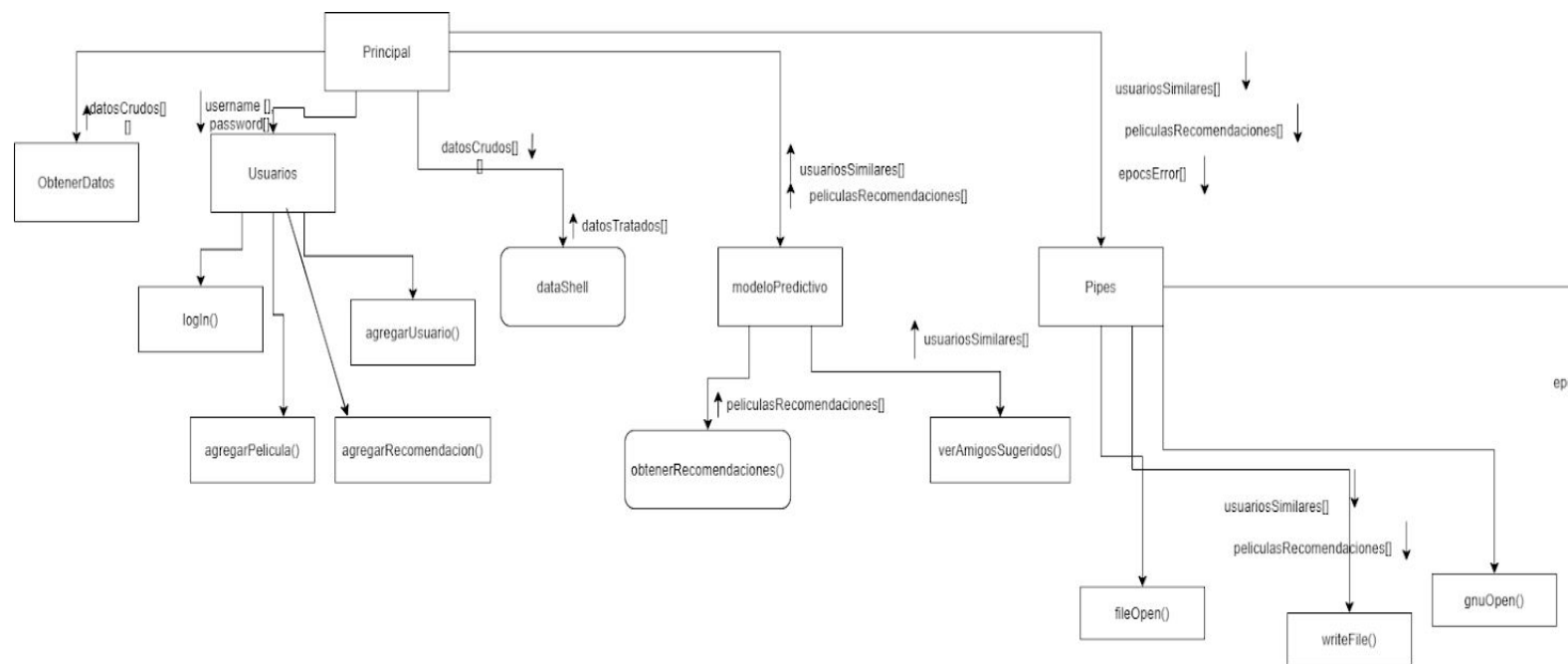
Utilizar estos pasos nos dejará generalizar hasta cierto punto todo el modelo predictivo, ya que mientras reciba los parámetros correctos, stochastic gradient descent aplicará esta técnica a **cualquier** conjunto de datos que reciba.

Por último, el manejo general del flujo de sistema utilizará el M.V.C. Esto nos permitirá que la vista solo se encargue de desplegar menús, gráficas, y recibir datos del usuario. El modelo se puede partir en 3 partes: el modelo general para controlar flujo de todo el comportamiento de procesamiento del sistema, el módulo cesarin que se encarga de controlar el flujo del modelo predictivo, y el módulo usuarios que encarga de manejar todo lo que tenga que ver con manejo de usuarios; esto para facilitar un entendimiento más sencillo del código. Por último, el controlador maneja el flujo general del programa, invocando tanto a los modelos como a la vista y asegurándose que los menús y sus órdenes de ejecución se respeten.

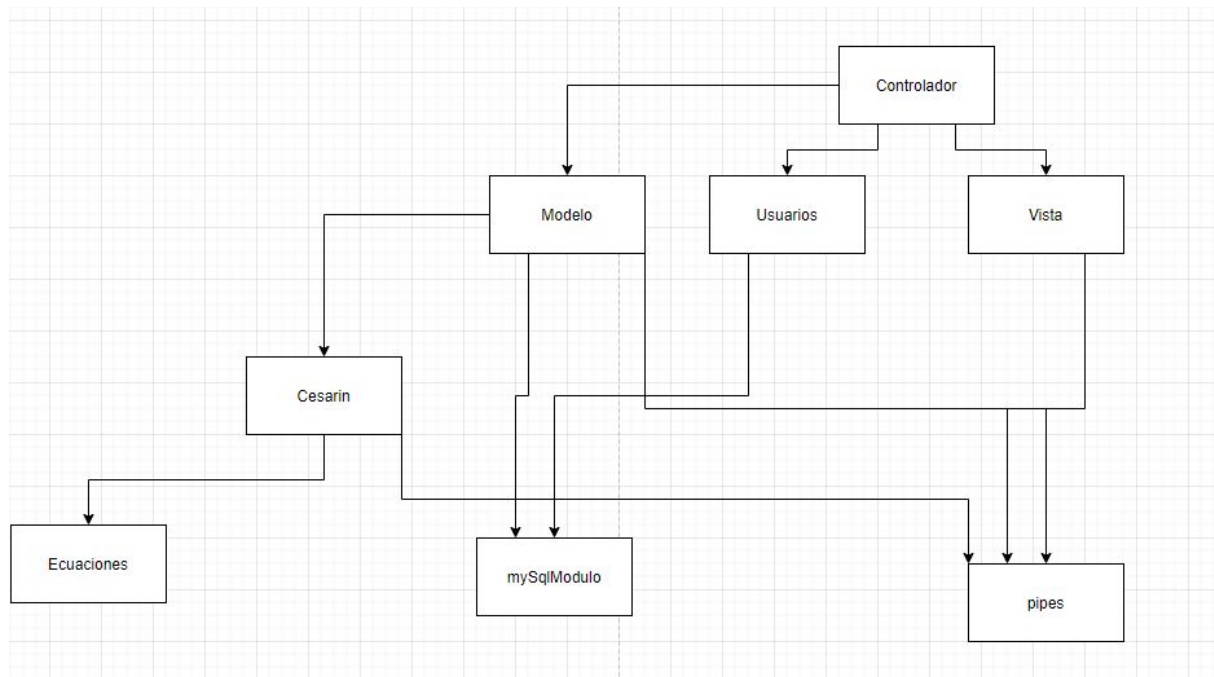
III. Diseño

Diagrama general del sistema:

Primer diseño:



Segundo diseño (final) para evitar dependencias horizontales:



Diseño bases de datos:

Tablas

usuarios

- id_usuario
- username
- password
- nombre
- edad
- genero

peliculas

- -
- titulo
- generoP
- clasificacion
- director
- actor_principal
- casa_productora
- duracion (minutos)
- año_publicacion

calificacion_peliculas

- id_usuario
- id_pelicula
- calificacion.

Tabla para datashell

- id_usuario
- edad
- drama
- accion
- animacion
- terror
- comedia
- G
- PG13
- R
- director
- actor_principal
- casa_productora
- duracion
- anio_publicacion
- calificacion (target)

Se obtiene con:

```
SELECT u.id_usuario,  
       u.edad,  
       p.id_peliculas,  
       p.titulo,  
       u.genero,  
       p.drama,  
       p.accion,  
       p.animacion,  
       p.terror,  
       p.comedia,  
       p.G,  
       p.PG13,
```

```

p.R,
p.actor_principal,
p.casa_productora,
p.duracion,
p.anio_publicacion,
c.calificacion
FROM usuarios u LEFT JOIN calificacion_peliculas c USING (id_usuario)
LEFT JOIN peliculas p ON (p.id_peliculas=c.id_pelicula);

```

géneros: comedia, terror, acción, animación, drama

clasificación: G, PG, PG-13, R

Se usará MySQL para manejar las bases de datos

MySQL Scripts

```

CREATE TABLE usuarios(
    id_usuario int AUTO_INCREMENT PRIMARY KEY,
    username varchar(50) NOT NULL,
    password varchar(50) NOT NULL,
    nombre varchar(50) NOT NULL,
    edad int NOT NULL,
    genero ENUM('M', 'F', 'Otro')
) CHARSET=latin1;

```

```

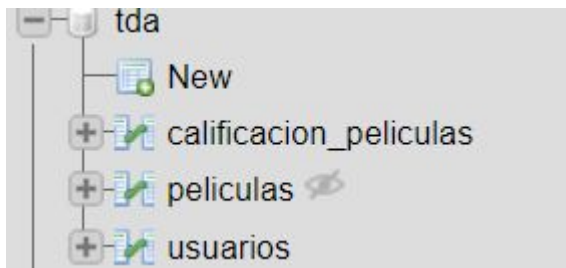
CREATE TABLE peliculas(
    id_peliculas int AUTO_INCREMENT PRIMARY KEY,
    titulo varchar(50) NOT NULL,
    drama int NOT NULL,
    accion int NOT NULL,
    animacion int NOT NULL,
    terror int NOT NULL,
    comedia int NOT NULL,
    G tinyint(1),
    PG13 tinyint(1),
    R tinyint(1),
    director varchar(50) NOT NULL,
    actor_principal varchar(50),
    casa_productora varchar(50),
    duracion INT NOT NULL,
    anio_publicacion int NOT NULL
) CHARSET=latin1;

```

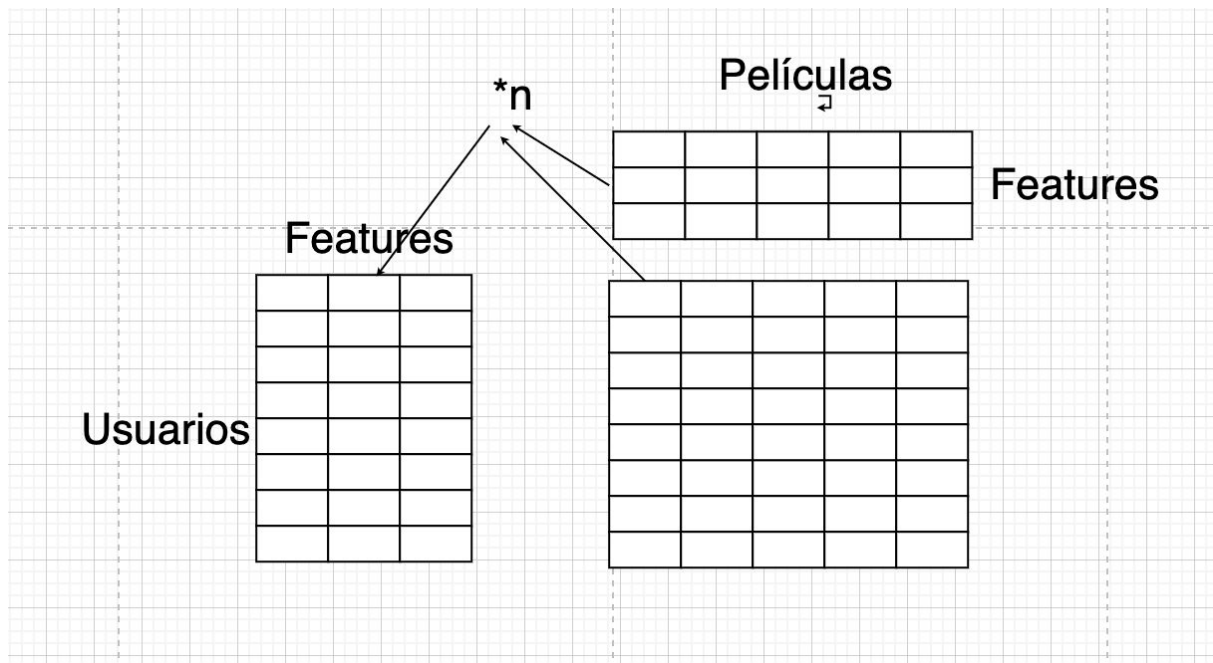
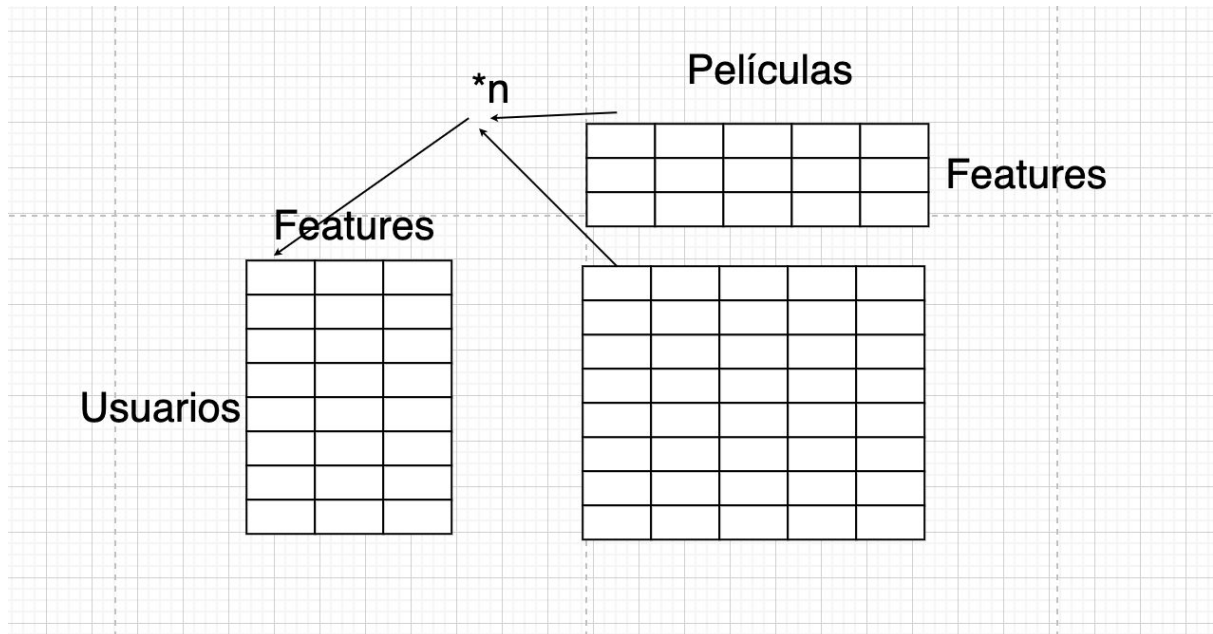
```

CREATE TABLE calificacion_peliculas(
    id_usuario int NOT NULL,
    id_pelicula int NOT NULL,
    calificacion int NOT NULL
) CHARSET=latin1;

```



Diseño Stochastic Gradient Descent:



Diseño gnuPlot:

gnuOpen():

- Será parte del módulo Pipes.
- Abrirá un pipe pointer a GnuPlot.
- Retornará un pipe pointer.

gnuPlot():

- Se encargará de graficar error vs epoc.
- Recibirá de argumento el nombre de un archivo.
- Recibirá de argumento un pipe pointer ya abierto.

Implementación:

```
FILE * pipesGnuOpen()
{
    return (popen("gnuplot -persist", "w"));
}

void pipesGnuPlot(FILE * gnupointer, char NombreArch[])
{
    char cont;
    fflush(stdin);
    fprintf(gnupointer, "set datafile separator '\\', '\\n' plot \"%s\" with lines\n", NombreArch);
    fflush(gnupointer);
    printf("Presiona enter para continuar...\n");
    scanf("%c", &cont);
    fflush(stdin);
}
```

Diseño Lectura/Escritura:

```
extern char ** mysql_readquery(char query[], MYSQL con, int *filas);
```

/*

Realiza el query solicitado y regresa un apuntador con el resultado en un arreglo de strings

@params char *query, MYSQL con, int *filas

@return resultados

*/

```
char ** mysql_readquery(char query[], MYSQL* con, int *filas)
```

```
{
    int fila;
    int totalfilas=0;
    char **resultados=NULL;
    if (mysql_query(con, query))
    {
        finish_with_error(con);
    }
}
```

```
MYSQL_RES *result = mysql_store_result(con);
```

```
if (result == NULL)
{
    finish_with_error(con);
}
```

```

}
MYSQL_ROW row;
// Cuento el total de filas.
while ((row = mysql_fetch_row(result)))
{
    totalfilas=totalfilas+1;
    /*for(int i = 0; i < num_fields; i++)
    {
        printf("%s ", row[i] ? row[i] : "NULL");
    }*/
}
mysql_free_result(result);

resultados=malloc(sizeof(char*)*totalfilas);
int i;
for(i=0; i<totalfilas;i++)
{
    resultados[i]=malloc(sizeof(char)*200);
}

//Ahora obtengo resultados bien.
if (mysql_query(con, query))
{
    finish_with_error(con);
}
MYSQL_RES *result2 = mysql_store_result(con);
if (result2 == NULL)
{
    finish_with_error(con);
}
num_fields=mysql_num_fields(result2);
MYSQL_ROW row2;
// Cuento el total de filas.
while ((row2 = mysql_fetch_row(result2)))
{
    char string[100];
    string[0]=0;
    for(int j = 0; j < num_fields; j++)
    {
        if(j>0)
        {
            strcat(string, ",");
        }
        strcat(string, row2[j]);
    }
    strcpy(resultados[i], string);
}
mysql_free_result(result2);

```



```

mysql_close(con);
*filas=totalfilas;
return resultados;
}

```

```

extern MYSQL con_mysql_startconnection();

```

```

/*

```

Inicia la conexión con la base de datos

```

@params

```

```

@return con

```

```

*/

```

```

MYSQL * mysql_startconnection(){

```

```

    MYSQL *mysql=NULL;;

```

```

    mysql_init(mysql);

```

```

    if(!mysql_real_connect(mysql, "localhost", "root", "", "tda", 0, NULL, 0))
    {

```

```

        printf("Error al conectarse a la base de datos %s\n", mysql_error(mysql));
        exit(1);
    }

```

```

    return mysql;
}

```

```

extern int mysql_doquery(char query[], MYSQL con);

```

```

/*

```

Realiza el query que inserta a la base, regresa si fue exitoso

```

@params char *query, MYSQL con

```

```

@return success

```

```

*/

```

```

int mysql_doquery(char query[], MYSQL con){

```

```

    if (mysql_query(con, query))
    {

```

```

        {
            finish_with_error(con);
            return 0;
        }

```

```

        mysql_close(con);

```

```

        return 1;
    }
}

```

Diseño resolución ecuaciones:

```

extern void ecuacionesRandom(int largo, int ancho, float *matriz[ancho]);

```

```

/*
    Asigna valores aleatorios entre 0 y 1 a toda la matriz.

    @params int largo, int ancho, float **matrix
    @return

*/

extern void ecuacionesPP(int largo, int ancho, float *matriz1[ancho], float *matriz2[ancho],
float *matrizFinal[ancho]);
/*
    Hace el producto punto entre matriz1 y matriz2 y asigna los valores a la matrizFinal

    @params int largo, int ancho, float **matriz1, float **matriz2, float **matrizFinal
    @return

*/

extern void ecuacionesCero(int largo, int ancho, float *matriz[ancho]);
/*
    Hace 0 todos los valores dentro de la matriz

    @params float **matriz
    @return

*/

extern void ecuacionesError(int largo, int ancho, float *matrizEntrenamiento[ancho], float
*matrizRandom[ancho], float *matrizError[ancho]);
/*
    Asigna a la matrizError el ECM entre las otras 2 matrices

    @params int largo, int ancho, float **matrizEntrenamiento, float **matrizRandom, float
**matrizError
    @return

*/

extern void ecuacionesCorregir(int largo, int ancho, float *matriz1[ancho], float
*mayriz2[ancho], float *matrizError[ancho]);
/*
    Actualizar matriz1 con stochastic gradient descent

    @params int largo, int ancho, float **matriz1, float **matriz2, float **matrizError
    @return

*/

```

```
EXTERN void ecuacionesStochasticGradientDescent(int largo, int ancho, int features,
int maxEpocs, float *errores, float *matriz1[features], float *matriz2[features],
float *matriz3[ancho], float *matriz4[ancho]);
```

```
/*
```

La función de ecuacionesStochasticGradientDescent manda a llamar a las respectivas funciones del módulo para que este puede hacer el modelo predictivo asociado al SGD.

```
@params int largo, int ancho, int features,
int maxEpocs, float *errores, float *matriz1[features], float *matriz2[features],
float *matriz3[ancho], float *matriz4[ancho]
```

```
@return
```

```
*/
```

Implementación:

```
void ecuacionesRandom(int largo, int ancho, float **matriz){
    srand (time(NULL));
    for(int i=0; i<largo; i++){
        for(int j=0; j<ancho; j++){
            matriz[i][j]=(rand()%100)/100;
        }
    }
}
```

```
void ecuacionesPP(int largo, int ancho, int features, float *matriz1[ancho], float
*matriz2[ancho], float **matrizFinal){
    for(i=0; i<largo; i++){
        for(int j=0; j<ancho; j++){
            for(int z=0; z<features; z++){
                matrizFinal[i][j]=matrizFinal[i][j]+(matriz1[i][j]*matriz2[z][j]);
            }
        }
    }
}
```

```
void ecuacionesCero(int largo, int ancho, float *matriz[ancho]){
    for(int i=0; i<largo; i++){
        for(int j=0; j<ancho; j++){
            matriz[i][j]=0;
        }
    }
}
```

```
void ecuacionesError(int largo, int ancho, float *matrizEntrenamiento[ancho], float
*matrizRandom[ancho], float *matrizError[ancho]){
    for(int i=0; i<largo; i++){
```

```

    for(int j=0; j<ancho; j++){
        matrizError[i][j]=((matrizEntrenamiento[i][j]-matrizRandom[i][j])^2)/2;
    }
}
}

void ecuacionesCorregir(int largo, int ancho, float *matriz1[ancho], float *matriz2[ancho], float
*matrizError[ancho]){
    for int(i=0; i<largo; i++){
        for(int j=0; j<ancho; j++){
            matriz1[i][j]=matriz1[i][j]+(n*matriz2[i][j]);
        }
    }
}

/*
    Función StochasticGradientDescent que usa lo de arriba
*/
void ecuacionesStochasticGradientDescent(int largo, int ancho, int features,
int maxEpocs,float *errores, float *matriz1[features], float *matriz2[features],
float *matriz3[ancho], float *matriz4[ancho]){
    float error=1000;
    float errorMax=100;
    float n=0.01;
    //PASO 1: Generar de forma aleatoria el contenido de nuestra matriz
    ecuacionesRandom(largo, features, matriz1);

    //PASO 2: Inicializar en cero toda la matriz a usar en el producto punto
    ecuacionesCero(largo, ancho, matriz3);

    //PASO 3: Generar el producto punto entre matriz1 y matriz2 y guardarlo en
    // la matriz3
    for(int i=0; i<maxEpocs||error<errorMax; i++){
        ecuacionesPP(largo, ancho, features, matriz1, matriz2, matriz3);

    //PASO 4: Obtenemos el error absoluto entre matriz4 y matriz3 (entrenamiento)
    error=ecuacionesError(largo, ancho, matriz4, matriz3);
    errores[i]=error;

    //PASO 5: Hacemos la actualización de matriz1 y matriz2
    ecuacionesCorregir(largo, ancho, features, matriz1, matriz2, matriz3, n);
    ecuacionesCorregir(largo, ancho, features, matriz2, matriz1, matriz2, n/10);
}

}

```

IV. Resultados

Para menú inicial (login y registrar):

Cuando ingresas opción inválida.

```
C:\Users\Mauricio\Desktop\5to Semestre\TDA\RepositorioModeloRecomendaciones\Codigo\ProyectoFinal\x64\Release\ProyectoFinal.exe
ERROR: opcion invalida.
Presiona <enter> para continuar...
Bienvenido a PelisChidas.c om!
    1.-Login.
    2.-Registrar.
Selecciona tu opcion:
v
```

```
ERROR: opcion invalida.
Presiona <enter> para continuar... █
```

Login.

Cuando ingresas un usuario que no existe o contraseña inválida:

```
Inicia sesion en PelisChidas.c om!
    Dame tu usuario: maudg00

    Dame tu contraseña: estoestamal
ERROR: esta combinacion de usuario/contrasena no existe.
Inicia sesion en PelisChidas.c om!
    Dame tu usuario:
```

Cuando es válido el login:

```
Inicia sesion en PelisChidas.c om!
    Dame tu usuario: maudg00

    Dame tu contraseña: hola123
Presiona <enter> para continuar... █
```

```
Bienvenido maudg00 !
    1.-Agregar pelicula.
    2.-Calificar pelicula.
    3.-Ver recomendaciones de peliculas.
    4.-Ver recomendaciones de amigos.
    5.-Ver todas las peliculas.
    6.-Salir.
Dame tu opcion: █
```

Registro.

Cuando tratas de registrar un usuario que ya existe:

```
Registrate a PelisChidas.c om!  
  Dame tu usuario: maudg00  
  
  Dame tu contrasena: nodebeservir  
  
  Dame tu nombre: Mauricio  
  
  Dame tu genero (M, F, O): M  
  
  Dame tu edad: 20  
ERROR: este usuario ya existe.  
Registrate a PelisChidas.c om!  
  Dame tu usuario: _
```

Cuando registras un usuario nuevo exitosamente:

```
Registrate a PelisChidas.c om!  
  Dame tu usuario: kirspirs  
  
  Dame tu contrasena: hola123  
  
  Dame tu nombre: Kirya  
  
  Dame tu genero (M, F, O): F  
  
  Dame tu edad: 20  
Presiona <enter> para continuar...
```

Para menú del programa una vez iniciada la sesión:

Para una opción inválida:

```
Bienvenido cesarin !  
  1.-Agregar pelicula.  
  2.-Calificar pelicula.  
  3.-Ver recomendaciones de peliculas.  
  4.-Ver recomendaciones de amigos.  
  5.-Ver todas las peliculas.  
  6.-Salir.  
Dame tu opcion: v  
ERROR: opcion equivocada.  
Presiona <enter> para continuar...
```

Para agregar película que ya existe:

```

Bienvenido cesarin !
1.-Agregar pelicula.
2.-Calificar pelicula.
3.-Ver recomendaciones de peliculas.
4.-Ver recomendaciones de amigos.
5.-Ver todas las peliculas.
6.-Salir.
Dame tu opcion: 1
4
Dime el nombre de la pelicula: Pulp Fiction
Dime el director: no servira
Dime el actor principal: no
Dime el casa productora: menos
Del 1 al 100 que tanto drama tiene: 99
Del 1 al 100 que tanta accion tiene: 99
Del 1 al 100 que tanta animacion tiene: 99
Del 1 al 100 que tanto terror tiene: 99
Del 1 al 100 que tanta comedia tiene: 99
Dime la duracion en minutos enteros: 99
Dime el anio de publicacion: 9999
Es clasificacion G? 1 si s, 0 si no: 1
ERROR: ya existe esta pelicula.
Presiona <enter> para continuar...

```

Para agregar película exitosamente:

```

Bienvenido cesarin !
1.-Agregar pelicula.
2.-Calificar pelicula.
3.-Ver recomendaciones de peliculas.
4.-Ver recomendaciones de amigos.
5.-Ver todas las peliculas.
6.-Salir.
Dame tu opcion: 1
4
Dime el nombre de la pelicula: Monsters Inc
Dime el director: Pete Docter
Dime el actor principal: John Goodman
Dime el casa productora: Pixar
Del 1 al 100 que tanto drama tiene: 75
Del 1 al 100 que tanta accion tiene: 40
Del 1 al 100 que tanta animacion tiene: 100
Del 1 al 100 que tanto terror tiene: 30
Del 1 al 100 que tanta comedia tiene: 95
Dime la duracion en minutos enteros: 92
Dime el anio de publicacion: 2001
Es clasificacion G? 1 si s, 0 si no: 1
92
La pelicula se subio exitosamente.
Presiona <enter> para continuar...

```

Para calificar película que no existe:


```
Bienvenido cesarin !
    1.-Agregar pelicula.
    2.-Calificar pelicula.
    3.-Ver recomendaciones de peliculas.
    4.-Ver recomendaciones de amigos.
    5.-Ver todas las peliculas.
    6.-Salir.
Dame tu opcion: 2
Calificar Pelicula.
Dime el nombre de la pelicula que quieres calificar: Nonsters Incoradspdsa
Del 1 al 5, cuanto le pones?: 5
ERROR: no existe la pelicula.
Presiona <enter> para continuar...
```

Para calificar película exitosamente:

```
Bienvenido cesarin !
    1.-Agregar pelicula.
    2.-Calificar pelicula.
    3.-Ver recomendaciones de peliculas.
    4.-Ver recomendaciones de amigos.
    5.-Ver todas las peliculas.
    6.-Salir.
Dame tu opcion: 2
Calificar Pelicula.
Dime el nombre de la pelicula que quieres calificar: Monsters Inc
Del 1 al 5, cuanto le pones?: 5
21
Se agrego exitosamente.
Presiona <enter> para continuar...
```

```
Bienvenido cesarin !
    1.-Agregar pelicula.
    2.-Calificar pelicula.
    3.-Ver recomendaciones de peliculas.
    4.-Ver recomendaciones de amigos.
    5.-Ver todas las peliculas.
    6.-Salir.
Dame tu opcion: 2
Calificar Pelicula.
Dime el nombre de la pelicula que quieres calificar: Toy Story
Del 1 al 5, cuanto le pones?: 5
3
Se agrego exitosamente.
Presiona <enter> para continuar...
```



```

Bienvenido cesarin !
    1.-Agregar pelicula.
    2.-Calificar pelicula.
    3.-Ver recomendaciones de peliculas.
    4.-Ver recomendaciones de amigos.
    5.-Ver todas las peliculas.
    6.-Salir.
Dame tu opcion: 2
Calificar Pelicula.
Dime el nombre de la pelicula que quieres calificar: Shrek 2
Del 1 al 5, cuanto le pones?: 5
10
Se agrego exitosamente.
Presiona <enter> para continuar...

```

```

Bienvenido cesarin !
    1.-Agregar pelicula.
    2.-Calificar pelicula.
    3.-Ver recomendaciones de peliculas.
    4.-Ver recomendaciones de amigos.
    5.-Ver todas las peliculas.
    6.-Salir.
Dame tu opcion: 2
Calificar Pelicula.
Dime el nombre de la pelicula que quieres calificar: Avengers Endgame
Del 1 al 5, cuanto le pones?: 5
15
Se agrego exitosamente.
Presiona <enter> para continuar...

```

Recomendaciones de amigos:

Con todo lo que calificamos de cesarin (películas animadas y de superhéroes), debería de recomendar Los Increíbles.

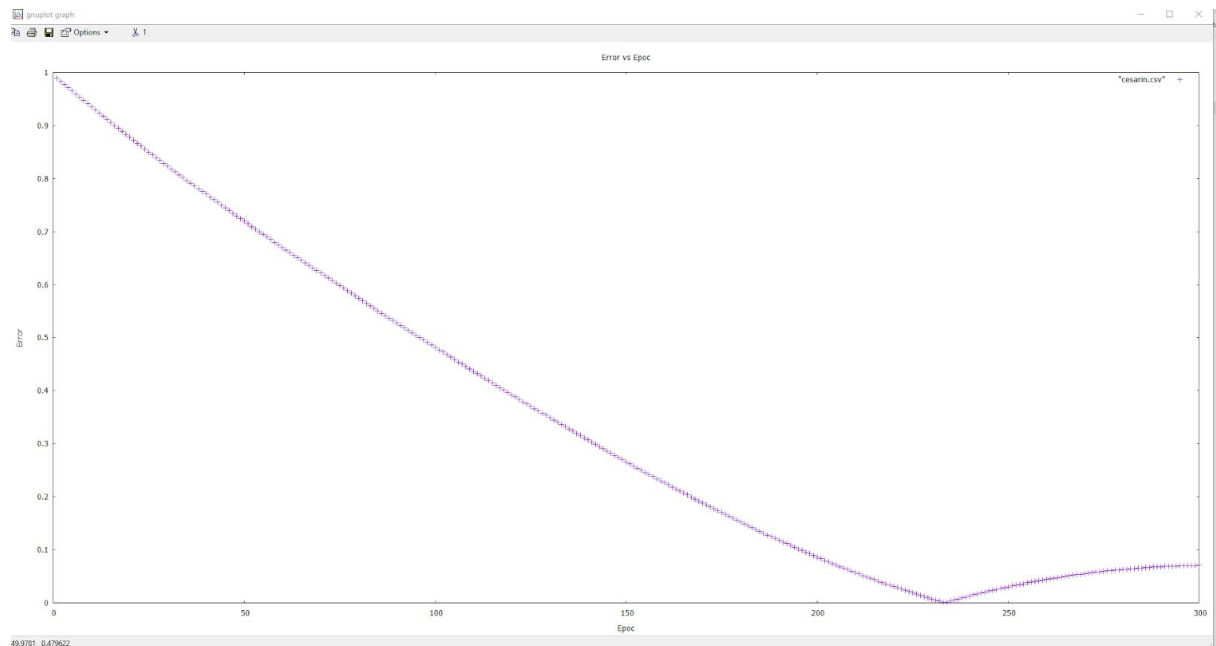
```

Monsters Inc = 100.000000
The Emoji Movie = 91.744560
Los Increibles = 89.205597
Shrek 2 = 88.824265
Wallace & Gromit: The Curse of the Were-Rabbit = 85.898941
The Good Dinosaur = 81.282639
Toy Story = 72.105743
Avengers Endgame = 60.538864
Grown Ups = 45.777157
Jack Y Jill = 43.090824
Star Wars = 42.206371
The Last Air Bender = 39.467262
Knives Out = 38.240002
Joker = 36.318199
Pulp Fiction = 35.543430
Grown Ups 2 = 35.204453
Inglorious Bastard = 34.523792
Annabelle = 28.471983
Parasite = 27.704556
The Conjuring = 27.069813
A Nightmare on Elm Street = 18.544514
Presiona enter para continuar...

```

Efectivamente, recomendó The Emoji Movie y los increíbles por la animación, con un 90% de chance de que te guste relativo al 100% que te gusta de Monsters Inc.

Error vs Epoc:



Recomendaciones amigos:

Mau igual le puso 5 estrellas a todas las animadas (menos Emoji Movie, a esa 1) entonces debería de ser igual a cesarin.

```

Bienvenido cesarin !
    1.-Agregar pelicula.
    2.-Calificar pelicula.
    3.-Ver recomendaciones de peliculas.
    4.-Ver recomendaciones de amigos.
    5.-Ver todas las peliculas.
    6.-Salir.
Dame tu opcion: 4
Ver Recomendaciones A.
maudg00 = 100.000000
ftapia = 55.938137
berny99gr = 14.320471
kirspirs = 8.055229
Presiona <enter> para continuar...
  
```

Ver todas las películas:

```

1.-Agregar película.
2.-Calificar película.
3.-Ver recomendaciones de películas.
4.-Ver recomendaciones de amigos.
5.-Ver todas las películas.
6.-Salir.
Dame tu opción: 5
Ver todas las películas.
1,Pulp Fiction,85,75,0,5,5,0,0,1,Quentin Tarantino,Samuel L. Jackson,A Band Apart,154,1994
2,Inglorious Bastard,85,80,0,0,0,0,0,1,Quentin Tarantino,Brad Pitt,Universal Pictures,153,2009
3,Toy Story,20,10,100,10,70,1,0,0,John Lasseter,Tom Hanks,Pixar,81,1995
4,The Good Dinosaur,60,40,100,5,55,1,0,0,Peter Sohn,Raymond Ochoa,Pixar,95,2015
5,The Conjuring,80,10,10,95,0,0,0,1,James Wan,Patrick Wilson,Safran Company,112,2013
6,A Nightmare on Elm Street,55,15,0,100,0,0,0,1,Wes Craven,Johnny Depp,New Line Cinema,91,1984
7,Wallace & Gromit: The Curse of the Were-Rabbit,45,50,100,25,70,1,0,0,Nick Park,Peter Sallis,DreamWorks Pictures,85,2005
8,Star Wars,60,70,20,0,20,1,0,0,George Lucas,Mark Hamill,Lucasfilm,121,1997
9,Grown Ups 2,10,10,5,0,80,0,1,0,Dennis Dugan,Adam Sandler,Columbia Pictures,101,2013
10,Shrek 2,70,40,100,0,70,1,0,0,Andrew Adamson,Mike Meyers,Dreamworks,93,2004
11,Knives Out,80,20,0,0,45,0,1,0,Rian Johnson,Ana de Armas,FilmNation Entertainment,130,2019
12,Parasite,90,30,0,50,0,0,0,1,Bong Joong-ho,Song Kang-ho,Barunson E&A,132,2019
13,The Last Air Bender,20,65,20,0,40,0,1,0,M. Night Shyamalan,Noah Ringer ,Nickelodeon Movies,103,2010
14,Jack Y Jill,20,10,10,0,90,0,1,0,Dennis Dugan,Adam Sandler,Columbia Pictures,91,2011
15,Avengers Endgame,75,95,0,0,70,0,1,0,J & A Russo,Robert Downey Jr.,Disney,182,2019
16,Joker,90,75,0,50,0,0,0,1,Todd Phillips,Joaquin Phoenix,Warner Brothers,124,2019
17,The Emoji Movie,30,75,95,0,90,1,0,0,Tony Leondis,T.J Miller,Sony Pictures Animations,90,2017
18,Grown Ups,50,0,0,0,95,0,1,0,Dennis Dugan,Adam Sandler,Columbia Pictures,102,2010
19,Annabelle,70,50,0,95,0,0,1,0,John R. Leonetti,Annabelle Wallis,Atomic Monster Productions,99,2014
20,Los Increíbles,50,95,95,0,60,1,0,0,Brad Bird,Holly Hunter,Pixar,116,2004
21,Monsters Inc,75,40,100,30,95,1,0,0,Pete Docter,John Goodman,Pixar,92,2001
Presiona <enter> para continuar... █

```

Salir:

```

Bienvenido cesarin i
1.-Agregar película.
2.-Calificar película.
3.-Ver recomendaciones de películas.
4.-Ver recomendaciones de amigos.
5.-Ver todas las películas.
6.-Salir.
Dame tu opción: 6
Salir.
Presiona <enter> para continuar...

C:\Users\Mauricio\Desktop\Sto Semestre\TDA\RepositoryoModeloRecomendaciones\Codigo\ProyectoFinal\64\Release\ProyectoFinal.exe (proceso 15952) se cerró con el código 0.
Para cerrar automáticamente la consola cuando se detiene la depuración, habilite Herramientas ->Opciones ->Depuración ->Cerrar la consola automáticamente al detenerse la depuración.
Presione cualquier tecla para cerrar esta ventana. . .

```