



Fernanda Pinedo | Oscar
Flores | Violeta García

Métodos de Inteligencia
Artificial

PROYECTO I

INDICE:

• Introducción.	2.
• Objetivos	2.
• Base de Datos	2.
• Seleccin de Variables	3.
○ Primer clasificador	
○ Segundo clasificador	
○ Tercer clasificador	
• Obtención de Clasificadores	6.
○ Primer clasificador	
○ Segundo clasificador	
○ Tercer clasificador	
• Predicciones	10.
○ Primer clasificador	
○ Segundo clasificador	
○ Tercer clasificador	
• Conclusiones	11.
• Anexos	12.
○ Códigos	
▪ Funciones	
• Referencias	16.

1. Introducción

Se puede distinguir entre dos tipos de clasificaciones, la supervisada y la no supervisada. La clasificación no supervisada es un proceso mediante el cual se definen clases de objetos caracterizados por distintas variables (reales o discretas) sin conocer a qué clase pertenece cada objeto, es decir, la clasificación supervisada busca crear clases de equivalencia en donde los elementos que pertenezcan a la misma clase presenten ciertas características en común (homogeneidad); pero a su vez las distintas clases deben ser disjuntas (heterogeneidad).

En la clasificación supervisada, también conocida como reconocimiento de patrones, cada objeto está caracterizado por distintas variables, reales o discretas, así como la clase a la que pertenece cada objeto. Por lo tanto, el objetivo de la clasificación supervisada consiste en encontrar un modelo para asignar un nuevo objeto a una de las categorías existentes en base a sus características. La muestra de objetos que sirve para construir el clasificador se le denomina conjunto de entrenamiento o *training*, ya que a partir de ella se determina la estructura y los parámetros del clasificador para posteriormente utilizarlos en el conjunto prueba o *testing*.

La clasificación supervisada juega un papel fundamental en las finanzas, ya que si se tienen distintos datos financieros de una empresa (variables) se puede determinar si ésta es candidata para sufrir una quiebra (entrar en default) o no. Por lo que, para realizar una buena clasificación, no es suficiente utilizar todos los datos que se dispone, sino hacer una selección de variables, dando así una limpieza previa a la base de datos y hacer una selección de las variables que influyan más en el fenómeno a caracterizar.

2. Objetivos

- Realizar una limpieza y tratamiento de datos a utilizar
- Realizar la selección de variables
- Obtener un clasificador
- Realizar una predicción

3. Base de Datos

Respecto a la base de datos obtenida del Excel *Datos_Proyecto2.xlsx*, se eliminó la primera fila ya que, en la última variable de default, no tenía ninguna clasificación (0 o 1). De la misma manera se eliminó la primera columna de fechas, debido a que no representa ningún valor relevante para el modelo.

data											
311x10 double											
	1	2	3	4	5	6	7	8	9	10	
1	9.8785e-04	0.4751	62.7405	42.6000	1.1000	58.5000	1.8300e+09	1.5400e+09	72.4900	0	
2	9.3180e-04	0.4757	65.3705	41.2000	1.2000	57.5000	1.6900e+09	1.3900e+09	77.5000	0	
3	8.9677e-04	0.5023	65.1305	41.6000	1.2000	57.1000	1.6800e+09	1.4600e+09	80.5000	1	
4	0.0010	0.5345	64.0405	41.4000	1.2000	56.3000	1.8300e+09	1.4700e+09	80.5000	0	
5	0.0011	0.6343	57.2805	41.4000	1.4000	55	1.7000e+09	1.3900e+09	84.6300	0	

Igualmente, notamos que había mucha diferencia entre los valores en materia de escala, por lo que se decidió normalizar los datos. Restándole a cada dato la media de la variable donde se encontraba el dato y dividiendo entre la desviación estándar.

4. Selección de variables

El proyecto presentado a continuación se realizó con la base de datos llamada *Datos_Proyecto2.xlsx* la cual cuenta con un total de 11 variables y 313 filas, de las cuales 10 variables son independientes y la última variable, que se encuentra en la última columna es dependiente.

Utilizando la fórmula de combinaciones con el uso de factoriales, se puede cuantificar el total de selecciones de variables posibles:

$$f(k, n) = \sum_{i=1}^n \frac{n!}{(n-k)! k!}$$

Donde,

k = elementos a considerar en el modelo

n = total de variables disponibles

Sabemos que existe un total de variables de 10, tomando en cuenta entonces esto, lo que cambiaría sería el número de elementos a considerar, por lo que la fórmula quedaría así

$$f(k) = \sum_{i=1}^{10} \frac{10!}{(10-k)! k!}$$

Por lo que el total de número de combinaciones posibles sería la suma de los resultados de la sumatoria considerando de 1 a 10 elementos, como se puede observar en la tabla a continuación:

k = número de elementos a considerar	Combinaciones posibles
1	10
2	45
3	120

4	210
5	252
6	210
7	120
8	45
9	10
10	1

Por lo tanto, el total de combinaciones o selecciones posibles para el problema sería 1,023.

4.1. Selección de variables del Primer Clasificador

Para el primer clasificador seleccionamos las variables de la manera que se ve a continuación, eliminando las variables cuyas correlaciones en valor absoluto fueran las 21 más grandes, o más cercanas a 1, quedándonos así con *Share Prices*, *Currency Exchange rates*, *Relative consumer prices indices 2005-100*, *Retail trade volume*, *International trade in goods – exports*, *CETES* y eliminando así las variables de *Hourly earnings, manufacturing*, *Index of Industrial production*, *Internal trade in goods – imports*.

Tabla	Share Prices	onal units	pemer price	incail trade	volumings, manu	industrial pril	trade in goc	trade in goc	CETES	
Share Prices	1	0.75427436	0.28672485	0.72245653	0.8773721	0.81047265	0.94048071	0.93391239	0.50962063	6.83531422
Currency exchange rates, National units per U	0.75427436	1	0.48726095	0.75761941	0.95145744	0.94371337	0.89608795	0.89572199	0.67551855	7.36165401
Relative consumer price indices, 2005=100	0.28672485	0.48726095	1	0.72203051	0.53288386	0.65207731	0.45455392	0.50150623	0.81024681	5.44728444
Retail trade volume	0.72245653	0.75761941	0.72203051	1	0.81712571	0.86136594	0.79104625	0.82499456	0.86642001	7.36305893
Hourly earnings, manufacturing	0.8773721	0.95145744	0.53288386	0.81712571	1	0.96275042	0.96877345	0.97235644	0.68128339	7.76400282
Index of industrial production	0.81047265	0.94371337	0.65207731	0.86136594	0.96275042	1	0.94186913	0.95425779	0.75373464	7.88024125
International trade in goods - exports	0.94048071	0.89608795	0.45455392	0.79104625	0.96877345	0.94186913	1	0.99667019	0.62029236	7.60977396
International trade in goods - imports	0.93391239	0.89572199	0.50150623	0.82499456	0.97235644	0.95425779	0.99667019	1	0.6559973	7.73541688
CETES	0.50962063	0.67551855	0.81024681	0.86642001	0.68128339	0.75373464	0.62029236	0.6559973	1	6.57311368

4.2. Selección de variables del Segundo Clasificador

Para el segundo clasificador seleccionamos las variables de la manera que se ve a continuación, eliminando en este caso las 37 variables que tuvieran el valor absoluto de la correlación es cercano a 1. quedándonos así con *Share Prices*, *Currency Exchange rates*, *Relative consumer prices indices 2005-100*, *Retail trade volume*, *International trade in goods – exports*, *CETES* y eliminando así las variables de *Hourly earnings, manufacturing*, *Index of Industrial production*, *Internal trade in goods – imports*

Tabla	Share Prices	onal units	pemer price	incaill trade	volumings, manu	industrial prvl	trade in goc	trade in goc	CETES	
Share Prices	1	0.75427436	0.28672485	0.72245653	0.8773721	0.81047265	0.94048071	0.93391239	0.50962063	6.83531422
Currency exchange rates, National units per U	0.75427436	1	0.48726095	0.75761941	0.95145744	0.94371337	0.89608795	0.89572199	0.67551855	7.36165401
Relative consumer price indices, 2005=100	0.28672485	0.48726095	1	0.72203051	0.53288386	0.65207731	0.45455392	0.50150623	0.81024681	5.44728444
Retail trade volume	0.72245653	0.75761941	0.72203051	1	0.81712571	0.86136594	0.79104625	0.82499456	0.86642001	7.36305893
Hourly earnings, manufacturing	0.8773721	0.95145744	0.53288386	0.81712571	1	0.96275042	0.96877345	0.97235644	0.68128339	7.76400282
Index of industrial production	0.81047265	0.94371337	0.65207731	0.86136594	0.96275042	1	0.94186913	0.95425779	0.75373464	7.88024125
International trade in goods - exports	0.94048071	0.89608795	0.45455392	0.79104625	0.96877345	0.94186913	1	0.99667019	0.62029236	7.60977396
International trade in goods - imports	0.93391239	0.89572199	0.50150623	0.82499456	0.97235644	0.95425779	0.99667019	1	0.6559973	7.73541688
CETES	0.50962063	0.67551855	0.81024681	0.86642001	0.68128339	0.75373464	0.62029236	0.6559973	1	6.57311368

4.3. Selección de variables del Tercer Clasificador

Para el tercero y último clasificador seleccionamos las mismas variables que en el clasificador 1, con la gran diferencia de cambiar el grado del modelo.

Tabla	Share Prices	onal units	pemer price	incaill trade	volumings, manu	industrial prvl	trade in goc	trade in goc	CETES	
Share Prices	1	0.75427436	0.28672485	0.72245653	0.8773721	0.81047265	0.94048071	0.93391239	0.50962063	6.83531422
Currency exchange rates, National units per U	0.75427436	1	0.48726095	0.75761941	0.95145744	0.94371337	0.89608795	0.89572199	0.67551855	7.36165401
Relative consumer price indices, 2005=100	0.28672485	0.48726095	1	0.72203051	0.53288386	0.65207731	0.45455392	0.50150623	0.81024681	5.44728444
Retail trade volume	0.72245653	0.75761941	0.72203051	1	0.81712571	0.86136594	0.79104625	0.82499456	0.86642001	7.36305893
Hourly earnings, manufacturing	0.8773721	0.95145744	0.53288386	0.81712571	1	0.96275042	0.96877345	0.97235644	0.68128339	7.76400282
Index of industrial production	0.81047265	0.94371337	0.65207731	0.86136594	0.96275042	1	0.94186913	0.95425779	0.75373464	7.88024125
International trade in goods - exports	0.94048071	0.89608795	0.45455392	0.79104625	0.96877345	0.94186913	1	0.99667019	0.62029236	7.60977396
International trade in goods - imports	0.93391239	0.89572199	0.50150623	0.82499456	0.97235644	0.95425779	0.99667019	1	0.6559973	7.73541688
CETES	0.50962063	0.67551855	0.81024681	0.86642001	0.68128339	0.75373464	0.62029236	0.6559973	1	6.57311368

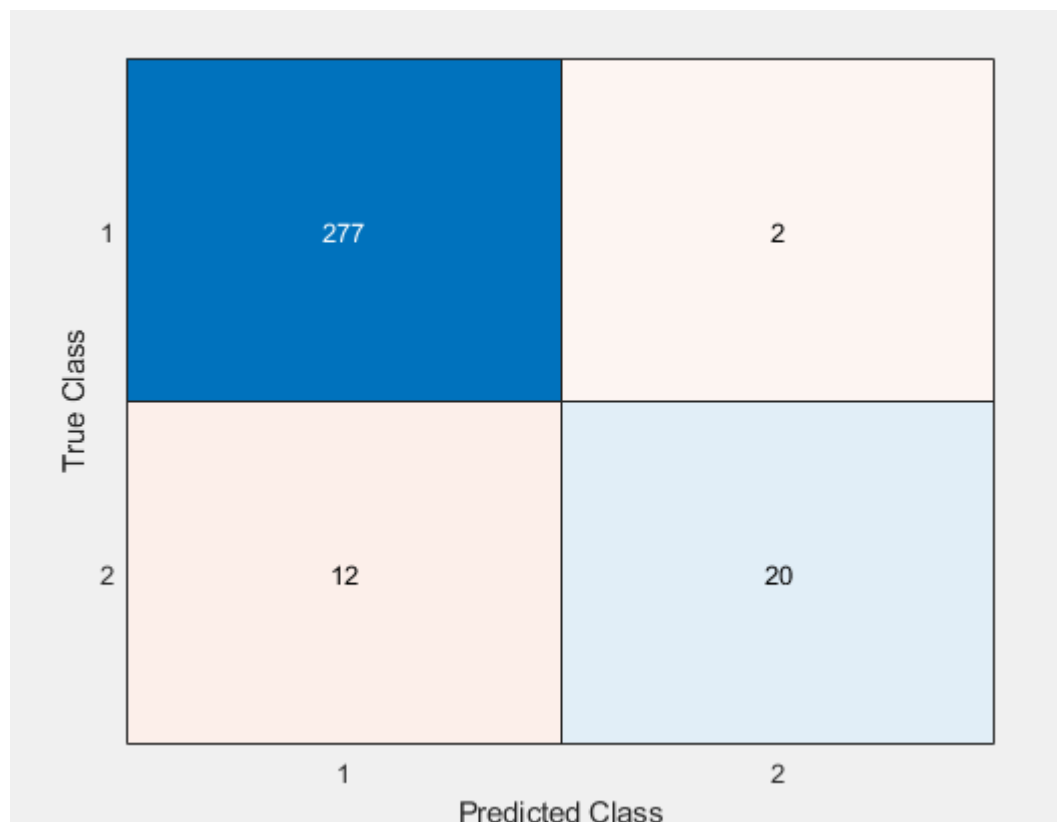
Se puede observar que la base de datos tiene demasiadas variables, sin embargo, algunas de ellas están muy correlacionadas, especialmente las de *Hourly earnings, manufacturing, Index of Industrial production, Internal trade in goods – exports, y Internal trade in goods – imports*. Por lo tanto, no se considera necesario mantener todas estas variables para la clasificación y predicción de los datos. Además de esto, se puede destacar que la suma de sus correlaciones con todas las demás variables (las marcadas de color verde en la columna sin nombre) son las que tienen los valores más altos en comparación con todas las demás.

En el caso de *Relative consumer prices indices 2005-100* y *CETES* se puede apreciar que se tiene una correlación muy baja en comparación con las demás variables. Por este motivo en ninguna seleccion de variables fue necesario eliminar estas dos y el precio de los activos (Esta variable en ningún momento se consideró borrarla, así fuera possible, debido a que es la ‘única’ información totalmente diferente y particular a las empresas que se están evaluando.

5. Obtención de clasificadores

Para la obtención de clasificadores se utiliza un modelo basado en regresión logística conocido como perceptrón simple, en dónde se pretende clasificar las bases de datos en dos categorías. Para este caso en particular, debido a que solamente se tienen dos categorías como resultados, no es necesario hacer particiones de 1vsAll. Se utilizan todos los datos para entrenar el modelo (no se da la oportunidad de evaluar con un 'Test' la calidad del modelo) debido a que se considera que existen muy pocos datos de los cuales aprender, esto tiene implicaciones fuertes, si en algún momento se decide probar la calidad del modelo con el desempeño del test, este último sería demasiado malo, debido principalmente a la cantidad de empresas en calidad de quiebra que tenemos registradas (aquellas con '1' en la columna de 'default').

5.1. Primer Clasificador



Grado 3, usando variables: [1 2 3 4 7 9]

Utilizando las variables de Para este modelo se utilizó un polinomio grado 3, obteniendo como resultados las métricas de clasificación

Accu_train =

0.9550

Prec_train =

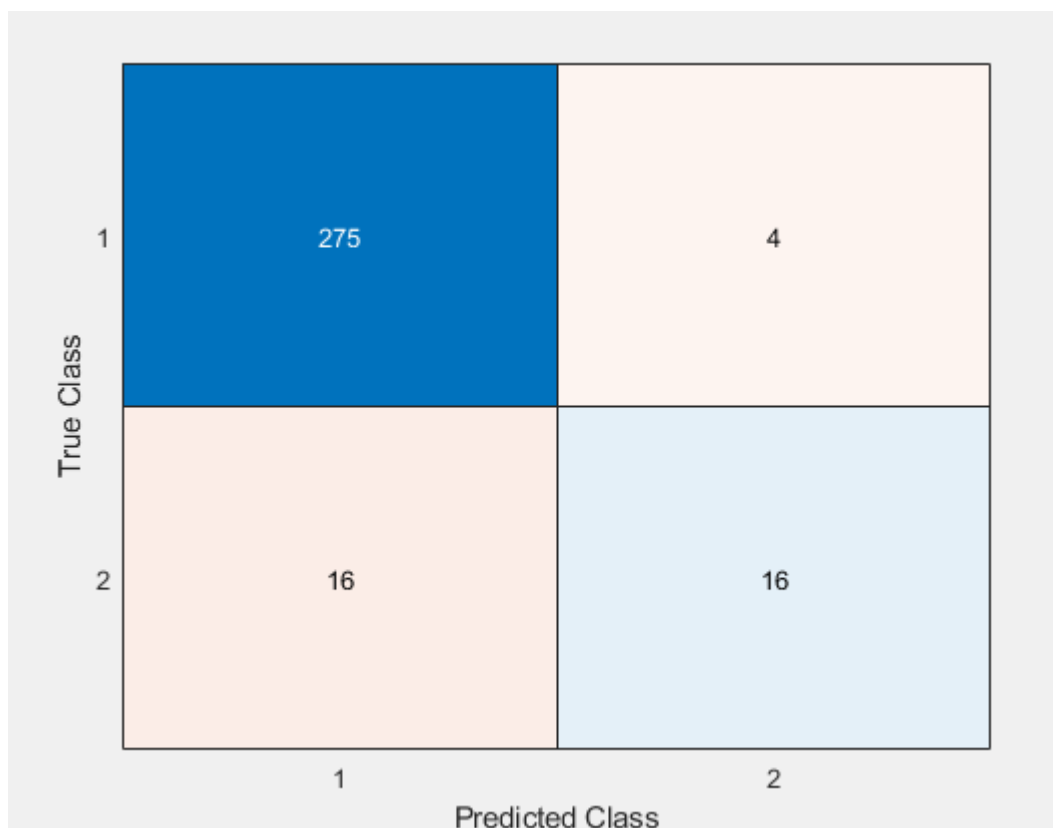
0.6250

Rec_train =

0.9091

Se puede observar que el desempeño en el entrenamiento parece ser bastante bueno en la exactitud y en el Recall, no siendo muy preciso. Este tipo de modelo sería bueno si estuviéramos negando créditos a las empresas que están por quebrar, se negarían algunos créditos a las empresas que no quebrarían (mala precisión), sin embargo, tendríamos una seguridad de que la mayoría de las empresas a las que le prestamos no entrarán en bancarrota (buen recall).

5.2. Segundo Clasificador



Grado 3, usando variables: [1 2 3 7 9]

Utilizando un modelo de grado 3 con las variables listadas en el apartado anterior.

Accu_train =

0.9357

Prec_train =

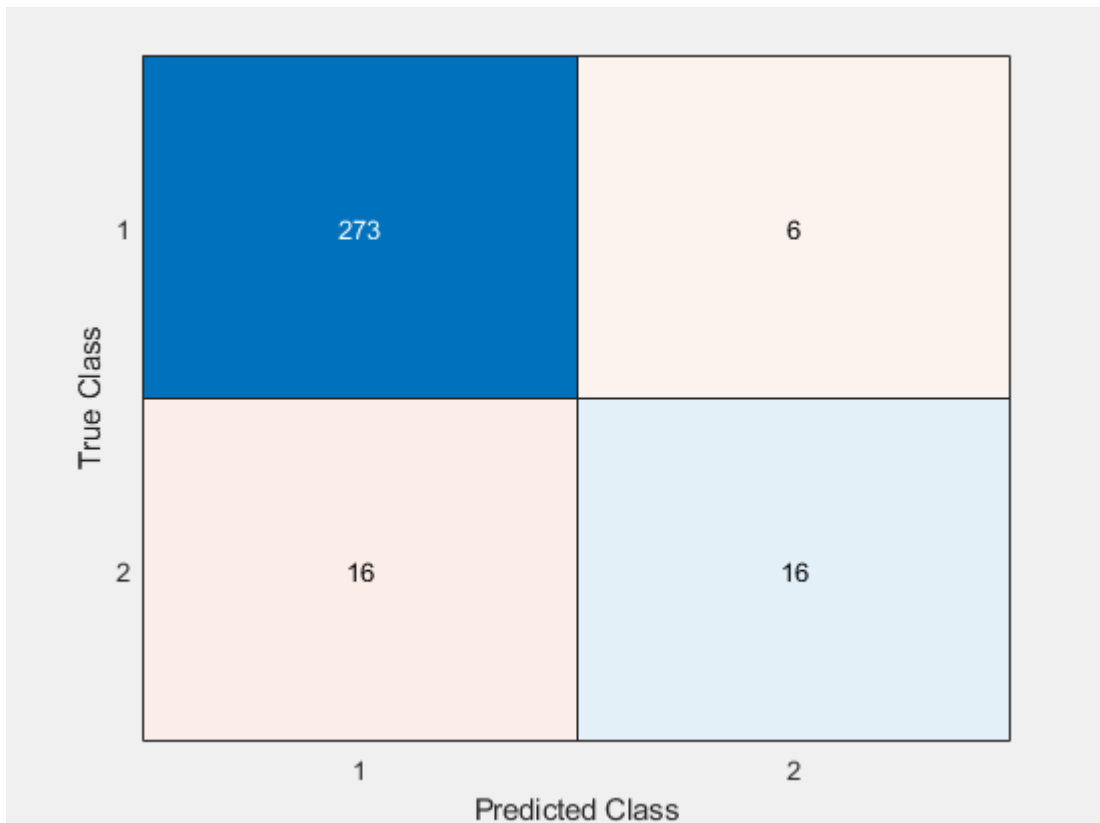
0.5000

Rec_train =

0.8000

De igual manera que el obtenido en el segundo clasificador; se obtiene una buena exactitud, un Recall decente y una precisión dudosa. A pesar de esto, podemos confiar un poco en los resultados obtenidos debido a que son algo similares al primer modelo.

5.3. Tercer Clasificador



Modelo de grado 4, utilizando los datos [1 2 3 4 7 9]

Accu_train =

0.9293

Prec_train =

0.5000

Rec_train =

0.7273

Para este modelo los resultados fueron un poco menos satisfactorios que en el modelo 2 y mucho menos satisfactorios que el primer modelo. A pesar de las diferencias existentes entre las métricas obtenidas, las predicciones hechas por este modelo fueron en algún punto similares a las obtenidas por el primer modelo y en parte también por el segundo.

6. Predicciones

Modelo1:

Yg_test =

1
0
1
0

Modelo2:

Yg_test =

0
0
1
0

Modelo3:

Yg_test =

1
0
0
0

El primer modelo, que es el modelo que más confianza genera tiene resultados bastante similares a los generados por el modelo 2 y 3. Si se tuviera que tomar una decisión sobre a cuál modelo o a cual predicción se le hace caso, se tomaría la elegida por el modelo 1. Son varios factores los que influyen en esta decisión, como primera de ellas podemos declarar que es la que mejores métricas ha conseguido. Además, el modelo 2 y 3 coinciden en que la empresa 3 y la empresa 4 no quebrarían. Otro punto importante es que el modelo 1 considera que las empresas 1 y 3 tienen un alto riesgo de que las empresas quiebren, mientras que el modelo 2 solamente considera a la empresa 3 y el modelo 3 solamente considera a la empresa 1.

Recordando las fortalezas de los modelos propuestos (un buen recall) podemos asumir que la capacidad de predicción para estos casos podrá clasificar como empresas con alta probabilidad de quiebra a quienes no necesariamente quiebren, sin embargo, dará seguridad con las otras dos empresas restantes de que esas no entrarán a bancarrota.

7. Resultados

Como resultados podemos declarar que el modelo1 es en el que confiamos un poco más, esto debido a que tiene como predicciones factores comunes al modelo 2 y modelo 3. Además de esto se puede observar que los resultados en las métricas de desempeño fueron mejores en el primer modelo que en los dos modelos siguientes.

La mayor cantidad de parámetros a estimar es de 115, siendo una cantidad de parámetros posibles a estimar. Para otros modelos de grado mayor y con una cantidad de variables más grandes se llegarían a estimar una cantidad mayor de parámetros (esto sería un problema serio, debido a que la cantidad de parámetros a estimar podría llegar a ser mayor que la cantidad de datos disponibles.)

Como resultado final de la predicción, tomando en consideración lo mencionado en el apartado anterior, se tomará por buena [1 0 1 0].

8. Conclusiones

El perceptrón es un método de clasificación supervisada, en este tipo de métodos se otorgan datos previamente clasificados y de los cuales se deben encontrar patrones para que en siguientes datos no-clasificados puedan ser predichos por un modelo propuesto. En este punto de proponer modelos es cuando son necesarias las métricas de desempeño para obtener un modelo que sea suficientemente bueno con las predicciones sin llegar a sobre-ajustar los datos (Este ultimo tema resultaría en otro problema a la hora de hacer predicciones, debido a que las predicciones mismas estarían sesgadas por algunos datos 'engañosos').

9. Anexos

9.1. Códigos

```
% P R O Y E C T O    1

% Modelo 1.

clear all;
close all;
clc;
%% Cargar los datos
data = xlsread('Datos_Proyecto2.xls', 'datos'); %Carga los datos del
archivo
TEST1 = xlsread('Datos_Proyecto2.xls', 'prediccion'); %Carga los datos
del archivo

%% Limpieza de datos
data = data(logical(1-sum(isnan(data),2)),:); % Eliminar todas las filas
que contengan NaN
%Normalización de datos
for i=1:size(data,2)-1
    TEST1(:,i) = (TEST1(:,i)-mean(data(:,i)))/std(data(:,i));
    data(:,i) = (data(:,i)-mean(data(:,i)))/std(data(:,i)); % Lo
estandarizamos debido a que tenemos variables muy grandes y otras muy
pequeñas.
end

%%
%Matriz de correlación
Correlacion = corr(data(:,1:end-1));

%%
X1 = data(:, [1 2 3 4 7 9]);
TEST1 = TEST1(:, [1 2 3 4 7 9]);
Y1 = data(:,end);

%% Regresión Logística
ngradol = 4;
X1a=func_polinomio(X1,ngradol);
W1=zeros(size(X1a,2),1); %Pesos iniciales %iniciar de ceros (tamaño(xa,
num columnas),en columna)
[J,dJdW]=fun_costob(W1,X1a,Y1); %Calculo de J y W

options=optimset('GradObj','on','MaxIter',1000);
[Wopt1,Jopt]=fminunc(@(W) fun_costob(W,X1a,Y1),W1,options);

% %% TRAIN
V1=X1a*Wopt1;
Yg_trainl=1./(1+exp(-V1));
Yg_trainl=round(Yg_trainl); %Redondea a unos y ceros pero deja en números
```

```

[Accu_train, Prec_train, Rec_train] = desempenio(Yg_train1,Y1) %
Desempenio.
%% Graficar matriz de confusión
confmat_train = confusionmat(Y1, Yg_train1)
confusionchart(confmat_train)

%% TEST
V1=func_polinomio(TEST1,ngrado1)*Wopt1;
Yg_test1=1./(1+exp(-V1));

Yg_test1=round(Yg_test1)
%%

% Modelo 2.

%% Cargar los datos
data = xlsread('Datos_Proyecto2.xls', 'datos'); %Carga los datos del
archivo
TEST2 = xlsread('Datos_Proyecto2.xls', 'prediccion'); %Carga los datos
del archivo

%% Limpieza de datos
data = data(logical(1-sum(isnan(data),2)),:); % Eliminar todas las filas
que contengan NaN
%Normalización de datos
for i=1:size(data,2)-1
    TEST2(:,i) = (TEST2(:,i)-mean(data(:,i)))/std(data(:,i));
    data(:,i) = (data(:,i)-mean(data(:,i)))/std(data(:,i)); % Lo
estandarizamos debido a que tenemos variables muy grandes y otras muy
pequeñas.
end

%%
%Matriz de correlación
Correlacion = corr(data(:,1:end-1));

%%
X2 = data(:, [1 2 3 7 9]);
TEST2 = TEST2(:, [1 2 3 7 9]);
Y2 = data(:,end);

%% Regresión Logística
ngrado2 = 4;
X2a=func_polinomio(X2,ngrado2);
W2=zeros(size(X2a,2),1); %Pesos iniciales %iniciar de ceros (tamaño(xa,
num columnas),en columna)
[J,dJdW]=fun_costob(W2,X2a,Y2); %Calculo de J y W

options=optimset('GradObj','on','MaxIter',1000);
[Wopt2,Jopt]=fminunc(@(W) fun_costob(W,X2a,Y2),W2,options);

% %% TRAIN
V2=X2a*Wopt2;
Yg_train2=1./(1+exp(-V2));
Yg_train2=round(Yg_train2); %Redondea a unos y ceros pero deja en números

```

```

[Accu_train, Prec_train, Rec_train] = desempenio(Yg_train2,Y2) %
Desempenio.
% %% Graficar matriz de confusión
confmat_train = confusionmat(Y2, Yg_train2)
confusionchart(confmat_train)

%% TEST
V2=func_polinomio(TEST2,ngrado2)*Wopt2;
Yg_test2=1./(1+exp(-V2));

Yg_test2=round(Yg_test2)
%%

% Modelo 3.

%% Cargar los datos
data = xlsread('Datos_Proyecto2.xls', 'datos'); %Carga los datos del
archivo
TEST3 = xlsread('Datos_Proyecto2.xls', 'prediccion'); %Carga los datos
del archivo

%% Limpieza de datos
data = data(logical(1-sum(isnan(data),2)),:); % Eliminar todas las filas
que contengan NaN
%Normalización de datos
for i=1:size(data,2)-1
    TEST3(:,i) = (TEST3(:,i)-mean(data(:,i)))/std(data(:,i));
    data(:,i) = (data(:,i)-mean(data(:,i)))/std(data(:,i)); % Lo
estandarizamos debido a que tenemos variables muy grandes y otras muy
pequeñas.
end

%%
%Matriz de correlación
Correlacion = corr(data(:,1:end-1));

%%
X3 = data(:, [3 7 9]);
TEST3 = TEST3(:, [3 7 9]);
Y3 = data(:,end);

%% Regresión Logística
ngrado3 = 4;
X3a=func_polinomio(X3,ngrado3);
W3=zeros(size(X3a,2),1); %Pesos iniciales %iniciar de ceros (tamaño(xa,
num columnas),en columna)
[J,dJdW]=fun_costob(W3,X3a,Y3); %Calculo de J y W

options=optimset('GradObj','on','MaxIter',1000);
[Wopt3,Jopt]=fminunc(@(W) fun_costob(W,X3a,Y3),W3,options);

% %% TRAIN
V3=X3a*Wopt3;
Yg_train3=1./(1+exp(-V3));

```

```

Yg_train3=round(Yg_train3); %Redondea a unos y ceros pero deja en números

[Accu_train, Prec_train, Rec_train] = desempenio(Yg_train3,Y3) %
Desempenio.
% %% Graficar matriz de confusión
confmat_train = confusionmat(Y3, Yg_train3)
confusionchart(confmat_train)

%% TEST
V3=func_polinomio(TEST3, ngrado3)*Wopt3;
Yg_test3=1./(1+exp(-V3));

Yg_test3=round(Yg_test3)

```

9.1.1. Funciones utilizadas

1.- Función desempeño

```

function [Exactitud, Precision, Recall] = desempenio(Yhat,Y)
% Función que da como resultado la exactitud, precision y recall de los
% datos estimados vs los datos reales. Recibe únicamente datos binarios.

conf = confusionmat(Yhat,Y);

tn = conf(1);
fn = conf(2);
fp = conf(3);
tp = conf(4);

Exactitud = (tp + tn)/(tp+tn+fp+fn);
Precision = tp/(tp+fp);
Recall = tp/(tp+fn);
end

```

2.- Función polinomio

```

function [Xp,coef]=func_polinomio (X,ngrado)
% Función que crea una matriz con los valores recibidos en X y los eleva
% a
% la potencia ngrado. Los resultados los regresa como una matriz con la
% estructura siguiente [1 x1 x2 x1x2 x1^2 x2^2 ....]
%
% Xp=func_polinomio (X,ngrado)

Xp=ones(size(X,1),1);
nvar=size(X,2);
coef=zeros(1,nvar);

for g=1:ngrado

```



```

% Obtener la tabla de coeficientes
Atemp=[(g-1:-1:1)' [1:1:g-1]'];
A=[];
for i=1:nvar-1
    for j=i+1:nvar
        Btemp=zeros(g-1,nvar);
        Btemp(:,i)=Atemp(:,1);
        Btemp(:,j)=Atemp(:,2);
        A=[A; Btemp];
    end
end
A=[A;g*eye(nvar)];

%% elevar al grado ngrado
for k=1:size(A,1)
    temp=ones(size(X,1),1);
    for j=1:nvar
        temp=temp.*(X(:,j).^A(k,j));
    end
    Xp=[Xp temp];
end
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
coef=[coef;A];
end
coef=coef';

```

3.- Función Costo

```

function [J,dJdW]=fun_costob(W,Xa,Y)

V=Xa*W;
Yg=1./(1+exp(-V)); %Y estimada

n=size(Xa,1); %Cantidad de elementos

J=sum(-Y.*log(Yg)-(1-Y).*log(1-Yg))/n;
%función de costo del Perceptrón

E=Yg-Y; %Error
dJdW=(E'*Xa)'/n; %Derivada de J respecto a W
end

```

9.2. Referencias

- (n.d.). Retrieved from http://www.montereyinstitute.org/courses/Algebra1/COURSE_TEXT_RESOURCE/U12_L2_T3_text_final_es.html.