Minería de Datos
Aplicación a la Fiabilidad Crediticia

Edgard Kenny Venegas Palacios[†] y Joseph Luis Kahn Casapía[‡]
Escuela Profesional de Matemática. Facultad de Ciencias.
Universidad Nacional de Ingeniería;
[†]*kenny@imca.edu.pe,* [‡]*jkahn@uni.pe*

Recibido el 7 de Setiembre del 2015; aceptado el 21 de Setiembre del 2015

Los modelos de clasificación en Minería de Datos, enfocados a predecir si un cliente de un banco es fiable o no para recibir un crédito, presentan un score y precisión con entrenamiento simple para RL: 0.21% y 0.79%, NB: 0.74% y 0.76%, AD: 0.70% y 0.69%, RNA: 0.74% y 0.73%, con entrenamiento cruzado para RL: 0.20% y 0.79%, NB: 0.73% y 0.72%, AD: 0.68% y 0.72%, RNA: 0.73% y 0.75%. Además de utilizar RL para encontrar relaciones entre las variables de la base de datos disponible.

Palabras Claves: Minería de datos, regresión lineal, métodos bayesiano, árboles de decisión, redes neuronales artificiales.

Classification models in Data Mining, focused on predicting whether a customer of a bank is reliable or not for credit, have a score and accuracy with simple training for RL: 0.21% and 0.79%, NB: 0.74% and 0.76%, AD: 0.70% and 0.69%, RNA: 0.74% and 0.73%, cross training RL: 0.20% and 0.79%, NB: 0.73% and 0.72%, AD: 0.68% and 0.72%, RNA: 0.73% and 0.75%. In addition to using RL to find relationships between variables of the database available.

Keywords: Data mining, linear regression, bayesian methods, decision trees, artificial neural networks.

1 Introducción

Un crédito bancario es un voto de confianza que un cliente recibe al obtener dinero de una entidad financiera, ya sea pública o privada. Por supuesto esa confianza se basa en que el cliente pruebe su solvencia (se pide por ejemplo que acredite ingresos suficientes y que sea propietario de inmueble). Mediante el crédito el cliente obtiene disponibilidad de efectivo y el Banco los intereses por el uso del dinero. El problema a tratar, es que decisión debe tomar la entidad financiera cuando un cliente solicita un crédito bancario basándose en la solvencia del cliente, antecedentes crediticios, etc. para minimizar el riesgo de que el cliente no pueda pagar las cuotas fijadas, clasificándolo en cliente bueno o malo. Los criterios para analizar el riesgo crediticio han sido variables a través del tiempo. La Minería de Datos es un conjunto de técnicas de análisis de datos que permiten: extraer patrones, tendencias y regularidades para describir y comprender mejor los datos y extraer patrones y tendencias para predecir comportamientos futuros. Expondremos técnicas de minería de datos como:

- Modelos de Regresión que se basan generalmente en expresar modelos y patrones mediante fórmulas algebraicas, funciones lineales, funciones no lineales, distribuciones o valores agregados estadísticos tales como medias varianzas, correlaciones, etc.

- Modelos Bayesianos que se basan en estimar la probabilidad de pertenencia a una clase o grupo, mediante la estimación de las probabilidades condicionales inversas, o apriori, utilizando para ello el teorema de Bayes. Algoritmos muy populares son el clasificador bayesiano naive, los métodos basados en máxima verisimilitud y el algoritmo EM.

- Árboles de decisión que se basan en dos tipo de algoritmos: los algoritmos denominados "divide y vencerás", como el $ID3/C4.5$ o el $CART$, y los algoritmos denominados "separa y vencerás", como el $CN2$

- Redes Neuronales Artificiales que aprender un modelo mediante el entrenamiento de los pesos que conectan conjunto de nodos o neuronas. La topología de la red y los pesos de las conexiones determinan el patrón aprendido. Existen innumerables variantes de organización: Perceptrón simple, redes multicapas, redes de base radial, red de Kohonen, y el más conocido es el de retropropagación(*backpropagation*).

Aplicaremos la minería de datos al problema de un banco al determinar la fiabilidad de un cliente que solicita un crédito bancario, utilizando una base de datos obtenida de *UCI Machine Learning Repository* [14] (*german.data*), utilizando técnicas de regresión lineal, métodos bayesianos, árboles de decisión y redes neuronales artificiales. Primero analizando el problema y los datos

para la creación del modelo, posteriormente la implementación utilizando librerías especializadas en el lenguaje de programación Python. Concluyendo con el análisis de los resultados y haciendo una comparación de los resultados obtenidos para cada modelo, con el fin de determinar que técnica es la más adecuada para el problema estudiado.

2 Estado del Arte

Los modelos de fiabilidad crediticio nacen alrededor de los años 1950 cuando Bill Fair y Earl Isaac fundan su compañía dedicada a apoyar las actividades de importantes empresas financieras y de ventas al menudeo [1]. Posteriormente, en los años 1960 se inicia el periodo en el que se desarrolla la industria de las tarjetas de crédito con lo cual los bancos ven una gran posibilidad de empezar utilizar modelos de fiabilidad [1]. En [2] y [1] se presenta un Estado del Arte en fiabilidad crediticia y fiabilidad comportamental. El artículo se concentra principalmente en mostrar lo que se ha hecho en el área desde la perspectiva de riesgo. Se presenta la historia del fiabilidad crediticia. Se vislumbran algunas de las variables que suelen ser importantes en un modelo como las características de la persona, el capital o valor solicitado, la capacidad de pago, las condiciones del mercado. Se mencionan técnicas empleadas en modelos de fiabilidad como la Regresión Logística, Árboles de Decisión, Redes Neuronales, Algoritmos Genéticos y algunos modelos híbridos que aprovechan características de más de un modelo como la combinación de la Programación Lineal con el uso de Redes Neuronales. Por otra parte, se analiza la fiabilidad comportamental o score aplicado sobre comportamientos en grupos de poblaciones. En este sentido se emplean técnicas que describen cómo es el comportamiento de un cliente de acuerdo a los rasgos de la población que lo rodean.

Los modelos más importantes son la Regresión Logística y el Análisis Discriminante Lineal. [3] establece que el Análisis Discriminante (Discriminant Analysis) y la Regresión Logística (Logistic Regression) como dos de las técnicas más empleadas en los modelos de fiabilidad crediticia. El autor muestra que estas técnicas tienen ciertas deficiencias cuando existen datos con una alta dimensionalidad, es decir muchos atributos a analizar, cuando la muestra o el número de ejemplos es bastante reducido, la selección de las variables y la incapacidad para modelar información no lineal. Similarmente, se han desarrollado trabajos que hacen importantes comparaciones entre estos dos modelos y otros mucho más elaborados. [4] hace una comparación entre el Análisis Discriminante, el Análisis de Probabilidad Unitaria (Probit Analysis) y las Redes Neuronales Artificiales (Artificial Neural Networks). De la misma manera, [5], [6], [7], [8] hacen comparaciones con otros modelos propuestos similares en donde los autores exponen las desventajas ante la falta de precisión de el Análisis Discriminante y la Regresión Logística.

Han surgido nuevos modelos que están siendo utilizados en la industria. Estos modelos son más eficientes que la Regresión Logística y el Análisis Discriminante Lineal y sus variantes. Entre los más usados se encuentra las Redes Neuronales Artificiales. [9] hace una comparación entre cinco modelos de redes neuronales: Multi-Layer Perceptron (MLP), Mixture Of Experts (MOE), Función de Base Radial (Radial Base Function, RBF), Learning Vector Quantization (LVQ) y la Resonancia Adaptativa Difusa (Fuzzy Adaptive Resonance, FAR); además del uso de las Máquinas Vectoriales de Soporte (Support Vector Machines, SVM) prometen mostrar mejores resultados de clasificación y predicción que como lo vienen mostrando las redes neuronales. [10] y [11] utilizan Máquinas Vectoriales de Soporte y proponen modelos de fiabilidad crediticia que, basados en sus experimentos, tienen un mejor desempeño y exactitud en la predicción que las redes neuronales artificiales.

Se han desarrollado nuevas técnicas que tienen un mejor desempeño. Algnas de estas técnicas consisten en la construcción de modelos híbridos que fusionen características de técnicas estadísticas y de aprendizaje de máquina. [6] desarrolla un modelo híbrido que emplea redes neuronales artificiales y curvas de regresión adaptativas multivariadas (Multivariate Adaptative Regression Splines,MARS). Similarmente, [12] presenta una arquitectura híbrida para generar un modelo de fiabilidad crediticia. Se fundamente en dos herramientas ampliamente utilizadas en la minería de datos: clustering y redes Neuronales. [7] muestra una combinación de las redes neuronales con el Análisis Discriminante. En [13] se comparan diferentes métodos estadísticos que han sido utilizados para predecir la insolvencia. Se distinguen los principales temas en la evaluación de los ingresos futuros utilizando los estados financieros, exponiendo los métodos de análisis discriminante simple (ADS), análisis discriminante múltiple (ADM), regresión logística (LR), algoritmo de partición sucesiva (RPA), técnica de escalamiento multidimensional (MDS), modelos de redes neuronales (NN), conjuntos aproximados (RS), utilites additives Discriminantes (UTADIS).

Además, existen muchas compañías que han implementado satisfactoriamente modelos de fiabilidad crediticia. Sin embargo, desde la perspectiva comercial, se han venido constituyendo empresas dedicadas a proveer soluciones de alto impacto desde el punto de vista de Inteligencia de Negocios (Business Intelligence, BI), Minería de Datos, Análisis Financiero, Toma de Decisiones, entre otros. Algunas de las compañías que más éxito han tenido en la implementación de modelos de fiabilidad crediticia son: Fair Isaac (http://www.fairisaac.com), SPSS (http://www.spss.com), SAS (http://www.sas.com). Estas compañías son capaces de ofrecer soluciones inteligentes que puedan apoyar de una manera ágil, cada una de las actividades de cualquier empresa en cualquier industria. Sin embargo, últimamente con los avances técnológicos y la competitividad, desde el punto de vista de desarrollo de software, casas desarrolladoras están ofreciendo soluciones fácilmente adaptables a las complejas industrias. Se tiene el caso de: IBM (http://www.ibm.com),

Oracle (http://www.oracle.com). Por otra parte, la lista de empresas que han desarrollado y adquirido modelos a fin de atender su operación diaria se podría considerar interminable. Sin embargo, algunos casos de éxito son entidades bancarias como por ejemplo HSBC, y franquicias proveedoras de tarjetas de crédito como: Master Card, American Express, Visa.

3 Técnicas

Cada uno de esto paradigmas incluye diferentes algoritmos y variaciones de los mismos, asi como otro tipo de restricciones que hacen que la efectividad del algoritmo depende del dominio de aplicacion, no existe lo que podıamos llamar el metodo universal aplicable a todo tipo de problema.

3.1 Regresion Lineal

La función de regresión más simple es la lineal (**RL**), cada variable explicativa participa de forma aditiva y constante para todo el dominio observado en la formación de la respuesta

$$E\left[y_i|x_{i1},\ldots,x_{in}\right] = \beta_0 + \beta_1 x_{i1} + \ldots + \beta_p x_{ip}.$$

Por tanto el modelo de regresión lineal se escribe

$$y_i = \beta_0 + \beta_1 x_{i1} + \ldots + \beta_n x_{in} + \epsilon_i.$$

La suposición de linealidad puede parecer muy restrictiva pero en realidad no lo es tanto. Si el número de individuos con que se cuenta es pequeño, los modelos lineales muestran una capacidad de generalización mayor que otros modelos más sofisticados y flexibles.

3.2 Naïve Bayes

Sin duda alguna se trata del modelo más simple de clasificación con redes bayesianas. En este caso, la estructura de la red es fija y solo necesitamos aprender los parámetros probabilísticos. El fundamento principal del clasificador Naïve Bayes (**NB**) es la suposición de que todos los atributos son independientes conocido el valor de la variable clase. A pesar de asumir esta suposición es sin duda bastante fuerte y poco realista en la mayoría de los casos, se trata de uno de los clasificadores más utilizados. Como un ejemplo de problema en el que el clasificador *NB* se esta mostrando como una de las técnicas más eficaces, podemos citar la lucha contra el correo basura o *spam*.

La hipótesis de independencia asumida por el clasificador *NB* da lugar a un modelo gráfico probabilístico en el que existe un único nodo raíz (la clase), y en el que todos los atributos son nodos hojas que tienen como único padre a la variable clase, gráficamente es la estructura de la figura 1.

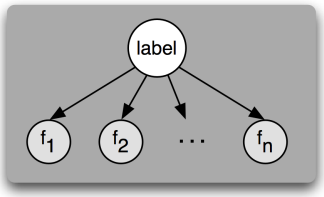

Figure 1: Grafo con independencia de los atributos.

Debido a la hipótesis de independencia usada en el NB, la expresión para obtener la hipótesis de probabilidad a posteriori dado por los atributos, es conocida como la hipótesis máxima a posteriori o hipótesis MAP (en inglés, *maximum a posteriori*) queda como sigue:

$$\begin{aligned} c_{MAP} &= \arg\max_{c\in\Omega_C} P\{A_1,\ldots,A_n|c\}P\{c\} \\ &= \arg\max_{c\in\Omega_C} P\{c\}\prod_{i=1}^{n} P\{A_i|c\} \end{aligned}$$

Por tanto, los parámetros que tenemos que estimar son $P\{A_i|c\}$ para cada atributo y la probabilidad a priori de la variable clase $P\{c\}$.

3.3 Árbol de decisión

Un árbol de decisión (**AD**) es un conjunto de condiciones organizadas en una estructura jerárquica, de tal manera que la decisión final a tomar se puede determinar siguiendo las condiciones que se cumplen desde la raíz del árbol hasta alguna de sus hojas. Una de las grandes ventajas de los árboles de decisión es que en su forma más general, las opciones a partir de una determinada condición son excluyentes. Esto permite analizar una situación y siguiendo el árbol de decisión apropiadamente, llegar a una sola acción o decisión a tomar.

Son muy útiles para encontrar estructuras en espacio de alta dimensionalidad y en problemas que mezclan datos categóricos y numéricos. Esta técnica se usa en tareas de clasificación, agrupamiento y regresión. Los árboles de decisión para predecir variables categóricas reciben el nombre de árboles de clasificación, ya que distribuyen las instancias en clases. Cuando los árboles de decisión se usan para predecir variables continuas se llaman árboles de regresión.

Los árboles de decisión siguen el método de "divide y vencerás" para partir el espacio del problema en subconjuntos. En la figura 2, encima del nodo raíz del árbol tenemos el problema a resolver. Los nodos internos (nodos de decisión) corresponden a particiones sobre atributos particulares y los arcos que emana de un nodo corresponden a los posibles valores del atributo considerado en ese nodo. Cada arco conduce a otro nodo de decisión o a un nodo hoja. Los nodos hojas representan la predicción o clase del problema para todas aquellas instancias que alcanzan esa hoja. Para clasificar una instancia desconocida se recorre el árbol de arriba hacia abajo de acuerdo a los valores de los atributos probados en cada nodo y, cuando se llega a una hoja, la instancia se clasifica con la clase

indicada por esa hoja.

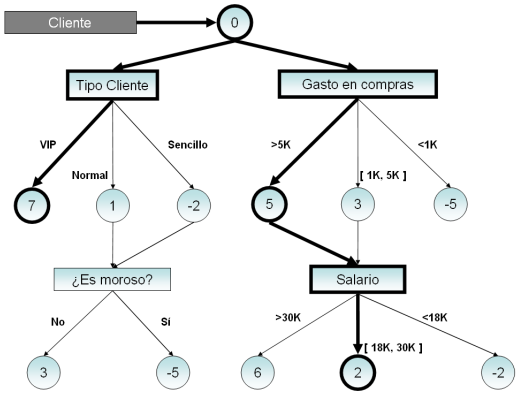

Figure 2: Árbol de decisión.

3.4 Red Neuronal

Normalmente modelamos una neurona biológica de la manera que se muestra en la figura 3, esta figura incluye una entrada externa adicional, denominada polarización o "bias" y denotada por θ_i, cuya finalidad es la de poder aumentar o disminuir el umbral de exitación de la neurona dependiendo de si es un valor positivo o negativo, respectivamente.

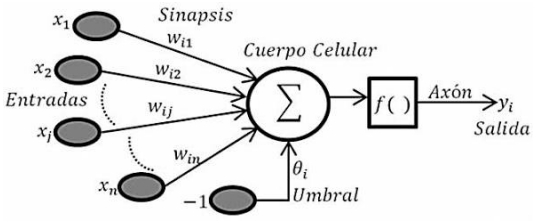

Figure 3: Representación de una neurona artificial.

Las entradas se presentan por el vector de entrada x, y el rendimiento de la sinapsis se modela mediante un vector de pesos w. Entonces el valor de salida de estas neurona viene dado por

$$y = f\left(\sum_{i=1}^{n} w_i x_i\right) = f(w^T x)$$

donde f es la función de activación. Cuando tenemos una red de neuronas, las salidas de unas se conectan con las entradas de otras.

Por tanto, podemos ver que una única neurona es una unidad de procesamiento muy simple. Se considera que el potencial de las redes neuronales artificiales proviene de la capacidad que proporciona el empleo de muchas de estas unidades simples y robustas al actuar en paralelo.

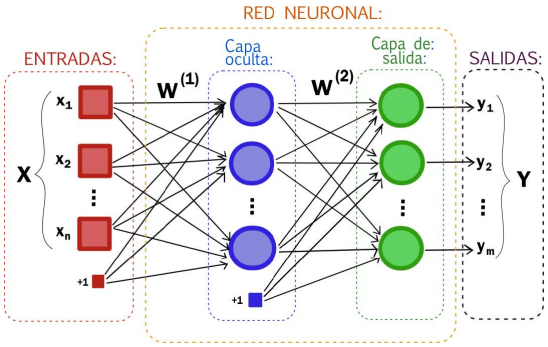

Figure 4: Red Neuronal Artificial.

En la figura 4 se observa un conjunto de entrada, el vector de entrada x, accediendo a la red desde el lado izquierdo y se propaga a través de la red hasta que la activación alcanza la capa de salida. Las capas intermedias son conocidas como las capas ocultas y que son invisibles desde fuera de la red. Hay dos modos de trabajar en una *RNA*:

- Modo de transferencia de la activación: cuando la activación es transmitida por toda la red. Este es el modo de funcionamiento o de aplicación y esta asociado a la operación de propagación hacia adelante.
- Modo de aprendizaje: cuando la red se organiza normalmente a partir de la transferencia de activación más reciente.

4 Datos

La base de datos *German Credit* correspondiente a créditos otorgados por un banco en el año 1994 en Alemania, la cual esta disponible en *UCI Machine Learning Repository* [14] de la University of California, Irvine, School of Information and Computer Sciences, a través del siguiente link: `https://archive.ics.uci.edu/ml/datasets/Statlog+\%28German+Credit+Data%29`

Ahora, veamos la descripción del conjunto de datos *German Credit*.

1. Título: Los datos de crédito alemana
2. Fuente de información:
   Profesor Dr. Hans Hofmann
   Institut f"ur Statistik und "Okonometrie
   Universit" at Hamburgo
   FB Wirtschaftswissenschaften
   Von-Melle-Park 5
   2000 Hamburgo 13.
3. Número de Instancias: 1000
   Se proporcionan dos conjuntos de datos. el conjunto de datos original, en la forma prevista por el profesor Hofmann, contiene atributos categóricos / simbólicos y se encuentra en el archivo *"german.data"*. Para algoritmos que requieren atributos numéricos, de la Universidad de Strathclyde producido el archivo *"german.data numérico"*. Este archivo se ha editado y varias variables indicadoras

añadido para que sea adecuado para algoritmos que no pueden hacer frente a las variables categóricas. Varios atributos que son ordenados por categorías (como atributo 17) han sido codificadas como enteros. Esta era la forma utilizada por Statlog.

4. Número de Atributos german.data: 20 (7 numérica, 13 categórica)
   Número de Atributos german.data-numeric: 24 (24 numérica)

5. Descripción Atributos de data de créditos alemán:

   - Atributo 1: (cualitativa) Estado de cuenta de cheques existente:
     A11: ... < 0 DM.
     A12: $0 \leqslant$ ... < 200 DM.
     A13: ...$\geqslant 200$ DM / asignaciones salariales para al menos 1 año.
     A14: Ninguna cuenta de cheques.
   - Atributo 2: (numérica) Duración en meses del atributo 1.
   - Atributo 3: (cualitativa) Historial de crédito:
     A30: No hay créditos tomados / todos los créditos pagados debidamente.
     A31: Todos los créditos en este banco pagado debidamente.
     A32: Créditos existentes pagadas volver debidamente hasta ahora.
     A33: Retraso en el pago de en el pasado.
     A34: Descripción crítica / otros créditos existentes (no en este banco).
   - Atributo 4: (cualitativa) Propósito:
     A40: Coche (nuevo).
     A41: Coche (utilizado).
     A42: Muebles / equipamiento.
     A43: Radio / televisión.
     A44: Electrodomésticos.
     A45: Reparaciones.
     A46: Educación.
     A47: Vacaciones.
     A48: Reentrenamiento.
     A49: Negocio.
     A410: Otros.
   - Atributo 5: (numérica) Cantidad de crédito.
   - Atributo 6: (cualitativa) Cuenta de ahorros / bonos:
     A61: ... < 100 DM.
     A62: $100 \leqslant$ ... < 500 DM.
     A63: $500 \leqslant$ ... < 1000 DM.
     A64: ...$\geqslant 1000$ DM.
     A65: Desconocido / no cuenta de ahorro.
   - Atributo 7: (cualitativa) Empleo actual desde:
     A71: Desempleados.
     A72: ... < 1 año.
     A73: $1 \leqslant$... < 4 años.
     A74: $4 \leqslant$ ... < 7 años.
     A75: ...$\geqslant 7$ años.
   - Atributo 8: (numérica) Tasa de Cuota en porcentaje de la renta disponible.
   - Atributo 9: (cualitativa) Estado civil y el sexo:
     A91: Hombre: Divorciado / separado.
     A92: Mujer: Divorciado / separado / casó.
     A93: Hombre: soltero.
     A94: Hombre: casado / viudo.
     A95: Mujer: soltero.
   - Atributo 10: (cualitativa) Otros deudores / garantes:
     A101: Ninguno.
     A102: Co-solicitante.
     A103: Garante.
   - Atributo 11: (numérica) Residencia actual desde (año).
   - Atributo 12: (cualitativa) Propiedad:
     A121: Inmobiliario.
     A122: Si no A121: la construcción de un acuerdo de ahorro de la sociedad / seguro de vida.
     A123: Si no A121 / A122: coche u otro, no en el atributo 6.
     A124: Desconocido / no propiedad.
   - Atributo 13: (numérica) Edad en años.
   - Atributo 14: (cualitativa) Otros planes de pago:
     A141: Banco.
     A142: Las tiendas.
     A143: Ninguno.
   - Atributo 15: (cualitativa) Alojamiento:
     A151: Alquiler.
     A152: Propio.
     A153: Gratis.
   - Atributo 16: (numérica) Número de créditos existentes en este banco.
   - Atributo 17: (cualitativa) Trabajo:
     A171: No calificada desempleados / - no residente.
     A172: No calificada - residente.
     A173: Trabajador cualificado / oficial.
     A174: Gestión / cuenta propia / empleado altamente cualificado / oficial.
   - Atributo 18: (numérica) Número de personas que son responsables de proporcionar la pensión alimenticia.
   - Atributo 19: (cualitativa) Teléfono:
     A191: Ninguno.
     A192: Sí, registrada bajo el nombre de los clientes.
   - Atributo 20: (cualitativa) Trabajador extranjero
     A201: Sí.
     A202: No.
   - Atributo 21: (numérica) Cliente fiable para concederle crdito o no:
     1 : Bueno.
     2 : Malo.

6. En el punto 3 se hace la observación que la data "german.data-numeric" se edita y agrega indicadores. Los indicadores se agregan de las columnas

16 y 17 para el atributo 4 que tiene 3 opción es decir si las columnas 16 y 17 son ceros se considera la tercera opción, de igual manera para las columnas 18 y 19 con el atributo 10, las columnas 20 y 21 con el atributo 15 y las columnas 22, 23 y 24 con atributo 17.

5 Implementación

Para la implementación de los métodos aplicados en la minería de datos se utiliza el lenguaje de programación Python, que cuenta con librerías especializadas en los métodos expuestos. Scikit-learn es una librería de código abierto de *machine learning* para el lenguaje de programación Python. Cuenta con diversos clasificación, regresión y algoritmos de agrupamiento que incluye máquinas de vectores soporte, bosques aleatorios, k-means y DBSCAN, y está diseñado para interactuar con las librerías numéricas y científicas de Python como NumPy, MatPlotLib y SciPy

La página web de scikit-learn es `http://scikit-learn.org/dev/index.html`, donde esta como realizar la instalación de la librería y toda la documentación, para los métodos de regresión lineal, naïve bayes y árboles de decisión. Para el método de redes neuronales artificiales utilizamos la librería scikit-neuralnetwork página `http://scikit-neuralnetwork.readthedocs.org/en/latest/guide_beginners.html` Ahora, veamos los pasos que necesitamos para la implementación:

- Primero necesitamos las librerias necesarias: scikit-learn (modelos de regresión lineal, naïve bayes y árboles de decisión), scikit-neuralnetwork (modelo de redes neuronales artificiales), numpy, matplotlib, pandas y random.
- Cargar los datos del archivo *"german.data-numeric"* y definir los conjuntos de entrenamiento y prueba utilizando.
- Inicializar los modelos de regresión lineal, naïve bayes, árboles de decisión y redes neuronales artificiales.
- Entrenar cada uno de los modelos inicializados.
- Con el modelo entrenado utilizar el conjunto de prueba para predecir de manera simple y cruzada sus posibles valores.
- Calcular el *score* de cada modelo, donde su mejor puntuación posible es de 1.0 y puede ser negativo (debido a que el modelo puede ser malo). Un modelo que siempre predice el valor esperado sin tener en cuenta las características de entrada, obtendría una puntuación de 0.0.
- Calcular el error cuadrático de cada uno de los modelos.
- Para los modelo de regresión lineal los valores predecidos son valores reales pero los valores deseados son binarios (1 y 2), por lo cual debemos transformar lo valores predecido, para lo cual diremos que si es menor que 1.5 el valor es 1 y si es mayor el valor es 2.
- Se puede imprimir un reporte de las predicciones realizadas y la matriz de confusión del conjunto de prueba.
- Guardar en un archivo .txt los valores predecidos del conjunto de prueba con la validación simple y cruzada.
- Utilizar la librería matplotlib para plotear la comparación de las predicciones con los valores conocidos del conjunto de prueba.

Las implementaciones se pueden visualizar y descargar en el siguiente repositorio de github: `https://github.com/Kanino19/MineriaDeDatos.git`

6 Estudio de las Variables

Un estudio preliminar de las variables para verificar si existe alguna dependencia o correlación se puede utilizar para identificar que variables utilizar en la siguiente fase, aplicación del modelo predictivo que se elija, lo cual puede reducir la cantidad de variables a utilizar lo cual reducirá el tiempo que le llevara a la implementación del modelo predictivo estudiar los datos y tener un resultados final. Con respecto a las variables de *german.data* haremos un estudio preliminar de las variables $A1$ con $A3$ y $A3$ con $A11$.

```
jkahn@Sofia:~$ python Regre_compa.py
Metodo de Regresion Lineal para Comparacion

******** Entrenamiento Simple ********
Coeficiente: 1.4856
Score: 0.26
Suma residual de cuadrados: 387.73
```

Figure 5: Resultados de la comparacion entre la variable A1 y A3

```
jkahn@Sofia:~$ python Regre_compa.py
Metodo de Regresion Lineal para Comparacion

******** Entrenamiento Simple ********
Coeficiente: 1.4856
Score: 0.26
Suma residual de cuadrados: 387.73
```

Figure 6: Resultados de la comparacion entre la variable A3 y A11

Es claro por el gráfico 7 entre $A1$ y $A3$ podemos apreciar una relación importante, pero respecto al gráfico 8 entre $A3$ y $A11$ no se aprecia ninguna relación.

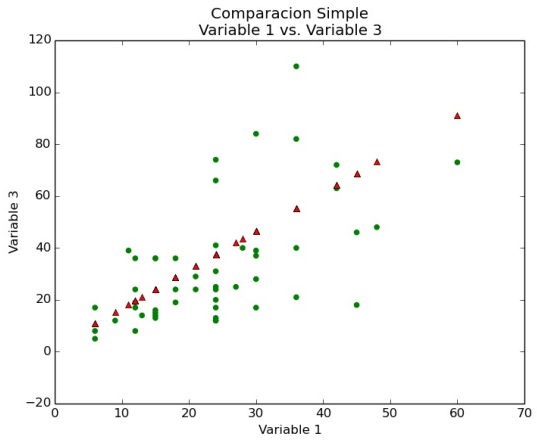

Figure 7: Variable A1 vs. Variable A3

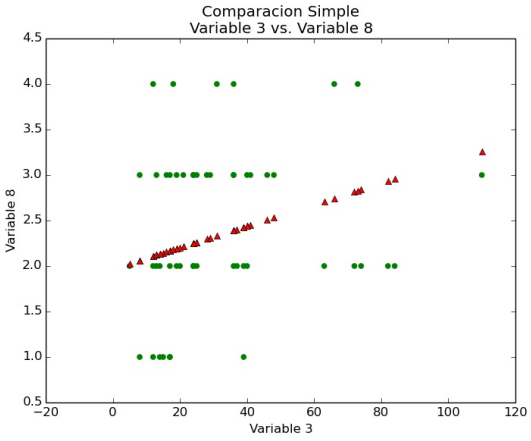

Figure 8: Variable A3 vs. Variable A11

Mediante este estudio previo para encontrar relación entre las variables, se puede reducir la cantidad que se utilizaran al crear el modelo de minería de datos, lo cual reducirá el tiempo de ejecución. Para los entrenamiento posteriores se utilizara la totalidad de las variables, debido a la poca cantidad de registros con las que se dispone.

7 Resultados

Los siguientes resultados corresponden a la implementación para cada método, regresión lineal en las figuras 9, 10 y 11, naïve bayes en las figuras 12, 13 y 14, árbol de decisión en las figuras 15, 16 y 17, y red neuronal artificial en las figuras 18, 19 y 20. Se utiliza la base de datos "*german.data-numeric*". Realizando dos tipos de entrenamiento del modelo validad simple y cruzada, para el primero se considera un 5% de los registros al azar para probar el modelos, es decir un 95% para el entrenamiento. Para el segundo se considera la validación con 10 pliegues y después de la predicción se considera el 5% de la validación simple, para realizar una comparación entre las validaciones.

```
jkahn@Sofia:~$ python Regresion.py
Metodo de Regresion Lineal

******** Entrenamiento Simple ********
Score: 0.21
Suma residual de cuadrados: 0.16
             precision    recall  f1-score   support

   C. Bueno       0.82      0.92      0.87        36
    C. Malo       0.70      0.50      0.58        14

avg / total       0.79      0.80      0.79        50

          Predic C. Bueno  Predic C. Malo
C. Bueno               33               3
C. Malo                 7               7

******** Entrenamiento 10-Cruzado ********
Score: 0.20
Suma residual de cuadrados: 0.17
             precision    recall  f1-score   support

   C. Bueno       0.82      0.92      0.87        36
    C. Malo       0.70      0.50      0.58        14

avg / total       0.79      0.80      0.79        50

          Predic C. Bueno  Predic C. Malo
C. Bueno               33               3
C. Malo                 7               7
jkahn@Sofia:~$
```

Figure 9: Resultados de Regresión Lineal

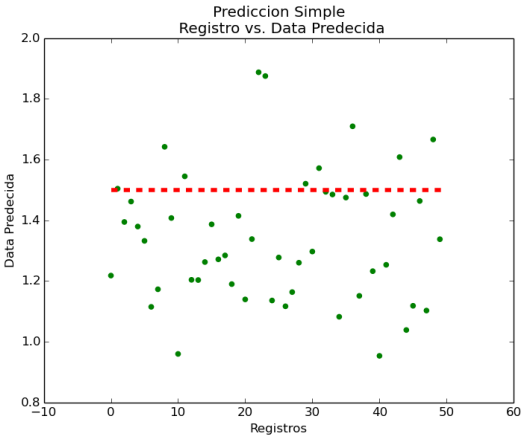

Figure 10: Registro vs. Data predecida

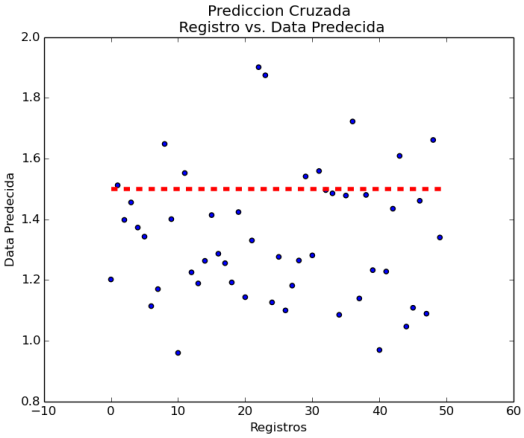

Figure 11: Gráficos de regresión lineal Registro vs. Data predecida

```
jkahn@Sofia:~$ python Naivebayes.py
Metodo de Naive Bayes

******** Entrenamiento Simple ********
Score: 0.74
Suma residual de cuadrados: 0.26
             precision    recall  f1-score   support

   C. Bueno       0.85      0.78      0.81        36
    C. Malo       0.53      0.64      0.58        14

avg / total       0.76      0.74      0.75        50

          Predic C. Bueno  Predic C. Malo
C. Bueno               28               8
C. Malo                 5               9

******** Entrenamiento 10-Cruzado ********
Score: 0.73
Suma residual de cuadrados: 0.27
             precision    recall  f1-score   support

   C. Bueno       0.82      0.75      0.78        36
    C. Malo       0.47      0.57      0.52        14

avg / total       0.72      0.70      0.71        50

          Predic C. Bueno  Predic C. Malo
C. Bueno               27               9
C. Malo                 6               8
jkahn@Sofia:~$
```

Figure 12: Resultados de Naïve Bayes

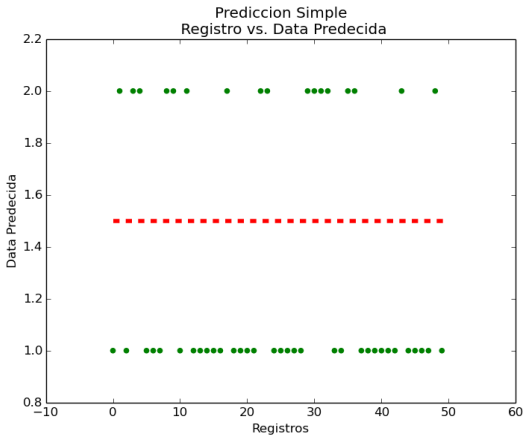

Figure 13: Gráficos de Naïve Bayes Registro vs. Data predecida

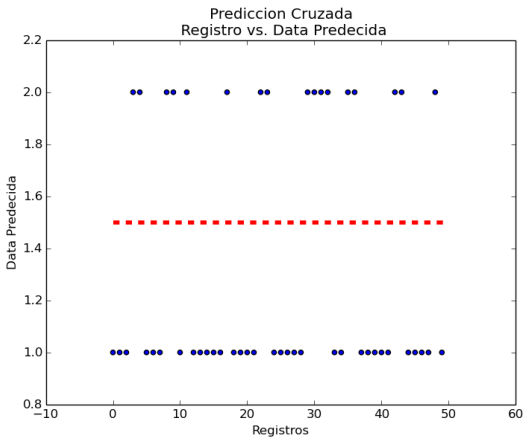

Figure 14: Gráficos de Naïve Bayes Registro vs. Data predecida

```
jkahn@Sofia:~$ python Arbol.py
Metodo de Arboles de Decision

******** Entrenamiento simple ********
Score: 0.70
Suma residual de cuadrados: 0.30
             precision    recall  f1-score   support

   C. Bueno       0.78      0.81      0.79        36
    C. Malo       0.46      0.43      0.44        14

avg / total       0.69      0.70      0.70        50

          Predic C. Bueno  Predic C. Malo
C. Bueno               29               7
C. Malo                 8               6

******** Entrenamiento cruzado ********
Score: 0.68
Suma residual de cuadrados: 0.32
             precision    recall  f1-score   support

   C. Bueno       0.81      0.81      0.81        36
    C. Malo       0.50      0.50      0.50        14

avg / total       0.72      0.72      0.72        50

          Predic C. Bueno  Predic C. Malo
C. Bueno               29               7
C. Malo                 7               7
jkahn@Sofia:~$
```

Figure 15: Resultados de Árboles de Decisión

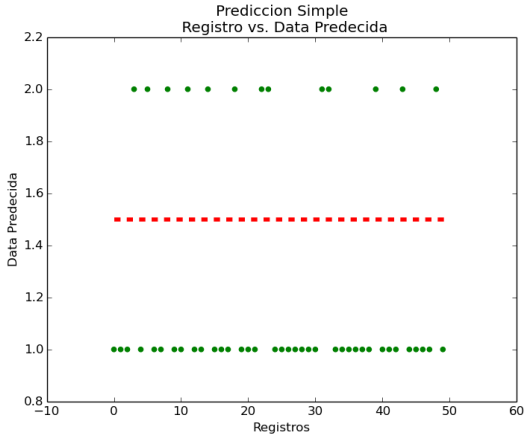

Figure 16: Gráficos de Árboles de Decisión Registro vs. Data predecida

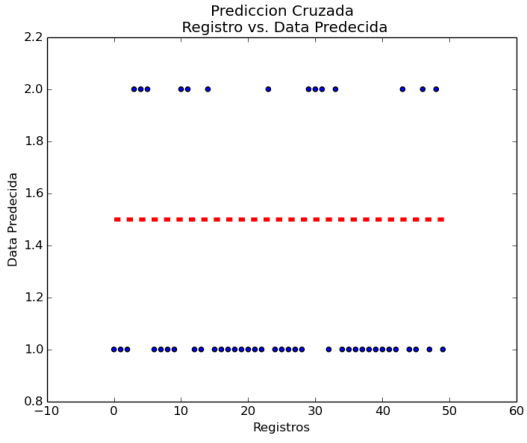

Figure 17: Gráficos de Árboles de Decisión Registro vs. Data predecida

```
jkahn@Sofia:~$ python Back.py
Metodo de Backpropagation
No handlers could be found for logger "sknn"

******** Entrenamiento simple ********
Score: 0.74
Suma residual de cuadrados: 0.39
             precision    recall  f1-score   support

   C. Bueno       0.81      0.83      0.82        36
    C. Malo       0.54      0.50      0.52        14

avg / total       0.73      0.74      0.74        50

          Predic C. Bueno  Predic C. Malo
C. Bueno               30               6
C. Malo                 7               7

******** Entrenamiento cruzado ********
Score: 0.73
Suma residual de cuadrados: 0.37
             precision    recall  f1-score   support

   C. Bueno       0.83      0.81      0.82        36
    C. Malo       0.53      0.57      0.55        14

avg / total       0.75      0.74      0.74        50

          Predic C. Bueno  Predic C. Malo
C. Bueno               29               7
C. Malo                 6               8
jkahn@Sofia:~$
```

Figure 18: Resultados de Backpropagation

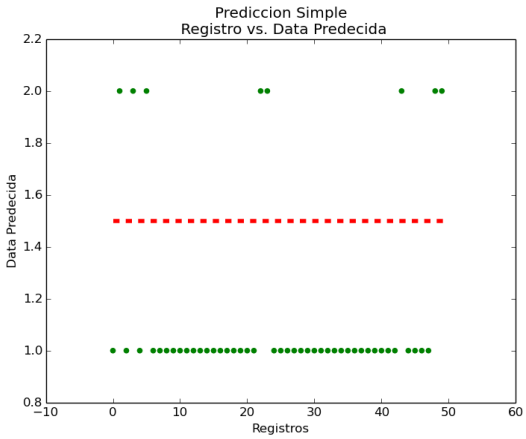

Figure 19: Gráficos de Backpropagation Registro vs. Data predecida

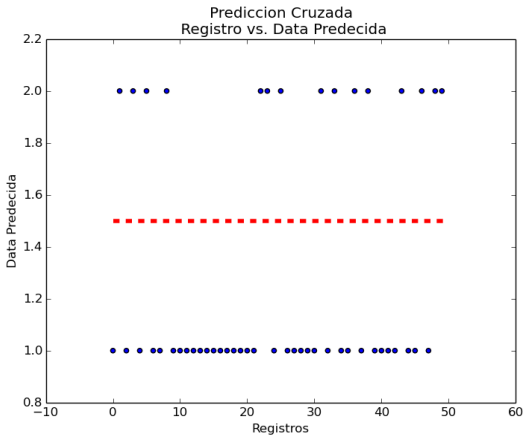

Figure 20: Gráficos de Backpropagation Registro vs. Data predecida

8 Análisis de los Resultados

Dividiremos el análisis de los resultados de las implementaciones para el conjunto de prueba anteriores en dos, la comparación entre validación simple y cruzada para cada modelo, y la comparación entre los modelos para determinar el más adecuado en la predicción de fiabilidad crediticia con los datos disponibles. Luego el análisis de cada resulta obtenido por cada modelos.

- **Validación Simple vs. Cruzado**
  Para cada uno de los modelos implementados se observa que el score con la validación simple es aproximada a la validación cruzada con 10 pliegues, pero mayor (cuadro 1). Para el modelo de RL y NB la precisión es mayor igual con la validación simple que con la validación cruzada, a diferencia de los modelos de AD y RNA (cuadro 3). Al igual que el score y la precisión la suma residual es aproximada, pero menor valor con la validación simple que con la cruzada (cuadro 2), lo cual quiere decir que los valores predecidos son mas fiables con la validación simple que con la validación cruzada. En el modelo de RL el score y la precisión para la validación simple y cruzada tienen valores iguales, respectivamente, pero la suma residual no lo es, debido a que los valores predecidos para el modelo de regresión lineal son continuos y no binarios, a los cuales se les aplica un tratamiento posterior para clasificarlos de forma binaria. Además para los métodos la cantidad de predicciones de clientes buenos y malos pueden coincidir pero no ser las mismas predicciones correcta, lo cual se puede evidenciar en los cuadros 9, 12, 15 y 18.

- **Comparación de los modelos**
  Los modelos con mayor score son NB, AD y RNA, con valores entre 0.65 y 0.75, siendo con mayor score los métodos de NB y RNA a diferencia de la RL que su score esta entre 0.20 y 0.21 (cuadro 1). Las precisiones para todos los modelos se encuentran entre 0.69 y 0.79, siendo con mayor precisión los modelos de RL y NB (cuadro 3). La suma residual se encuentra entre 0.16 y 0.39.

Además, los modelos expuestos e implementados predicen mejor a los clientes buenos que a los clientes malos, con porcentajes de aciertos entre 0.75 y 0.92 de predicción correctas de clientes buenos (cuadros 5) y porcentajes de aciertos entre 0.43 y 0.64 de predicción correctas de clientes malos (cuadros 6), por lo cual en ponderado el porcentaje de acierto esta entre 0.70 y 0.80, siendo el mayor el modelo de RL, NB y RNA con la validación simple(cuadro 4).

Validación	RL	NB	AD	RNA
Simple	0.21	0.74	0.70	0.74
Cruzada	0.20	0.73	0.68	0.73

Table 1: Score

Validación	RL	NB	AD	RNA
Simple	0.16	0.26	0.30	0.39
Cruzada	0.17	0.27	0.32	0.37

Table 2: Suma residual de cuadrados

Validación	RL	NB	AD	RNA
Simple	0.79	0.76	0.69	0.73
Cruzada	0.79	0.72	0.72	0.75

Table 3: Precisión

Validación	RL	NB	AD	RNA
Simple	0.80	0.74	0.70	0.74
Cruzada	0.80	0.70	0.72	0.74

Table 4: Porcentaje de predicciones correctas

Validación	RL	NB	AD	RNA
Simple	0.92	0.78	0.81	0.83
Cruzada	0.92	0.75	0.81	0.81

Table 5: Porcentaje de predicciones correctas cliente bueno

Validación	RL	NB	AD	RNA
Simple	0.50	0.64	0.43	0.50
Cruzada	0.50	0.57	0.50	0.57

Table 6: Porcentaje de predicciones correctas cliente malo

9 Conclusiones

En general, los modelos aquí presentados predicen mejor a los cliente buenos que a los cliente malos, lo cual nos indica que es más adecuado considerar el criterio de si el cliente es clasificado como cliente bueno.

Los modelos con mayor score y precisión, y menor suma residual son los más fiables, y los que se deben considerar al momento de la elección del método a utilizar en la predicción de fiabilidad crediticia. Por lo cual, los mejores modelos para predecir la clase de cliente bueno son Naïve Bayes y Redes Neuronales artificiales, mientras que el modelos de Naïve Bayes predice mejor la clase de clientes malos. A lo cual se suma que los modelos expuestos tienen una mayor fiabilidad al entrenarlos con el método de validación simple.

Sin olvidar que los resultados obtenidos son basados en la utilización de todas las variables expresadas en la data "german.data-numeric", lo cual da una mayor grado de libertad a la hora de predecir la fiabilidad crediticia de un cliente. Así para trabajos futuros, hacer un análisis de las variables que tienen mayor relevancia dentro de la data para reducir la cantidad de variables con las que se trabaja, es decir reducir el grado de libertad del problema.

Referencias

1. L. C. Thomas. A survey of credit and behavioural scoring: forecasting financial risk of lending to consumers. *International Journal of Forecasting*, 16:149–172, 2000.
2. Andrés Yesid Ramírez A. Técnicas de minería de datos aplicadas a la construcción de modelos de score crediticio: Estado del arte. *Universidad Nacional de Colombia*, 2007.
3. Y. Yang. Adaptive credit scoring with kernel learning methods. *European Journal of Operational Research*, 183:1521–1536, 2007.
4. H. Abdou, J. Pointon, and A. El-Masry. Neural nets versus conventional techniques in credit scoring in egyptian banking. *Expert Systems with Applications*, In Press:1606, 2007.
5. V. S. Desai, J. N. Crook, and G. A. Overstreet. A comparison of neural networks and linear scoring models in the credit union environment. *European Journal of Operational Research*, 95:24–37, 1996.
6. T. S. Lee and I. F. Chen. A two-stage hybrid credit scoring model using artificial neural networks and multivariate adaptive regression splines. *Expert Systems with Applications*, 28:743–752, 2005.
7. T. S. Lee, C.-C. Chiu, C.-J. Lu, and I.-F. Chen. Credit scoring using the hybrid neural discriminant technique. *Expert Systems with Applications*, 23:245–254, 2002.
8. R. Malhotra and D. K. Malhotra. Evaluating consumer loans using neural networks. *Omega*, 31:83–96, 2003.
9. D. West. Neural network credit scoring models. *Computers and Operations Research*, 27:1131–1152, 2000.
10. S. T. Li, W. Shiue, and M. H. Huang. The evaluation of consumer loans using support vector machines. *Expert Systems with Applications*, 30:772–782, 2006.
11. C. L. Huang, M. C. Chen, and C. J. Wang. Credit scoring with a data mining approach based on support vector machines. *Expert Systems with Applications*, 33:847–856, 2006.
12. N. C. Hsieh. Hybrid mining approach in the design of credit scoring models. *Expert Systems with Applications*, 28:655–665, 2005.
13. Petya Platikanova. El análisis económico-financiero: Estado del arte. *Revista de Contabilidad y Dirección*, 2:95–120, 2005.
14. M. Lichman. UCI machine learning repository, 2013.