**MEMORIA: Ganancias y gastos de familias filipinas**

Artem Vartanov y Mario Baldocchi

Titulos a tamaño 14, texto tamaño 12

**Introducción**

Los [datos](https://www.kaggle.com/datasets/grosvenpaul/family-income-and-expenditure) que estudiaremos reflejan las ganancias y gastos de diversas familias filipinas, además de ciertos aspectos importantes a la hora de reforzar nuestra idea sobre el poder adquisitivo de las mismas.

Esta base de datos se podría emplear en diversos ámbitos laborales, pero supondremos que nuestro cliente es un banco, al que le interesa poder saber cuándo conceder un préstamo o denegarlo. Para poder hacer esto último, una de las principales variables a tener en cuenta es el dinero total que ingresa cada unidad familiar. Sin embargo, en Filipinas mucha gente trabaja en negro, por lo que no puede uno realmente fiarse de lo que el cliente deposita en la cuenta. Es por ello que, basándonos en los gastos y otros datos personales, trataremos de predecir el sueldo

anual (en pesos filipinos, que a 30/11/2023 el cambio es de 1₱ = 0,017€) de una unidad familiar.

**Descripción de los datos**

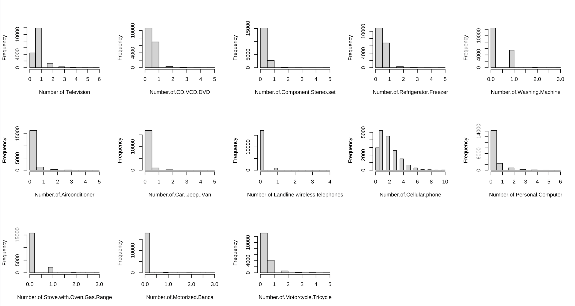
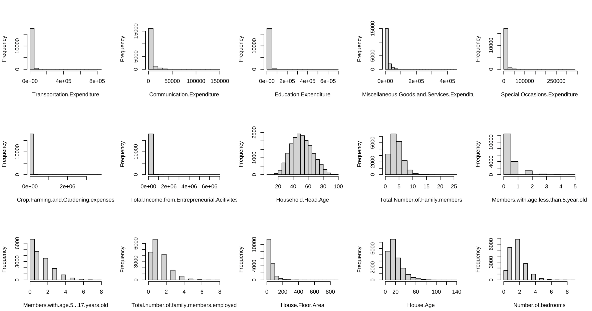
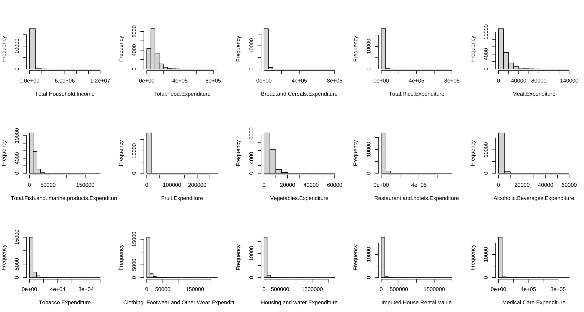
La base de datos se conforma de 18022 clientes y 60 variables, de las cuales 43 son nominales y 17 categóricas. Entre las distintas variables nominales se puede observar que aproximadamente un 25% están relacionadas con desembolsos monetarios, como por ejemplo los gastos médicos, en ropa o en comida. El resto describen la familia y la vivienda en la que residen, así como datos personales de la persona con mayor sueldo.

**Preprocesado**

Una vez importados los datos, lo primero que debe hacerse es echar un vistazo por encima. De lo primero que se percata uno, es que hay variables categóricas "escondidas", entre las que se encuentra, por ejemplo, Electricity. Y es que, aunque en esta columna solo haya números (0 y 1), estos son utilizados para indicarnos si el cliente en cuestión tiene electricidad en casa o no, es decir, se podría sustituir el 1 por un "Sí" y el 0 por un "No". Consecuentemente, hay que ir columna por columna comprobando que ninguna variable categórica se haga pasar por otro tipo de variable, y en el caso de localizar alguna, realizaremos su conversión a factor con as.factor <líneas 20-36>.

Seguimos analizando. Vemos que hay solo dos columnas que tienen valores N/A´s, *Household.Head.Occupation* y *Household.Head.Class.of.Worker,* que representan el trabajo del miembro de la familia con las mayores ganancias y para quién trabaja respectivamente. Ante esta situación se nos presentan varias opciones. Por un lado, se puede optar por la eliminación de todas aquellas filas que contengan un N/A (3355 en total). Por el otro, podría ser interesante prescindir de ambas columnas. En este caso, elegiremos la segunda opción por dos motivos principales. En primer lugar, tanto la una como la otra no aportan información relevante frente al resto de variables, por lo que es preferible perder dicha información a "sacrificar" un cliente. En segundo lugar, al ser dos variables categóricas no formarán parte del modelo de regresión, el cual es una de nuestras prioridades. Es verdad que si estuviésemos tratando con un banco que facilitase muy pocos datos de los clientes (no es nuestro caso), cabría la posibilidad de optar por quitar todas aquellas filas con algún N/A para posteriormente convertir las variables categóricas en "dummy", pudiendo así ajustar mejor el modelo de regresión.

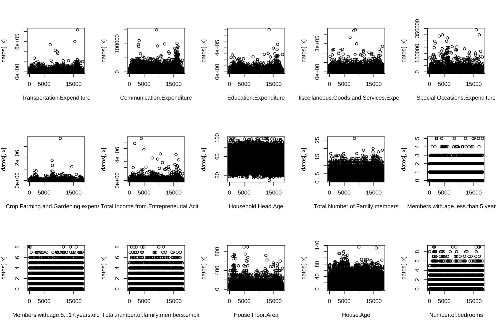
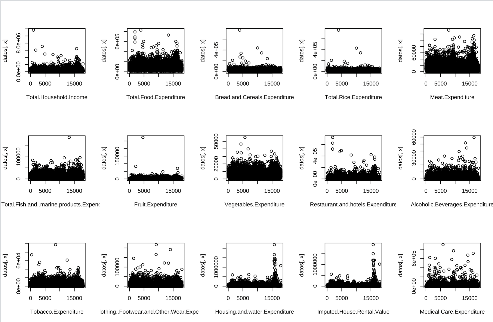
Para finalizar con el preprocesado, es recomendable prestar atención en cómo se distribuyen los datos. Para ello, podemos comenzar generando un histograma de cada una de las variables nominales.

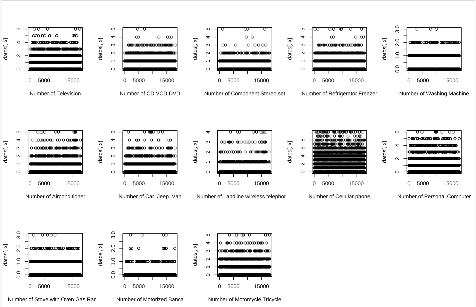


*(Histogramas de todas las variables nominales)*

De esta manera, podemos percatarnos, por ejemplo, de que todos aquellos atributos que representan algún tipo de gasto tienen algo en común: son asimétricos hacia la derecha. Esto es algo completamente lógico, pues el gasto que realiza una familia solo puede encontrarse en el intervalo (0, ∞).

Además de la simetría, es de especial interés conocer si existen datos atípicos o no. Para esto último, en vez de generar histogramas, generaremos un diagrama de dispersión.





*(Diagramas de dispersión de todas las variables nominales)*

Como era de esperar, diversos atributos como presentan outliers. Al estar tratando con variables simétricas y asimétricas, la eliminación de los datos atípicos puede resultar compleja. En el caso de que la variable sea simétrica, simplemente se extraerán todos aquellos datos que no se encuentren en el intervalo

[q1 - coefIQR\*iqr, q3 + coefIQR\*iqr], donde coefIQR es igual a 1,5. Por otro lado, si la variable no es simétrica, hay que realizar un paso previo. Este consiste en transformar la variable, mediante box-cox, a una normal.

Una vez realizados todos los pasos anteriormente mencionados, nos queda una base de datos limpia de N/A´s y outliers, donde la categoría de todas las variables es la correcta.

**Regresión Logística**

La semana pasada se ha producido uno de los peores terremotos de la historia de Filipinas, dejando gran parte del país destrozado. Consecuentemente, muchas familias han perdido sus casas. Este acontecimiento ha hecho que el banco haya visto una oportunidad de oro para aumentar su capital. Es por ello que ha puesto en marcha el PRVF (plan para la reconstrucción de viviendas en Filipinas), un préstamo de 90000₱ (1485€) al que cualquier familia filipina puede acceder siempre y cuando cumpla una condición: la renta per cápita mensual (ingresos mensuales de la familia / número de miembros en la familia) debe ser mayor o igual al 5% (4500₱) del préstamo.

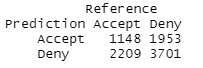
Para poder realizar el modelo de regresión logística utilizaremos la misma base de datos que hasta ahora, con una pequeña diferencia, y es que *Total Household Income* se transforma en una variable categórica, donde los únicos posibles valores que puede tomar son “Accept”, si la familia cumple la condición, o “Deny”, en caso contrario.

Antes de comenzar a entrenar el modelo, vamos a dividir nuestros datos en dos conjuntos:

-train: se utiliza para entrenar el modelo.

-test: se emplea para comprobar el modelo.

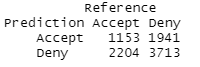
En el primer modelo vamos a tener en cuenta todos los atributos nominales disponibles en la base de datos. El modelo resultante se aproxima al lanzamiento de una moneda, por lo que no es interesante.





*(Datos del modelo)*

Para tratar de mejorarlo, podemos quitar las variables con un p-valor muy alto. Sin embargo, esta acción tan solo mejora el modelo en un 0,19%, tal y como se puede ver en la siguiente imagen.

****

****

*(Datos del modelo)*

Antes de conformarnos y quedarnos con este modelo, comprobemos si hay uno mejor. Para ello aprovechémonos de la validación cruzada, la cual realizaremos con 20 grupos (k = 20). Se puede observar como el modelo ha mejorado muy notablemente.

****

****

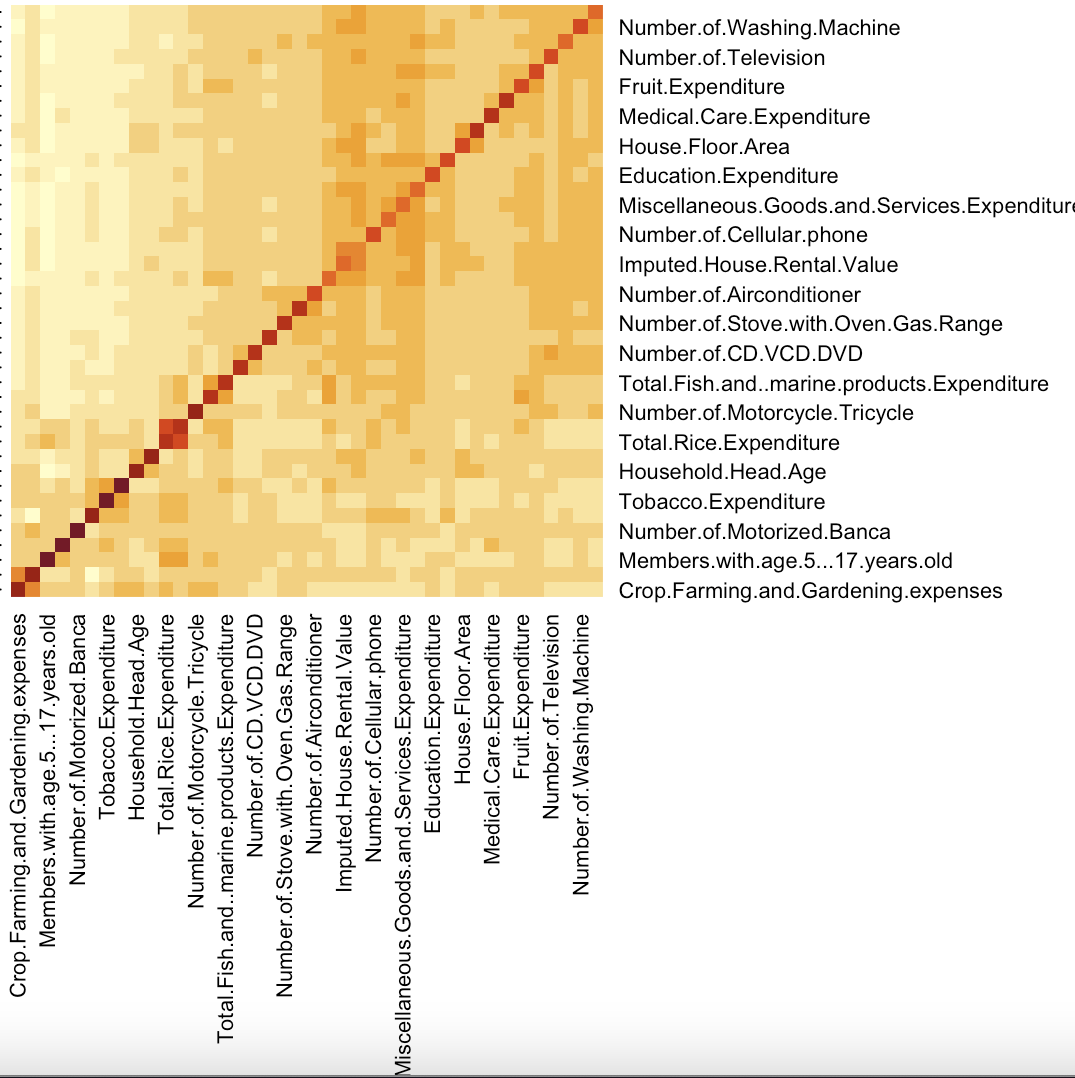
*(Datos del modelo obtenido mediante validación cruzada)*

Y es que no solo es bueno por la exactitud a la hora de clasificar a los clientes del banco, sino también porque tiene una mayor tendencia a clasificar a los “Accept” como “Deny” que los “Deny” como “Accept”. Esto último es de especial relevancia para el banco, pues

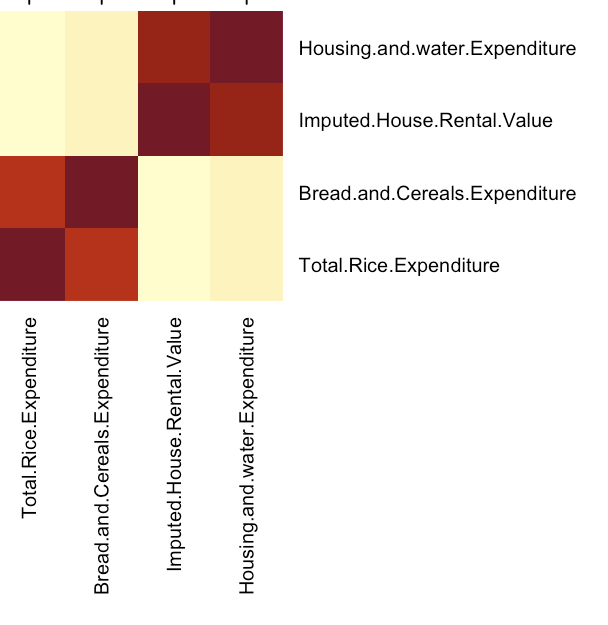
**Correlaciones**

explicar por que nos quedamos con correlacion de pearson, porque muestras grandes y al parecer no son normales

Al construir el modelo de regresión con las variables numéricas, es necesario eliminar las columnas categóricas. Antes de comenzar a estudiar las correlaciones, ese seleccionan todas aquellas variables que cuyas las variables correlacionadas con la variable respuesta. Pero antes de ello, primero quitamos las columnas no numericas.

\*\*\*FALTA TEXTO\*\*\*

Estudiaremos la correlación entre todas las variables, menos la respuesta. Viendo el mapa de calor de la derecha, es muy difícil sacar conclusiones. Nos quedamos aquellas que columnas tengan correlacion mayor de 0.7, siendo esta relación fuerte. Nos salen las siguientes columnas:

Housing.and.water.Expenditure, Imputed.House.Rental.Value, Bread.and.Cereals.Expenditure, Total.Rice.Expenditure.

A la derecha esta el mapa de calor, vemos que están altamente correlacionadas por parejas.

Quitamos al miembro de la pareja, que tenga MENOR correlación con la variable respuesta:

Total.Rice.Expenditure e Imputed.House.Rental.Value

Al analizar otra vez la tabla de correlaciones, vemos que ya no hay columnas cuyo coeficiente de correlacion de Spearman es mayor que 0.7 en valor absoluto.

Por último, de las columnas que quedaron en el dataframe sin autocorrelaciones, seleccionamos las que mas se correlacionan con la variable respuesta(> 0.7 en abs). Las guardamos para usarlas adelante

**Regresión lineal múltiple**

**Modelo completo (RLM\_FULL)**

Generamos el modelo de regresión múltiple que usará todas las variable numéricas y todo el conjunto de datos (incluyendo columnas autocorrelacionadas). En la *Figura 1* aparece el summary del modelo completo. Vemos que no todas las variables tienen el p-valor < 0.05 (valor prefijado para rechazar hipótesis nulas). Esto quiere decir que no podemos estar seguros de que el coeficiente de esa variable no sea 0, en otras palabras, que debería estar en el modelo esa variable.

Cabe destacar que el R2 ajustado es 0.7981 que mide la cantidad de variabilidad explicada por el modelo, teniendo en cuenta la penalización por introducir más variables en el modelo(usamos este criterio para poder contrastarlo con modelos posteriores, pues tendrán diferente número de variables). En este caso, como tenemos 42 predictores, la penalización es bastante alta.

Después de calcular la raíz cuadrada del error del modelo nos sale 132910.2, lo cual parece ser un error crítico.

**Modelo mejorado (RLM\_FIT)**

Primero intentamos construir un modelo a partir de las columnas altamente correlacionadas con la variable respuesta.

Despues intentamos construir un modelo con la funcion regsubsets, que nos devuelve para cada posible numero de variables la mejor selección de variables para disminuir el error. Ademas, junto con regsubsets le aplicamos regsubsets con 4 grupos, para evitar overfitting

Explicar lo de los tests de normalidad

Explicat box-cox

Meter mas criterios, cp de mallows, R cuadrado ajustado

**Modelos regularizados**

**Regularización de Tíjonov (Ridge)**

Esta regularización añade una penalización al método de mínimos cuadrados para hacer que los valores de coeficientes se acerquen a 0, pero no llega a eliminar predictores. Realizamos validación cruzada para encontrar el valor del parámetro necesario lambda para minimizar el error. Con este modelo nos ha dado un error cuad medio de xxx.

**Regularización Lasso**

Esta regularización añade una penalización al método de mínimos cuadrados para hacer que los valores de coeficientes se acerquen a 0, llegando a eliminar los predictores. Realizamos validación cruzada para encontrar el valor del parámetro necesario lambda para minimizar el error. Esto es favorable para nosotros, pues, además de predecir mejor que el modelo Ridge, nos quedamos con meno variables 28 de 38.

**Regularización Elastic**

Esta regularización realiza una mezcla entre los dos métodos anteriores(intentando sacar lo mejor de ambos métodos). El parámetro alpha es el que mide la inclinación hacia uno u otro. Si alpha = 1, entonces es lo mismo que Lasso, en caso de que alpha = 0, se trata de Ridge.

Lo que hacemos es generar modelos para 20 alphas que van de 0 a 1 con paso 0.05 (incluimos Lasso y Ridge). Realizamos validación cruzada para encontrar el valor del parámetro necesario lambda para minimizar el error, de modo que nos queda una tabla de 20 filas y 3 columnas: el alpha del modelo, el lambda que minimiza el error para el alpha dado, y el error del modelo teniendo a los dos anteriores como parámetros. Encontramos el error minimo de esa tabla que es ...

**Modelos PLS y PCR**

**PCR**

En este caso vamos a dejar las columnas autocorrelacionadas, pues no suponen un problema para estos métodos. A partir de la validación cruzada, queremos encontrar el modelo con el menor número de componentes que mejor explique la variabilidad de los datos. Nos da el modelo con xxx componentes.

**PLS**

En este caso vamos a dejar las columnas autocorrelacionadas, pues no suponen un problema para estos métodos. A partir de la validación cruzada, queremos encontrar el modelo con el menor número de componentes que mejor explique la variabilidad de los datos. Nos da el modelo con xxx componentes.

Tabla de resultados

|  |  |  |
| --- | --- | --- |
| MODELO | ERROR | N componentes |
| RLM\_FULL | N/A | N/A |
| N/A | N/A | N/A |