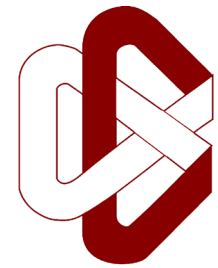


CIMAT

Centro de Investigación en Matemáticas, A.C.

Imputación del Ingreso en la Encuesta Nacional de Ocupación y Empleo

T E S I S

Que para obtener el grado de

Maestro en Ciencias en Estadística Oficial

P r e s e n t a

José de Jesús Luján Salazar

Director de Tesis:

Dr. Johan Van Horebeek

Guanajuato, Gto. Agosto de 2009

Integrantes del jurado:

M.C. Sergio Nava Muñoz

Presidente

Dra. Guillermina Eslava Gómez

Vocal

Dr. Johan Jozef Lode Van Horebeek

Vocal y director de tesis

Firma

A mi esposa

Nora

A mis hijos

Bruno de Jesús
y Ángela Montserrat

Agradecimientos

Gracias a el Dr. Johan Van Horebeek por su consejo y dirección en la elaboración de la tesis. Sus cualidades personales y técnicas fueron decisivas para la obtención de esta meta. Para él mi admiración y respeto.

Agradezco infinitamente a mi familia por su apoyo incondicional y por suplir las ausencias. A lo largo de toda la maestría compartimos el esfuerzo y cansancio. Es un logro también compartido. Para ellos mi más profundo amor.

Al Instituto Nacional de Estadística y Geografía (INEGI) que nos dio la oportunidad de participar en este programa.

A mis padres porque su ejemplo ha sido mi guía.

A mis hermanos por su respaldo y compañía en todos mis proyectos.

A los profesores Graciela González, Rogelio Ramos y Sergio Nava por compartir con nosotros su conocimiento pero por encima de todo por su gran calidad humana.

Resumen

En la Encuesta Nacional de Ocupación y Empleo (ENOE) una de las variables más importantes es el ingreso obtenido por la realización de un trabajo. La presencia de no respuestas de esta variable a nivel individual representa desafíos estadísticos y computacionales.

En la primera parte de tesis se presente un análisis exploratorio que permite entender mejor la relación que existe entre el ingreso y el resto de las variables presentes en el cuestionario. Para ese fin se explica el marco general de la encuesta y se desarrollan varios algoritmos que son capaces de manipular el volumen de datos que el ENOE genera utilizando como herramienta el paquete estadístico R.

En la segunda parte, se describen y se comparan de manera extensa las técnicas más importantes de imputación; son aplicadas para predecir ingreso promedio e indicadores derivados.

Índice general

1

2

3

4

5

6

7

A

B

1. Introducción

El análisis y recuperación de datos omitidos es un área de la estadística bien conocida. La manera habitual de tipificar la no respuesta se divide en: a) no respuesta total y b) no respuesta individual (*item nonresponse*). El primero se refiere a los casos en donde carecemos por completo de la información del objeto seleccionado en la muestra y normalmente se trata con técnicas que ajustan a los factores de expansión, por ejemplo una familia que se encuentra de vacaciones fuera del Estado durante el periodo de levantamiento, en consecuencia, no se puede captar los datos relativos a sus residentes. El segundo suele ser más variado y complejo, ocurre cuando carecemos de información en algunas de las preguntas que se realizan al elemento en muestra y se origina porque el informante se rehúsa contestar, desconoce el dato que se le solicita o porque la respuesta obtenida no satisface las restricciones que establece la validación, para ilustrar esta situación imaginemos un niño de 12 años que reporta como su último grado escolar aprobado el primer año de preparatoria, en nuestro país es algo que el sistema educativo no permite y en consecuencia la validación rechaza el dato. Para resolver la no respuesta individual usualmente se emplean algunas de las técnicas de imputación las cuales revisaremos en el capítulo 5.

En el INEGI, la Dirección de Marcos de Muestreo y Diseño Estadístico (DMMDE) es el área responsable de generar y actualizar el Marco Nacional de Viviendas (marco de muestreo), de la selección de muestras, ajuste de factores de expansión (fde) y cálculo de precisiones estadísticas. Cuando inicia un proyecto, colaboradores de la DMMDE analizan los resultados, los niveles de desagregación geográfica y la precisión que los usuarios desean obtener para proponer un tamaño de muestra, enseguida se selecciona la muestra, se recolecta la información en campo, se captura y válida; entonces, los datos son enviados de regreso a la DMMDE para efectuar el ajuste a los fde y, por último, son entregados al área de procesamiento para generar la estimación de resultados (tabulados, frecuencias, indicadores).

El ajuste de los fde buscaba en principio resolver la no respuesta total; sin embargo, recientemente se le ha agregado la responsabilidad de hacer que las estimaciones de los totales poblacionales sean comparables entre los diferentes proyectos que realiza el Instituto. Las cifras de referencia se llaman proyecciones de población y las establece un comité interinstitucional conformado por el Consejo Nacional de Población (CONAPO) y el INEGI. Los factores utilizados en este estudio cuentan con las dos correcciones. Una manera de ilustrar el ajuste del fde por no respuesta es interpolar linealmente del total de viviendas habitadas que tiene la i-ésima UPM entre las que respondieron la encuesta.

Sea:

$$f_i = \frac{1}{\pi_i} = \frac{N_i}{n_i}$$

$$f_i^* = f_i \frac{n_i'}{n_i^*}$$

Donde: f_i es el factor de expansión, N_i total de viviendas que conforma la i-ésima UPM, n_i número de viviendas seleccionadas de la i-ésima UPM, f_i^* factor de expansión ajustado por la no respuesta, n_i' es el conteo de viviendas habitadas (de las n_i seleccionadas inicialmente es probable que algunas no tengan uso habitacional, por ejemplo, son empleadas como negocio exclusivamente, están deshabitadas o demolidas), finalmente n_i^* es el número de viviendas en las que efectivamente obtuvimos respuesta.

En cuanto a la no respuesta individual, la Encuesta Nacional de Ocupación y Empleo (ENOE) utiliza la imputación deductiva [1] y aunque es la ideal, según *Kalton* (1986), en la realidad se puede aplicar en una cantidad limitada de características, en los casos restantes se asigna con un código especial (generalmente 9, 99 ó 9's) que identifica a los valores *No especificados*.

El objetivo principal de este estudio es implementar y comparar una serie de métodos de imputación que predigan los valores perdidos (no respuesta individual) referentes al ingreso por trabajo, obtenidos por la ENOE en particular con los datos de la ciudad de León Guanajuato. El reto más importante cuando hablamos de imputación se refiere a la posibilidad de utilizar la información adicional que se captó del caso en cuestión. Los datos que se disponen en la encuesta pueden verse desde distintos enfoques, por ejemplo, los ocupados se clasifican según su posición en el empleo: patrones, cuentas propias, subordinados, trabajador sin pago y otros trabajadores; para decidir a qué clase pertenece una persona en particular se tiene que analizar la respuesta en las preguntas: `p3, p3d, p3h, p3g1_1, p6a1` y `p6_7`, entonces se tiene que decidir si se trabaja con las preguntas individuales o como concepto (variable calculada). Si es como concepto, entonces, se analiza si se puede considerar como categórica o como ordinal.

Las preguntas del cuestionario y las variables calculadas son más de trescientas y 150 mil casos a nivel nacional, de tal suerte que el archivo tipo texto con toda la información superaba los 200 MB y se tiene documentado que R comienza a tener problemas con el manejo de la memoria RAM con archivos de este tamaño, por lo que el siguiente paso fue familiarizarse con las técnicas para el manejo de conjuntos grandes de datos (`filehash, RODBC, ff, Biglm`).

Una vez cargada la información en R se realizó un primer análisis exploratorio para desechar a las variables que evidentemente no se relacionaban con el ingreso. Esta etapa requirió de mucho trabajo ya que la gran mayoría de la información de la encuesta es categórica y en ocasiones con catálogos extensos (SCIAN y CMO), otro problema fueron las secuencias ya que las preguntas se aplican bajo determinadas secuencias. Por ejemplo:

la pregunta 3K. ¿El contrato es …. 1. Temporal? …2. De planta o base?... tiene sentido aplicarla a los trabajadores subordinados pero no para los patrones o a los que se dedican a actividades por su cuenta (ver secuencias de 3e y 3g). Cuando una pregunta "No aplica" el valor por default en la ENOE es un espacio en blanco y cuando se carga en R se interpreta como "Not available / Missing value (NA)" entonces fue necesario recodificar las variables para evitar problemas al momento de procesar (na.rm = TRUE) o sobre las conclusiones. El resultado de este proceso fue la reducción a poco más de cien campos.

En el capítulo cuatro se realizó un segundo análisis más detallado para agrupar variables, transformarlas y comparar su capacidad de predicción individual y en conjunto. Se identificaron 57 variables que podrían usarse como auxiliares para ajustar los métodos de imputación. Como preparativo para aplicar las técnicas de imputación en el capítulo 5 se emplearon los métodos de regresión *Lasso y stepwise* para reducir el número de variables logrando obtener el conjunto definitivo de datos, el cual constaba de 29 campos.

Los algoritmos de imputación que fueron tomados de la literatura [2] [3] [4] son los siguientes: a) Media general, b) Media por clases, c) K-Vecinos más cercanos, d) Hot-deck, e) Regresión lineal, f) Redes Neuronales (NN) y g) Regresión-knn. Para evaluar los distintos métodos se propuso trabajar con un *conjunto de datos para la simulación (CDS)* que incluyó exclusivamente los casos en que se había obtenido el dato del ingreso laboral, lo que representa aproximadamente 90% del total. Con base en las variables auxiliares se determinó el patrón que seguía la no respuesta para realizar 100 muestras replicando el patrón sobre el CDS, de tal manera que se generaron cien parejas de conjuntos de entrenamiento y prueba para evaluar los métodos de imputación.

Se utilizó el error cuadrático medio para seleccionar la mejor configuración de cada técnica (por ejemplo en k-vecinos elegimos la k que minimizaba el error), una vez ubicada se contrastaron sus resultados contra los obtenidos por el resto de los métodos. En nuestro ejercicio el que logró el menor error fue Regresión-knn.

En el capítulo 6 se seleccionaron un grupo de parámetros para observar el efecto que la imputación tendría sobre los resultados reales. Los parámetros estimados (ver tabla 6.2) fueron los siguientes: cuantiles y media de los ingresos, promedio y mediana del ingreso por hora trabajada y el coeficiente Gini.

El coeficiente Gini es quizás la medida más ampliamente utilizada de desigualdad de ingresos (Gini 1914). El índice de Gini es una medida relativa de desigualdad de los ingresos ya que depende solamente del ingreso. Se han desarrollado un número importante de representaciones para expresar el cálculo y su forma gráfica [5].

En las oficinas de estadística, el ingreso laboral es una de las variables más importantes y por ello es frecuentemente objeto de esfuerzos para predecir los valores perdidos, ya que se reúnen en ella tasas altas de no respuesta y mucho interés en sus resultados. En particular, en la ENOE se utiliza para generar una serie de indicadores relativos a la subocupación, a la estratificación de las percepciones y a la condición en que las personas desarrollan sus actividades económicas, lo que sirve para describir las características que tiene el mercado laboral y para realizar diagnósticos que dirijan las políticas públicas.

2. La Encuesta Nacional de Ocupación y Empleo

La Encuesta Nacional de Ocupación y Empleo (ENOE) tiene por objetivo recolectar y presentar información estadística sobre las características ocupacionales de la población a nivel nacional, así como su interacción con otras variables demográficas y económicas que permitan profundizar en el análisis de los aspectos laborales. Es el resultado de cinco años (2000-2004) de revisión metodológica y operativa de sus dos antecesoras: la Encuesta Nacional de Empleo (ENE) y, la Encuesta Nacional de Empleo Urbano (ENEU).

La ENOE mantiene como su eje rector los marcos establecidos por la Organización Internacional del Trabajo (OIT) e incorpora las especificaciones que la Organización para la Cooperación y el Desarrollo Económico (OCDE) emite sobre la Población Desocupada (PD) con el objetivo de dar un tratamiento adecuado a algunos subuniversos que se encontraban entre la ocupación y desocupación [6] .

Cabe mencionar que las encuestas sobre la fuerza laboral en el país tienen una larga tradición, la cual comienza en 1972 con el levantamiento de la Encuesta Nacional de los Hogares (ENH); de 1973 a 1974, la Encuesta Continua de Mano de Obra (ECMO); de 1974 a 1984 la Encuesta Continua sobre Ocupación (ECSO); de 1983 a 2004, la Encuesta Nacional de Empleo Urbano (ENEU) y de 1991 a 2004 la Encuesta Nacional de Empleo (ENE).

2.1. Diseño conceptual

La población objetivo de la encuesta está conformada por todas las personas que residen habitualmente en las viviendas seleccionadas. Los instrumentos de captación más importantes son el Cuestionario Sociodemográfico (CS) y el Cuestionario de Ocupación y Empleo COE. La información referente a los hogares y a las características sociodemográficas de los residentes de la vivienda se registra en el CS, mientras que el COE contiene las variables que identifican en primer lugar la condición de actividad y en segundo término preguntas específicas para cada uno de los universos. Es necesario mencionar que el COE se levanta a la población de 12 y más años de edad, sin embargo, por cuestiones metodológicas del marco de la OCDE se publica para la población de 14 y más años de edad.

2.1.1. Principales universos

La encuesta capta tres objetos de estudio: i) viviendas, ii) hogares y iii) personas. A su vez las personas se ordenan y jerarquizan en grandes universos como aparece en la siguiente figura:

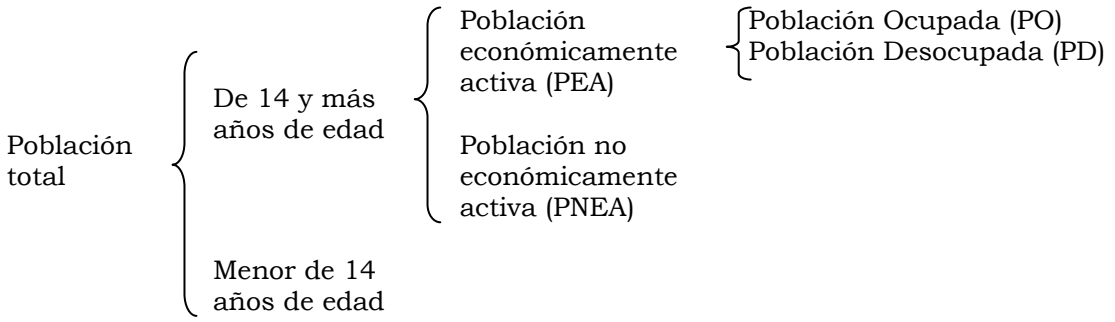


Figura 2.1: Jerarquía de los universos para la población total.

A su vez, la PO por su posición en el trabajo se desagrega de la siguiente manera:

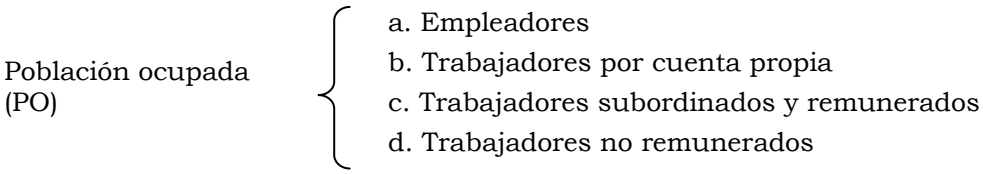


Figura 2.2: Categorías de la posición en el trabajo de la población ocupada.

El presente estudio se centra en el subuniverso de la PO que percibe un sueldo, salario o ganancia por el desarrollo de su actividades laborales, de tal forma que de las cuatro categorías que se muestran en la figura 2.2 debemos excluir la última *d. Trabajadores no remunerados* ya que son ocupados que no reciben pago por su trabajo, asimismo se dejarán de lado los trabajadores que desempeñan su labor en el extranjero.

2.1.2. Cobertura temática

Para la población en general se recolectan las siguientes variables sociodemográficas:

a. Condición de residencia.
b. Parentesco.
c. Sexo y edad.
d. Fecha y lugar de nacimiento.
e. Educativas (a la población de 5 y más años de edad).
f. Número de hijos (a mujeres de 12 y más años de edad).
g. Estado conyugal (a las personas de 12 y más años de edad).

La información que se capta de la población ocupada es la siguiente:

a. Ocupación.
b. Condición de multiocupación.
c. Datos de trabajo secundario (ocupación, sector de actividad, acceso a la atención médica).
d. Posición en la ocupación.
e. Tamaño del establecimiento.
f. Tipo de contrato.
g. Afiliación sindical.
h. Sector de propiedad.
i. Sector de actividad.
j. Disponibilidad de local.
k. Tipo de local.
l. Horas trabajadas.
m. Remuneraciones al trabajo, formas y periodo de pago.
n. Prestaciones laborales.
o. Acceso a servicio médico.
p. Datos sobre trabajo secundario.
q. Presión laboral.
r. Regularidad en el trabajo principal.

Para el grupo de personas que están en condición de desocupación:

a. Tipo de trabajo buscado.
b. Duración de la desocupación.
c. Experiencia laboral.
d. Razones de la desocupación.
e. Posición en la última ocupación.
f. Ocupación en el último trabajo.
g. Sector de actividad en el último trabajo.

Las variables que se disponen para la población no económicamente activa (PNEA) son:

a. Razones de no actividad.
b. Motivos de desaliento.
c. Experiencia laboral.
d. Razones de abandono del último trabajo.
e. Posición en la última ocupación.
f. Ocupación.
g. Sector de actividad.

2.2. Diseño muestral

El marco muestral empleado para la selección de viviendas es el Marco Nacional de Vivienda (MNV) que elabora el INEGI a partir de los censos y conteos de población y vivienda levantados alternativamente cada cinco años. Las viviendas se agrupan para formar Unidades Primarias de Muestreo (UPM), tomando como base la información disponible en el cuestionario censal, se realiza una estratificación socioeconómica y finalmente se selecciona aproximadamente el 10% de las UPM, las cuales constituyen el MNV. Es necesario comentar que dicho marco se utiliza para todas las encuestas de hogares que levanta el Instituto.

La unidad de selección de la encuesta es la vivienda y las unidades de análisis es tanto el hogar como las personas que residen en la vivienda. El tamaño de muestra es de 120 mil viviendas, las cuales se levantan durante 13 semanas (un trimestre). El tamaño de la muestra se calculó para generar estimaciones de la Tasa de Desempleo (TD) porque es uno de los indicadores claves y que exige mayor tamaño de muestra ya que su magnitud es pequeña (de 1 a 6% de la PEA).

El esquema de muestreo utilizado es complejo, pues emplea dos etapas de selección para reunir la colección de viviendas que finalmente serán visitadas; además, como mencionamos es estratificado y por conglomerados. En la primera etapa se elige alrededor de 10% de las UPM que conforman el territorio nacional, el método de selección de la primer etapa es Proporcional al Tamaño Sistemático (PPTS). La segunda etapa hace uso del método de selección Sistemático y usualmente toma grupos de 5 y 20 viviendas por UPM en localidades urbanas y rurales respectivamente.

Adicionalmente la muestra está dividida en cinco grupos de viviendas llamados Panel, cada uno de los cuales es homogéneo con respecto a los demás. Cada panel permanece en la muestra durante cinco trimestres, en cada trimestre es remplazado solamente un panel y cuatro permanecen. Esta característica posibilita realizar estudios longitudinales con una separación temporal de uno a cuatro trimestres. En la siguiente figura se muestra cómo se realiza el reemplazo de los paneles a través del paso de los trimestres.

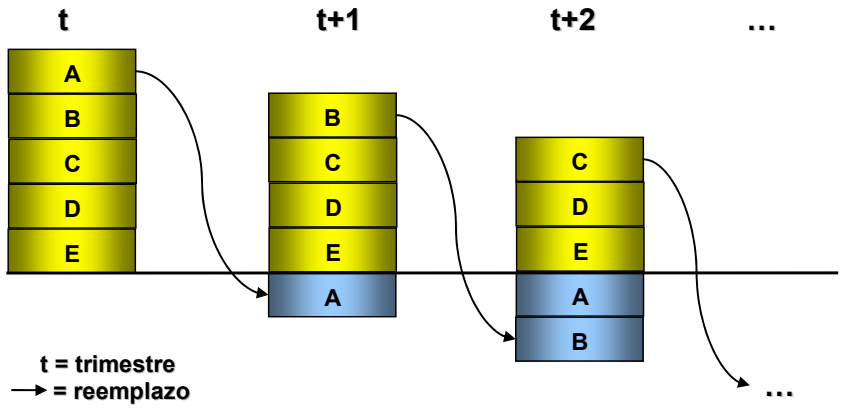


Figura 2.3. Diagrama de rotación de los paneles de muestra de la ENOE.

2.3. La variable ingreso por trabajo

En el glosario de la Encuesta encontramos la siguiente definición de la variable Ingreso por trabajo: "Percepción monetaria y/o en especie que recibió u obtuvo la población ocupada por el desempeño de su ocupación. Se considera sólo el ingreso neto, es decir, la cantidad de dinero que reciben los ocupados, libre de descuentos por pago de impuestos, cuotas sindicales y/o cuotas a una institución de seguridad social, en el caso de los trabajadores subordinados o de los gastos de operación de la unidad económica, en el caso de los trabajadores independientes".

La versión operativa del concepto arriba mencionado se plasma en las preguntas 6b y 6c del COE, también se debe comentar que el ingreso se capta teniendo como referencia el trabajo principal y puede ser reportado por cualquier miembro del hogar de 15 y más años de edad que conozca la información.

En la siguiente figura se muestra la pregunta que se emplea para captar el salario del trabajo.

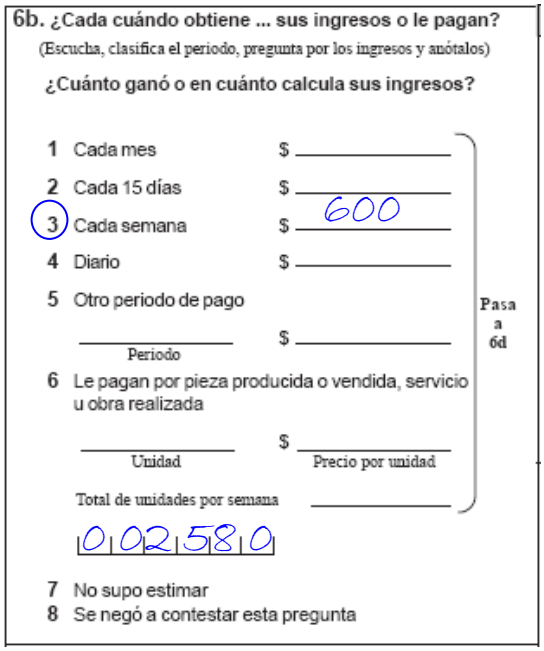


Figura 2.4. Pregunta 6b del COE.

Este reactivo capta dos datos: i) el periodo de pago y ii) los ingresos, el entrevistador tiene que realizar la conversión a ingresos mensuales, por ejemplo, si reportan que le pagan semanalmente $600, el ingreso mensual resultaría de multiplicar ($600/semana)x(4.3 semanas/mes) = $2,580/mes.

Sin embargo, si el informante desconoce el ingreso o se niega a contestar, se busca rescatar la información a través de rangos de salarios mínimos.

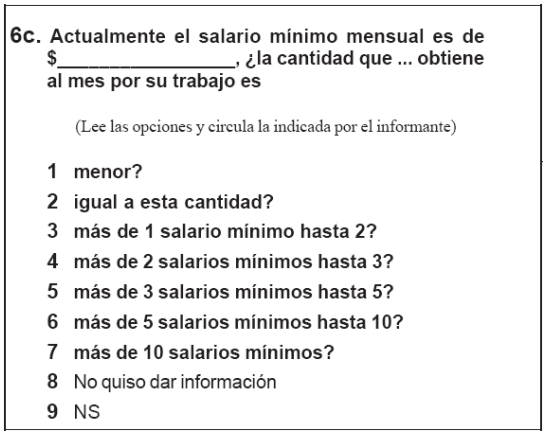


Figura 2.5. Pregunta 6c del COE.

A continuación se presenta el desagregado típico de la variable ingresos para la PO de 14 y más años de edad. Observamos que de los 40.6 millones de ocupados, 3.8 declararon no recibir ingresos y 2.2 no lo especificaron. Por lo que los deja con un universo efectivo de 34 534 157 personas con ingresos especificados, de los cuales 32.8 millones reportaron el ingreso directo a través de la pregunta 6b, lo que representa un 95% y el restante 5% (1.8 millones) se captó por intervalos de salarios mínimos en la pregunta 6c.

INDICADORES ESTRATÉGICOS DE OCUPACIÓN Y EMPLEO

Total

INDICADOR	2005 Trimestre I		
	Total	Pregunta 6b	Pregunta 6c
3.3 Nivel de ingresos	**40 575 874**	**32 762 109**	**1 772 048**
Hasta un salario mínimo	5 945 681	5 626 803	318 878
Más de 1 hasta 2 salarios mínimos	9 688 832	9 117 260	571 572
Más de 2 hasta 3 salarios mínimos	7 734 190	7 261 319	472 871
Más de 3 hasta 5 salarios mínimos	7 087 046	6 840 295	246 751
Más de 5 salarios mínimos	4 078 408	3 916 432	161 976
No recibe ingresos	3 832 662		
No especificado [1]	2 209 055		

[1] Incluye a los trabajadores en el extranjero

Cuadro 2.1. Estratos de ingreso en el trimestre I de 2005.

El rubro *No especificado* está integrado por los que contestaron las opciones 8 y 9 de la pregunta 6c y por los trabajadores en el extranjero.

Los valores expandidos nos indican que 2.2 millones de personas ocupadas carecen de información en la variable ingresos. Los casos no especificados representan 5.4% de la PO.

3. Manejo de bases de datos grandes en R

La carga de datos dentro de los paquetes estadísticos y la posterior exportación de las salidas a otras aplicaciones para formatear los reportes de resultados pueden ser tareas frustrantes y en ocasiones pueden tomar más tiempo que el mismo análisis estadístico [7].

Normalmente los paquetes estadísticos no tienen implementadas todas las instrucciones necesarias para el manejo de bases de datos de gran escala. Los responsables de coordinar el desarrollo de funciones para el acceso de datos sugieren el uso de herramientas que en este renglón ofrecen más prestaciones que R. Citan por ejemplo los comentarios de Therneau y Grambsch (2000) que prefieren manipular los datos con SAS y después realizar el análisis de supervivencia en S.

No obstante en los años recientes se han liberado una serie de paquetes de funciones que permiten el acceso y operación con grandes bases de datos (ver paquetes *Biglm, filehash*, RODBC, ROracle, RMySQL y DBI en la página oficial de R http://www.r-project.org) y otros que permiten utilizar la funcionalidad desarrollada en lenguajes como Java, Perl y Python directamente al código de R (ver SJava, RSPerl y RSPython del proyecto Omegahat y rJava de R-project).

La imputación que se plantea en este estudio debería estar incorporado al proceso de generación de información de la ENOE, en ese sentido se debería proponer también una solución informática para el acceso a la información que no modifique la manera en que los datos están estructurados. En este capítulo describiremos los métodos disponibles en R y algunas aplicaciones que están en operación en el Instituto.

3.1. Alternativas para el acceso de datos

3.1.1. Archivos de texto

La manera más sencilla de cargar datos a R es a través de archivos de texto, siempre y cuando su tamaño sea pequeño o mediano (menor de 200MB, durante los ejercicios que se realizaron en este proyecto se empezaba a tener problemas que superaban este tamaño lo que coincide con [7] R Data Import/Export pagina 13). La función más importante para leer archivos es `scan` aunque tienen muchos parámetros, en ciertas ocasiones vale la pena detenerse un poco y escribir esta comando con todos sus detalles.

Es muy común que la distribución de datos se haga en formatos propietarios como: Excel, SPSS, STATA o DBF, por mencionar algunos; en la mayoría de los casos es posible generar un archivo de texto con la información, sin embargo, en otros, esto no es posible y por eso se han desarrollado funciones para entrar directamente a los archivos: read.DIF, read.spss, read.ssd o read.dbf. A continuación se presenta un ejemplo para leer un archivo de texto con el comando scan.

Ejemplo:

```
cat("TITLE extra line", "2 3 5 7", "11 13 17",
     file="ex.data", sep="\n") #Crea un archive de texto con
tres líneas.
pp <- scan("ex.data", skip = 1, quiet= TRUE) #Lee el archivo,
con skip se especifica el número de filas que omite al inicio
del archivo.
scan("ex.data", skip = 1)
Read 7 items
[1]  2  3  5  7 11 13 17
```

Podemos leer un grupo de líneas o registros especificando la opción nlines.

```
scan("ex.data", skip = 1, nlines=1)#lee una línea después de
saltar 1
Read 4 items
[1] 2 3 5 7
```

La opción `what` se utiliza para especificar los tipos de dato que tiene cada columna, los tipos aceptados son `(logical, integer, numeric, complex, character)`. Si `what` es una lista, se asume que las líneas del archivo de datos son registros que contienen tantos campos como elementos en la lista.

```
scan("ex.data", what = list("","","")) # flush es F -> lee el
dato "7" lo que genera un warning porque el número de
elementos leídos no es múltiplo de las columnas.
Read 4 records
[[1]]
[1] "TITLE" "2"     "7"     "17"
[[2]]
[1] "extra" "3"     "11"    ""
[[3]]
[1] "line" "5"    "13"   ""

scan("ex.data", what = list("","",""), flush = TRUE)
Read 3 records
[[1]]
[1] "TITLE" "2"     "11"
[[2]]
[1] "extra" "3"     "13"
[[3]]
[1] "line" "5"    "17"
unlink("ex.data") # borra el archivo
```

Aunque el comando `scan` tiene mucha versatilidad para el manejo de archivos y es más eficiente en términos del tiempo requerido para cargar un archivo determinado, en un buen número de casos resulta más fácil utilizar funciones que están preconfiguradas para leer archivos con algún formato conocido.

Entre las funciones de este tipo podemos mencionar a `read.table`, `read.csv` o `read.delim`, los cuales leen archivos con formato de tabla y que, por cierto, internamente

llaman a la función `scan`. El primero recupera una tabla en donde se tiene que especificar el carácter que separa cada columna, el segundo está preparado para leer archivos que tienen el popular formato *CSV* (*comma-separated values*) y el último se enfoca en archivos en donde el tabulador es el carácter que separa cada columna. Para los tres comandos el "." separa los enteros de los decimales cuando el campo es numérico, las versiones `read.csv2` y `read.delim2` asume por default la "," justo como se utiliza en el continente Europeo.

Para ejemplificar su funcionamiento utilizaremos el conjunto de datos en formato csv que contiene información sobre una muestra de hombres enfermos del corazón en Sudáfrica[1]. Una vez que los datos fueron descargados y depositados en un medio de almacenamiento local se puede leer en R de la siguiente manera:

```
datos <- read.csv (file="SAheart.data",header=TRUE)
datos[1:5,]
  row.names sbp tobacco  ldl adiposity famhist typea obesity alcohol age chd
1         1 160   12.00 5.73     23.11 Present    49   25.30   97.20  52   1
2         2 144    0.01 4.41     28.61  Absent    55   28.87    2.06  63   1
3         3 118    0.08 3.48     32.28 Present    52   29.14    3.81  46   0
4         4 170    7.50 6.41     38.03 Present    51   31.99   24.26  58   1
5         5 134   13.60 3.50     27.78 Present    60   25.99   57.34  49   1
```

Observamos que la primer columna corresponde al nombre o número de cada caso, aunque por sí misma no es una característica, se puede asignar directamente a la propiedad `row.names` del objeto `data.frame` resultante.

```
datos <- read.csv (file="SAheart.data",
                   header=TRUE, row.names=1)
```

La función permite también obtener el conjunto de datos directamente de una dirección URL, lo cual facilita la ejecución de scripts desde cualquier equipo con conexión a Internet. Para permitir el acceso de las columnas del `data.frame` como si fueran variables independientes se utiliza el comando `attach`.

```
datos <- read.csv (file="http://www-stat-class.stanford.edu/
~tibs/ElemStatLearn/datasets/SAheart.data",header=TRUE,row.names=1)
attach (datos)
```

El objeto resultante de ejecutar el comando `read.csv` es un data.frame el cual almacena tablas cuyas columnas pueden ser de tipo distinto (numeric, logic, string y factor)

Importando otros tipos de archivos

Las opciones en que pueden organizarse los datos son muy variados y responder a situaciones como accesibilidad, conocimientos técnicos por parte de quien genera el

[1] El archivo se analiza en el libro *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*. Trevor Hastie, Robert Tibshirani y Jerome Friedman. Los datos se pueden obtener en: http://www-stat-class.stanford.edu/~tibs/ElemStatLearn/datasets/SAheart.data; y su descripción: http://www-stat-class.stanford.edu/~tibs/ElemStatLearn/datasets/SAheart.info

conjunto de datos, los tipos de datos o la dimensión (n,p). Cuando se busca que desde **R** se acceda a los datos se puede utilizar una amplia gama de opciones, por supuesto, siempre es mejor utilizar los formatos más universales para facilitar el acceso. En la siguiente tabla se listan algunos de los comandos más utilizados para introducir datos a **R**.

Comando	Descripción
Archivos de texto	
`scan(base)`	Función genérica para leer datos de un archivo o de la consola, devuelve un vector o una lista.
`read.table(utils)`	Lee un archivo con formato de tabla y lo vacía en un *data frame.*
`read.csv`	Lee un archivo con las variables separadas por comas.
`read.delim`	Recupera un archivo delimitado por tabulador.
`read.DIF(utils)`	Capta información a partir de una hoja de cálculo.
`read.fortran(utils)`	Lee archivos con formato Fortran.
`read.ftable(stats)`	Manipula tablas de contingencias provenientes de archivos planos.
`data.entry(utils)`	Interfaz gráfica para la edición de datos.
`read.fwf(utils)`	Recupera datos de un archivo con variables de ancho fijo.
`read.dcf(base)`	Lee y escribe datos en formato DCF.
`read.arff(foreign)`	Lee datos de archivos con la especificación ARFF (*Machine Learning Project*).
`read.xls (xlsReadWrite)`	Lee un archivo de MS Excel 97-2003.
`read.dbf(foreign)`	Lee archivos DBF.
Paquetería estadística y matemática	
`read.dta(foreign)`	Interpreta los archivos *binaries* de Stata.
`read.epiinfo(foreign)`	Recupera archivos de datos de Epi Info.
`read.mtp(foreign)`	Lee hojas de trabajo Minitab.
`read.octave(foreign)`	Lee archivos de texto de Octave.
`read.spss(foreign)`	Interpreta archivos de datos de SPSS.
`read.ssd(foreign)`	Obtiene *data frames* desde archivos de datos de SAS, vía read.xport.
`read.xport(foreign)`	Lee librerías de SAS en formato XPORT.
Interacción con manejadores de datos	
`sqlFetch(RODBC)`	Lee tablas desde bases de datos ODBC.
`sqlQuery(RODBC)`	Ejecuta una sentencia SQL.
`dbFetch (filehash)/dbMultiFetch`	Extrae una lista de características de la base de datos.

Tabla 3.1. Comandos para el manejo de archivos de datos.

En algunos casos el tamaño de la base de datos inclina a los usuarios a depositar la información en Sistemas Manejadores de Base de datos (DBMS por sus siglas en inglés), los cuales proporcionan una plataforma con mayores prestaciones de seguridad y repertorio de operaciones. El acceso concurrente a los datos es una característica generalmente necesaria en conjuntos de datos de esta naturaleza.

Se han desarrollado algunos paquetes que establecen una interfaz directa entre R y los DBMS comercialmente más usados, algunos de los que podemos mencionar son: ROracle, RMySQL y RSQLite. El sistema cliente-servidor ODBC es ampliamente utilizado en ambientes Windows y Unix para conectarse a una gran variedad de servicios, algunos de los que se han probado son: Microsoft SQL Server, Access, MySQL and PostgreSQL en Windows and MySQL, Oracle, PostgreSQL y SQLite en Linux. Además ODBC proporciona accesibilidad para archivos de Microsoft Excel, DBF (Microsoft FoxPro), texto y csv, por mencionar algunos.

3.1.2. Archivos dbf

Los archivos DBF tienen su origen en el programa dBASE, que se considera el primer sistema manejador de base de datos que fue ampliamente usado en microcomputadoras. La empresa Ashton-Tate desarrolló la primera versión para CP/M y poco tiempo después para las plataformas de Macintosh, Unix, VMS y MS-DOS. La legendaria versión III Plus significó todo un hito en los productos de software para computadoras personales. En 1991 fue vendido a Borland y a dataBased Intelligence en 1999.

El formato del archivo dbf es una mezcla de datos binarios y de texto, el encabezado contiene información binaria y los registros son almacenados en caracteres ASCII. La mayoría de la información sobre los censos, encuestas y registros administrativos que entrega el INEGI lo hace con el formato DBF versión 2.5, ya que la mayoría de los paquetes estadísticos, hojas de cálculo, SMBD y plataformas de desarrollo pueden recuperar la información contenida en estos archivos.

```
library(foreign)
datos.dbf <- read.dbf("D:/Imputacion/BD/ENOE-
105/vivT105.DBF")
dim (datos.dbf)
[1] 120221     21
```

El comando `read.dbf` está incluido en el paquete `foreign` y se puede especificar la opción `as.is` que controla la conversión de las variables que son de tipo string (character) al tipo factor de R. El archivo que se cargó en las líneas anteriores ocupa 8.5MB en disco duro, pero el conteo de memoria RAM libre después de efectuar la lectura del archivo bajó en 80MB. El objeto resultante de la lectura del archivo `dbf` es un `data.frame,` por lo que se pueden usar todas las propiedades aplicables.

El paquete `foreign` incluye la función `write.dbf` para descargar variables tipo data.frame a ficheros con formato dbf. Es básica en el sentido de que no se permite hacer operaciones de reemplazo de valores, alta o eliminación de registros sobre los archivos.

```
write.dbf(datos.dbf, "x.dbf", factor2char=T, max_nchar = 254)
```

3.1.3. Manejo de datos con el paquete `filehash`

`Filehash` fue desarrollado por [9] Roger Peng, movido por la dificultad que representa el manejo de grandes bases de datos en el paquete R. La razón por lo que el manejo de archivos grandes se hace pesado es porque R requiere que los objetos que maneja estén cargados completamente en memoria, para conjuntos pequeños no existe inconveniente, todas las operaciones se pueden realizar con eficiencia ya que la memoria disponible en la mayoría de los equipos soporta el requerimiento de espacio, sin embargo con archivos grandes el sistema se puede volver lento o simplemente no podrá ser cargado ya que no se cuenta con la memoria para cargar de una vez todas las variables que componen la base de datos.

Roger Peng utiliza la siguiente definición para referirse a bases de datos grandes: "es cualquier conjunto de datos que no puede ser cargado a R como un único objeto por limitaciones en el espacio de memoria".

El paquete implementa una base de datos que utiliza variables simples para permitir el acceso a los datos. Cuando se realiza una solicitud al valor asociado a una determinada variable, es trabajo de la base de datos empatar la variable con el valor y regresarlo.

Con el paquete `filehash` los datos son almacenados en un archivo en disco duro en lugar de la memoria. Cuando el usuario solicita un dato asociado a una variable, `filehash` encuentra el objeto en disco duro, lo carga a memoria y lo devuelve al usuario.

La implementación de base de datos que tiene el paquete `filehash` permite el manejo completo de operaciones de lectura y escritura a través de archivos que son consultados por medio de nombres de campo (key), adicionalmente la información se accede directamente del disco en lugar de la memoria.

Crear una base de datos filehash

Las bases de datos pueden ser creadas con el paquete `filehash` con la función dbCreate. Únicamente requiere el argumento nombre de la base de datos.

```
>  library(filehash)
Simple key-value database (1.0-1 2007-08-13)
>  dbCreate("miBD")  # Crea la base de datos
[1] TRUE
>  db <- dbInit("miBD") # Toma el nombre de la base de datos
y regresa un objeto heredado de la clase "filehash"
```

Una vez que la base ha sido creada se utiliza la función dbInit, que regresa un objeto S4 heredado de la clase `filehash`.

***Accesando a* una base de datos `filehash`**

El repertorio de funciones para interactuar con los archivos `filehash` está compuesto por: `dbFetch, dbInsert, dbExists, dbList` y `dbDelete,` todas ellas genéricas que aplican procedimientos específicos para la base de datos sobre la que se ejecutan. Para agregar datos a nuestro archivo usemos el siguiente comando.

```
dbInsert (db, "sexo",
            as.factor(sample(c("Mujer","Hombre"),
            100,prob=c(.52,.48),replace=TRUE)))
dbInsert (db, "edad",
           sample(12:100,100,
           prob = dnorm(12:100,mean=25,sd=20),
           replace=TRUE))
dbInsert (db, "Ciudades",
           as.factor(c("Cd. de México",
           "Guadalajara","Monterrey",
           "Guanajuato","Aguascalientes" )))
```

Se agregaron tres variables, la primera corresponde a 100 datos tipo factor que dan cuenta de la característica sexo, la segunda corresponde a 100 edades con dominio de 12 a 100 y la última es una lista de cinco ciudades, observe que no es necesario que todas las variables tengan el mismo número de elementos. Ahora podemos recuperar esos valores con la instrucción dbFetch.

```
edad <- dbFetch (db,"edad")
mean (edad)
[1] 33.73
```

Es normal que se requiera una lista de variables para ejecutar algún proceso.

```
ed.sex <-dbMultiFetch (db,c("sexo","edad") )
boxplot(edad ~ sexo, data=ed.sex)
```

Para listar las variables que contiene la base de datos se ejecuta.

```
dbList(db)
[1] "Ciudades" "edad"     "sexo"
```

Se pueden eliminar características con el siguiente comando:

```
dbDelete (db,”Ciudades”)
dbExists (db,”Ciudades”)
[1] FALSE
```

Lo más ortodoxo es utilizar las funciones `dbInsert` y `dbFetch`, sin embargo resulta más sencillo utilizar el estándar de R para el manejo de variables internas `$, [[` y `[`.

```
db$a <- rnorm(100, 1)
mean(db$a)
```

```
[1] 1.011141
> mean(db[["a"]])
[1] 1.011141
table(db[["sexo"]])

Hombre  Mujer
    43     57
table(db$sexo)

Hombre  Mujer
    43     57
dbList(db)
[1] "Ciudades" "edad"     "a"        "sexo"
```

Se incluye un método para soportar la función genérica `with`, la cual trabaja con listas o ambientes. Las tres instrucciones que a continuación se presentan obtienen los mismos resultados:

```
with (db, c(Edad = mean(edad), a = mean(a)))
     Edad         a
33.730000  1.086689
sapply(db[c("edad", "a")], mean)
     edad         a
33.730000  1.086689
unlist(lapply(db[c("edad", "a")], mean))
     edad         a
33.730000  1.086689
```

El paquete `filehash` define un método para el comando `lapply` que permite al usuario aplicar una función sobre todos los elementos de la base de datos directamente. El método esencialmente cicla a través de todas las variables almacenadas en la BD, carga cada objeto de manera independiente y aplica la función definida a cada objeto.

Carga de bases de datos `filehash`

Una manera alterna de trabajar con bases de datos `filehash` es cargar los nombres de las variables en el ambiente y accesarlas directamente, sin tener que utilizar las funciones `$`, `[[` y `[`. para ejecutar operaciones con variables internas. La función dbLoad del paquete `filehash` trabaja parecido al estándar `load` de **R,** excepto porque carga vínculos activos en el ambiente, en lugar de los datos reales. El vínculo activo se crea a través de la función `makeActiveBinding` del paquete **base;** dbLoad toma la BD `filehash` y crea símbolos en el ambiente que corresponden a las variables almacenadas en la BD. Conceptualmente hablando los vínculos activos son como punteros dirigidos hacia la base de datos. A continuación se presenta un pequeño ejemplo para ilustrar la manera en que trabajan los vínculos activos.

```
dbCreate("testDB")
[1] TRUE
```

```
db <- dbInit("testDB")
db$x <- rnorm(100)
db$y <- runif(100)
db$a <- letters
dbLoad(db)
ls()
[1] "a" "db" "x" "y"
```

Observe que aparecen objetos adicionales en nuestro *workspace*. De cualquier manera los valores de las variables (a, x, y) no están almacenadas en memoria, lo están en el disco duro. Cuando alguno de los objetos es accesado, los valores son automáticamente cargados desde la BD.

```
 mean(y)
[1] 0.5118129
 sort(a)
[1] "a" "b" "c" "d" "e" "f" "g" "h" "i" "j" "k" "l" "m" "n"
[15] "o" "p" "q" "r" "s" "t" "u" "v" "w" "x" "y" "z"
```

Si asignamos valores diferentes a alguno de esos objetos o variables, la base de datos se actualiza a través del mecanismo de los vínculos activos.

```
 y <- rnorm(100, 2)
 mean(y)
[1] 2.010489
```

Si desechamos la variable que apunta a la base de datos y después la recargamos, observaremos que los valores actualizados de "y" persisten.

```
 rm(list = ls())
 db <- dbInit("testDB")
 dbLoad(db)
 ls()
[1] "a" "db" "x" "y"
 mean(y)
[1] 2.010489
```

Tal vez una desventaja del enfoque de los vínculos activos que se implementan en este paquete es que cada vez que un objeto es accesado, los datos deben ser recargados en la memoria. Este comportamiento es diferente a la propuesta implementada en g.data con la asignación retardada en donde los datos deben ser cargados una vez y a partir de entonces permanecen en memoria.

Actualmente el paquete `filehash` puede representar base de datos en dos formatos. El formato por default se llama "DB1" y almacena las llaves y los valores en un solo archivo. Por experiencia, este formato trabaja bien, pero en la mayoría de los casos pueden resultar un poco lenta su inicialización cuando hay muchas (>>1,000) variables (keys). El objeto "`filehash`" en R almacena un mapa que asocia cada variable con una dirección, la cual nos indica el lugar en la base de datos en donde comienza la información de cada variable.

Antes de leer los datos se realiza una verificación para cerciorarse que el mapa está actualizado.

El segundo formato se llama "RDS" y almacena las variables como archivos separados en el disco en un directorio con el mismo nombre que la base de datos.

Este formato es el más directo y simple de los formatos disponibles. Cuando una petición se hace para una llave específica, `filehash` encuentra el archivo apropiado en el directorio y lee el archivo en **R**. El único detalle que se presenta en sistemas operativos que utilizan nombres de archivos sensibles a mayúsculas y minúsculas es que pueden colisionar. Una ventaja de este formato es que la mayor parte del trabajo de la organización está delegado al sistema de archivos.

3.1.4. Ejemplo con los datos de la ENOE

La Encuesta Nacional de Ocupación y Empleo (ENOE) levanta información trimestralmente en 120 mil viviendas y se utiliza para generar resultados sobre el nivel de desocupación y caracterizar el empleo en el país. La muestra está diseñada para que se generen estimaciones a nivel i. Nacional, ii. Entidad federativa, iii. 32 ciudades autorepresentadas, iv. Localidades de 100 000 y más habitantes, v. Localidades de 15 000 a 99 999 habitantes, vi. Localidades de 2 500 a 14 999 habitantes y vii Localidades con menos de 2 500 habitantes.

En la siguiente figura se presenta el esquema de la base de datos de la encuesta, para simplificar el diagrama se muestran únicamente los campos llave con los que se establecen las relaciones. Existen cuatro tablas principales, Vivienda, Hogar, Datos sociodemográficos y, por último, el cuestionario de ocupación y empleo (COE). Adicionalmente se genera una tabla con los registros de la población de 12 y más años, la cual incluye variables calculadas para facilitar la explotación. Debido a las limitaciones del formato **dbf,** el **coe** está dividido en dos tablas, ya que cuenta con 317 campos.

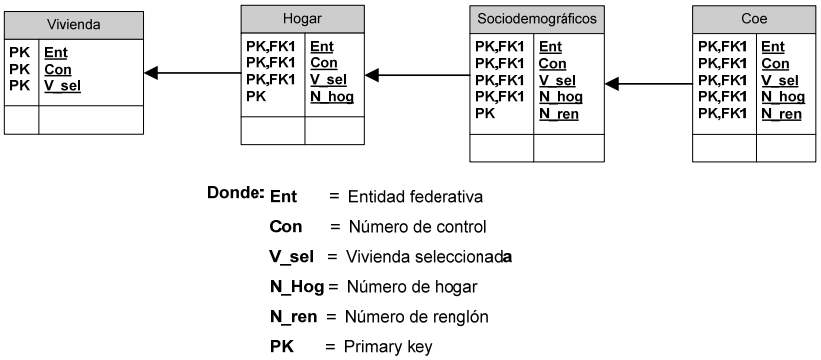


Figura 3.1. Modelo de la base de datos ENOE.

El objetivo de este ejemplo es predecir los ingresos no especificados de la población ocupada (PO), por lo que utilizaremos la tabla del COE del primer trimestre de 2005 (105) la cual contiene datos de las personas de 12 y más años de edad. El total de registros son 311 mil y ocupa 540 MB en disco duro, sin embargo, del total de registros, 57% corresponde a la población económicamente activa (PEA) y de ellos 96% son PO, por lo que de manera global tomaremos el 55% de los registros.

El primer asunto que se tuvo que resolver fue la separación de los datos, ya que estaban almacenados en distintas tablas. Se optó por generar un programa en MS FoxPro para realizar el empate de las tablas, filtrar los registros y generar un archivo de salida que pudiera ser leído fácilmente en R. A continuación se presenta un esquema que describe el proceso inicial de depuración:

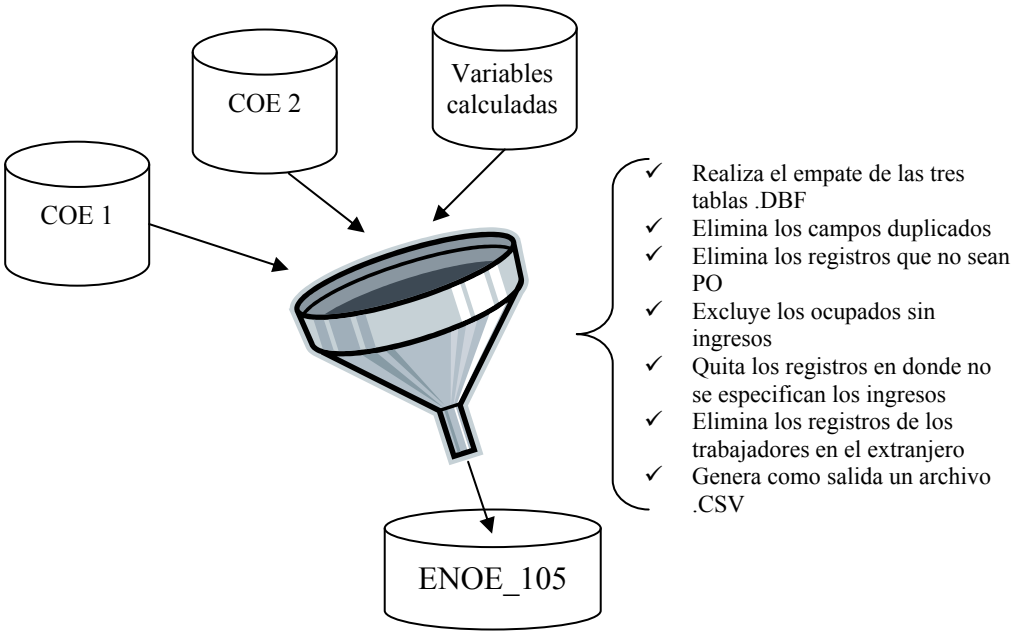


Figura 3.2. Primer proceso de empate y depuración de la información.

El siguiente paso consistió en tomar el archivo `.csv` resultante de la depuración inicial para crear la nueva base de datos con formato `filehash`. A continuación se muestran las líneas de código empleadas:

```
# Se cargan la tabla de datos en formato CSV a un dataframe
bd_105 <-
"d:\\Imputacion\\Estrategicos\\Proceso1\\enoe_105.csv"
datos <- read.csv (bd_105,header=T)

# Descarga la información contenida en el dataframe a un
archivo filehash.
db <- dumpDF(datos,
 dbName = "d:\\Imputacion\\filehash\\bd\\enoe_105")
```

En este momento ya se pueden ingresar datos de la base de la siguiente manera:

```
# Inicializa la base de datos
base_105 <- "d:\\Imputacion\\filehash\\bd\\enoe_105"
db <- dbInit (base_105)
```

El objeto `db` permite introducir los datos y ejecutar operaciones sobre ellos, por ejemplo en la siguiente instrucción con ayuda de la función `with` realizamos una gráfica condicionada de los LogIngresos contra el Estrato sociodemográfico y los años de instrucción formal.

```
with (db,coplot( jitter(EST) ~ jitter(A_ACUM) | log(P6B2),
                 xlab="Años de instrucción",
                 ylab="Estrato sociodemográfico",
                 overlap=.15))
```

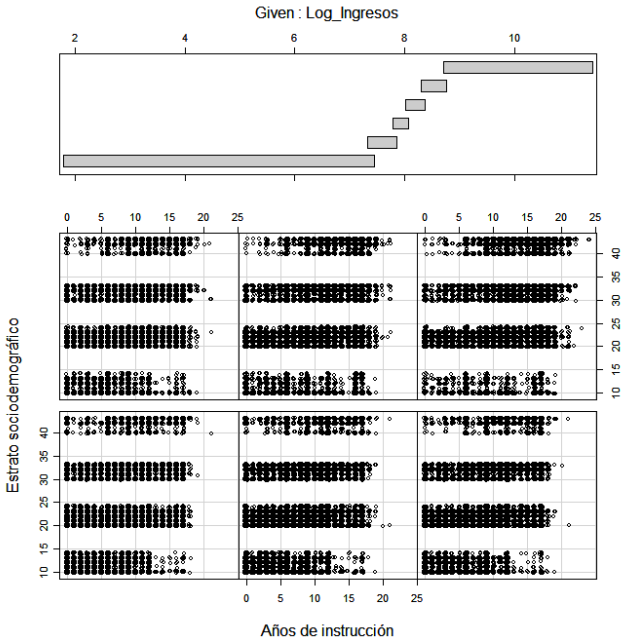


Figura 3.3. Gráfica condicional de ingresos vs. estrato socioeconómico y años de instrucción.

Para aplicar una regresión de los ingresos contra una serie de predictores ejecutamos la siguiente instrucción. Se requiere de aproximadamente 400MB de memoria para ejecutarla

```
fit <- with(db, lm(log(P6B2) ~ EDA+N_INS+EST+S_SOC+PO_S_SS+
            A_IS+P_SS+P3L2+A_ACUM+P3M4+P3L1+P3M1+P6B1+
            P3K1+P6D+P4+T_LOC+CP_ANC+P3+P5C_HMA+P5C_HMI+
            P5C_HJU+P5C_HLU+P5C_HVI+E_JOR6+T_MER21+CD_A+
            T_UE9+P5C_THRS+P11_H5+S_ACT+POS+S_ACT4+P4G+SEX+
            P4E+P4B+P5G14+P5+P3A+P3R+P5A+P3O+P4A+P10A4+
            P11_5+PO_SUB+P5C_TDIA+EMP_A_SSS+P3Q+NEG_A_CP+
            EMP_A_TD+SUB_REM+P4C+P4C+P5D+P6_7+NEG_A_PAS+
            P3S+EMP_A_PAS+P_LAB+T_CON+POA+P3I+P3N+PO_PR2+
            EMP_A_CSS+P3J+PO_PR+P8_4+NEG_A_CE+PER_A_SSS+
            P7+ABAN_A_TD+T_CON8+ABAN_A_SSS+P5C_HDO+
```

```
            PER_A_PAS+PER_A_TD+P5C_HSA+ABAN_A_PAS+
            ABAN_A_CSS+PER_A_CSS+P9+P3R_ANIO+FIN_N_PAS+N_HOG,
            weights=FAC))
summary (fit)

Residuals:
       Min         1Q     Median         3Q        Max
-150.47467   -3.95169    0.04048    4.27501  203.76165

Coefficients: (2 not defined because of singularities)
              Estimate Std. Error t value Pr(>|t|)
(Intercept)  8.004e+00  1.401e-01  57.118  < 2e-16 ***
EDA          2.866e-03  1.547e-04  18.527  < 2e-16 ***
N_INS        8.616e-02  2.792e-03  30.862  < 2e-16 ***
EST          1.818e-02  2.980e-04  61.025  < 2e-16 ***
S_SOC       -1.951e-01  5.224e-02  -3.735 0.000188 ***

.........
.........

P9           2.833e-03  3.216e-03   0.881 0.378360
---
Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 10.04 on 137579 degrees of freedom
Multiple R-Squared: 0.4875,     Adjusted R-squared: 0.4872
F-statistic:  1558 on 84 and 137579 DF,  p-value: < 2.2e-16
```

Se efectuó una regresión a partir de un objeto tipo data frame que tomaba toda la base de datos.

```
id <- function (x) return (as.vector(x))
# Vaciamos toda la bd a un objeto tipo data frame
datosm <- as.data.frame(lapply(db,id))
fit <- lm(log(P6B2) ~ EDA+N_INS+..............+N_HOG,
          weights=FAC,data=datosm)
```

La memoria que se requiere para ejecutar la regresión con data frame y con `filehash` es prácticamente la misma (400 MB aproximadamente), la diferencia consiste en que en el primer caso se necesita cargar de manera anticipada los datos a la memoria. Inicialmente se intentó subir los 300 campos pero precisaba 350 MB y si los agregamos a los 400 del proceso `lm` ocasionaba que la regresión se detuviera porque no alcanzaba la memoria de la computadora. Por lo que fue necesario crear un archivo que en lugar de tener todos los campos, sólo tuviera los 88 involucrados en la regresión.

Si deseamos accesar a las variables (keys) usando su nombre directamente y sin necesidad de utilizar las funciones de acceso: `[[`, `[` y `$`, se puede usar la función dbLoad para colocar los nombres de las variables en el ambiente.

```
# Carga la base de datos al ambiente
```

```
dbLoad(db)

# Lista los campos que contiene la BD
dbList(db)
```

En el siguiente gráfico vemos la distribución del ingreso con respecto a la edad del trabajador. Obsérvese que las variables se accesan directamente.

```
boxplot (log(P6B2) ~ EDA,
         main="Población ocupada por ingresos y edad",
         xlab="Edad", ylab="Logaritmo de los ingresos")
```

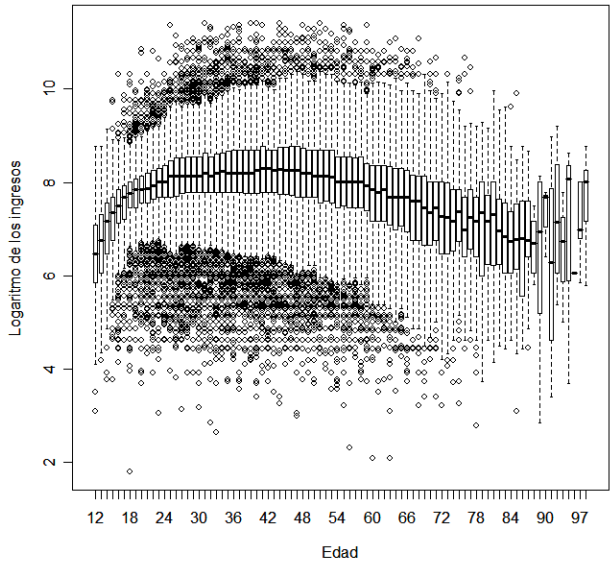


Figura 3.4. Diagrama de caja de los ingresos laborales de la población ocupada por año de edad.

3.1.5. Open database connectivity (ODBC)

En los apartados anteriores (3.1.1, 3.1.2 y 3.1.3) exploramos las formas más familiares que tiene R para leer datos y aunque son suficientes para la mayoría de las tareas no podemos dejar de notar algunas limitantes como las siguientes: son pocos tipos de datos soportados, tiene problemas para operar con archivos grandes (unos cuantos cientos de mega bytes) [7] (pag. 13) y es difícil implementar el acceso concurrente.

Por otro lado, los sistemas manejadores de base de datos (DBMS por sus siglas en inglés) están pensados como su nombre lo dice para administrar el diseño, carga y acceso de la información depositada bajo su resguardo. Entre sus bondades se encuentran la implementación de reglas de seguridad para evitar operaciones no autorizadas, brinda mecanismos para el acceso concurrente, cuenta con rutinas eficientes para consultar bases de datos grandes y cuenta con estrategias de almacenamiento más organizadas que las estructuras utilizadas por R (tablas bidimensionales). A continuación haremos un breve recuento de los DBMS y las alternativas para interactuar con ellos desde R.

Tradicionalmente han existido dos tipos de DBMS: por un lado, los grandes y costosos (Informix, Oracle, Sybase, DB/2 y SQL Server), por otro, los de uso académico o para el uso de volúmenes pequeños de información como MySQL, PostgreSQL y Access. Lo anterior marcado también por el énfasis en las características de seguridad de datos. Actualmente la separación entre estos dos grupos se ha ido reduciendo gracias a la nueva versión de PostgreSQL OpenSource y las versiones "libres" de Informix, Oracle y Sybase, que pueden correr en el sistema operativo Linux [7].

Open Database Connectivity (ODBC) es un estándar para el acceso a todas estas fuentes de datos y fue desarrollado por Microsoft Corporation, el objetivo de ODBC es hacer posible el acceso a cualquier dato desde cualquier aplicación, sin importar el Sistema de Manejador de Bases de Datos (DBMS por sus siglas en inglés) que almacene los datos. ODBC logra esto al insertar una capa intermedia llamada manejador de Bases de Datos, entre la aplicación y el DBMS, el propósito de esta capa es traducir las consultas de datos de la aplicación en comandos que el DBMS entienda. Para que esto funcione tanto la aplicación como el DBMS deben ser compatibles con ODBC, esto o la aplicación debe ser capaz de producir comandos ODBC y el DBMS debe ser capaz de responder a ellos. [9]

Paquetes interfaz para bases de datos en R

Hay varios paquetes disponibles en CRAN que permiten a R comunicarse con algunas DBMS. Proporcionan diferentes niveles de abstracción. Algunos proveen medios para copiar conjuntos completos de datos hacia o desde la base de datos. Todos los paquetes tienen funciones para seleccionar datos a través de consultas SQL y para recuperar el resultado ya sean tablas completas o en secciones. Todos, a excepción de RODBC, se restringen a un solo DBMS.

RODBC

El paquete RODBC provee una interface para las fuentes de datos que soportan el estándar ODBC. Permite que el código de R entre en diferentes sistemas de bases de datos, se puede ejecutar en Unix/Linux y Windows. La mayoría de los sistemas de bases de datos proveen soporte para ODBC. Hasta el momento se han hecho pruebas con Microsoft SQL Server, Access, MySQL y PostgreSQL en Windows y Oracle, PostgreSQL y SQLite en Linux.

En el sistema operativo Windows, el soporte para ODBC está instalado por default y las últimas versiones disponibles se encuentran en http://www.microsoft.com/data/odbc/ como parte del MDAC. En Unix/Linux se requiere un controlador ODBC como unixODBC o iODBC y un controlador en el sistema de base de datos. El proyecto FeeODBC es un repositorio con información relacionada al ODBC.

Windows provee controladores no solamente para DBMSs sino también para hojas de cálculo como Excel, archivos DBase (DBF) y hasta para archivos de texto. Es posible tener conexiones simultáneas. Una conexión se puede abrir llamando a odbcConnect o odbcDriverConnect, el cual regresa un objeto que puede ser usado para accesos subsecuentes a la base de datos.

Se puede cerrar una conexión llamando a odbcClose, y cuando se pide concluir la sesión de R. Los detalles de las tablas disponibles de una cierta conexión se pueden obtener usando sqlTables. La función sqlSave copia en un *data frame* de R a una tabla en la base de datos, y sqlFetch copia una tabla en la base de datos a un *data frame* de R.

Una consulta Sql puede enviarse a la base de datos llamando la función sqlQuery, lo que regresará un *data frame* R. sqlCopy envía una consulta a la base de datos y guarda el resultado en una tabla en la BD. Un control más fino de la consulta se puede lograr con las funciones odbcQuery y después sqlGetResults para extraer los resultados.

A continuación se presenta un ejemplo usando PostgreSQL, para el cual el controlador ODBC convierte los nombres de las columnas del *data frame* a minúsculas. Usamos la base de datos testdb, la cual creamos con anticipación, su DSN (*data source name*) está definido en '~/.odbc.ini' si usamos unixODBC, aunque exactamente el mismo código se usaría con MyODBC para acceder a bases de datos MySQL con Linux o Windows (donde MySql convierte también los nombres a minúsculas). En Windows, los DSN son configurados en una pantalla del Panel de Control (*Data sources* ODBC) en la sección Herramientas administrativas en 2000/XP).

```
> library(RODBC)
## "tolower" Indica que cambien los nombres a minúsculas
> channel <- odbcConnect("testdb", uid="ripley",
case="tolower")

## Carga a un data frame la información que se vaciará en la
base de datos
> data(USArrests)
> sqlSave(channel, USArrests, rownames = "state", addPK =
TRUE)
> rm(USArrests)

## Lista las tablas en la base de datos
> sqlTables(channel)
TABLE_QUALIFIER TABLE_OWNER TABLE_NAME TABLE_TYPE REMARKS
1 usarrests TABLE

## Muestra en pantalla la información solicitada
> sqlFetch(channel, "USArrests", rownames = "state")
murder assault urbanpop rape
Alabama 13.2 236 58 21.2
Alaska 10.0 263 48 44.5
...

## Una consulta SQL, originalmente en una línea
> sqlQuery(channel, "select state, murder from USArrests
                     where rape > 30 order by murder")
state murder
1 Colorado 7.9
2 Arizona 8.1
3 California 9.0
```

```
4 Alaska 10.0
5 New Mexico 11.4
6 Michigan 12.1
7 Nevada 12.2
8 Florida 15.4

## Elimina la tabla
> sqlDrop(channel, "USArrests")

## Cierra la conexión
> odbcClose(channel)
```

Enseguida mostramos un ejemplo sencillo para leer hojas de cálculo con ODBC en Windows:

```
> library(RODBC)
> channel <- odbcConnectExcel("d:/est_n10500.xls")
## list the spreadsheets
> sqlTables(channel)

        TABLE_CAT TABLE_SCHEM               TABLE_NAME   TABLE_TYPE REMARKS
1 d:\\est_n10500       <NA>                Hombres$ SYSTEM TABLE    <NA>
2 d:\\est_n10500       <NA>                Mujeres$ SYSTEM TABLE    <NA>
3 d:\\est_n10500       <NA>                  TOTAL$ SYSTEM TABLE    <NA>
4 d:\\est_n10500       <NA> Hombres$Área_de_impresió        TABLE    <NA>
5 d:\\est_n10500       <NA> Hombres$Títulos_a_imprim        TABLE    <NA>
6 d:\\est_n10500       <NA> Mujeres$Área_de_impresió        TABLE    <NA>
7 d:\\est_n10500       <NA> Mujeres$Títulos_a_imprim        TABLE    <NA>
8 d:\\est_n10500       <NA>   TOTAL$Área_de_impresió        TABLE    <NA>
9 d:\\est_n10500       <NA>   TOTAL$Títulos_a_imprim        TABLE    <NA>
## Se presentan dos opciones para obtener el contenido de la
hoja llamada TOTAL.
> shTOTAL <- sqlFetch(channel, "TOTAL")
> shTOTAL <- sqlQuery(channel, "select * from [TOTAL$]")
> odbcClose(channel)
```

Observe que la especificación de la tabla es diferente entre el nombre devuelto por sqlTables y el empleado en la función sqlFetch que es capaz de mapear la diferencia. La conexión ODBC con las hojas de cálculo de Excel es de sólo lectura y no se puede alterar el contenido.

3.1.6. Paquete `ff`: redireccionamiento de memoria principal

Una de las limitaciones de R es que sólo puede direccionar objetos que caben en el espacio disponible de la memoria virtual, actualmente de 2 a 4 GB en sistemas de 32 bits. En consecuencia, teóricamente la longitud máxima de un vector es $2^{31} - 1$. La limitante en sistemas de 64 bits es menos severa (GB) pero sigue siendo una limitante y no puede arreglárselas con conjuntos de datos muy grandes (R Development Core Team 2007b).

El paquete ff [23] está diseñado para superar esta limitación ya que extiende el sistema de R al hacer uso de un nuevo tipo de contenedor, el cual permite trabajar con archivos

almacenados en memoria persistente (Disco duro, CD, DVD, etc.) como si se encontraran en memoria principal. Las funciones internas del paquete construyen los archivos empleando el formato binario nativo de R.

Desde el punto de vista del usuario, los objetos ff aparecerán como vectores o arreglos ordinarios de R que serán accesados usando los operadores comunes, aunque de hecho no están completamente residentes en memoria. El intercambio de datos entre la memoria virtual y el dispositivo de almacenamiento es alcanzado vía direccionamiento de páginas de memoria de archivos binarios.

El paquete ff comprende dos capas: una de bajo nivel escrita en C++ y una de alto en R. La versión actual es la 1.0 (junio 30 de 2007) y fue preparada para el concurso de programación de useR! 2007. En este momento, el soporte está limitado a datos numéricos (datos de doble precisión).

Manejo del paquete ff

Las funciones `ff` y `ffm` son empleadas para abrir y crear los archivos. Ambas funciones requieren el argumento `file` que especifica el nombre del archivo. Cuando se especifica el argumento `length` (para `ff`) o `dim` (para `ffm`) se crea un nuevo archivo, en otro caso se abre el archivo existente, por ejemplo, un archivo con longitud de 10 se crea con:

```
 library("ff")
 foo1 <- ff("foo1", length = 10)
 foo1
$ff.attributes
class file pagesize readonly
"ff" "foo1" "65536" "FALSE"
$first.values
[1] 0 0 0 0 0 0 0 0 0 0
```

Las operaciones de lectura y escritura de objetos ff se pueden realizar con los operadores "`[]`" y "`[]<-`". Por default, los valores de un objeto ff son iniciados con cero.

Los elementos de foo1 pueden modificarse con el operador "`[]<-`". Por ejemplo los primeros 10 elementos del conjunto de datos *rivers* que contienen la longitud de los 141 ríos de Estados Unidos pueden ser almacenados en un objeto ff de la siguiente forma:

```
data("rivers")
foo1[1:10] <- rivers[1:10]
```

En esta etapa debe notar que foo1 es un objeto ff mientras que foo1[…] regresa un vector normal de R.

```
foo1
$ff.attributes
class file pagesize readonly
"ff" "foo1" "65536" "FALSE"
$first.values
```

```
[1] 735 320 325 392 524 450 1459 135 465 600
# Ahora como vector
foo1[1:10]
[1] 735 320 325 392 524 450 1459 135 465 600
```

El paquete provee métodos para devolver la dimensión y la longitud del arreglo.

```
length(foo1)
[1] 10
length(foo1[1:10])
[1] 10
```

De manera equivalente, se pueden seleccionar muestras a partir de objetos ff.

```
 set.seed(1337)
 sample(foo1, 5, replace = FALSE)
[1] 735 392 524 450 600
```

El archivo es manejado por R como un puntero externo. Para una referencia más clara, el colector de basura (*garbage collector*) gc puede usarse:

```
rm (foo1)
gc()
```

Al llamar gc() elimina la referencia al archivo, pero no lo borra del disco duro. Entonces los datos siguen presentes, el archivo puede ser abierto nuevamente, lo cual podemos hacer con la función ff sin especificar el argumento de longitud:

```
> foo1 <- ff("foo1")
> foo1
```

Con `ffm` se pueden crear arreglos multidimensionales, por ejemplo, para crear y almacenar un objeto ffm con el conjunto de datos “cars”:

```
foo2 <- ffm("foo2", dim = c(50, 2))
data("cars")
foo2[1:50, 1] <- cars[, 1]
foo2[1:50, 2] <- cars[, 2]
```

Nuevamente, foo2 regresa un objeto ffm mientras que al usar los operadores de índice (“`[]`”) devuelve una matriz o un vector, por ejemplo, no existe un método disponible para graficar objetos ffm. Sin embargo, se puede realizar un scatterplot utilizando los operadores índice:

```
plot(foo2[, 1], foo2[, 2], pch = 16, las = 1)
# O de manera equivalente:
plot(foo2[, 1:2], pch = 16, las = 1)
```

Ejemplo ff

Para propósitos ilustrativos tomemos el Censo de Población de Estados Unidos de 2000. Los datos utilizados contienen variables demográficas basadas en las preguntas realizadas a las personas residentes de las viviendas. Los datos están disponibles en http://www2.census.gov/census_2000/datasets/. En este ejercicio utilizaremos la información del estado de Texas que contiene 2 465 variables para 750 624 viviendas.

Como archivo binario (8byte, doble), los datos ("P" tablas para Texas) ocupan aproximadamente 13.7 GB. Los datos están almacenados en un archivo llamado texas_p.ffd y puede ser accesado vía ffm:

```
txdata <- ffm("/tmp/texas_p")
```

Las variables edad mediana para todos los residentes de la vivienda, para hombres y para mujeres, así como el tamaño promedio de la familia son considerados en el ejemplo, las columnas con la información considerada son 393 a 395 y 696, respectivamente. Dado que el ejemplo es para realizar un análisis exploratorio, podemos utilizar una muestra en lugar del conjunto completo de datos, digamos 10 000 registros de las variables mencionadas.

```
set.seed(1337)
ind <- runique(10000, total = 750624)
agb <- txdata[ind, 393]
agm <- txdata[ind, 394]
agf <- txdata[ind, 395]
afs <- txdata[ind, 696]
```

Un análisis sencillo de las edades medianas implica eliminar los datos omitidos que han sido codificados como ceros. Entonces un `boxplot` de las edades medianas se puede construir de la siguiente forma:

```
 in.c <- agb != 0
 agm0 <- agm[in.c]
 agf0 <- agf[in.c]
 agb0 <- agb[in.c]
 summary(agm0)
Min. 1st Qu. Median Mean 3rd Qu. Max.
0.00 27.00 34.00 35.72 43.00 98.00
 summary(agf0)
Min. 1st Qu. Median Mean 3rd Qu. Max.
0.00 29.00 37.00 38.24 46.00 96.00
 boxplot(agb0, agm0, agf0, names = c("total", "male",
"female"),
+ ylab = "median age", las = 1)
```

Cuando se dibujan `scatterplots` de bases de datos grandes es aconsejable usar el paquete `rgl` [10] para realizar gráficas. Las funciones `rgl` para graficar proveen

visualizaciones mucho más eficientes para conjuntos grandes de datos que las que normalmente se utilizan en R.

Interacción con el paquete biglm

El paquete biglm provee la capacidad de ajustar modelos lineales generalizados para conjuntos grandes de datos (para ser precisos conjuntos de datos o que son más grandes que la memoria). Si usamos la función envolvente (*wrapper*) ff.data.frame, es posible usar objetos ffm como entrada para la función bigglm.

El siguiente ejemplo muestra en primer lugar cómo convertir el conjunto de datos *trees* en un objeto ffm. Después de aplicar la función ffm.data.frame, puede ser usado el comando bigglm justo como lo indica la documentación.

```
demo(ff.bigglm)
> # Carga el paquete 'biglm' y los datos 'trees'
> require("biglm")
[1] TRUE
> data("trees")
> # create ffm object and convert 'trees' data
> m <- ffm("foom.ff", c(31, 3))
> m[1:31, 1:3] <- trees[1:31, 1:3]
> # create a ffm.data.frame wrapper around the ffm object
> ffmdf <- ffm.data.frame(m, c("Girth", "Height", "Volume"))
> # define formula and fit the model
> fg <- log(Volume) ~ log(Girth) + log(Height)
> trees.out <- bigglm(fg, data = trees, chunksize = 10,
sandwich = TRUE)
> ffmdf.out <- bigglm(fg, data = ffmdf, chunksize = 10,
sandwich = TRUE)
> # show summaries of fitted models
> summary(trees.out)

Large data regression model: bigglm(formula = formula, data =
datafun, ...)
Sample size = 31
Coef (95% CI) SE p
(Intercept) -6.632 -8.087 -5.176 0.728 0
log(Girth) 1.983 1.871 2.094 0.056 0
log(Height) 1.117 0.733 1.501 0.192 0
Sandwich (model-robust) standard errors
> summary(ffmdf.out)

Large data regression model: bigglm(formula = formula, data =
datafun, ...)
Sample size = 31
Coef (95% CI) SE p
(Intercept) -6.632 -8.087 -5.176 0.728 0
log(Girth) 1.983 1.871 2.094 0.056 0
log(Height) 1.117 0.733 1.501 0.192 0
```

```
Sandwich (model-robust) standard errors
> # cleanup
> rm(m, ffmdf); invisible(gc(verbose = FALSE))
```

3.2. Técnicas de programación

En la sección anterior revisamos varias estrategias para entrar a los datos, sin embargo, no solamente es importante tener métodos adecuados para accesar los datos, sino también lo es tener una programación adecuada que facilite interactuar con la información.

Es muy importante que se conozcan las ventajas y desventajas de cualquier herramienta, de esa manera podremos aprovechar sus capacidades y evitar caminos demasiado largos para la solución de los problemas a los que nos enfrentamos.

3.2.1. Aprovechar la vectorización

Una de las características más importantes que tiene el paquete es el manejo vectorial de los objetos y de las operaciones, por ejemplo, pensemos en una operación que se tiene que efectuar a un vector de un millón de elementos, dicha operación consiste en recorrer las observaciones un lugar.

```
# Corrimiento con ciclo
> system.time({a<-(1:1000000); for ( i in 2 :1000000) a[i]<-
a[i-1]})
   user  system elapsed
   5.30    0.00    5.52
# Corrimiento con una operación vectorial
> system.time({ a<-1:1000000; a[2:1000000]<-a[1:(1000000-
1)]})
   user  system elapsed
   0.24    0.02    0.25
```

Observamos que el tiempo empleado por la operación vectorial es 50 veces menor a la prueba con un ciclo `for`.

En general es una buena práctica evitar los ciclos anidados, sin embargo, cuando se trata de bases de datos grandes, evitar esta práctica es clave ya que los procesos pueden tardar mucho o, simplemente, no completarse la ejecución por falta de memoria. Existen algunas alternativas como el comando `sapply`.

3.2.2. Administración de la memoria

R administra la memoria en modo *lazy*, lo que entre otras cosas significa que el usuario necesita liberar explícitamente la memoria, aquí un ejemplo:

```
# Asigna una parte de la memoria disponible a la variable a.
a<- 1:1000000;
```

```
# Elimina la variable a, sin embargo en este momento no se
registrará el cambio en el contador de la memoria libre.
rm(a);
# garbage collector: libera memoria apartada que ya no se usa
gc()
```

Hacer un uso racional de la memoria, implica crear sólo los objetos (variables) necesarios.

```
a<- 1:1000000
b<- a # en este momento no se copia el objeto a
b[1] <- 0 # en este momento se copia el objeto a
```

Recuerda que puedes modificar la memoria virtual que R tiene actualizando el apartado `memory.size()`. Un manejo adecuado de la memoria depende en mucho de la medida en qué tan bien estemos utilizando los comandos y las alternativas disponibles para la lectura de los datos que ofrece R. A continuación se listan algunas recomendaciones útiles para un buen uso de la memoria:

- Carga a memoria principal solamente las variables con las que se va a trabajar,
- Elimina los datos de memoria que ya no utilices (`rm(); gc()`).
- Si el tamaño de la base de datos es grande lo más seguro es que obtendrás mejores resultados si transformas tu archivo a formato:
  - Filehash.
  - ff.
  - SMBD y te conectas con RODBC, RORACLE o RMySql.
- Da preferencia a acceso de datos a través de comandos como: select y xxFetch, ya que nos ayudan a extraer únicamente las características que entrarán en juego en nuestros procesos.
- Aprovecha las ventajas de cada lenguaje y plataforma, recuerda que si vas a realizar varios procesos de transformación de variables, empate de tablas, procesamientos secuenciales a los archivos, lo más seguro es que éstas tareas se puedan realizar con una mayor agilidad desde el mismo SMBD o desde algún lenguaje especializado en manejo de Bases de Datos.

3.2.3. Usa los comandos adecuados.

Por otro lado, R dispone de una gran variedad de comandos, los cuales brindan la posibilidad de utilizarlos dependiendo de la situación específica. Es común que R contenga funciones que siendo más eficientes en términos del tiempo requerido para ejecutarse son las que necesitan que se les proporcione más información (parámetros) con respecto a la acción que van a realizar, un ejemplo claro que ilustra esta condición es el comando `scan` que es el comando genérico para leer un archivo independientemente de su formato y `read.table` que se especializa en archivos tipo tabla y requiere menos argumentos, pero más tiempo para recuperar los datos.

Obsérvese que entre más información se pase explícitamente a R más rápido se ejecutarán los comandos:

```
A <- matrix(scan("matrix.dat", n = 200*2000), 200, 2000,
byrow = TRUE)
A <- as.matrix(read.table("matrix.dat")) #muchisimo mas lento
```

3.2.4. Características del equipo

Se debe tomar en cuenta las limitaciones y características de la computadora en donde se está ejecutando el análisis. Una regla que se puede formular con respecto a la memoria es la siguiente: puedes cargar sin mayor problema datos hasta 20% del RAM. La capacidad, velocidad y cache del microprocesador son elementos que determinan la rapidez con que las operaciones aritméticas son ejecutadas, por lo que es importante contar con el hardware con las mejores prestaciones cuando se requiriera realizar cálculos estadísticos laboriosos.

Durante el desarrollo de este proyecto fue necesario ejecutar alrededor de 80 simulaciones que tardaban entre 12 y 24 horas. cada una; se decidió realizar los procesos por las noches empleando varios equipos de manera simultánea.

Recuerde que lo mejor es utilizar a R como plataforma de comunicación y para ejecutar procesos estadísticos.

3.3. Aplicación

La Encuesta Nacional sobre Violencia en el Noviazgo 2007 (ENVN) se levantó por iniciativa del Instituto Mexicano de la Juventud (IMJ) y con la colaboración del Instituto Nacional de Estadística, Geografía e Informática (INEGI) durante octubre y noviembre de 2007.

Los objetivos principales fueron i) generar información sobre la frecuencia y magnitud de la violencia que se da entre parejas no convivientes y de las características de la dinámica de las relaciones de noviazgo y ii) obtener información que oriente el diseño de acciones de política pública para la prevención, atención y erradicación de la violencia que se manifiesta en las relaciones de noviazgo entre la juventud mexicana.

Los resultados de la encuesta se entregaron en tabulados, presentaciones y la base de datos con los registros recolectados en campo en las 18,063 viviendas seleccionadas. El propósito del proyecto en el que se involucró a R era determinar el nivel de detalle en que se podían presentar las variables que entrarían dentro de los tabulados.

En el INEGI se utiliza normalmente el coeficiente de variación (CV) para evaluar la confiabilidad de una cifra determinada y se calcula de la siguiente manera:

$$cv = \frac{\sqrt{\hat{\sigma}^2}}{\hat{\theta}}$$

Dado que los diseños muestrales de las encuestas que realiza el instituto son complejos, utilizamos la técnica de conglomerados últimos para estimar la varianza $\hat{\sigma}^2$.

Si para una cantidad estimada $\hat{\theta}$, su CV asociado es menor a 0.15 entonces se considera de buena calidad, si oscila entre 0.15 a 0.20 es aceptable, pero si es superior a 0.20 se califica como no confiable.

El proceso común para la generación de tabulados es i) Diseñar un plan de tabulados inicial[2] ii) Elaborar las especificaciones funcionales iii) Procesar los tabulados iv) Calcular las precisiones $(\sigma^2, cv, deff)$ para la evaluación v) Editar y liberar los cuadros. La actividad iv se realiza en el área encargada de diseñar y realizar las muestras, lo que implica doble trabajo, pues ellos tienen que calcular los mismos estimadores, además de las precisiones, editar las salidas y enviar los resultados, lo cual lleva por lo general cuatro semanas.

Se observaba que el cuello de botella se presentaba entre las etapas iii) y iv) la cuestión es si podrían unirse de alguna forma para evitar el doble trabajo al estimar los totales, razones y proporciones, así como la edición necesaria para vaciar la información a los formatos. Existían muchas alternativas para resolver el problema. La primer opción son los paquetes estadísticos entre los que se diferencian los comerciales y los libres (GNU), por un lado el Instituto adquiere regularmente licencias de SPSS, sin embargo, no incluyen el módulo de muestras complejas (*complex samples*) necesario para calcular la varianza de nuestras encuestas; por el otro, está el software libre como R que representa una solución prácticamente gratuita pero tiene la desventaja de requerir personal entrenado que pueda elaborar un script para calcular las cifras y varianzas de un determinado tabulado.

Entonces surgió la idea de tener un sistema organizado por capas: la primera proporcionaría una interfaz gráfica al usuario con las facilidades de una herramienta comercial y abstraería la complejidad de la programación; en la segunda se utilizaría una herramienta GNU como motor para el procesamiento.

[2] El plan de tabulados es una serie de cuadros propuestos, que tiene un orden determinado y responde a las principales cuestiones que dieron origen al proyecto. El plan sólo incluye el formato sin cifras del tabulado y en algunas ocasiones especificaciones funcionales para construir los tabulados.

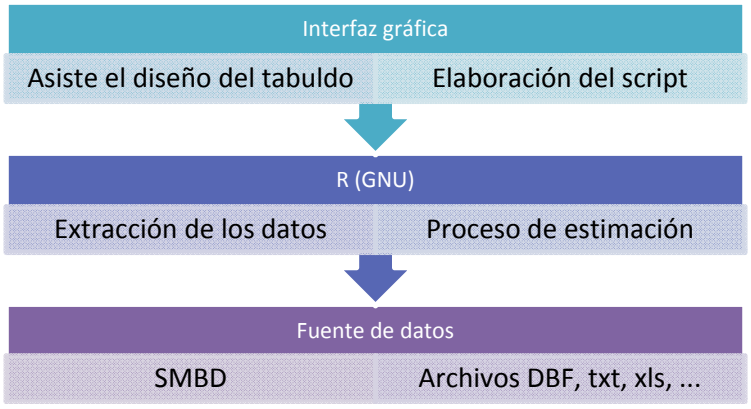


Figura 3.5. Diagrama de capas de SIDIGET.

Una de las tareas que significó un reto mayor fue establecer un mecanismo para que la interfaz gráfica lograra generar el script del tabulado que respondiera al diseño que el usuario realizara en determinado momento. Con base en experiencias anteriores, en particular de la ENDIREH 2006 (Encuesta Nacional sobre la Dinámica de las Relaciones) en la que se elaboraron manualmente los scripts del plan de 27 tabulados, ello nos permitió estructurar la secuencia de comandos y la identificación de patrones. Posteriormente se agrupo la secuencia de instrucciones en secciones que ahora conforman el script general de la aplicación.

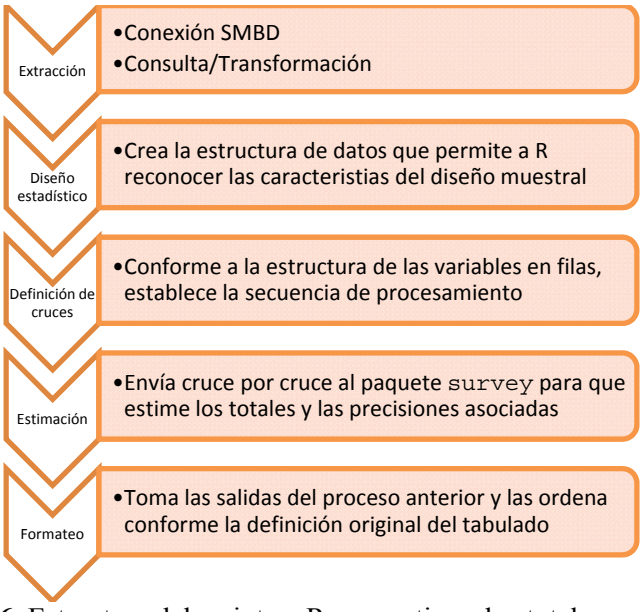


Figura 3.6. Estructura del script en R para estimar los totales y precisiones

Para que el sistema tuviera una mayor operatividad se desarrollaron procedimientos para almacenar, modificar y eliminar los tabulados. La información referente al diseño del tabulado se almacena en un SMBD, lo que permite el trabajo en equipo y evita problemas

de versiones de datos y programas que son comunes cuando los desarrolladores guardan su trabajo en carpetas locales y de red.

La operación de modificación posibilita la realización de ajustes a los tabulados, sobre todo, cuando después de una primera corrida resultaba que el coeficiente de variación identificaba una parte del tabulado como no confiable, lo que obligaba a realizar agrupaciones sobre determinadas variables y volver a ejecutar el tabulado, o en casos extremos, suprimirlo de la serie de tabulados.

A continuación se presenta el diagrama general de la aplicación a la que se nombró Sistema para el Diseño, Generación y Evaluación de Tabulados (SIDIGET).

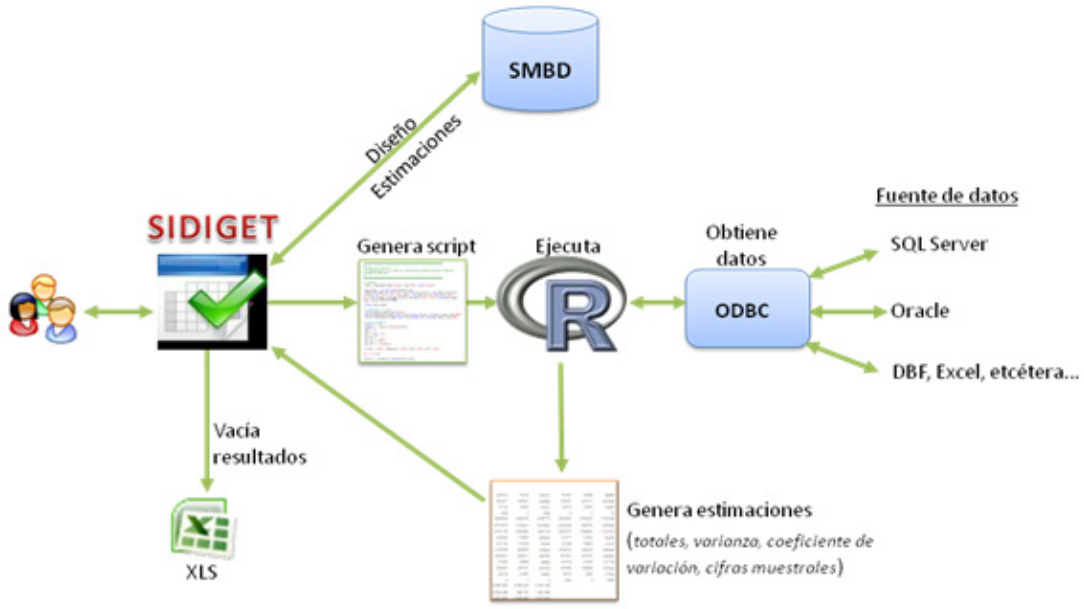


Figura 3.7. Diagrama del sistema SIDIGET

En el diagrama anterior observamos de izquierda a derecha a los usuarios que interactúan con la interfaz gráfica a la que denominamos SIDIGET. La interfaz recibe las especificaciones del tabulado: variables en filas y columnas y población objetivo (universo). La flecha dirigida hacia arriba indica que las estimaciones, precisiones y el diseño del cuadro son almacenados en un SMBD de manera que cuando el usuario requiera una modificación pueda recuperar el diseño anterior y realizace los ajustes necesarios sin tener que comenzar de cero. Cuando el usuario solicita que sean calculadas las cifras de su tabulado, SIDIGET genera y envía un script al paquete estadístico **R**.

El script está dividido en cinco bloques (ver figura 3.6) en primer lugar, a través del paquete RODBC se extrae la información de la fuente de datos seleccionada durante la etapa de diseño. Es importante destacar que el proceso de extracción de datos obtiene únicamente las características que entran en juego en el tabulado (variables en filas, columnas, universo, estrato, upm y factor de expansión), lo cual permite un uso racional de la memoria, evitando la carga de datos que sólo alentarían el proceso de estimación.

Una vez finalizada la extracción, se calculan, cruce por cruce, los totales, varianzas y coeficientes de variación utilizando la función `svyby` del paquete `survey` y se envían a un archivo txt, el cual es retomado y formateado por SIDIGET.

Figura 3.8. Salida del proceso de estimación de SIDIGET.

En la figura anterior, se muestra la pantalla de diseño, en el marco izquierdo se listan todas las variables de la tabla, en el cuadro titulado Universos se coloca una variable (binaria) que identifique los registros que entran en el tabulado o si apretamos el botón […] nos permitirá construir una expresión lógica que determine el universo. En el recuadro titulado Variables [COLUMAS] se listan las variables que, como su nombre lo indica, se desplegarán en las columnas del tabulado y el titulado Variables [FILAS] muestra los campos que se presentaran en los renglones. Finalmente en el cuadro inferior derecho se puede observar la realización de los totales del tabulado.

Para facilitar la rápida interpretación del CV se codificó en colores dependiendo del nivel que tiene en cada celda: Blanco: cv=[0,0.15), Amarillo: cv=[0.15,0.20] y Rojo: cv > 0.20. Aunque en la figura se presentan los totales, la aplicación tiene también la posibilidad de mostrar y exportar al formato Excel las cifras puntales de la varianza y del cv de cada celda.

En el anexo 3.1 se incluye un ejemplo del script que produce sidiget para que R genere las estimaciones del tabulado.

4. Análisis exploratorio

Este capítulo comienza con la descripción de los principales rasgos de la variable ingreso. En seguida se realizó una exploración para encontrar la relación que existe entre el ingreso y el resto de las variables del cuestionario. Durante el proceso fue necesario recodificar las variables para que los valores "No aplica" por pase de secuencia no se interpretarán erróneamente como datos perdidos (NA = *Not Available*). Se utilizaron técnicas para agrupar y selección de variables para reducir la dimensionalidad. Encontramos que el grupo de variables auxiliares contaban con distintos porcentajes de no respuesta parcial, la cual fue necesario resolver a través de procesos de imputación. Finalmente se analizaron algunas evidencias presentes en el conjunto de datos para identificar el tipo de no respuesta que se presentaba en la declaración de los ingresos.

4.1. El ingreso laboral

En México, como en otros países, el ingreso por trabajo tiene una distribución sesgada a la derecha, lo anterior implica retos para modelar su comportamiento. En el capítulo 2 observamos que el rango de la variable va de $1 a $999 998, de hecho el valor máximo corresponde a un código especial que identifica a las personas cuyo ingreso supera $999 997.

Durante el primer trimestre de 2005 (105) se captaron 151 220 COE que llegaron a la pregunta que capta los ingresos (P6B1). Un total de 137 664 reportaron el ingreso directamente y 13 556 (9%) no respondieron o se negaron a declarar el monto de ingresos que perciben por el desempeño de su trabajo.

P6B1. ¿Cada cuándo obtiene… sus ingresos o le pagan?			
	Total	**151 220**	
1	Cada mes	11 694	
2	Cada 15 días	35 489	
3	Cada semana	65 218	
4	Diario	19 668	
5	Otro periodo de pago	4 752	
6	Le pagan por pieza producida o vendida, servicio u obra realizada	843	
7	No supo estimar	11 412	13 556 registros
8	Se negó a contestar la pregunta	2 144	

Cuadro 4.1 Frecuencia de la pregunta 6B1 del COE.

Con el objeto de recuperar la información de las personas ubicadas en las opciones 7 y 8 de la pregunta P6B1, se aplica la pregunta P6C, en donde se le menciona al informante una serie de intervalos de salarios mínimos para que se autoclasifique en alguno de ellos. Con

esta pregunta (ver cuadro 4.2) se recuperó la información de 5 988 (44%), el resto 7 568 (56%) mantuvieron la negativa de proporcionar el dato.

P6C. Actualmente el salario mínimo mensual es de $____, ¿la cantidad que… obtiene al mes por su trabajo es			
	Total	**13 556**	
1	menor?	501	
2	igual a esta cantidad?	675	
3	más de 1 salario mínimo hasta 2?	1 806	
4	más de 2 salarios mínimos hasta 3?	1 427	
5	más de 3 salarios mínimos hasta 5?	935	
6	más de 5 salarios mínimos hasta 10?	439	
7	más de 10 salarios mínimos?	205	
8	No quiso dar información	1 919	7 568 registros
9	No especificado	5 649	

Cuadro 4.2 Frecuencia de la pregunta 6C del COE.

Las personas que responden a las opciones 1 a 7 de la pregunta P6C pueden ser objeto de una imputación especial ya que se cuenta con la ventaja adicional de conocer un intervalo acotado en donde ubicamos el ingreso del trabajador en cuestión (datos censurados). Sin embargo es necesario aclarar que el objetivo del presente estudio es recuperar el ingreso de los ocupados que en P6C están clasificados en las opciones 8 y 9.

4.1.1. La escala

El primer paso de la exploración fue la construcción de histogramas, diagramas de caja y dispersión, así como el ajuste de modelos de regresión lineal simple de la variable en la escala original. Sin embargo, los gráficos resultantes eran poco informativos, ya que prácticamente todas las observaciones se concentraban en el intervalo 1 a 20 000, cuando el rango completo de la variable es de 1 a 999 998. Por otro lado, observamos que la regresión lineal arrojaba heterocedasticidad en sus residuos (crecen conforme al ingreso) y que algunos casos de ingresos altos aparecían como puntos influyentes en la regresión. Encontramos que David, Little, Samuhel, Triest (1986) [13] en su artículo *Métodos alternos para la imputación del ingreso del CPS* proponen un modelo de regresión logarítmico, cuya variable dependiente $q = \ln(ingreso)$ se predice a partir de un conjunto de variables previamente definidas en Lillard and Willis (1978), Greenlees et al. (1982) y Betson and Van der Gaag (1983). Se utilizó el método de mínimos cuadrados para estimar los coeficientes de regresión y los registros seleccionados para ajustar el modelo fueron aquellos cuyo ingreso fue superior a $100 dólares.

Desde el punto de vista de la predicción es importante considerar aquellas observaciones que resultan excesivamente influyentes en la ecuación de regresión David, Little, Samuhel, Triest (1986). En la figura 4.1 se muestra una de las gráficas de residuales de la regresión de la variable ingreso explicada por los años de escolaridad. Durante la revisión de los ejercicios de regresión simple que se realizaron notamos que existen algunos registros que

inciden sistemáticamente. En la mayoría de los casos eran los que reportaban ingresos muy altos, para ilustrarlo observemos la siguiente figura en donde los registros que se resaltan en la gráfica son #60563, #128366 y #102345 que respectivamente tienen un ingreso de más de un millón, novecientos mil y setecientos cuarenta mil pesos al mes.

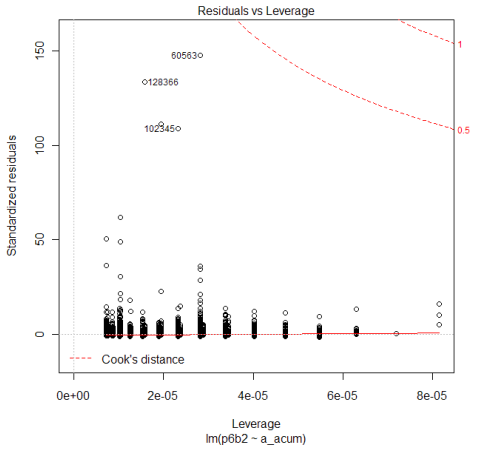


Figura 4.1 Gráfica de diagnostico de la regresión de los ingresos sobre los años de escolaridad.

La imputación que se busca hacer en este proyecto se enfoca principalmente en el tratamiento de observaciones *típicas* (aquellas que no se encuentran en las colas de la distribución). Lo que es excepcional por definición, es difícil de predecir. Proponemos que futuros estudios se enfoquen en la recuperación con técnicas especializadas de datos perdidos en los extremos de la distribución.

Con base en experiencias anteriores y en el primer análisis optamos por retomar la transformación logarítmica de la variable en cuestión porque ayuda a resolver el problema de heterocedasticidad y a evitar la posibilidad de predicciones con valores negativos. También utilizamos la idea de la seleccionar un subconjunto de registros, Little et al (1986) consideran que los registros que entran al ejercicio son aquellos que superan una cota inferior (> $100 dólares). No obstante en nuestra encuesta se presentan varios casos que son identificados como influyentes en la cola derecha.

Buscando referencias para identificar si un caso se puede considerar atípico o no encontramos que el sistema de validación de la ENOE consigna como datos sospechosos los que contienen 50 mil pesos o más y son revisados por un analista (humano). Sugerimos que este mismo tratamiento se dé a los registros que el procedimiento de imputación asigne ingresos muy altos y decida en consecuencia.

Valorando distintas alternativas para establecer el rango del ingreso que se consideraría en la tesis se optó por incluir en la base de datos a las personas que estuvieran entre el percentil 1 y 99 de los ingresos.

Para abrir los datos utilizamos el procedimiento visto en la sección 3.1.4. El paquete `survey` lo empleamos para la estimación de totales, percentiles, medias y razones tomando

en cuenta el diseño de la muestra. A continuación se incluye la serie de comandos para efectuar la exclusión de los casos que corresponden a los percentiles 1 y 99:

```
library (filehash)
library (survey)
   options(survey.lonely.psu="adjust")

archivo_fh_IN <-
"d:\\Imputacion\\Scripts_Def\\Datos_filehash\\enoe_105_ING"
archivo_fh_OUT <-
"d:\\Imputacion\\Scripts_Def\\Datos_filehash\\enoe_105_ING98"

db <- dbInit (archivo_fh_IN)  # Abre la base de datos con todos los
ocupados que especificaron los ingresos en 6B
dbLoad(db)
campos <- dbList(db)

# Crea la variable con el diseño de la muestra
disenio <- svydesign(id = ~UPM, strata = ~EST, weights = ~FAC,
           variables = as.formula(paste ('~',paste(campos,
           collapse="+"))) )
# Estima los cuantiles
ing_trunc <- svyquantile(P6B2, disenio, quantiles=c(0.01,0.99),
                         method = "linear", f = 1)
obs_fuera <- ((P6B2< ing_trunc[1]) | (P6B2> ing_trunc[2]))
nc <- length (campos)
dbCreate(archivo_fh_OUT)   # Crea un nuevo conjunto de datos fhash
db_out <- dbInit(archivo_fh_OUT)
# Vacía campo x campo las filas seleccionadas
for (i in 1:nc)
      dbInsert(db_out,campos[i],db[[campos[i]]][!obs_fuera])
ing_trunc
[1]   172 21500
```

En la última línea del código anterior podemos observar que el valor estimado para el percentil 1 y 99: P_1 = \$172 y P_{99} = \$21 500. El efecto de no tomar en cuenta los casos en que el ingreso reportado sea menor a \$172 y mayor a \$21 500 hace que la población expandida baje un 1.9% y los casos muestrales de 137 664 a 135 294 lo que representa un decremento de 1.7 por ciento.

4.1.2. Función de densidad y distribución de los ingresos

A continuación presentaremos los primeros elementos necesarios para el análisis exploratorio: la media, los cuantiles, histogramas y las funciones de densidad y distribución de la variable de interés.

```
library (Hmisc)
library (hexbin)
```

Para construir un histograma es necesario cargar en memoria los datos, crear una variable con las especificaciones del diseño muestral y finalmente ejecutar el comando `svyhist`.

```
#Abre la base
# TRUCADO el 1% de AMBAS COLAS
base_ok_105_ing <-
"d:\\Imputacion\\Scripts_Def\\Datos_filehash\\enoe_105_ING98"

db <- dbInit (base_ok_105_ing)
dbLoad(db)
(campos <- dbList(db))

# Histograma de la variable original
disenio98 <-  svydesign(id = ~UPM, strata = ~EST, weights = ~FAC,
   variables = as.formula(paste ('~',paste(campos,collapse="+"))) )

svyhist(~P6B2, disenio98, sub="(truncado 1%)",
              xlab="Pesos ($)", ylab="Densidad",
              freq=F) #col="lightgray",,ylim=c(0,0.6))
dens<-svysmooth(~P6B2, disenio98,bandwidth=700)
lines(dens,col="blue",lwd=2)
```

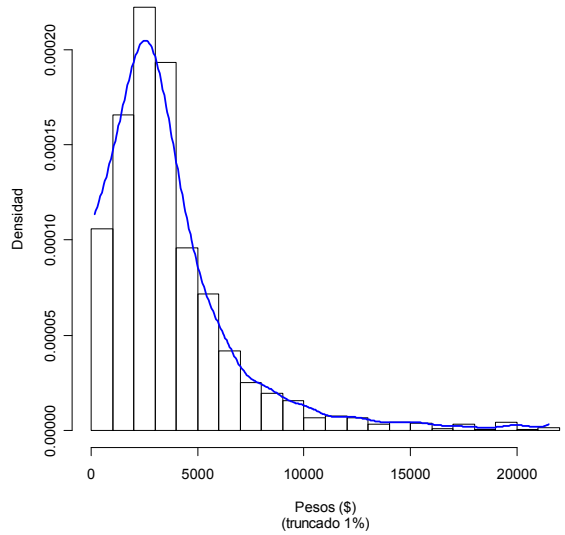


Figura 4.2 Histograma de los ingresos laborales.

Observamos que la distribución está sesgada a la derecha, el pico de la distribución se encuentra alrededor de los $3 000. Para ingresos superiores a diez mil pesos la frecuencia es marginal. En la siguiente tabla se presenta la estimación para los tres cuartiles del ingreso.

Cuartil	Ingreso mensual	Intervalo de confianza (α=.05)	
		L. inferior	L. superior
1er.	1 935	1 806	1 935
2do.	3 010	3 010	3 010
3er.	4 600	4 500	4 730

Cuadro 4.3 Cuartiles de los ingresos y su intervalo de confianza con nivel de significancia de 0.05.

A continuación se presenta la distribución acumulada de probabilidad y tres líneas verticales que corresponden a los cuartiles. Se pueden observar que existen algunos escalones visibles, los cuales son generados por respuestas muy frecuentes, por ejemplo, la mediana que se ubica en $3 010, si hacemos la reconversión a ingresos semanales entonces podemos ver que ($3 010/mes)/(4.3semana/mes) = $700/ semana es una repuesta común.

```
cdf.est <- svycdf(~P6B2, disenio98)
plot (cdf.est,main="Distribución de probabilidad",
      sub="Pesos ($)", xlim=c(0,21500))
Ing_q  <-  svyquantile(~P6B2, disenio98, quantiles=c(0.25,
    0.5, 0.75), alpha=0.05,ci=TRUE, method = "linear", f = 1)
abline (v=Ing_q$q, col=c(3,2,3), lty=2, lwd=2)
```

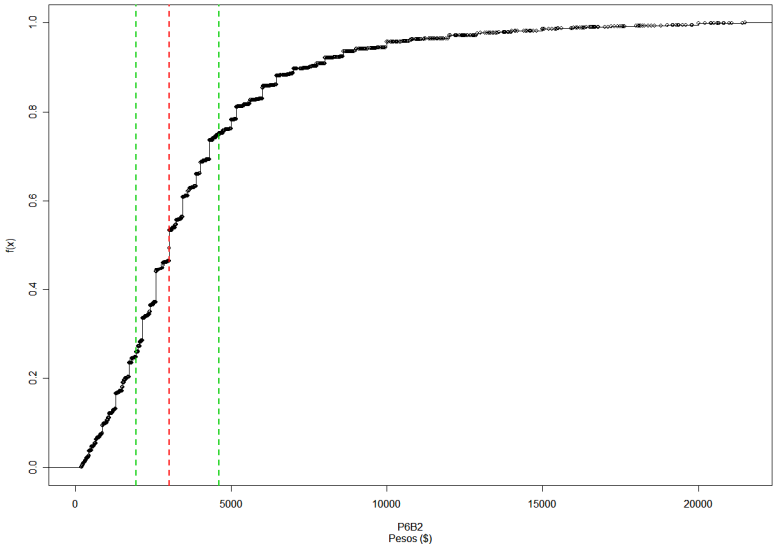


Figura 4.3 Distribución acumulada de probabilidad de los ingresos.

El ingreso mensual promedio de la población ocupada en el primer trimestre de 2005 se ubica en $3 797 con un error estándar asociado de $18.2, la moda se sitúa en los 2 580 pesos, lo que significa un salario semanal de $600.

```
media98   <- svymean (~P6B2, disenio98)
media98_H <- svymean (~P6B2, design=subset(disenio98,SEX==1))
media98_M <- svymean (~P6B2, design=subset(disenio98,SEX==2))
```

Para estimar la media cruzada por una segunda variable, en este caso sexo, se utiliza el comando `svyby` y con ello reemplazar las dos líneas anteriores:

```
media98HM <- svyby (~P6B2, ~SEX, disenio98, svymean)
```

Sexo	Ingreso promedio	Error Estándar	Coeficiente de variación
Total	3 797	18.2	0.005
Hombres	4 089	21.2	0.005
Mujeres	3 265	23.9	0.007

Cuadro 4.4. Estimación de la media y precisiones de los ingresos por sexo.

Se observa de inmediato que existe una diferencia marcada (25%), entre las percepciones mensuales netas de hombres y mujeres, haciendo patente un problema de género en el mercado laboral, sin embargo este hecho es derivado de la interacción de una serie de fenómenos sociales y económicos. La encuesta es capaz de proporcionar datos que ayudan a contextualizar y describir mejor esta situación, y sin el objetivo de trivializar un problema tan complejo me gustaría expresar comentarios sobre el particular, por ejemplo, si consultamos los resultados de la encuesta encontraremos que las horas a la semana que en promedio trabajan hombres y mujeres se ubica en 44.5 y 37.8 horas respectivamente, lo que significa una diferencia de 18 por ciento.

Por supuesto es válido preguntarnos ¿cuál es la causa por la que las mujeres trabajan menos tiempo? Intuitivamente sabemos que las mujeres dedican más tiempo atendiendo labores del hogar como un balance culturalmente aceptado, ya que los hombres salen a trabajar y las mujeres se quedan haciendo los trabajos que requiere el hogar. Pero si nos centramos en aquellas mujeres y hombres que trabajan, esperaríamos que ambos dediquen cantidades similares de tiempo a las actividades como lavar ropa, planchar, preparar y servir la comida –que la encuesta capta bajo el concepto *quehaceres del hogar*- ¡Encontraremos que en promedio las mujeres dedican 22.4 horas a la semana, mientras que los hombres sólo 4.9!.

Regresando a la exploración del ingreso, notamos que es útil contar con elementos que nos permitan observar el efecto que tiene el diseño de la muestra sobre las estimaciones, en ese sentido, la gráfica 4.4 muestra con línea punteada la estimación de la función de densidad de probabilidad de los ingresos para la muestra y la continua representa la función estimada. Notamos que la densidad para los ingresos expandidos se coloca por encima de la muestral en el intervalo $0 a 3 000 y apenas por debajo en el intervalo que va de los 5 a los 15 mil pesos.

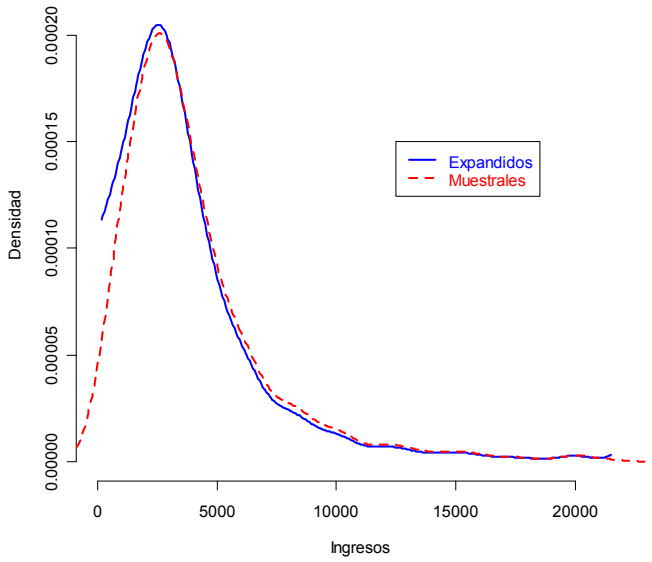


Figura 4.4 Función de densidad de probabilidad (KDE) del ingreso muestral y expandido.

En el cuadro 4.5 se presentan una serie de indicadores relativos a los ingresos y a las horas trabajadas, éstos resultados se presentan trimestralmente bajo el nombre de Indicadores Estratégicos y se pueden desagregar por entidad federativa y sexo para la población de 14 y más años de edad. Pueden existir pequeñas diferencias con los resultados que se generan en este documento porque trabajamos con la base de datos completa que incluye a los trabajadores de 12 y 13 años de edad.

4.1.3. Diagrama de dispersión

En la siguente figura observamos una matriz de diagramas de dispersión con una selección de variables relacionadas con el ingreso (logaritmo), en la diagonal inferior se observan los diagramas de dispersión utilizando la herramienta hexbin, la cual permite identificar la densidad de individuos en cada sección de la figura. El color azul obscuro son los niveles más bajos de concentración, posteriomente los azules más claros y los amarillos identifican las máximas concentraciones.

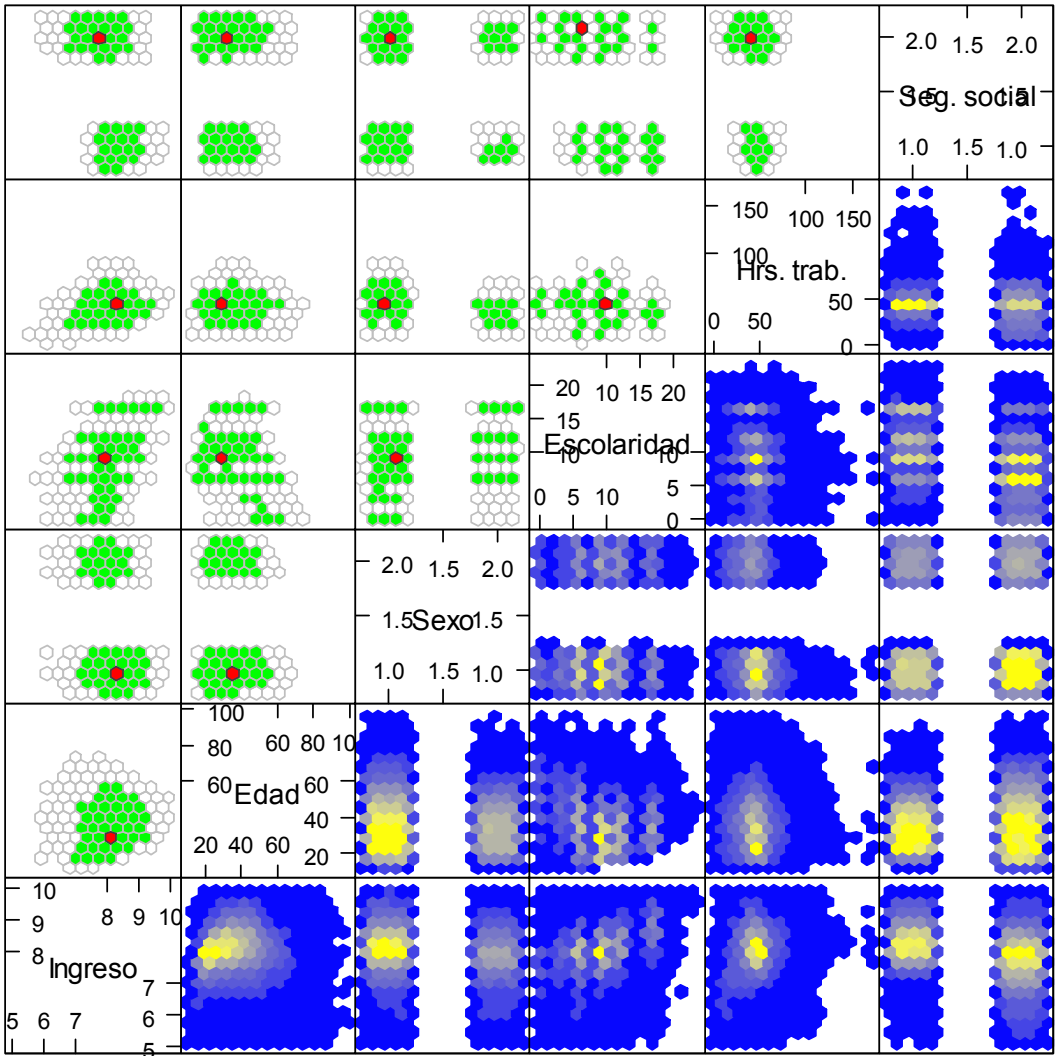


Figura 4.5 Matriz de diagramas de dispersión del logaritmo de los ingresos, edad, sexo, años acumulados en el sistema escolar formal, horas trabajadas y acceso a instituciones de seguridad social.

La diagonal superior representa un boxplot bidimensional, el punto en rojo identifica la mediana, los hexágonos de color verde (caja) agrupa los casos entre el primer y tercer cuartiles, finalmente la malla con fondo blanco (bigotes) delimita las observaciones con una distancia de hasta 1.5 veces el rango intercuartilico.

La variable Edad tiene un rango que va de 12 hasta 97 años, observamos que la mayor concentración se ubica en el grupo de 20 a 40 años de edad y que si bien se observa una tendencia, ésta no es lineal, más bien, parece una relación cuadrática. La variable sexo muestra que el grupo de hombres (Sexo==1) es más numeroso que el de mujeres (Sexo==2) y que, como lo vimos en el cuadro 4.4, la media de los ingresos es más alta para los primeros. En el caso de la variable Escolaridad que identifica los años cursados en el sistema escolarizado, notamos que existe una correlación positiva con los ingresos al igual

que sucede con la duración de la jornada semanal de trabajo en el rango de 20 a 60 horas. Finalmente para la variable Seguridad social se puede ver que el grupo de los ocupados que tiene acceso a instituciones de seguridad social (Seg.soc==1) es en promedio más alto y con menor varianza si los comparamos con los que carecen de esta prestación (Seg.soc==2).

4.1.4. Ingresos versus edad, nivel de instrucción y experiencia laboral

Podemos analizar con mayor detalle la relación existente entre el ingreso y la edad, por lo que para generar un diagrama de dispersión considerando el diseño muestral y para evitar la saturación de la gráfica volvemos a utilizar contenedores hexagonales (hexbin) y la escala de gris se refiere al número de trabajadores que agrupa.

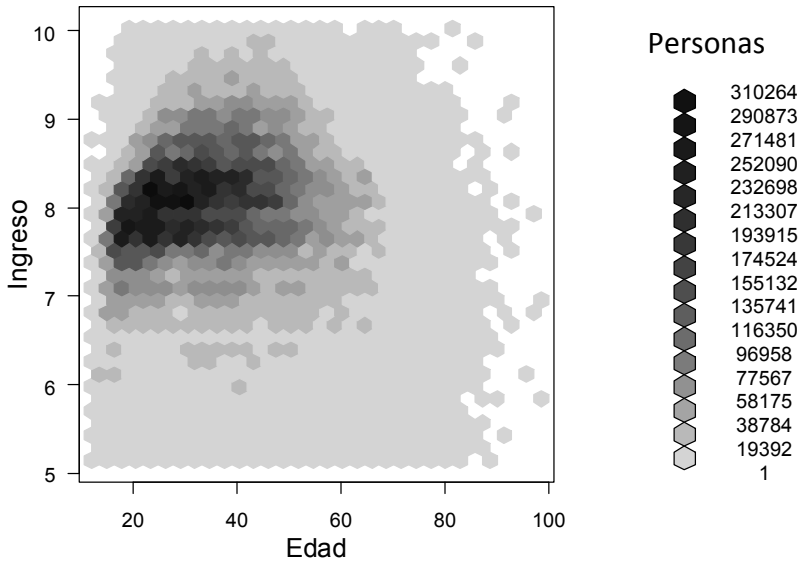


Figura 4.6 Diagrama de dispersión hexbin de los ingresos (logaritmo) y la edad del informante.

Se puede comparar la presentación de la gráfica 4.6 contra el anexo 4.2 que representa la salida obtenida con el comando `plot` y notar la mejora obtenida con el paquete `survey`. A continuación se incluye la instrucción para crear el gráfico.

```
svyplot (Log_ing~Edad, disenio, style="grayhex",
         main="Ingreso contra edad", xlab="Edad",
         ylab="Ingreso", legend=1)
```

En el diagrama de dispersión se observa una forma triangular en la aglomeración de los datos y que la edad para trabajar se extiende hasta los 60 años, la mayor concentración entre los 30 y 40, los ingresos alrededor de la media poblacional (media de los ingresos en escala logaritmica 7.9)

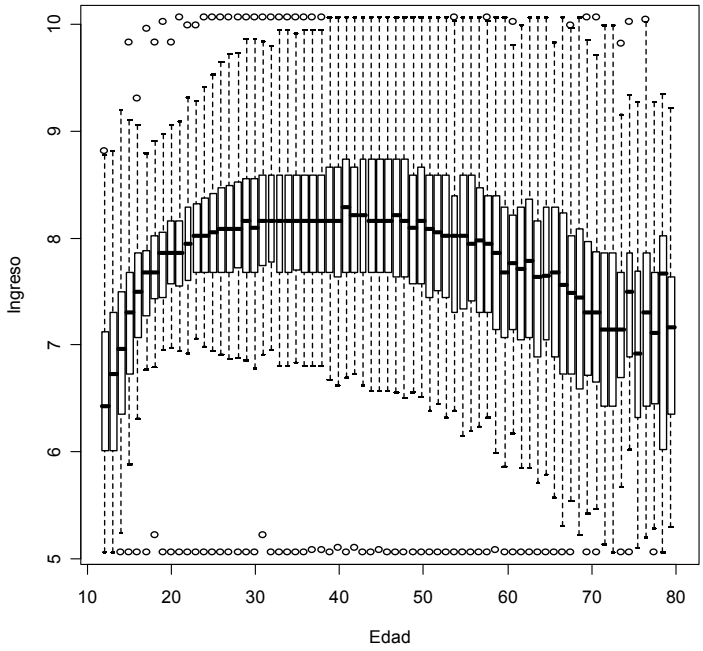


Figura 4.7 Diagrama de caja de los ingresos (logaritmo) y la edad del trabajador.

La gráfica 4.7 despliega de una manera alterna el cruce edad-ingreso. Se puede notar que la mediana sigue un patrón cuadrático aunque la variabilidad cubre prácticamente todo el rango de la variable. Se observa un ligero sesgo a la derecha en la distribución de los ingresos sobre todo en el rango de los 20 a los 60 años. Cabe comentar que el digrama de caja se realizó con estimaciones de los cuantiles que consideran el diseño muestral.

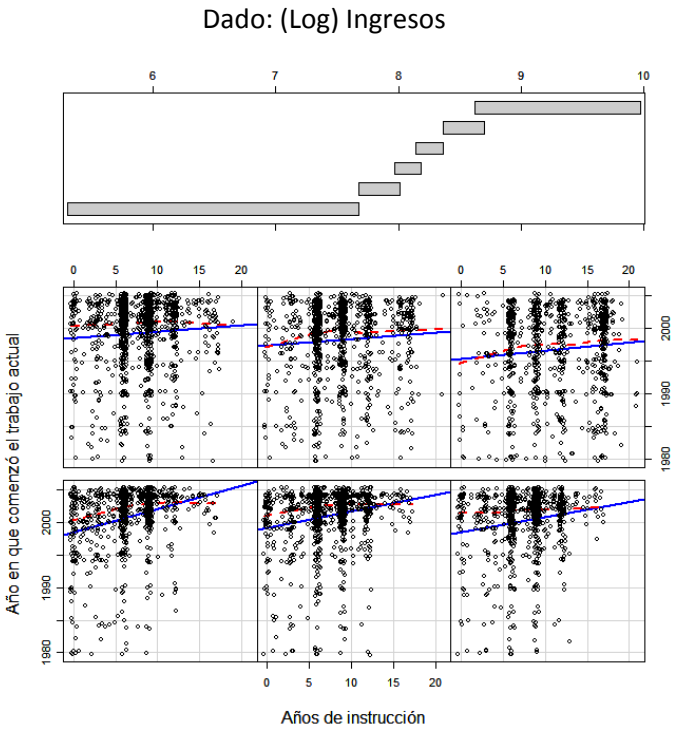


Figura 4.8 Diagrama condicionado de los años de instrucción escolar y el año que comenzó el trabajo actual dado los ingresos (logaritmo). El ejercicio de la gráfica es muestral.

El diagrama 4.8 inicia por dividir el rango de los ingresos en seis grupos que contiene el mismo número de observaciones con un traslape del 15%, como se puede observar en el recuadro superior. La parte inferior cuenta con un diagrama de dispersión para cada uno de los grupos construidos a partir de la estratificación de los ingresos, se lee de abajo hacia arriba y de izquierda a derecha. Se alcanza a notar estructuras verticales que identifican el término de los niveles: primaria, secundaria, preparatoria y licenciatura, a los 6, 9, 12 y 17 años de estudio aprobados respectivamente.

En el grupo de menores ingresos, se identifica con claridad a las personas que terminaron la primaria y secundaria y la línea de regresión indica que a mayor grado de estudios más reciente fue su inserción al empleo actual. Los grupos 2, 3 y 4 incluyen a más personas con estudios de preparatoria, aunque la pendiente de la recta de regresión disminuya. Finalmente en los grupos 5 y 6 cuentan con más personas con licenciatura terminada y con mayor antigüedad en el trabajo, por lo que podemos concluir que existe cierta correlación entre los ingresos y la escolaridad aunque en los grupos con mayores percepciones la experiencia tiene una importancia especial.

4.1.5. Resultados publicados

Es importante conocer el tipo de resultados que se generan a partir del ingreso y por eso mostramos a continuación un estracto de los Indicadores estratégicos. Podemos observar que el ingreso se presenta interactuando con otras variables (horas trabajadas la semana pasada y la posición en el trabajo). Es posible intuir que la no respuesta individual de cada viarable involucrada en la estimación puede generar efectos negativos para la calidad de las cifras reportadas.

INDICADOR	Estimación	E.E.	C.V. (%)	Intervalo de Confianza al 90% LIIC	LSIC
9. Promedios y medianas					
Edad de la población económicamente activa					
Promedio	36.931	0.064	0.173	36.826	37.036
Mediana	35.000	-	-	-	-
Años de escolaridad de la población económicamente activa					
Promedio	8.587	0.024	0.285	8.546	8.627
Mediana	9.000	-	-	-	-
Horas trabajadas a la semana por la población ocupada					
Promedio	42.702	0.094	0.219	42.548	42.856
Mediana	45.000	-	-	-	-
Ingreso (pesos) por hora trabajada de la población ocupada					
Promedio	24.676	0.262	1.061	24.246	25.107
Mediana	16.611	0.119	0.714	16.280	16.670
Empleadores					
Promedio	52.276	4.587	8.775	44.730	59.822
Mediana	29.070	0.675	2.321	27.780	30.000
Cuenta propia					
Promedio	21.385	0.324	1.516	20.852	21.919
Mediana	13.333	0.280	2.097	12.920	13.840
Cuenta propia en actividades no calificadas					
Promedio	19.248	0.283	1.472	18.782	19.714
Mediana	12.500	-	-	-	-
Trabajadores subordinados y remunerados asalariados					
Promedio	23.797	0.159	0.666	23.536	24.058
Mediana	16.667	-	-	-	-
Trabajadores subordinados y remunerados con percepciones no salariales					
Promedio	25.825	1.372	5.311	23.569	28.081
Mediana	15.504	0.304	1.960	15.000	16.000

Cuadro 4.5. Apartado de promedios y medianas de los indicadores estratégicos.

Observamos que los empleadores o patrones[3] son los que perciben el mayor ingreso por hora trabajada y en absolutos representa solamente el 5% de los ocupados. En segundo lugar encontramos a los trabajadores subordinados y remunerados con percepciones no salariales[4], todo apunta a que la magnitud del ingreso compensa la falta de prestaciones y de seguridad social que caracteriza a este universo (ver anexo 4.1). Los trabajadores subordinados y remunerados asalariados se encuentran en tercer lugar y agrupa a 6 de cada 10 ocupados, finalmente encontramos a las personas que se dedican a actividades por

[3] Empleadores o patrones: emplean al menos un trabajador remunerado asalariado.
[4] Se refiere a todas aquellas personas que en el desempeño de su actividad reconocen depender de un jefe o superior, pero sin recibir un salario como forma de pago, percibiendo otras modalidades tales como comisiones, honorarios, destajo, propinas, etcétera.

cuenta propia y cuyos ingresos se ubican apenas por encima de los $20/hora, esta categoría concentra el 25% de las plazas que existen en el mercado laboral mexicano.

4.2. Conjunto de variables auxiliares

La tarea más ardua del proyecto fue la conformación de un conjunto de datos que pudiera emplearse para ajustar modelos orientados a la predicción de los ingresos no especificados.

Comenzamos seleccionando los cuestionarios en que la respuesta de los ingresos fuera *directa.* Excluimos a los ocupados que no reciben salario, a los trabajadores en el extranjero y a los que hubieran declarado su ingreso a través de intervalos (P6C). Un total de 138 mil registros fueron escogidos cada uno con las 300 características disponibles en la ENOE.

El siguiente paso fue depurar las variables que se pueden agrupar de la siguiente forma: a) rubros de identificación y diseño muestral b) datos sociodemográficos c) información de la ocupación y empleo. Se refiere a las preguntas que se captan en el cuestionario de ocupación y empleo y d) variables calculadas. Son una serie de conceptos que se agregan a la base de datos para facilitar el proceso de generación de resultados.

Para ilustrar lo que son las variables calculadas, consideremos el campo A_acum que nos indica la cantidad de años que una persona ha cursado en el sistema educativo. Para calcularla es necesario contar con el nivel (primaria, secundaria, bachillerato, etcétera…) y el último grado aprobado, por ejemplo, si una persona concluyó tres semestres de licenciatura, el valor que tendría en el campo a_acum sería 13, ya que se acumularían 6 años de primaria + 3 de secundaria + 3 bachillerato + 1 de licenciatura.

También existe la variable calculada Nivel de instrucción (n_ins) de tipo categórica y que se calcula a partir de los mismas campos que a_acum. Sus clases son: 1) Primaria incompleta, 2) Primaria completa, 3) Secundaria completa, 4) Medio superior y superior y 5) No especificado.

Una de las cuestiones a resolver es cuál(es) de las cuatro variables utilizamos (nivel, grado, a_acum y n_ins). Parece evidente que a_acum es la más adecuada ya que resume las otras tres y que por su naturaleza discreta se puede aprovechar mejor en los métodos estadísticos. Desafortunadamente no fue tan sencillo evaluar todos los grupos de variables, al interior y apreciar la interacción con los otros grupos. De hecho, en el capítulo 5 podemos observar que la técnica *Stepwise* selecciona a a_acum y n_ins para entrar en el modelo, por otro lado regresión lasso es consecuente con la lógica y descarta n_ins.

La no respuesta parcial en prácticamente toda la base de datos fue otro problema que requirió solución, por ejemplo, el número de horas trabajadas la semana pasada, sus valores posibles son 0 a 168 horas y 999 que identifica los valores perdidos. El coeficiente de regresión Ingreso versus total de horas trabajadas presentaba un atenuamiento importante cuando estaban presentes los códigos 999 (ver figura 4.9).

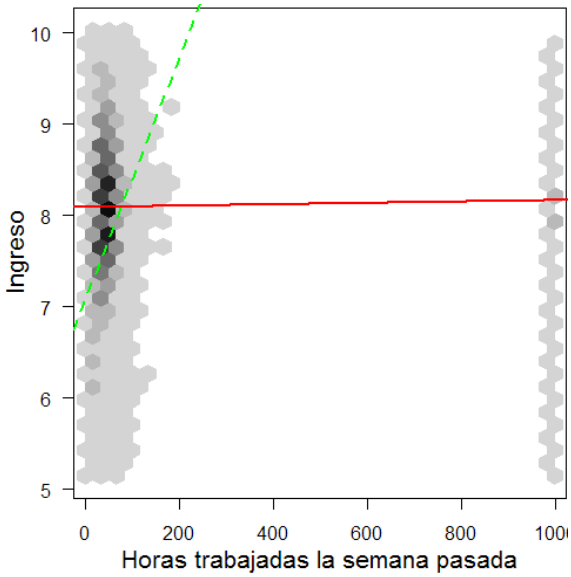


Figura 4.9 Efecto de la no respuesta parcial en las horas trabajadas cuando se relaciona con el ingreso. La línea sólida color rojo presenta la recta de regresión cuando las horas trabajadas incluyen los códigos que identifican la no respuesta parcial (999). La línea punteada en color verde muestra la recta de regresión de la variable imputada.

Se buscó que el proceso de imputación de las variables auxiliares fuera sencillo, pero que respetara en lo posible la relación que tenían con las otras preguntas. Fue necesario revisar los conceptos y tabulados de la encuesta para detectar cruces que tenían sentido, por ejemplo, para imputar el número de trabajadores de una empresa, se encontró que mantenía relación con el sector de actividad al que pertenecía (primario, secundario y terciario). Por lo tanto se optó por utilizar el método de imputación *media por clase*, en donde la clase fue el sector de actividad.

4.2.1. Depuración inicial de variables

El proceso de depuración de variables consistió, a grandes razgos, de los siguientes procesos:

a) Exclusión de las variables que aplicaba de manera exclusiva a poblaciones diferentes de los ocupados (ver figura 2.1). Por ejemplo: 2D. ¿A dónde acudió o qué hizo para buscar empleo?
b) Se eliminaron las variables calculadas que involucraron el ingreso en su construcción. Por ejemplo la Tasa de condiciones críticas de ocupación (TCCO) (ver anexo 4.3).
c) Reemplazar los códigos *No Aplica*[5] que utiliza la ENOE ya que R los interpretaba como `NA`=valores perdidos.
d) Imputar las variables auxiliares.

[5] Se refiere al código que se asigna para la ENOE cuando una pregunta no aplica por un pase de secuencia. Por ejemplo: El cuestionario indica que la pregunta sobre el nivel y grado de estudios sólo se aplique a las personas de 5 y más años, pero si es menor los campos quedan en blanco.

e) Se ajustó una regresión lineal de los ingresos contra cada variable auxiliar con el objeto de eliminar los campos que resultaran no significativos.
f) Se exploró la posibilidad de resumir grupos de variables, por ejemplo, las preguntas P3L y P3M del COE identifican 3 y 7 clases de prestación laboral, respectivamente; podríamos asignar dummies para cada clase, sin embargo ejecutamos componentes principales y regresión líneal simple y encontramos que era suficiente con la realización de una variable (P_lab) que indicara si tenía prestaciones o no.

En el cuadro 4.6 se presenta la tabla con la lista completa de variables resultantes del proceso de depuración. La columna *Nombre* se refiere al mnemómico o nombre de campo de la variable, Descripción es el nombre largo de la variable, la columna Tipo contiene una letra que identifica las siguientes clases (D) discretas, (O) ordinales, (C) categóricas y (R) para las reales.

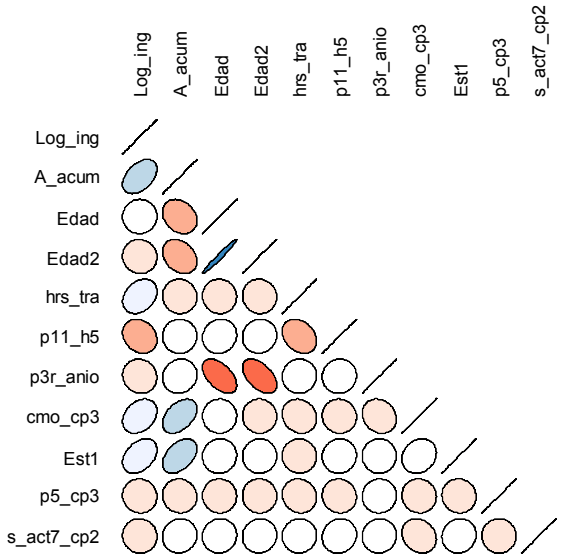


Figura 4.10 Matriz de correlación de una selección de variables numéricas de la ENOE.

En las encuestas de corte sociodemográfico la mayoria de las preguntas son categóricas y aunque existe una ampliga gama de técnicas para realizar análisis e inferencia categórica su uso implica ciertas limitaciones si la comparamos con el desarrollo que se puede hacer con las numéricas. Con ayuda del paquete `ellipse` construyó la gráfica 4.10 que muestra la correlación existente entre las variables númericas que fueron seleccionadas en el conjunto inicial. La inclinación y color de la elipse muestra el sentido e intensidad de la correlación. Si la relación es positiva entonces se pinta de azul y su inclinación es hacia la derecha, la fuerza se representa con colores más obscuros y por el adelgazamiento de la elipse. La corelación negativa se identifica con la escala de color rojo y la inclinación hacia la izquierda. Una breve descripción de las variables lo puede encontrar en la tabla 4.6.

Podemos notar que que las variables A_acum (escolaridad), hrs_tra (horas trabajadas), cmo_cp3 (ocupación) y est1 (estarato sociodemográfico) tienen corelación positiva con el ingreso. También podemos observar que la escolaridad tiene correlación más fuerte con el ingreso. La edad tiene una corelación menos significativa que el resto de las variables, de

hecho es positiva pero de magnitud muy pequeña, en la figura 4.7 podemos notar que la relación entre ambas variables es de segundo orden y por esa razón se agregó la variable edad2 (edad^2) que junto a p11_h5, p3r_anio, p5_cp3, s_act7_cp muestran signo negativo en la correlación. A grandes razgos significa que a partir de una cierta edad (40 años aproximadamente) el promedio del ingreso percibido por los trabajadores comienza a bajar, el mismo efecto tiene en el salario si los ciudadanos dedican más horas a los quehaceres del hogar (p11_h5) o tienen muchos años en su mismo puesto (p3r_anio) o no trabajan en el sector servicios (s_act7_cp).

Otras variables que tienen una relación que sobresale son: Edad y escolaridad de manera negativa, por lo que a mayor edad menor nivel de escolaridad. Horas trabajadas y quehaceres domesticos, entre más tiempo dedique al trabajo menos horas hace de quehaceres en el hogar. A mayor nivel socioeconómico (est1) más son los años que las personas pueden cursar en centros educativos (a_acum).

Nombre	Descripción	Tipo	Mín.	Máx.	No esp.	Catálogo	Tratamiento a no especificados	Observación
A_acum	Años de educación formal	D	0	23	99	0 a 23 años de instrucción	Se asignó la media por año cumplido a 254 casos	
N_ins	Nivel de instrucción	O	1	4	5	1=Prim Inc;2=Prim Com; 3=Sec. Com; 4=Medio S. y Sup; 5=No especificado (NS)	Se reasignaron 117 casos no especificados siguiendo lo recuperado en la variable A_acum	
S_soc	Acceso a instituciones de seguridad social	C	1	2		1 = Con acceso; 2 = Sin acceso		
Sexo	Sexo	C	1	2		1= Hombre; 2=Mujer		
Cp_anc	Trabajadores por cuenta propia en actividades no calificadas	C	0	1		1 = Sí; 0 = No		Sí=19.36%
Edad	Años cumplidos de edad	D	12	97	98	12 a 97 años cumplidos	Se asignó la media a 43 casos	
Edad2	Años cumplidos de edad al cuadrado		12^2	97^2				
hrs_tra	Total de horas trabajadas por semana	D	0	168		0 a 168 horas trabajadas	Se asignó la media por año de edad cumplido a 28 casos. Se rescató la información de 5755 registros con la información de horas habituales p5e	Generada a partir de 5c y 5e. Se consignó el valor 0 a aquéllos que no trabajaron la semana pasada y que no tienen un horario regular de trabajo
E_jor	Estratos de la duración de la jornada	O	1	5	6	1= No trabajó la sem. pasada; 2= de 1 a 14hrs ; 3= 15 a 34hrs; 4 =35 a 48 hrs; 5= 49 a 168; 6= No especificado	Al igual que el campo N_ins se recuperaron 6063 registros con la información procesada en hrs_tra	
Est1	Estrato socioeconómico (primer dígito)	O	1	4		1= Bajo; 2= Medio bajo; 3= Medio alto; 4= Alto		
Est2	Estrato socioeconómico	O	10	43				
p6b1	Periodo de pago	C	1	6		1=Mensual; 2=Quincenal; 3=Semanal; 4=Diario; 5=Otro periodo; 6=Destajo; 7=No supo estimar; 8=Se negó a contestar		
aban_a_pas	Abandonaron el año pasado un trabajo	C	0	1		1 = Sí; 0 = No		Sí=3.3%
emp_a_pas	Empezó el año pasado un trabajo	C	0	1		1 = Sí; 0 = No		Sí=16.31%
neg_a_cp	Personas que iniciaron un negocio CP el año pasado	C	0	1		1 = Sí; 0 = No		Sí=2.4%
per_a_pas	Personas que perdieron un empleo remunerado el año pasado	C	0	1		1 = Sí; 0 = No		Sí=2.3%
P_lab	Prestaciones laborales	C	0	2	3	0=No es trabajador subordinado; 1=Tiene prestaciones laborales; 2=No tiene prestaciones laborales; 3=No especificado	P_lab <- P_LAB, P_lab[P_lab!=1] <- 0	
T_con	Tipo de Contrato	C	1	4	5	1=Temporal; 2=Base; 3=Contrato NS; 4=Sin contrato;5=NS		
p4	Nombre de la empresa	C	1	3	9	1=Especifica el nombre; 2=No tiene nombre; 3=Es una unidad doméstica; 4 =Trabajador en el extranjero; 9=NS	Se recuperó con base en la pregunta 3a. Sí P3a=1 => (75.7,17.2,6.8)%; sí P3a=2 => (26.6,73.3,0.1)%	El código 4 se refiere a trabajadores en el extranjero a los que no se les preguntan los ingresos

Nombre	Descripción	Tipo	Mín.	Máx.	No esp.	Catálogo	Tratamiento a no especificados	Observación
T_loc	Tamaño de localidad (en habitantes)	O	1	4		1=100mil y +;2=De 15mil a 99,999; 3=De 2,500 a 14,999; 4=Menor de 2,500 habitantes		
s_act7_cp2	SCIAN: 2da. componente principal (7 grupos)	R	-0.7	0.7		(-0.04)s_act7_1+ (-0.05)s_act7_2+ (-.70)s_act7_3+ (0.71)s_act7_4+ (0.07)s_act7_5 + (-0.003)s_act7_6		Se selecciono este grupo de claves por ser el más numeroso
s_act7_cp3	SCIAN: 3er. componente principal (7 grupos)	R	-0.3	0.8		(0.41)s_act7_1 + (0.64)s_act7_2 + (-0.48)s_act7_3 + (-0.38)s_act7_4 + (-0.21)s_act7_5 + (0.02)s_act7_6		
cmo_cp3	CMO: 3er. componente principal	R	-0.4	0.8				
cmo_cp6	CMO: 6ta. componente principal	R	-0.1	1.1				
p11_h5	Horas dedicadas a realizar quehaceres en el hogar	D	1	98	99		Se asignó el promedio de horas dedicadas a quehaceres domésticos por sexo	
Mic_neg_sl	Identifica a los micronegocios sin local	C	0	1		1 = Sí; 0 = No		Se extrajo de la variable T_UE9 (ámbito y tamaño de la unidad económica)
Patrón	Identifica a los empleadores (con trabajadores a sueldo)	C	0	1		1 = Sí; 0 = No		Se extrajo de la variable POS (posición en el trabajo)
Sect_Inf	Identifica a los trabajadores en el sector informal	C						Se calcula con base en las preguntas 4c, 4e y 4g (T_MER21)
Sin_Fin_Lucro	Es una institución pública o una sin fines de lucro	C	0	1		1 = Sí; 0 = No		
Sec_Privado	Se trata de una actividad o negocio del sector privado	C	0	1		1 = Sí; 0 = No		
Trab_todo_anio	Trabaja todo el año	C	0	14		14 = Sí; 0 = No		Se asignó el valor 0 a los casos que tenían NA
p5_cp3	3er. componente principal de las preguntas 5, 5A y 5D		-1.2	1		(0.008)p5 + (.23)p5a + (.97)p5d	Se asignó a la clase más frecuente en cada pregunta	
Sindicalizado	¿En este empleo pertenece a algún sindicato?		0	1		0 = Sí; 1 = No	Se extrajo de la segunda clase de la pregunta 3I	
Tam_emp	Número de personas que laboran en la empresa o negocio		1	11	99	01= 1 per; 02= 2 a 5 per; 03= 6 a 10 per; 04= 11 a 15 per; 05= 16 a 20 per; 06= 21 a 30 per; 07= 31 a 50 per; 08= 51 a 100 per; 09= 101 a 250 per; 10= 251 a 500 per; 11= 501 y más per; 99= NS	Se recuperó la información no especificada de 3q con T_UE9 y de 3g con la posición en el trabajo	Se calculó en conjunto con las preguntas 3g y 3q
p3r_anio	¿En que año entró a trabajar?		1931	2005	9999		Se asignó la media por año cumplido a 206 casos	
p3r_anio2	En que año entró a trabajar? Al cuadrado							
p6_7	Sólo recibe sueldo, salario o jornal		0	7		7 = Sí; 0 = No	Se asignó 0 a las respuestas No y No aplica	
Busca_otro	Entonces ¿no ha tratado de buscar otro trabajo?		0	4		4 = Sí; 0 = No	Se asignó 0 a las respuestas No y No aplica	
Subocup	Subocupados		0	1		1 = Sí; 0 = No		
Primario	Sector primario		0	1		1 = Sí; 0 = No		
Secundario	Sector secundario		0	1		1 = Sí; 0 = No		
Terciario	Sector terciario		0	1		1 = Sí; 0 = No		
Jefe	Identifica al jefe del hogar		0	1		1 = Sí; 0 = No		Se tomó de Parentesco
Conyuge	Identifica al cónyuge del jefe del hogar		0	1		1 = Sí; 0 = No		Se tomó de Parentesco

Nombre	Descripción	Tipo	Mín.	Máx.	No esp.	Catálogo	Tratamiento a no especificados	Observación
IAGobierno	Institución administrada por el gobierno		0	1		1 = Sí; 0 = No		Se calculó a partir de P4D1
NoApoyoGob	En los últimos 3 meses no ha recibido apoyo por parte del gobierno		0	1		1 = Sí; 0 = No		
IDENTIFICACIÓN GEOGRÁFICA Y DEL MARCO DE MUESTREO								
Entidad	Entidad federativa		1	32				
Ciudad	Ciudad		1	86				
UPM	Unidad primaria de muestreo		8	426				
CON	Número de control		1	6042				
V_SEL	Vivienda seleccionada		1	23				
N_HOG	Número de hogar		1	5				
N_REN	Número de renglón		1	20				
FAC	Factor de expansión		5	4650				
Log_ing	Ingreso mensual (logaritmo)		5.147	9.976				

Cuadro 4.6 Lista de variables que conforman el conjunto de datos para modelar el ingreso de la ENOE 105

4.2.2. Explorando las variables seleccionadas

En este apartado incluimos un par de gráficas que muestran la relación que algunas variables auxiliares tiene con el ingreso.

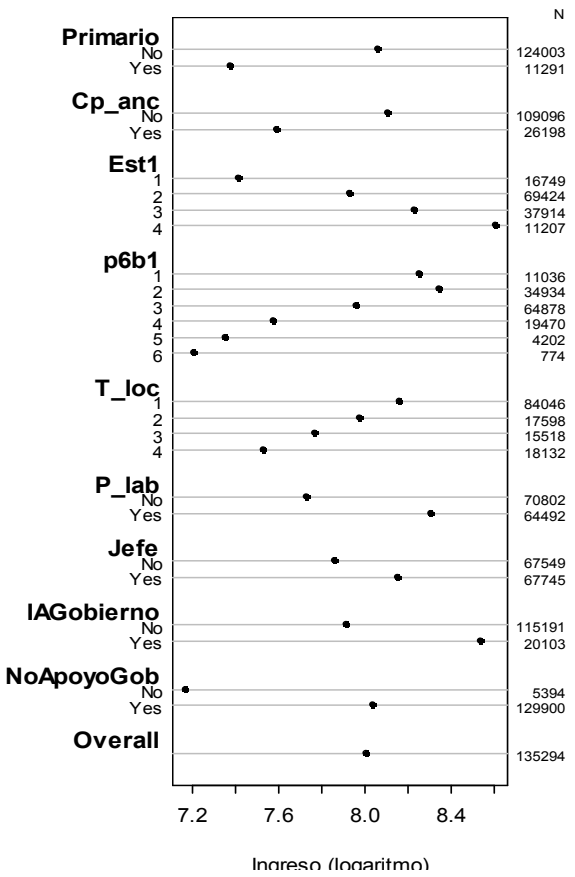


Figura 4.9 Media muestral de una selección de variables independientes.

En la figura de la izquierda se presenta la media de los ingresos para las variables Sector primario, Cuentas propias en actividades no calificadas, Estrato, Tamaño de localidad, Prestaciones laborales, ¿Es el jefe del hogar?, ¿Institución administrada por el gobierno? y, por último, la identificación de las personas que No recibieron apoyo del gobierno. En el anexo 4.4 incluye la media por estratos para las 46 variables seleccionadas para ajustar los modelos.

Observese que las personas con menos ingresos son aquéllas que desempeñan su oficio en el sector primario (agricultura, caza y pesca), que viven en zonas en estratos socioeconómicos bajos, los jornaleros o destajistas (reciben el salario diario, por pieza vendida o servicio realizado), los que habitan en zonas de baja densidad poblacional y finalmente las personas que reciben apoyos del gobierno (becas de capacitación, apoyos económicos para buscar trabajo, microcréditos, procampo, oportunidades, etcétera).

Árboles

Se ha observado que los árboles de decisión pueden ayudar a realizar inferencia a través de la interpretación de los segmentos y nodos que conforman el arbol. También que permite descubrir relaciones presentes en el conjunto de datos y que con frecuencia no son evidentes o fáciles de deducir con otras herramientas.

Un arbol de desición divide recursivamente el espacio al que pertenecen los datos. Emplea una función aditiva que se utiliza para generar las estimaciones.

$$\hat{y}(x) = \sum_i d_i I(x \in R_i)$$

Donde: d_i: es el valor asignado a la i-ésima region.
$I(x \in R_i)$: Es una variable indicadora que toma el valor de 0 sí el elemento x no pertenece a la region i, y 1 en caso de que pertenezca.

Para ajustar el arbol se utiliza una medida de impureza para evaluar la construcción de un arbol determinado, la cual se selecciona en función del problema que se desea atacar, por ejemplo: regresión ó clasificación. La medida de impureza para regresión es la siguiente:

$$I(R) = \frac{1}{\#R} \sum_{i:x_i \in R} (y_i - d)^2$$

En donde *#R* es el número de elementos que contiene la region *R*, *d* es el valor que se le ha asignado a la región *i*. El valor real que tiene la observación *i* es y_i.

Las instrucciones para ajustar un arbol en R se presentan a continuación.

```
dat <- as.data.frame(cbind(Log_ing, Edad, A_acum, N_ins, S_soc,
       Sexo, Cp_anc, hrs_tra, E_jor, Est2, p6b1, aban_a_pas,
       emp_a_pas, neg_a_cp, per_a_pas, P_lab, T_con, p4, T_loc,
       s_act7_cp2, s_act7_cp3, cmo_cp3, cmo_cp6, p11_h5,
       Mic_neg_sl, Patron, Sect_Inf, Sin_Fin_Lucro, Sec_Privado,
       Trab_todo_anio, p5_cp3, Sindicalizado, Tam_emp, p3r_anio,
       p6_7, Busca_otro, Subocup, Primario, Secundario, Terciario,
       Jefe, Conyuge, IAGobierno, NoApoyoGob, FAC)

require (rpart)
fit <- rpart(Log_ing ~ . , data=dat, method="anova", weights=FAC)
plot (fit)
text (fit)
```

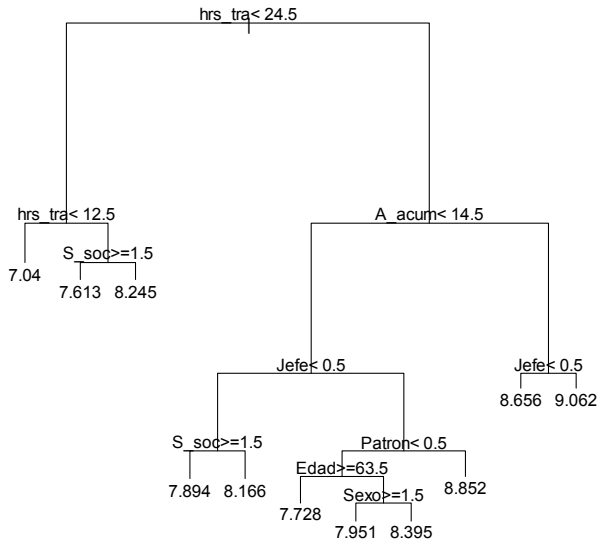


Figura 4.11 Árbol de decisión de las variables independientes seleccionadas contra el logaritmo de los ingresos.

Observamos que la variable `hrs_tra` horas trabajadas a la semana se ubica en el nodo raíz y divide a la población en aquéllos que trabajan media jornada (4 horas durante 6 días), en el segundo nivel de profundidad encontamos una división para aquéllos que laboran 12 horas o menos y cuyo ingreso promedio es el menor de todos los nodos terminales y equivale aproximadamente al pago de un salario mínimo ($45 diarios). Los que dedican a su actividad entre 13 y 14 horas y tienen acceso a instituciones de seguridad social (S_soc==1) tienen un ingreso semejante a la media nacional.
La rama de la derecha discrimina a la población dependiendo si tiene estudios de licenciatura (A_acum < 14.5), en el segundo nivel indaga si el ocupado en cuestión es jefe de su hogar (Jefe==1) o si lo es de su empresa (Patrón==1). De tal modo que el grupo que en promedio tiene el mayor ingreso promedio del árbol son los profesionistas (licenciatura, maestría y doctorado) que son jefes de su hogar. En los niveles más profundos del árbol se indaga sobre la edad y sexo del trabajador, los cuales decrementa su ingreso si son mayores de 63 años o son mujeres.

4.2.3. Distribución espacial

La ENOE esta diseñada para generar resultados a i)nivel nacional, ii) nacional con desglose a cuatro tamaños de localidad iii) estatal y iv) 32 de las principales ciudades. Resulta interesante observar como se distribuye la variable a lo largo del país, para tener una idea del similitudes y diferencias y en caso de que se juzge necesario se incorpore esa información al modelo de imputación.

Entidad federativa	Ingreso promedio	Error estándar
AGUASCALIENTES	4,010	64
BAJA CALIFORNIA	5,480	74
BAJA CALIFORNIA SUR	5,029	129
CAMPECHE	3,583	90
COAHUILA	4,222	105
COLIMA	4,090	91
CHIAPAS	2,306	59
CHIHUAHUA	4,587	132
DISTRITO FEDERAL	4,629	72
DURANGO	3,353	79
GUANAJUATO	3,643	78
GUERRERO	3,267	77
HIDALGO	3,284	90
JALISCO	4,023	76
MÉXICO	3,674	64
MICHOACÁN	3,396	113
MORELOS	3,340	69
NAYARIT	3,329	68
NUEVO LEÓN	4,915	62
OAXACA	2,884	75
PUEBLA	3,027	51
QUERÉTARO	4,067	75
QUINTANA ROO	4,537	81
SAN LUIS POTOSÍ	3,294	59
SINALOA	3,767	81
SONORA	4,407	123
TABASCO	3,663	99
TAMAULIPAS	4,091	98
TLAXCALA	2,978	52
VERACRUZ	3,205	83
YUCATÁN	3,192	69
ZACATECAS	3,443	106

Cuadro 4.7A Ingreso promedio y error estándar por entidad federativa. Fuente: ENOE 105.

Ciudad	Ingreso promedio	Error estándar
01 CD. DE MÉXICO	4,259	57
02 GUADALAJARA	4,567	60
03 MONTERREY	5,145	63
04 PUEBLA	4,103	65
05 LEÓN	4,237	48
07 SAN LUIS POTOSÍ	4,323	78
08 MÉRIDA	4,003	79
09 CHIHUAHUA	4,952	80
10 TAMPICO	4,403	89
12 VERACRUZ	3,933	81
13 ACAPULCO	3,639	72
14 AGUASCALIENTES	4,406	82
15 MORELIA	4,371	70
16 TOLUCA	4,242	90
17 SALTILLO	4,674	82
18 VILLAHERMOSA	5,109	94
19 TUXTLA GUTIÉRREZ	3,920	83
21 TIJUANA	5,861	93
24 CULIACÁN	4,667	75
25 HERMOSILLO	4,927	79
26 DURANGO	3,932	71
27 TEPIC	4,298	67
28 CAMPECHE	3,622	77
29 CUERNAVACA	3,814	96
31 OAXACA	3,992	78
32 ZACATECAS	4,592	93
33 COLIMA	4,353	80
36 QUERÉTARO	4,739	88
39 TLAXCALA	3,267	58
40 LA PAZ	5,044	95
41 CANCÚN	5,254	83
43 PACHUCA	4,417	90

Cuadro 4.7B Ingreso promedio y error estándar por ciudad con representatividad en la encuesta. Fuente: ENOE 105.

En la tabla anterior observamos que las Entidades con mayores ingresos promedio son Baja California, Baja California Sur y Nuevo León, seguramente la influencia del vecino país del norte impacta a la alza las percepciones de los trabajadores en esa zona del país. Los salarios más bajos se otorgan en el sur y centro: Chiapas, Oaxaca y Tlaxcala.

En cuanto a las ciudades observamos que los mejores salarios se dan en Tijuana, Cancún y Monterrey, resulta consistente en dos de tres casos con lo observado con las Entidades. Por otra parte, los menores ingresos se ubican en Tlaxcala, Campeche y Acapulco.

4.3. La no respuesta

Según Kalton y Krasprzyk [1] los datos perdidos (missing data) en las encuestas se deben esencialmente a la no respuesta total e individual/parcial, la primera se refiere a la falta de información en todas las preguntas que componen el cuestionario y se presenta en aquellas

situaciones en donde el informante se niega a contestar el cuestionario o en casos en donde no se puede localizar a la vivienda seleccionada o a las personas que la habitan, la segunda sucede cuando el informante desconoce o se niega a responder uno o un grupo de los reactivos que constituyen el instrumento o cuando uno de los valores registrados en el cuestionario es rechazado por una regla de validación (edit).

Cabe aclarar que existe diferencia entre la no respuesta individual y la parcial. La individual señala la carencia de información sobre un reactivo en particular y la parcial se refiere a los cuestionarios en que un grupo de preguntas tienen respuesta y el resto no, a lo largo del documento nos referimos a los dos términos de manera indistinta.

Causas de la pérdida de datos	Omisión Negativa Desconocimiento Eliminado porque no satisface los criterios de validación

Cuadro 4.8 Causas de la pérdida de datos.

Usualmente los casos de no respuesta total contienen únicamente la información relativa a la ubicación geográfica y al marco de muestreo (Estrato, UPM y USM). Los aspectos más relevantes de esta información se toman en cuenta para ajustar los factores de expansión y así compensar la pérdida de casos completos.

Por otro lado, la no respuesta parcial implica un reto importante ya que resulta deseable incorporar la información que ha sido captada en el resto de las preguntas del cuestionario. Se han desarrollado una amplia gama de métodos para imputar datos de este tipo. En el siguiente capítulo encontraremos una breve descripción de las técnicas principales.

4.3.1. Tipos de no respuesta parcial

La no respuesta parcial puede clasificarse dependiendo del patrón que presenta, Frank Harrell [2] realiza la siguiente agrupación:

1. Completamente aleatoria (MCAR por sus siglas en inglés). Las razones para la pérdida de información no están relacionadas con las características o respuestas del tema, lo anterior incluye el valor omitido, si se conociera. Un ejemplo es la pérdida de los resultados de un análisis de laboratorio porque el tubo con la muestra se cayó al suelo.

2. Aleatoria (MAR por sus siglas en inglés). Los datos no son perdidos de manera aleatoria, sin embargo la probabilidad que un valor se pierda depende de valores de variables que de hecho se midieron. Por ejemplo, considere una encuesta en donde las mujeres son menos propensas a responder su ingreso personal (pero la probabilidad de responder es independiente a su ingreso actual). Si nosotros conocemos el sexo de cada persona y tenemos los niveles de ingreso para algunas mujeres se podría realizar una estimación insesgada del ingreso por sexo. Los datos tipo MAR y MCAR también son llamados *no respuesta ignorable*.

3. Informativa (IM por sus siglas en inglés). Los elementos son más probables de perderse si los valores verdaderos de la variable en cuestión son sistemáticamente más altos o bajos. Un ejemplo de ello es cuando los informantes con ingresos bajos o muy altos son menos propensos a declarar su ingreso personal en una entrevista. IM es también llamada no respuesta no ignorable.

La no respuesta IM es el tipo más difícil de manejar. En algunos casos no hay solución para recuperar los valores omitidos, aun cuando existe la manera de proporcionar evidencia de que se trata de datos tipo IM. Se tienen que hacer consideraciones relativas al muestreo y al tipo de variable para proponer un modelo que busque asignar valores a respuestas IM. Por otro lado, MCAR es el caso más fácil de manejar. Nuestra habilidad de corregir y analizar datos tipo MAR depende del poder predictivo de las variables auxiliares. La mayoría de los métodos disponibles para tratar con datos perdidos asumen que los datos son de tipo MAR.

Entonces resultaría benéfico identificar el tipo de no respuesta al que nos enfrentamos, ya que de él dependerá los métodos que podemos emplear para recuperar la información.

Para indentificar el tipo de no respuesta debemos formular las siguientes preguntas:

No	Pregunta	MCAR	MAR	IM
1	¿La información perdida no se relaciona con las características o respuestas obtenidas en el cuestionario?	✓	×	×
2	¿La probabilidad que un valor se pierda depende de valores de variables que efectivamente se midieron?	×	✓	=
3	¿Los elementos son más probables de perderse si los valores verdaderos de la variable en cuestión son sistemáticamente más altos o bajos?	×	×	✓

Cuadro 4.9 Preguntas para identificar el tipo de no respuesta.

El signo "=" en la segunda pregunta se refiere a que puede presentarse o no cuando el tipo de no respuesta es IM. Ahora bien, podríamos aprovechar el hecho que las dos primeras preguntas son escencialmente excluyentes y si encontramos un ejemplo que apoye una afirmación a la pregunta dos, entonces, podemos descartar que se trate del tipo MCAR.

En la sección 2.3. La variable ingreso por trabajo se dijo que la tasa general de no respuesta de las percepciones netas del trabajo se ubica en 5.4% de la población ocupada, a partir de este porcentaje inicial podemos comenzar a indagar sobre los patrones que sigue para subgrupos de la población.

4.3.2. Estratificación del Marco Nacional de Viviendas

El Marco Nacional de Viviendas (MNV) es el instrumento que el INEGI utiliza como fuente para la selección de muestras que son utilizadas en todas sus encuestas en hogares. Se realiza a partir de los listados de viviendas elaborados durante los censos de población que en nuestro país se realizan cada cinco años. Durante su construcción las viviendas se aglutinan en bloques *homogéneos* llamados unidades primarias de muestreo (UPM). Cada bloque es clasificado como miembro de un cierto estrato socioeconómico. El personal que labora en la ENOE está firmemente convencido que existe una relación entre el estrato socioeconómico y los porcentajes obtenidos de no respuesta parcial y total. A continuación se describe brevemente el proceso de estratificación.

El MNV [12] se estratificó con base en una serie de variables provenientes del cuestionario censal (ver cuadro 4.10), las cuales fueron agregadas y promediadas hasta el nivel de UPM, enseguida se utilizó la técnica k-medias para agrupar las unidades primarias de muestreo en bloques cuya varianza sea mínima, pero que la diferencia en sus medias sea máxima. Las UPM se concentraron en cuatro bloques identificados como estrato socioeconómico bajo, medio bajo, medio alto y alto. Una segunda etapa consistió en realizar una segunda división al interior de los estratos previamente definidos, dando como resultado un subestrato que sigue la misma lógica ordinal.

INDICADORES EMPLEADOS EN LA ESTRATIFICACIÓN POR ÁMBITO DE ESTUDIO

DESCRIPCIÓN DEL INDICADOR	ÁMBITO DE ESTUDIO			
	NACIONAL	URBANO ALTO	COMPLEMEN-TO URBANO	RURAL
% DE VIVIENDA:				
QUE DISPONEN DE AGUA ENTUBADA DENTRO DE LA VIVIENDA		X		
CON DRENAJE	X	X		X
CON ELECTRICIDAD				X
QUE DISPONEN DE AGUA, LUZ, Y DRENAJE	X	X	X	X
CON PISO DIFERENTE DE TIERRA	X			X
CON PAREDES DE MATERIAL SÓLIDO				X
CON COCINA EXCLUSIVA	X	X	X	X
SIN HACINAMIENTO	X	X	X	X
CON SERVICIO SANITARIO EXCLUSIVO CON CONEXIÓN DE AGUA		X		
CON SERVICIO SANITARIO EXCLUSIVO CON ADMISIÓN DE AGUA	X		X	
QUE UTILIZAN GAS PARA COCINAR	X			X
CON RADIO O RADIOGRABADORA	X			X
CON TELEVISIÓN	X			
CON REFRIGERADOR	X	X	X	
CON LICUADORA	X			X
CON AUTOMÓVIL O CAMIONETA PROPIOS	X	X	X	
CON VIDEOCASETERA			X	
CON LAVADORA	X		X	
CON TELÉFONO		X	X	
CON BOILER		X	X	
CON CUATRO BIENES (TELÉFONO, REFRIGERADOR, LAVADORA Y BOILER)		X		
CON CUATRO BIENES (RADIO, TELEVISIÓN, LICUADORA Y REFRIGERADOR)	X		X	
CON EL MÍNIMO EQUIPAMIENTO (RADIO O TELEVISIÓN Y LICUADORA)				X
% DE POBLACIÓN:				
DERECHOHABIENTE A SERVICIO DE SALUD		X	X	
DE 6 A 17 AÑOS QUE ASISTE A LA ESCUELA	X	X	X	
DE 6 A 14 AÑOS QUE ASISTE A LA ESCUELA				X
DE 15 AÑOS Y MÁS ALFABETA	X			X
DE 15 AÑOS Y MÁS CON POSTPRIMARIA	X	X	X	X
GRADO PROMEDIO DE ESCOLARIDAD	X	X	X	X
OCUPADA QUE GANA MÁS DE 2.5 SALARIOS MÍNIMOS	X	X	X	X
OCUPADA QUE GANA MÁS DE 5 SALARIOS MÍNIMOS	X	X	X	
FEMENINA DE 12 AÑOS Y MÁS ECONÓMICAMENTE ACTIVA.	X	X	X	
ECONÓMICAMENTE ACTIVA DE 20 A 49 AÑOS	X	X	X	
OTROS:				
PORCENTAJE DE HOGARES EN EL DECIL NACIONAL 8, 9 Y 10	X	X	X	
RELACIÓN DE DEPENDENCIA ECONÓMICA	X	X	X	

Cuadro 4.10 Variables utilizadas para estratificar las UPM, según el tamaño de localidad.

La variable estrato socioeconómico es importante pues permite a los procedimientos de selección de las muestras generar estimadores con menor varianza (si los estimadores están correlacionados con las variables empleadas en su construcción). Para el proceso de imputación resultaría util si se lográ verificar su relación con la no respuesta y los niveles de ingreso.

4.3.3. El estrato socioeconómico y la no respuesta

Como logramos observar en el apartado anterior la estratificación del MNV incluye características que podemos agrupar en desarrollo humano, económico y urbano de las viviendas que conforman las UPM. De manera particular, si logramos identificar una relación entre el estrato y la no respuesta o el estrato y los ingresos, nos podría servir en el primer caso para identificar el patrón que sigue la no respuesta y en el segundo para utilizarla como variable auxiliar para recuperar los ingresos perdidos.

Estrato socioeconómico		Total		¿Cómo declaró los ingresos?					
				Directamente		Por intervalos		No especificado	
		Absoluto	Relativo	Absoluto	Relativo	Absoluto	Relativo	Absoluto	Relativo
Total		**36 694 268**	**100.0**	**32 887 240**	**89.6**	**1 773 163**	**4.8**	**2 033 865**	**5.5**
Bajo	10	1 912 205	100.0	1 835 356	96.0	31 163	1.6	45 686	2.4
	11	689 265	100.0	653 506	94.8	13 925	2.0	21 834	3.2
	12	1 566 404	100.0	1 477 140	94.3	25 676	1.6	63 588	4.1
	13	1 416 156	100.0	1 337 929	94.5	28 744	2.0	49 483	3.5
	14	432 328	100.0	412 059	95.3	5 643	1.3	14 626	3.4
Medio-bajo	20	8 031 783	100.0	7 536 969	93.8	189 648	2.4	305 166	3.8
	21	2 048 597	100.0	1 894 768	92.5	77 283	3.8	76 546	3.7
	22	3 752 098	100.0	3 383 323	90.2	185 899	5.0	182 876	4.9
	23	4 108 198	100.0	3 547 644	86.4	278 273	6.8	282 281	6.9
	24	1 352 762	100.0	1 138 295	84.1	143 824	10.6	70 643	5.2
Medio-alto	30	2 221 299	100.0	2 055 833	92.6	55 997	2.5	109 469	4.9
	31	312 329	100.0	288 925	92.5	8 726	2.8	14 678	4.7
	32	2 749 790	100.0	2 394 125	87.1	186 242	6.8	169 423	6.2
	33	2 995 918	100.0	2 538 292	84.7	228 264	7.6	229 362	7.7
Alto	40	396 952	100.0	356 244	89.7	20 428	5.1	20 280	5.1
	41	163 853	100.0	126 831	77.4	14 525	8.9	22 497	13.7
	42	1 152 348	100.0	906 748	78.7	115 374	10.0	130 226	11.3
	43	1 391 983	100.0	1 003 253	72.1	163 529	11.7	225 201	16.2

Cuadro 4.11 Población ocupada por estrato socioeconómico según forma en que declaró su ingreso.

En la tabla anterior se presentan las estimaciones para la población ocupada desagregada por estrato socioeconómico y la forma en que reportaron el ingreso. Vemos que de manera global, 89.6% de los ocupados reportaron su salario de manera directa, 4.8% a través de intervalos de salarios mínimos y el restante 5.5% no lo reveló. El comportamiento de la respuesta por intervalos tiene fluctuaciones importantes. El mínimo se presenta en el estrato 14 con 1.3% y el máximo en 43 con 11.7%, para la no respuesta el mínimo se coloca en el estrato 10 con 2.4% y el tope se localiza nuevamente en el 43, aunque existen altibajos se nota una tendencia creciente a la par de la variable estrato.

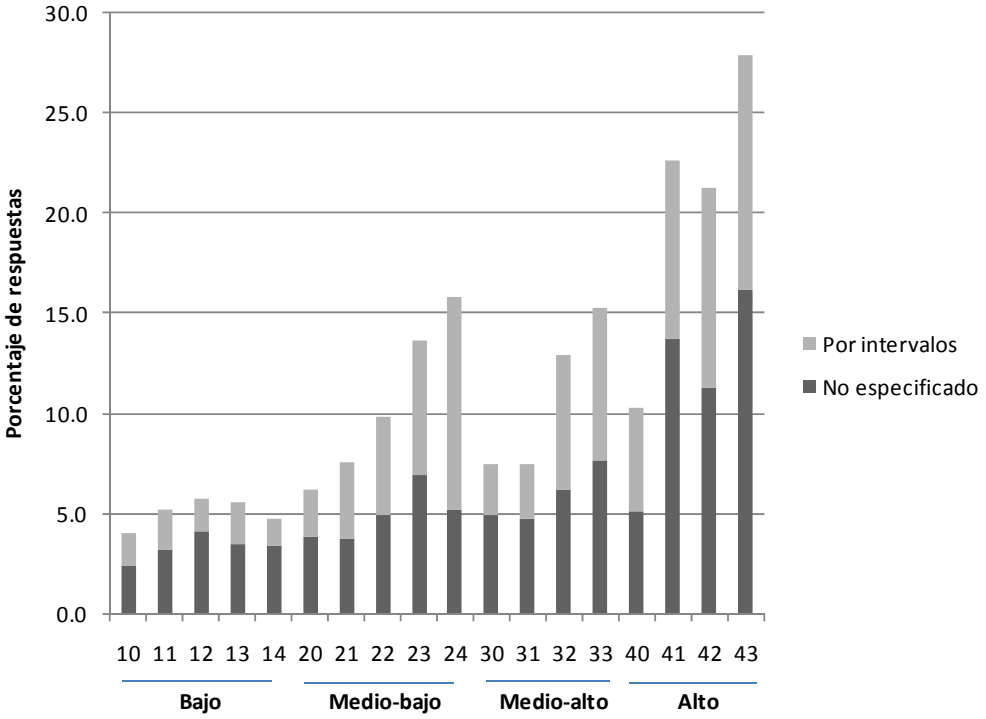


Figura 4.13 Porcentaje de los informantes que respondieron su ingreso por intervalos de salarios mínimos y simplemente no lo especificaron.

Se observa en la gráfica el porcentaje acumulado de las personas que no declararon directamente su ingreso. Confirmamos que la magnitud va en aumento conforme al estrato socioeconómico, el máximo se encuentra en el estrato Alto y los valores con menor magnitud se localizan en el Bajo, por otro lado no se observa mucha diferencia en la respuesta de los estratos Medio-bajo y Medio-alto.

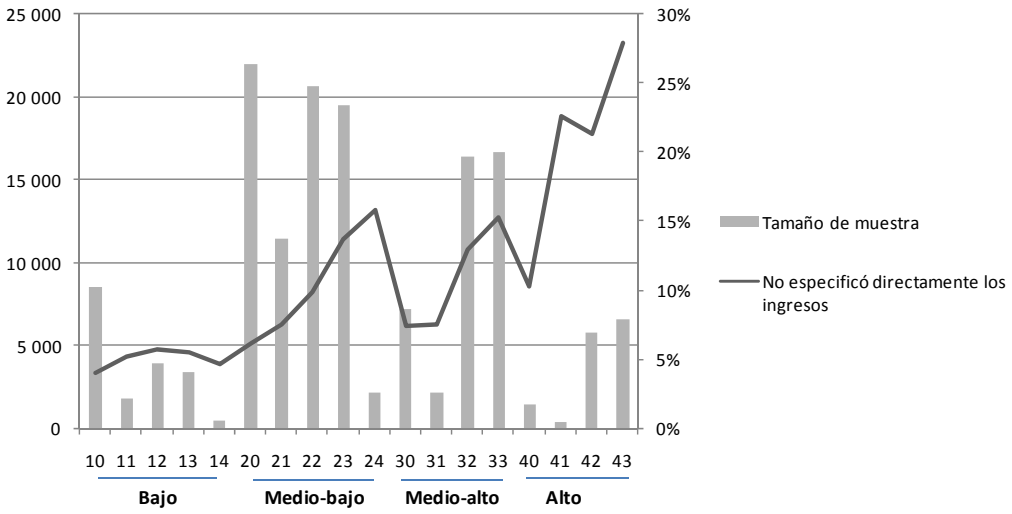


Figura 4.14 Población ocupada entrevistada por estrato socioeconómico y porcentaje de personas que no especificó directamente su ingreso.

La gráfica 4.13 se construyó para observar el efecto combinado del tamaño de muestra y el porcentaje de no respuesta en la variable en cuestion. En términos absolutos el estrato Medio-bajo es el que tiene mayor peso ya que su tasa de no respuesta directa está cerca de 10% y es el grupo con más casos registrados de personal ocupado, no obstante, las altas tasas del estrato con los ingresos más altos podría generar observaciones influyentes al mezclar los ingresos multiplicados por el factor de expansión.

Con la evidencia recabada observamos que efectivamente la probabilidad que un valor se pierda depende de valores de variables que se midieron, en particular el estrato. **Entonces la respuesta a la pregunta 2 es afirmativa y podemos decir que la no respuesta de la variable ingreso no es MCAR**

Enfrentar la pregunta 3 *¿Los elementos son más probables de perderse si los valores verdaderos de la variable en cuestión son sistemáticamente más altos o bajos?* es un asunto complicado ya que para responderla necesitariamos los valores reales de los casos en los que no tenemos información y se requeriría un operativo de levantamiento especial para recuperar el dato insistiendo con el informante, el libro de Sharon Lohr [13] de muestreo sugiere algunos procedimientos para llevar a cavo este tipo de recolección. Lo que si podemos hacer es explorar que sucede con la estimación de los ingresos en relación con el estrato socioeconómico.

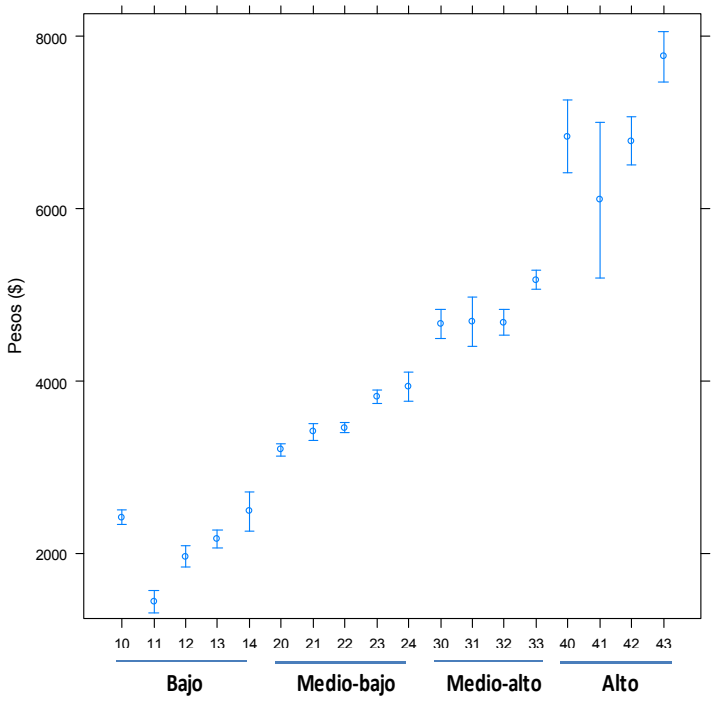


Figura 4.15 Estimación de la media de los ingresos e intervalos de confianza (α=0.05) por estrato socioeconómico.

En la figura anterior observamos que el ingreso promedio tiene una correlación positiva con respecto al estrato socioeconómico. La amplitud de los intervalos se ve afectada principalmente por el tamaño de muestra en cada estrato y por la magnitud de la estimación. El ingreso medio más bajo se encuentra en el estrato 11 con un monto de $1 442 y el máximo en el 43 con $7 765; en términos porcentuales la diferencia es de 438 por ciento.

Si nos ubicamos en un estrato específico, por ejemplo, 24 podremos notar que su media coincide con la que se estima para la población total; sin embargo, con la información disponible *no podemos decir si al interior de este grupo la no respuesta es sistemáticamente más baja o alta*, aun cuando su tasa de respuesta directa sea 84.2%, no obstante, si observamos la gráfica 4.13 y 4.14 *llegaremos a la conclusión de que a mayores ingresos más alta es la tasa de no respuesta*.

La evidencia encontrada no permite desechar la teoría de que la no respuesta sea del tipo IM, al menos que logremos deshacernos de la idea de que fuera MCAR y consigamos acumular datos que nos permitan suponer que la no respuesta es –al menos- tipo MAR.

4.4. Comentarios finales del capítulo

En este capítulo tuvimos la oportunidad de revisar de manera general la distribución de los ingresos y logramos identificar su relación con otras variables. Resolvimos una amplia gama de situaciones conflictivas con los datos de la encuesta (*no aplica y datos perdidos*) y se buscó obtener la mayor información disponible de las variables categóricas.

Elegimos una subpoblación homogénea (ciudad de León, Guanajuato) para que fuera más propicio para el método de predicción atendiendo las acciones que se observaron en experiencias anteriores. Finalmente exploramos la no respuesta con el objetivo de esbozar su patrón y tipo, lo que nos permitirá seleccionar y configurar las técnicas de imputación.

5. Imputación

Existen diferentes métodos para imputar valores perdidos. En la práctica los usuarios de la información con frecuencia excluyen del análisis aquellos registros que carezcan de respuesta en una determinada variable. Este es llamado el método del caso útil (*available case method*). Esos usuarios o investigadores implícitamente asumen que los registros que dejan fuera no aportan información adicional a la que contiene el conjunto de datos con información completa. La mayoría de las ocasiones este supuesto no se cumple.

Podemos decir que en general existen tres tipos de imputación: deductiva, determinística y estocástica. La imputación deductiva (ID) se refiere a imputaciones donde el valor asignado es deducido a partir de información conocida. Por ejemplo la edad de una persona puede obtenerse a partir de la fecha de nacimiento, el ingreso total puede calcularse a partir de la suma de los distintos componentes del ingreso. En una encuesta tipo panel el año de nacimiento de una persona es constante, si en un levantamiento no podemos obtener respuesta en este reactivo podemos esperar la siguiente edición para recuperarlo y asignarlo hacia atrás o adelante según corresponda. Aunque ID es una técnica correcta y frecuentemente utilizada, desde el punto de vista metodológico no es tan importante como los otros dos tipos de imputación.

La imputación determinística y estocástica puede ser usada si el valor correcto no puede ser deducido. Entonces una predicción tiene que hacerse para el valor perdido. Una imputación estocástica requiere la generación de números aleatorios que pertenezcan a la distribución de probabilidad de la variable. Por ejemplo la imputación *hot-deck* puede ser determinística o estocástica. Si un registro es seleccionado aleatoriamente del grupo de registros donantes o si son ordenados aleatoriamente entonces el método *hot-deck* es un método estocástico, de otra manera el *hot-deck* será determinístico.

Un ejemplo de imputación determinística es la imputación de la media, calculada a partir de los valores conocidos y asignada a los registros con valores perdidos en la misma variable. Parece lógico pensar que se obtendrían mejores resultados si se crearan grupos de registros similares en donde la media de dicho grupo sea imputada en lugar de la media general. Otro ejemplo es la regresión que puede ser vista como una generalización de la imputación de la media por grupos.

Una desventaja de la imputación determinística es que la distribución de la variable que se imputa generalmente se afecta, ya sea por que aparecen uno o más picos o porque la varianza se subestima lo que produce distorsiones en su distribución. Para mantener la varianza y covarianza de los datos imputados, es conveniente agregar un componente aleatorio a las técnicas de imputación determinística y así obtener la contraparte estocástica. Una práctica común es agregar un componente aleatorio que se suma como un residual a la predicción efectuada por la parte determinística. Este residual puede ser seleccionado de una distribución normal con media cero y varianza σ^2. Usualmente la sigma cuadrada es igual que la varianza de los residuales de la regresión que se realiza entre la variable a imputar y las variables las auxiliares.

Este capítulo se compone de cinco secciones, en la primera se dará una breve descripción de los métodos de imputación. En la segunda se hace un breve resumen y aplica las técnicas *stepwise* y regresión lasso como procedimientos para la reducción de las variables de la encuesta. La tercera sección incluye el criterio para comparar el desempeño de las técnicas de imputación. En el apartado cuatro se detallan los métodos que se proponen en esta tesis. También se presentan los resultados obtenidos de los ejercicios realizados, empleando para cada método distintas configuraciones y cuando era factible utilizando el conjunto de datos completo y reducido. Finalmente la sección cinco expone los resultados de las mejores configuraciones detectadas para cada técnica de imputación.

5.1. Métodos de imputación

Durante los últimos 50 años se han desarrollado una amplia variedad de métodos de imputación para asignar valores a las preguntas que han quedado sin respuesta en encuestas y censos. La mayor parte de los métodos pueden expresarse con la siguiente fórmula [1]:

$$\hat{y}_{mi} = \beta_{r0} + \sum_j \beta_{rj} z_{mij} + \hat{e}_{mi} \tag{3}$$

Donde $\hat{y}_{mi}$es el valor imputado para el i-ésimo registro con el valor perdido de y, z_{mij} son los valores pertenecientes a las variables auxiliares de dicho registro, β_{r0} y β_{rj} son los coeficientes de regresión de y en x para los informantes que respondieron a la pregunta y $\hat{e}_{mi}$ son los residuales elegidos acorde al esquema específico de el método de imputación en particular.

A continuación se lista una serie de procedimientos de imputación propuestos en Kalton y Kasprzyk (1986) y Laaksonen (Stat Finland 2000) [25].

1. Deductiva o lógica
2. Media general
3. Media por clases
4. Media condicional
5. *Cold-deck*
6. *Hot-deck*
    - 6.1. Aleatorio simple
    - 6.2. Aleatorio por clases
    - 6.3. Secuencial
    - 6.4. Jerárquico
    - 6.5. En función de distancia
7. Regresión
8. Múltiple

1. Imputación deductiva o lógica. En ocasiones la respuesta para una pregunta en particular puede ser deducida a partir del patrón de respuestas obtenidas del mismo informante. Es a juicio de Kalton y Kasprzyk la forma ideal de imputación, sin embargo, lo normal es que a la mayoría de las preguntas no se les pueda aplicar este procedimiento. Es común que la imputación deductiva se realice mientras se ejecuta la validación (edits) ya que su formato es: *if (condición) then (asignación)*.

2. Media general. Todos los valores faltantes de una variable se reemplazan con el valor de la media de las unidades observadas.

3. Media por clases. La muestra se agrupa en clases según lo determinen los valores de las variables auxiliares elegidas. Al interior de cada grupo la media de los que responden es asignada a los valores faltantes.

4. Media condicional. Es el reemplazo de los valores perdidos por medias que se condicionan al resto de variables con respuesta. Si las variables Y_1, .., Y_k pertenecen a una distribución normal multivariada con media μ y la matriz de covarianza Σ. Entonces las variables con valores perdidos tienen regresores lineales en las variables observadas, los coeficientes de regresión que son funciones bien conocidas de μ y Σ. El método fue propuesto por Buck (1960) y primero estimó μ y Σ de la muestra, ambos fueron calculados a partir de los casos completos, y entonces se usaron para predecir vía regresión lineal los valores perdidos como función de las variables con respuesta. El cálculo de diferentes regresores lineales para cada patrón de datos perdidos sería formidable, aunque se puede hacer utilizando otros criterios de agrupamiento. Little y Rubin (1986) [11].

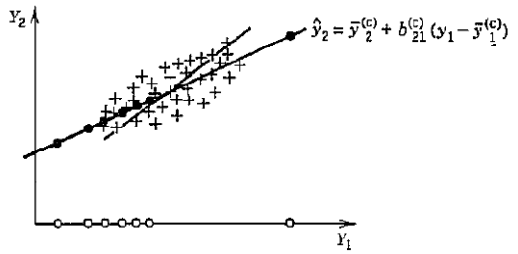


Figura 5.1 Método de Buck con dos variables (K=2)

En la figura anterior podemos ver el método de Buck ilustrado para dos variables. Los puntos marcados con "+" representan los casos en que ambas Y_1 y Y_2 son observadas. Esos puntos son usados para calcular la línea de regresión por cuadrados mínimos de Y_2 sobre Y_1. $\hat{y}_2 = \bar{y}_2^{(c)} + b_{21}^{(c)}\left(y_1 + \bar{y}_1^{(c)}\right)$. Donde el superíndice c identifica a los casos completos. Casos con Y_1 observado pero que Y_2 está perdido se representa por círculos en el eje Y_1. El procedimiento de Buck los remplaza por puntos que fueron colocados encima de la línea de regresión. Los casos con Y_2 observada y Y_1 pérdida se podrían imputar en la línea de regresión Y_1 sobre Y_2, que es la otra línea en el diagrama. Los promedios de los valores observados e imputados generados por este procedimiento son estimadores consistentes bajo el supuesto MCAR.

5. *Cold-deck*: Se define un conjunto de casos donantes con base en fuentes de información externa: levantamientos anteriores de la encuesta, distribución de frecuencias, valores establecidos por expertos del tema. El procedimiento asigna a los campos sin información el valor del donante, los casos se pueden agrupar en clases o estratos. La calidad de la imputación depende directamente de la información considerada para construir los registros donantes.

6. *Hot-deck*. En este método se sustituyen los valores individuales extraídos de unidades observadas (donantes) en la mayoría de los casos similares. Es una práctica muy difundida que involucra cada vez más esquemas elaborados para la selección de los casos similares Rubin (1987) y se puede definir con el método en el cual el valor imputado se selecciona de una distribución estimada. Normalmente la distribución empírica es la empleada para reproducir el patrón observado. El procedimiento tiene ventajas como: genera valores que encajan con los aceptados, conserva la distribución y no requiere supuestos fuertes para estimar. Algunas desventajas son: distorsiona la relación con el resto de las variables y carece de un elemento probabilístico.

6.1. *Hot-deck*: Aleatorio simple. Un donante es elegido aleatoriamente del conjunto total de casos que contestaron, el donador seleccionado concede el valor de su característica en cuestión para que sea asignado al caso que no tiene respuesta. Receptores y donantes son tomados de la misma encuesta y corresponde al método *Hot-deck* más sencillo.

6.2. *Hot-deck*: Aleatorio por clases. Este método selecciona aleatoriamente un caso con respuesta dentro de una clase o grupo definido para la imputación, el valor del registro seleccionado es asignado al caso que no tiene respuesta.

6.3. *Hot-deck*: Secuencial. El procedimiento inicia con el establecimiento de un conjunto de clases de imputación. Cada clase dispone de un valor de referencia (VR), se inicializa tomando el valor de una edición anterior de la encuesta o a través de *Cold-deck*, los casos al interior de cada clase se ordenan de cierta manera considerando las variables auxiliares. La ejecución del proceso ocurre de manera ordenada, tomando registro a registro y sí un caso determinado no presenta respuesta se imputa con el contenido de VR, pero si el registro tiene valor entonces, éste es almacenado por VR y podrá ser empleado para imputar el siguiente registro. Este método tiene la desventaja de que ante una racha de registros a imputar se utiliza el mismo donante, y que existe cierta dificultad en establecer los valores iníciales de cada clase y por otro lado su ventaja estriba en que los donadores son más cercanos gracias al ordenamiento, y su ejecución secuencial produce una autocorrelación positiva, aunque este beneficio no es medular para las estimaciones.

Clase h	Variable de interes Y	
No. de registro	Antes de imputar	Después de imputar
1	NA	*yh*
2	45	45
3	34	34
4	NA	34
5	30	30
6	56	56
7	47	47
8	40	40
9	NA	40
10	NA	40

Nota: *yh*: Valor de inicio asignado por *cold-deck* para la clase h.
NA: "Not available", identifica los registros sin respuesta.

Cuadro 5.1 Ejemplo de imputación *hot-deck* secuencial.

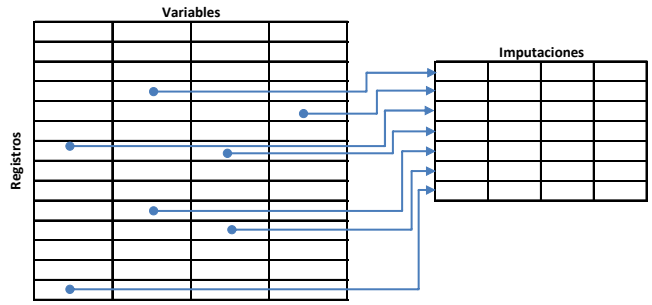
En el cuadro anterior observamos que el primer caso de la clase h no tiene información por lo tanto se le asigna el VR inicial establecido a través de un procedimiento *cold-deck*. El caso 4 recibe el valor del registro anterior. Los registros 9 y 10 forman una racha de no respuesta y por tanto toman el dato captado en el renglón 8.

6.4. *Hot-deck*: Jerárquico. Las desventajas del método secuencial son resueltas por el jerárquico, una forma de *hot-deck* desarrollada para las preguntas del Suplemento de Ingresos del mes de Marzo de la CPS (*Current Populaton Survey* realizada por BLS). El procedimiento ordena los registros con y sin respuesta en un amplio número de clases de imputación elaboradas a partir de una categorización detallada de un gran conjunto de variables auxiliares. Los que no responden son empatados con los que si respondieron a partir de un enfoque jerárquico, en ese sentido, si un empate no puede realizarse con las clases de imputación iníciales, entonces las clases son colapsadas y el empate es realizado a un nivel de detalle más bajo. Para ver más detalles consulte Coder (1978) y Welniak y Coder (1980) [1].

6.5. *Hot-deck*: en función de distancia. Es un procedimiento no paramétrico basado en la identificación de casos cercanos en un determinado espacio. Sobresalen dos elementos a considerar en este método i) la medida de distancia, entre las más comunes se encuentran la Euclidiana y de Mahalanobis Sande (1979); Vacek y Ashikago (1980) [1] y se calcula a partir de las variables auxiliares, y ii) el número de (k) vecinos a considerar, una manera de elegir el mejor k es a través de validación cruzada. Por otro lado la función para agregar los k-vecinos seleccionados depende del tipo de variable, si es categórica se puede elegir la case que tenga mayor frecuencia (por supuesto debemos decidir que sucede en caso de empate), pero si es numérica lo más común es optar por la media común o la ponderada.

7. Regresión. Este método usa la información de los casos con respuesta para predecir el valor de la variable que requiere imputación, apoyándose en un grupo de variables auxiliares. El valor imputado puede ser el valor predicho o el predicho más el residual y así volverlo un método estocástico. Si el valor imputado es el valor predicho (con un solo patrón) entonces el resultado coincide el método 4. Media condicional.

8. Múltiple. Propuesto por Donald B. Rubin en su libro Imputación Múltiple (1987). Es una técnica que remplaza cada valor perdido o deficiente (que no cumple con las reglas de validación) con dos o más valores aceptables que representan una distribución de posibilidades.



Cada vector renglón de imputación es de longitud m, donde
El modelo para la primer imputación = ⋯
El modelo para la segunda imputación = ⋯
:
El modelo para la m-ésima imputación = ⋯

Figura 5.2 IM Conjunto de datos con m imputaciones para cada dato perdido.

La idea detrás de la imputación múltiple es que para cada dato perdido se imputa varias veces el valor, digamos *m*-veces, en lugar de solo una, como lo podemos ver en la figura 5.2 IM. Esos *m* valores están ordenados de tal manera que el primer conjunto de valores imputados (primer columna) es usado para formar el primer conjunto de datos completos y así sucesivamente. Si utilizamos imputación múltiple con valores pequeños para *m*, digamos entre 2 y 10 se pensaría que los datos tienen tasas pequeñas de no respuesta.

Las *m* imputaciones de cada dato perdido crea *m* conjuntos de datos completos. Cada conjunto completo de datos es analizado usando procedimientos estadísticos estándar (para datos completos) justo como si los datos imputados fueran reales. Los resultados se combinan para producir estimaciones e intervalos de confianza de las variables con datos perdidos.

Existen algunas ventajas importantes de la imputación múltiple sobre la imputación simple. Por ejemplo cuando las imputaciones se realizan de forma aleatoria como un intento de representar la distribución de los datos. El resultado deriva en un incremento de la eficiencia de la estimación (ver capítulo 4 del libro Imputación Múltiple).

5.2. Reducción del conjunto de datos

La meta de la imputación es predecir el ingreso lo mejor posible con la información disponible y los métodos propuestos en teoría deben incluir tantas variables auxiliares como sea posible y por supuesto que estén relacionadas con el ingreso. Sin embargo el incremento de variables puede generar problemas al momento de estimar/ajustar el modelo; por ejemplo si se trabaja con un modelo de regresión, puede surgir multicolonealidad [4], Little (1993).

De cualquier manera para minimizar los costos de procesamiento cuando la imputación es implementada, los censos y encuestas más importantes en Estados Unidos de Norteamérica buscan modelos con el máximo poder predictivo y un mínimo de variables. Frank E. Harrell [2] sugiere los siguientes puntos a considerar en la reducción de variables:

1. Utiliza la literatura disponible para eliminar las variables sin importancia.
2. Elimina las variables que tienen distribuciones muy extrañas.
3. Elimina predictores que tienen un alto número de elementos sin respuesta (*missing data*)
4. Utiliza métodos estadísticos.

Durante el desarrollo del capítulo anterior se realizó una selección de los campos contenidos en la base de datos de la ENOE, comenzamos con 300 variables y se descartaron utilizando principalmente los criterios 2 y 3, en algunos casos se agruparon variables utilizando componentes principales (criterio 4), o seleccionando a un representante hasta llegar a las 46 que ahora tenemos. Sin embargo para disponer de un conjunto reducido de variables con igual o mayor poder predictivo se optó por emplear la técnica *lasso* propuesta por Tibshirani (1996).

Greenless et al (1982) [11] propusieron formar una serie de subpoblaciones en la CPS para ejecutar el proceso de imputación hot-deck dentro de cada una. Uno de los objetivos que buscaban era obtenr grupos más homogéneos en donde las carácteristicas de las personas sean semejantes. En este estudio remtomamos esta idea, en un inicio se penso seleccionar a una entidad federativa completa, no obstante se ha encontrado que el comportamiento qe tiene el mercado de trabajo en las ciudades dista mucho del que existe en las zonas rurales.

Fue entonces que colocamos la atención en las 32 ciudades autorepresentadas de la encuesta. Apreciamos que la Ciudad de León, Guanajuato cuenta con las características que distingue a una buena parte de las zonas urbanas de la mesa central del país con un desarrollo económico y social arriba de la media nacional. Además cuenta con la menor estimación de la varianza para la media de los ingresos (ver tabla 4.7B) por lo que los ejercicios de imputación se realizarán sobre la capital económica de Guanajuato.

5.2.1. Regresión *Stepwise*

Es un método alterno para la selección de variables que identifica buenos (pero no necesariamente los mejores) modelos seleccionando un subconjuntos de variables, que utiliza considerablemente menos poder computacional que otros métodos alternos basados en regresión. Ese método es referido como método de regresión *stepwise*. El conjunto de características seleccionadas son identificadas secuencialmente al agregar o eliminar variables. [14]

Fordward stepwise: Elige modelos agregando una variable a la vez a un conjunto elegido previamente. Se inicia eligiendo el primer subconjunto constituido por la variable independiente que explique la mayor cantidad de variación de la variable dependiente. Será la que tenga la mayor correlación simple con Y. A cada paso que le preceda se elegirá a la

variable que mayor reducción signifique a la suma de cuadrados del error (SCE), sin un criterio de paro, se continuará hasta que todas las variables sean incorporadas al modelo.

Backward elimination: Todas las variables elegidas para el modelo inicial y en cada paso una variable se identifica para excluirla del modelo, el criterio es la que cause un menor incremento en la SCE. Semejante al procedimiento anterior se concluye con todas las variables fuera del modelo.

Ni la selección forward ni la eliminación *backward* toman en cuenta el efecto que la eliminación o adición de una variable puede tener en la contribución del resto de variables en el modelo. Una variable agregada en los primeros pasos del algoritmo puede tener un efecto insignificante después de que otras variables hayan sido agregadas, o variables previamente eliminadas pueden ser importantes después de que otras hayan sido eliminadas del modelo. La selección de variables comúnmente llamada *stepwise regression* es una selección forward que verifica a cada paso la importancia de todas las variables previamente incluidas. Si la SCE parcial para cualquier variable previamente incluida no cumple un cierto criterio mínimo para pertenecer al modelo, entonces, el procedimiento cambia a eliminación *backward* y las variables son eliminadas una a la vez hasta que todas las variables restantes cumplan el criterio mínimo, entonces la selección forward regresa.

Las funciones implementadas en los paquetes estadísticos normalmente incluyen un criterio para terminar el proceso de selección, en el caso de selección forward el criterio más común es la tasa de reducción en el SCE causada por la siguiente variable candidata a ser considerada. Este criterio puede expresarse en términos de un valor crítico de entrada "*F-to-enter*" o en términos de un "nivel de significancia para entrar" (SLE por sus siglas en Inglés), donde F es la "F-test" de la suma parcial de cuadrados (SPC) de la variable que está siendo considerada. La selección forward termina cuando las variables fuera del modelo no cumplen con el criterio para entrar.

La regla de paro en la eliminación *backward* es el "*F-test*" de la variable con el menor SPC de las características que permanecen en el modelo. Nuevamente, este criterio puede traducirse en términos de un criterio "*F-to-stay*" o en un "nivel de significancia para permanecer" (SLS), el proceso *backward* concluye cuando todas las variables que permanecen en el modelo cumplen con el criterio para subsistir.

La regla de paro para la selección *stepwise* utiliza ambos criterios forward y *backward*. El proceso de selección de variables concluye cuando todas las variables en el modelo cumplen con el criterio para permanecer y variables fuera del modelo no cumplen el criterio para entrar.

5.2.2. Regresión Lasso

La regresión *Lasso* (*Least absolute shrinkage and selection operator*) es un método de reducción y selección de modelos para regresiones lineales. Se relaciona con los métodos *soft-thresholding* de coeficientes, regresión *stagewise* (*fw*) y de *boosting*. Minimiza la suma de los cuadrados sujeto a la restricción de que la suma del valor absoluto de los coeficientes sea menor que una constante. El objetivo de esta restricción es favorecer soluciones con

algunos coeficientes exactamente en cero. Entonces podríamos seleccionar el grupo de variables con coeficiente mayor a cero como el conjunto reducido. La idea de *lasso* es un tanto general y puede ser aplicada a una amplia gama de modelos estadísticos [15].

Breve explicación de la Regresión *Lasso*

Dado un conjunto de mediciones $x_1, x_2, \ldots, x_p$ y una variable de salida y, el método *lasso* ajusta un modelo línea:

$$\hat{y} = \hat{\beta}_0 + \hat{\beta}_1 x_1 + \hat{\beta}_2 x_2 + \cdots + \hat{\beta}_p x_p$$

El criterio que usa es:

$$min\left\{\sum_{i=1}^{N}\left(y_i - \sum_j \beta_j x_{ij}\right)^2\right\} \quad ; \quad \text{sujeto a } \sum_j |\beta_j| \leq s \tag{2}$$

La primera suma se realiza sobre todas las observaciones en el archivo de datos. El valor *s* parámetro que templa el ajuste. Cuando s es demasiado grande, la restricción no tiene efecto y la solución es la misma que se obtiene con la regresión lineal múltiple de cuadrados mínimos de *y* sobre $x_1, x_2, \ldots, x_p$

Sin embargo cuando valores más pequeños de *s (s>=0)* la solución es una versión recortada de las estimaciones de cuadrados mínimos. Con frecuencia algunos de los coeficientes β_j son cero. Elegir *s* es como elegir el número de predictores a utilizar en un modelo de regresión, y la validación cruzada es una buena herramienta para estimar el mejor valor para *s*.

Los cálculos para encontrar la solución *lasso* es un problema de programación cuadrática, y puede ser abordado por algoritmos estándares de análisis numéricos. Para facilitar los cálculos *lasso* también se pueden ver como una variante de *least angle regression* (*LARS*). Este algoritmo explota la estructura especial del problema *lasso* y provee un camino eficiente para calcular soluciones simultáneas para todos los valores de *s*. Hastie, Thibshirani y Friedman (pag. 74) [18] y [21]

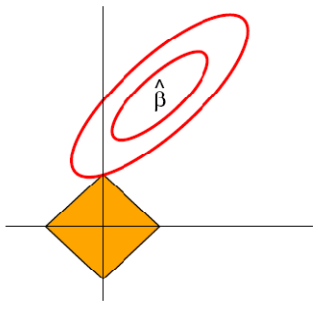


Figura 5.3 Gráfica de la estimación de los coeficientes de regresión lasso.

Los contornos elípticos de la función que estima los coeficientes de regresión, el centro corresponde a la estimación de mínimos cuadrados ordinarios y la región definida por la restricción es el rombo amarillo. La solución lasso es el primer lugar que el contorno elíptico toca el rombo, en ocasiones esto ocurre en una esquina lo cual significa que algún coeficiente será igual a cero.

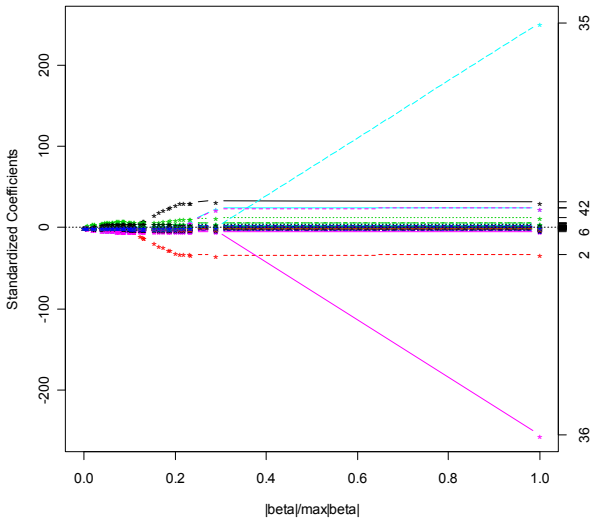


Figura 5.4 Coeficientes de regresión estandarizados para todos los valores de *s*.

En la gráfica anterior se presenta la magnitud que toman los coeficientes de regresión conforme aumenta el parámetro de restricción, hasta que finalmente concluye cuando $|beta|/max|beta| = 1$. Sobresalen dos variables la primera #35 se refiere al año en que comenzó el trabajo actual, y la #36 es el cuadrado de la anterior ya que durante la exploración observamos que el comportamiento observado asemeja a una función de segundo orden.

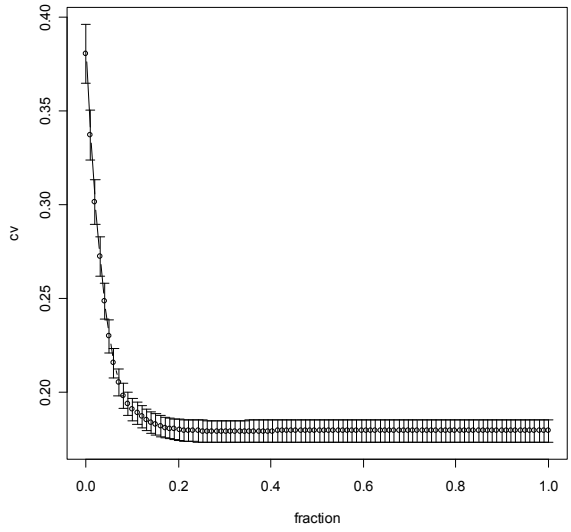


Figura 5.5 Validación cruzada de la regresión lasso.

El artículo de Tibshirani (1996) [15] menciona que podemos utilizar como herramienta la validación cruzada para fijar la fracción (relacionada con el parámetro s) de manera que determinemos las variables que conformarán el conjunto reducido. Entonces se puede notar que cuando la fracción va de 0.0 a 0.1 el error decrece rápidamente. En el intervalo 0.1 a 0.2 continua mejorando la precisión aunque es proporcionalmente es mucho menor. Cuando la fracción supera 0.2 el error prácticamente permanece constante. Para este ejercicio elegimos la fracción=0.1 para conformar el subconjunto de 29 variables.

Una vez establecida la fracción se estimaron la magnitud de los coeficientes y el orden en que entraron las variables hasta cumplir con la restricción (tabla 5.2). En las gráficas 5.6 se puede observar las 29 características seleccionadas con la regresión lasso y para tener una referencia adicional aplicaremos la técnica stepwise a los datos y compararemos los resultados.

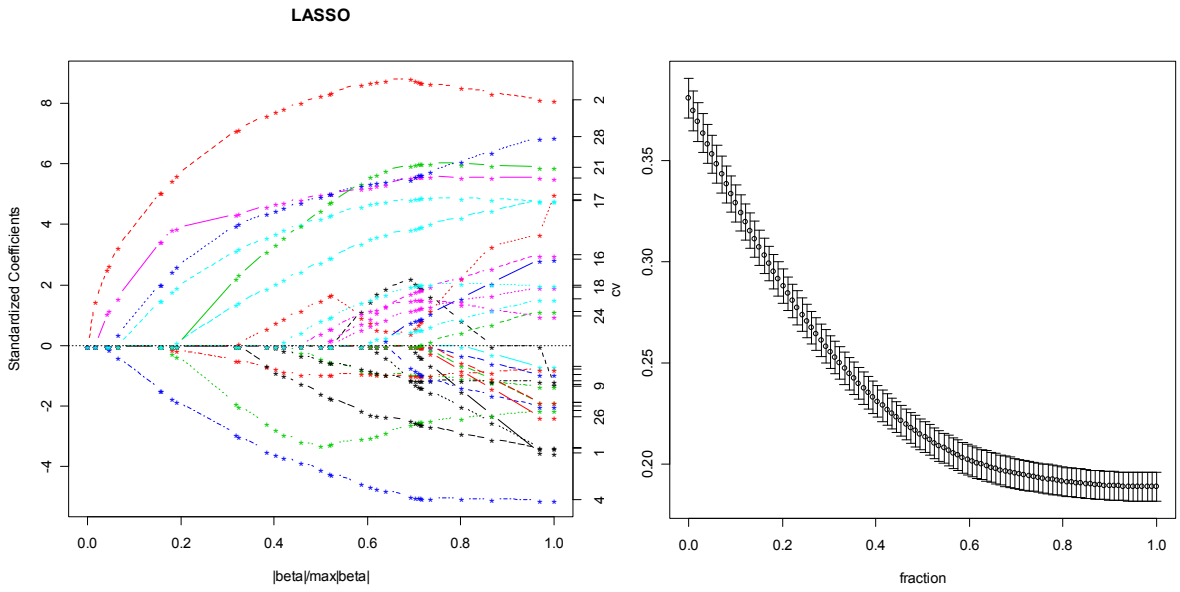


Figura 5.6ª **Figura 5.6b**

Gráfica de coeficientes estandarizados y el resultado de la validación cruzada para el subconjunto de 29 variables seleccionado.

No. col.	Nombre	LASSO		STEPWISE		RESULTADO		
		Coeficiente	Entrada	Estimate	t value	Lasso	SW	Lasso vs. SW
3	A_acum	3.01E-02	1	4.26E-02	8.7	✓	✓	✓
9	E_jor	1.12E-01	2	1.14E-01	6.1	✓	✓	✓
6	Sexo	-1.60E-01	3	-1.74E-01	-8.0	✓	✓	✓
43	Jefe	1.91E-01	4	1.21E-01	6.6	✓	✓	✓
23	cmo_cp3	2.21E-01	5	2.03E-01	8.2	✓	✓	✓
19	p4	-2.42E-02	6	0.00E+00		✓	✗	✗
5	S_soc	-7.19E-02	7	-3.71E-02	-1.6	✓	✓	✓
8	hrs_tra	4.70E-03	8	4.93E-03	4.8	✓	✓	✓
27	Patron	4.34E-01	9	4.12E-01	11.7	✓	✓	✓
11	Est2	5.93E-03	10	8.40E-03	7.5	✓	✓	✓
36	p3r_anio2	-1.55E-06	11	-1.70E-04	-2.3	✓	✓	✓
25	p11_h5	-1.50E-03	12	-3.85E-03	-4.4	✓	✓	✓
31	Trab_todo_anio	8.08E-02	13	4.78E-02	2.4	✓	✓	✓
12	p6b1	-2.34E-02	14	-2.97E-02	-2.9	✓	✓	✓
34	Tam_emp	6.70E-03	15	7.96E-03	2.4	✓	✓	✓
17	P_lab	7.16E-02	16	8.50E-02	3.7	✓	✓	✓
24	cmo_cp6	1.91E-01	17	2.06E-01	3.9	✓	✓	✓
10	Est1	1.37E-02	18	0.00E+00		✓	✗	✗
45	IAGobierno	6.11E-02	19	7.40E-02	2.3	✓	✓	✓
18	T_con	-1.96E-02	20	-2.81E-02	-4.9	✓	✓	✓
22	s_act7_cp3	9.65E-02	21	1.36E-01	5.2	✓	✓	✓
28	Sect_Inf	-5.19E-02	22	-9.82E-02	-3.7	✓	✓	✓
2	Edad2	-2.31E-05	23	-4.39E-04	-14.9	✓	✓	✓
38	Busca_otro	3.87E-02	24	1.10E-01	3.3	✓	✓	✓
37	p6_7	-3.13E-02	25	-5.97E-02	-3.2	✓	✓	✓
14	emp_a_pas	-1.45E-02	26	0.00E+00		✓	✗	✗
21	s_act7_cp2	-2.24E-02	27	0.00E+00		✓	✗	✗
26	Mic_neg_sl	-2.74E-02	28	-6.53E-02	-2.3	✓	✓	✓
15	neg_a_cp	-8.42E-03	29	-9.87E-02	-2.0	✓	✓	✓
44	Conyuge	0.00E+00	30	0.00E+00		✗	✗	✓
40	Primario	0.00E+00	31	5.44E-01	2.1	✗	✓	✗
1	Edad	0.00E+00	32	3.48E-02	13.7	✗	✓	✗
16	per_a_pas	0.00E+00	33	5.97E-02	1.5	✗	✓	✗
20	T_loc	0.00E+00	34	4.12E-02	1.8	✗	✓	✗
41	Secundario	0.00E+00	35	7.90E-01	3.4	✗	✓	✗
46	NoApoyoGob	0.00E+00	36	1.35E-01	1.5	✗	✓	✗
33	Sindicalizado	0.00E+00	38	0.00E+00		✗	✗	✓
39	Subocup	0.00E+00	39	0.00E+00		✗	✗	✓
29	Sin_Fin_Lucro	0.00E+00	40	0.00E+00		✗	✗	✓
4	N_ins	0.00E+00	41	-6.24E-02	-3.2	✗	✓	✗
13	aban_a_pas	0.00E+00	42	0.00E+00		✗	✗	✓
30	Sec_Privado	0.00E+00	43	0.00E+00		✗	✗	✓
7	Cp_anc	0.00E+00	44	0.00E+00		✗	✗	✓
42	Terciario	0.00E+00	46	7.45E-01	3.2	✗	✓	✗
32	p5_cp3	0.00E+00	47	0.00E+00		✗	✗	✓
35	p3r_anio	0.00E+00	50	6.68E-01	2.2	✗	✓	✗
					TOTAL	29	34	33

Cuadro 5.2

En el cuadro 5.2 se resumen los resultados de los métodos *Lasso* y *Stepwise*. De las 46 variables *Lasso*, selecciona 29 con la fracción colocada en 0.1 y *stepwise* por el criterio *AIC* incluye en el modelo 34 características. Observamos que son 29 las más importantes para *Lasso*, solo cuatro no lo son para *stepwise*, de manera global podemos decir que el resultado es semejante para ambos métodos.

5.3. Criterio de evaluación

Uno de los productos de la exploración (capítulo 4) fue la obtención de un conjunto acotado de datos que nos servirá para ajustar una serie de métodos de imputación. Recordemos que se tomaron los 4 279 registros pertenecientes a la Ciudad de León, Guanajuato; se seleccionaron 55 variables de las 300 disponibles originalmente. Ocho corresponden a la identificación geográfica y del marco muestral, 46 columnas con información propia del cuestionario y por último una con el logaritmo del ingreso.

Ahora bien un punto esencial en la comparación de los distintos métodos de predicción es la manera en que se instrumenta su evaluación. Idealmente nos gustaría ajustar un modelo con la población ocupada que proporcionó el monto de su ingreso y utilizarlo para recuperar el ingreso del conjunto de datos que no lo tiene.

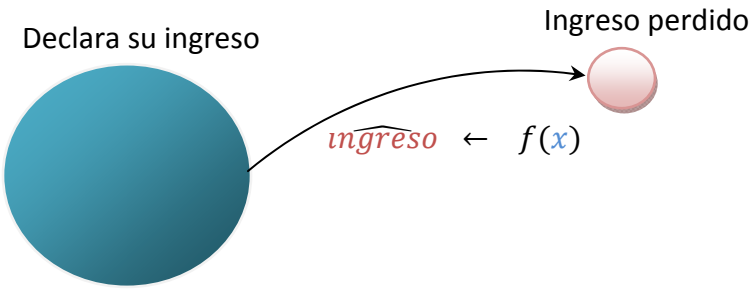


Figura 5.7

El problema consiste principalmente en la carencia de un indicador que para cada caso nos sirva de referencia y así determinar que tan alejada esta la predicción del valor real ($error = \hat{y}_k - y_k$). Entonces debemos seleccionar una herramienta que describa el ingreso de las personas que respondieron el ingreso y compararla con los resultados predichos.

El procedimiento que proponemos para evaluar los métodos es realizar *k-muestras aleatorias de validación* a partir de los casos que respondieron el ingreso, dividiendo el conjunto original en dos grupos: 1) de entrenamiento (*training*) y 2) de prueba (*test*). El método se ajusta con el conjunto de entrenamiento y se evalúa con el de prueba.

5.3.1 Selección del conjunto de prueba

Recordemos que una de las conclusiones del capítulo anterior fue que podíamos suponer que la no respuesta del ingreso era tipo MAR. Entonces se requiere reconocer el patrón en que los ingresos se pierden y para ello se efectuaron diversos ejercicios para identificar una serie de clases que discriminaran los niveles de respuesta. A continuación se presenta el cuadro generado a partir de los datos de la encuesta y que proponemos para seleccionar el conjunto de prueba:

Grupos de edad por nivel de instrucción		Estrato socioeconómico				
		Total	Bajo	Medio-bajo	Medio-alto	Alto
Total		0.05	0.03	0.04	0.05	0.09
Jovenes	Hasta secundaria	0.03	0.02	0.03	0.03	0.03
Jovenes	Medio superior y superior	0.06	0.02	0.05	0.06	0.08
Adultos	Hasta secundaria	0.05	0.04	0.05	0.05	0.08
Adultos	Medio superior y superior	0.08	0.05	0.06	0.07	0.11

Cuadro 5.3 Probabilidad de no respuesta muestral del ingreso laboral por estrato, edad y nivel de instrucción.

Las variables involucradas en la tabla son: edad, nivel de instrucción y estrato socioeconómico. La población joven y adulta se refiere a los trabajadores con 35 años o menos, y, 36 y más respectivamente. Los cortes de las variables se definieron con base en dos criterios, la división balanceada de la población y la diferencia en las tasas de respuesta.

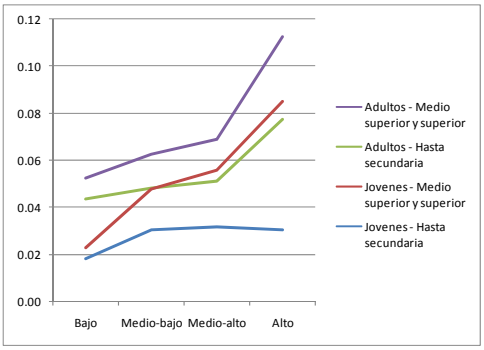


Figura 5.8 Probabilidad de no respuesta por estrato socioeconómico, edad y nivel de instrucción.

5.3.2 Realización de las muestras

A partir de la matriz de datos con los 4 279 casos de trabajadores que declararon el ingreso, se seleccionaron 100 muestras, las cuales se utilizarán para ajustar y evaluar los métodos de imputación. Para tal efecto se desarrollaron las funciones necesarias para recuperar el patrón de no respuesta existente en la ciudad de León, Gto. (tasa de no respuesta global 6%)

El esquema de muestreo elegido para replicar el tipo de no respuesta MAR fue estratificado monoetápico, dentro de cada estrato se utilizó el muestreo Bernoulli para seleccionar a los casos que entrarían en grupo de prueba. Los casos se agruparon en 16 estratos o clases (ver tabla 4.11), entonces sean $I_{11},\ldots, I_{EN}$ las variables indicadoras (VI), donde E corresponde al estrato y N el número de casos dentro de cada estrato, de tal forma que las indicadoras son iid en cada clase. Las VI señalan si el elemento ek pertenece a la muestra s y se define de la siguiente forma:

$$I_{ek} = \begin{cases} 1\ si\ \ ek\ \in\ \ s \\ 0\ si\ \ ek\ \notin\ \ s \end{cases}$$

Si π_e es una constante que satisface $0 < \pi_e < 1$ cada I_{ek} sigue la misma distribución Bernoulli [16].

$$\Pr(I_{ek} = 1) = \pi_e \quad ; \quad \Pr(I_{ek} = 0) = 1 - \pi_e$$

En particular para la ciudad de León, Guanajuato los valores de π_e para cada estrato se especifican en la siguiente tabla:

Grupos de edad por nivel de instrucción		Estrato socioeconómico				
		Total	Bajo	Medio-bajo	Medio-alto	Alto
Total		0.06	0.00	0.04	0.06	0.19
Jovenes	Hasta secundaria	0.03	0.00	0.04	0.02	0.05
Jovenes	Medio superior y superior	0.09	0.00	0.07	0.07	0.18
Adultos	Hasta secundaria	0.05	0.00	0.03	0.06	0.22
Adultos	Medio superior y superior	0.15	0.00	0.09	0.11	0.22

El procedimiento de selección se implemento de la siguiente forma:

```
muestra_MAR <- function (patron,Est1,Edad,N_ins) {  #
Selecciona una muestra siguiendo un patrón MAR
   num_reg<- length (Est1)
   #direcc es una variable de direccionamiento entre los
registros y la tabla de probabilidades
   #Est1 : 1= Bajo; 2 = Medio-bajo; 3=Medio-alto; 4=Alto
   #Edad : 1= Jovenes; 2 =Adultos
   #N_ins: 1= Hasta secundaria; 2 = Medio superior y superior
   direcc <- ((Est1-1)*4) + ((ifelse(Edad <= 35, 1,  2)-1)*2)
+ ifelse(N_ins <= 3, 1,  2)

   prXreg <- patron[direcc]    # Toma la dirección calculada
a cada registro y trae su probabilidad de no respuesta
   aleatorios <- runif (num_reg)   # Genera N números
aleatorios con ~ uniforme 0,1.
   # Compara el número aleatorio contra la probabilidad de
que dicho registro no responda sus ingresos
   # Si el número aleatorio es inferior o igual a la
probabilidad de no respuesta se selecciona en el grupo de
prueba
   # Sí seleccion == 1 entonces es elemento seleccionado para
TEST
   #    seleccion == 0 entonces es elemento seleccionado para
TRAIN
   seleccion <- aleatorios <= prXreg
   cuantos <- sum (seleccion)
   total   <- sum (FAC)
   tnr     <- cuantos / num_reg
   tnr_e   <- sum(seleccion*FAC)/sum(FAC)
return (list(sel=seleccion,n=cuantos,pnr=tnr,pnr_e=tnr_e))}
```

Una de las características del muestreo Bernoulli es que el tamaño de muestra es aleatorio y en la siguiente gráfica mostramos para cada clase la proporción de elementos seleccionados para servir como grupo de prueba en las 100 replicas realizadas. En promedio el tamaño de muestra (conjunto de prueba) fue 251.9 registros y en término porcentual se ubicó en 5.8 con respecto al total de registros.

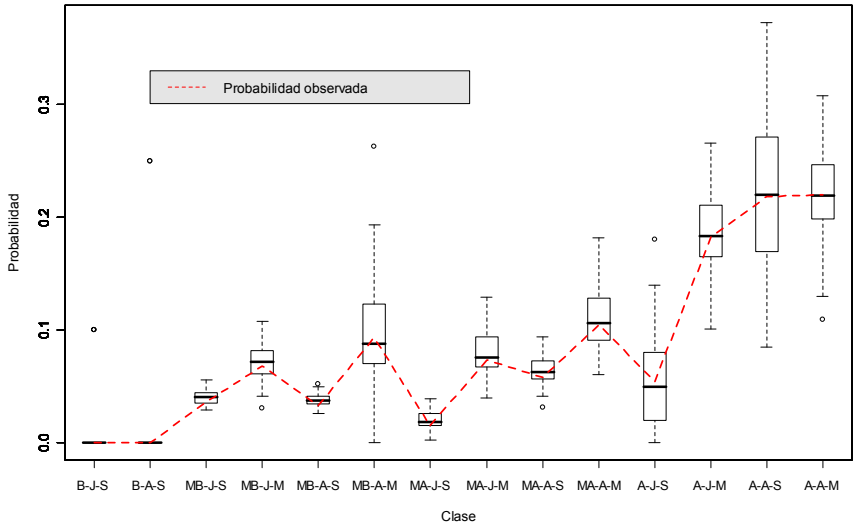


Figura 5.9 Diagrama de caja de la probabilidad de no respuesta por clase.

En la figura 5.9 podemos ver la tasa de no respuesta simulada para cada clase (conjunto de prueba / número de casos totales) para la ciudad de León Gto. Las etiquetas en el eje de las abscisas (x) se compone del formato EE-D-N, donde: EE: Estrato socioeconómico y sus entradas son: B = Bajo, MB= Medio Bajo, MA= Medio alto y A = Alto. D: Grupos de edad, sus entradas: J = Joven y A = Adulto; finalmente N: Nivel de instrucción, sus entradas: S = Hasta secundaria y M = Medio superior y superior.

Con línea punteada (roja) se presenta la probabilidad de no respuesta observada para cada una de las clases y los diagramas de caja resumen el comportamiento de las 100 muestras realizadas.

5.3.3 Indicador para la evaluación

Se propone que la evaluación se realice estimando el error cuadrático medio (ECM) del conjunto de prueba:

$$e_k = w_k\,(y_k - \hat{y}_k)$$

$$ECM(\hat{y}) = \frac{1}{N}\sum_{k=1}^{n} e_k^2 \qquad (1)$$

Donde:

e_k: Error ponderado del k-ésimo caso
w_k: Factor de expansión
$\hat{y}_k$: Estimación del ingreso

y_k: Valor real
$N = \sum_{k=1}^{n} w_k$: Población total estimada

5.4. Experimentos

Se seleccionaron siete métodos para simular la imputación del ingreso y se dispusieron cuatro versiones del conjunto de datos, el primero corresponde a la tabla completa, el segundo a la selección efectuada con regresión *lasso* (29 variables), el tercero y cuarto son la versión estandarizada de los dos primeros. La relación completa se presenta en el siguiente cuadro:

Método	*Conjunto de datos*			
	Escala original		Datos estandarizados	
	Completo (OE)	Reducido (OR)	Completo (EC)	Reducido (ER)
A. Media general	✓			
B. Media por clases			✓	✓
C. K-Vecinos más cercanos	✓	✓	✓[1]	✓[1]
D. Hot-deck	✓	✓	✓	✓
E. Regresión lineal	✓	✓		
F. Redes Neuronales (NN)	✓	✓	✓	✓
G. Regresión-knn			✓	✓

Cuadro 5.4 Métodos empleados para la simulación de la imputación de los ingresos.
[1] Se experimento con dos funciones de agregación media general y ponderada por la distancia.

5.4.1. Media general

La media general sólo depende de la variable ingreso en los casos en que se obtuvo respuesta. Lo que equivale a que de la ecuación (3) fijemos en cero $\hat{e}_{mi}$ y β_{rj} y la media muestral se asigne a , β_{r0} entonces el modelo se reduce a $\hat{y}_{mi} = \bar{y}_r$

Para evaluar las predicciones individuales se propuso calcular el error cuadrático medio (ECM) definido en la ecuación (1). Se aplicó el procedimiento sobre las 100 muestras seleccionadas y se obtuvo un ECM promedio de 59.7 y su desviación estándar (DE) se coloco en 7.6. Es necesario comentar que se efectuó la simulación considerando la media muestral y expandida y los resultados obtenidos fueron esencialmente los mismos.

5.4.2. Media por clase

Esta técnica a diferencia de la anterior utiliza información captada en otros reactivos de la encuesta. Las nueva definición para el modelo (3) es construir las $z_j's$ tal que correspondan a las variables dummys que identifican a las clases y nuevamente fijando $\hat{e}_{mi} = 0$ entonces la ecuación se simplifica $\hat{y}_{mi} = \bar{y}_{rh}$ en donde *h* corresponde a la clase.

El elemento fundamental consiste en establecer las clases en que se agrupa a la población, las cuales pueden definirse por un criterio conceptual que incorpore información a priori y posteriori de los datos que genera la encuesta o fuentes de datos alternas. Ya que para la ENOE se carece de información que ayude a construir las clases se ha optado por generarlas automáticamente apoyándonos para ello en el método de agrupamiento llamado k-medias.

Agrupamiento K-Medias.

Es un algoritmo muy popular propuesto por MacQueen (1967) es muy eficiente y frecuentemente usado en proyectos de agrupación de gran escala [17].

El algoritmo comienza de dos posibles maneras, la primera asignando cada caso a uno de los K grupos predeterminados y entonces se calculan los K centroides o la segunda es pre-especificando los centroides. Entonces de manera iterativa el algoritmo busca minimizar la suma de cuadrados del error (SCE) a través de la reasignación de los casos en los grupos. El criterio de paro es cuando ya no se reduce el valor de SCE. En la siguiente tabla se encuentra el algoritmo detallado.

1. Define las entradas: $\mathcal{L} = \{x_i, i = 1,2, \dots, n\}$, K = número de grupos

2. Realiza uno de los siguientes arranques:
   a. A partir de una asignación aleatoria inicial de los casos hacia los K grupos, calcula los centroides, $\bar{x}_k, k = 1,2, \dots, K$
   b. Pre-especifica los centroides iniciales, $\bar{x}_k, k = 1,2, \dots, K$

3. Calcula el cuadrado de la distancia euclidiana de cada elemento con su centroide:
$$SCE = \sum_{k=1}^{K} \sum_{c(i)=k} (x_i - \bar{x}_k)^t (x_i - \bar{x}_k)$$
   Donde $\bar{x}_k$ es el k-ésimo centroide y $c(i)$ es el grupo que contiene a x_i

4. Reasigna cada elemento a su centroide de grupo más cercano, eso hará que SCE reduzca su magnitud. Actualiza los centroides de grupo después de cada reasignación.

5. Repite los pasos 3 y 4 hasta que no más reasignaciones tengan lugar.

Cuadro5.5 Algoritmo de agrupamiento K-medias

Un parámetro que es desconocido en esta técnica es el número de grupos que realmente existen en la población y por ende no se puede fijar k en un inició, es por ello que se realizaron ensayos con un conjunto de valores para seleccionar el parámetro k.

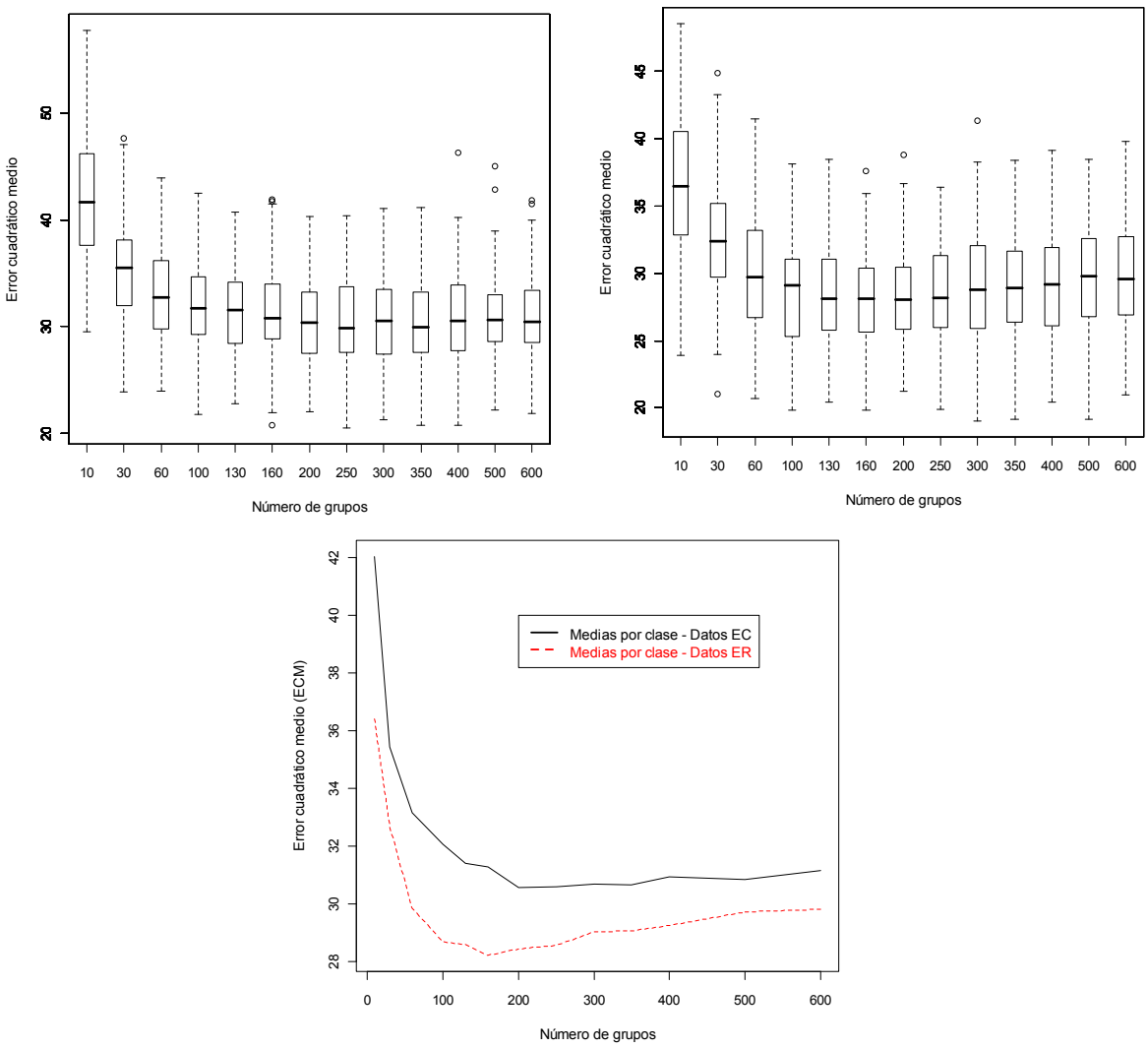


Figura 5.10 Error Cuadrático Medio generado por imputación de la media por clases, arriba a la izquierda para el conjunto de datos estandarizado en su versión completa (EC) se muestra el diagrama de caja de los ECM para las cien muestras seleccionadas y los valores de prueba para k (Número de grupos), a la derecha los resultados para el conjunto ER. Abajo se puede observar el promedio de los ECM de los dos experimentos.

Con base en las gráficas anteriores optamos por seleccionar el modelo con 160 clases (k) ya que tiene en promedio el menor ECM con 28.22 y una DE igual a 3.8 lo que representa una mejora substancial si lo comparamos con la media general.

5.4.3. K-Vecinos más cercanos

Este método es comúnmente identificado por sus siglas en inglés *knn (k-nearest-neighbor)* el cual localiza en función de la distancia a k vecinos de una observación en particular. Entran aquí en juego el conjunto de variables auxiliares seleccionadas, el valor que toma k y la función de agregación que resume la característica que donara los vecinos del caso con

el ingreso perdido. Las funciones de agregación dependen del tipo de variable, si es numérica la más común es el promedio, si es categórica la disponga de la mayor frecuencia.

Para este experimento utilizaremos como técnica para resumir la información de los k-vecinos el promedio simple y ponderado en función de la distancia. (se modificó la función de predicción del paquete *knnflex* para dar soporte al promedio ponderado ver anexo 5.1 `knn.predict.weighted`). Con respecto a la distancia lo más normal es que se utilice la distancia euclidiana $d_i = \|x_i - x_0\|$ una vez estandarizados los datos (media=0 y varianza=1) [18].

Si el método de agregación es el promedio simple, entonces llegamos a la misma solución que medias por clase en donde h corresponde al grupo de vecinos más cercanos, pero si el promedio es ponderado, entonces se agrega un residual $\hat{e}_{mih}$ que dependerá de las distancias de los vecinos con el registro que se desea imputar.

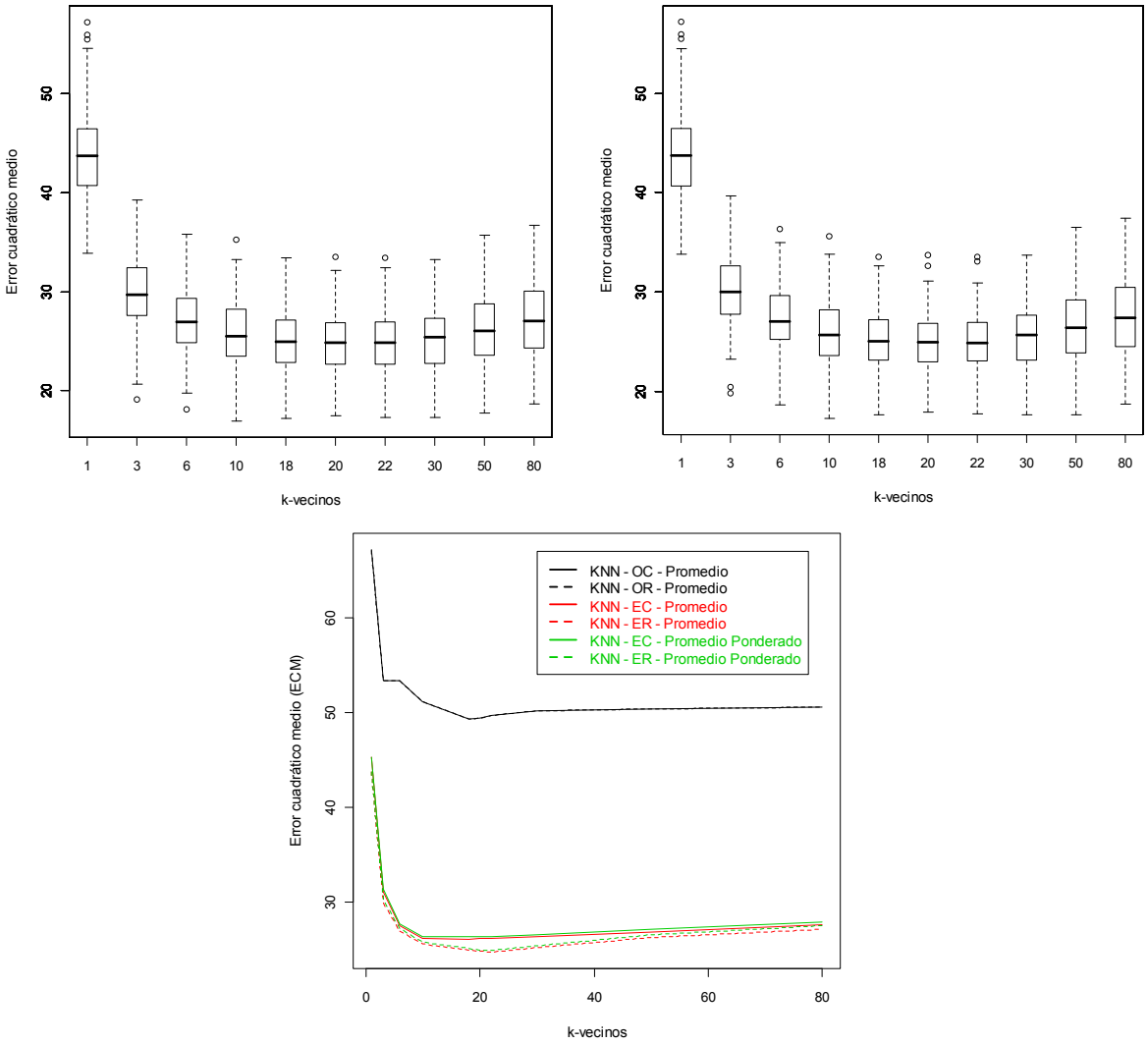


Figura 5.11 Error Cuadrático Medio del método *Knn* para imputar el ingreso. Arriba se encuentra la salida de las dos corridas que utilizaron el conjunto de datos ER. A la izquierda con la función de agregación promedio simple y a la derecha empleando promedio ponderado (distancia). Abajo se observa el ECM promedio de las seis corridas.

En la figura anterior observamos en la parte superior las dos mejores corridas, ambas con el conjunto ER, se puede apreciar que no existe diferencia importante en la tendencia ni en la variabilidad por lo que elegimos como representante el método más simple que utiliza el promedio general con k=20. Para estos parámetros se observo un ECM promedio de 24.8 con desviación estándar igual a 3.0.

5.4.4. *Hot-deck*

Como se describe previamente, existe una amplia gama de métodos también llamados de donante que difieren principalmente en la manera que selecciona al registro donador. En este experimento retomaremos la implementación *hot-deck* que Stefano M. Iacus (2006) propone, la cual emplea una matriz de proximidad construida con la técnica *Random Recursive Partitioning (RRP)* [3].

El algoritmo RRP y la matriz de proximidad

El algoritmo RRP es un procedimiento Monte Carlo sobre el conjunto de particiones recursivas de los datos el cual genera una medida de proximidad/disimilaridad y hace uso del método de árboles de regresión (RT). Cada observación i es asignada una variable respuesta ficticia $U_i \sim U(0,1)$. El árbol de regresión que modela U en función de las covariables $(X_1, X_2, \dots, X_p)$ da como resultado una partición de los datos aleatoria y recursiva. La medida de proximidad π desde esta partición aleatoria se obtiene de la siguiente manera: $\pi_{ij} = 1$ para todas las observaciones i y j en la misma hoja; $\pi_{ij} = 0$ de cualquier otra forma. Esta partición y medida de proximidad depende por completo de la realización del vector aleatorio $(U_1, \dots, U_n)$: Por lo tanto el procedimiento replicado R veces y la medida final de proximidad es determinada por el promedio de $\pi_{ij}{}'s$ obtenido en todas las replicas.

Denotaremos $x_i = (x_{i1}, \dots, x_{ip})$ al vector de covariables $(X_1, X_2, \dots, X_p)$ para las observaciones $i = 1, \dots, n$, donde n es el tamaño de muestra. Entonces el algoritmo RRP es el siguiente:

```
Mientras ( r <=R)
```
Genera n números aleatorios $u_1, \dots, u_n \sim U(0,1)$

Ajusta un árbol de regresión $u_i \sim (x_{i1}, \dots, x_{ip})$

Asigna a $\pi_{ij}^{(r)} = 1$ si la observación *i* y *j* en el mismo nodo terminal; $\pi_{ij} = 0$ de cualquier otra forma

```
Fin mientras
```
Asigna Π y Δ de la siguiente manera:

$$\Pi = \left[\pi_{ij} = \frac{1}{R}\sum_{r=1}^{R} \pi_{ij}^{(r)}\right] \quad y \quad \Delta = 1 - \Pi$$

Llamaremos a Π matriz de proximidad-RRP y a Δ la matriz de disimilaridad-RRP

El algoritmo para la imputación

La propuesta realizada por Iacus (2006) es un método *hot-deck* en función de distancia (4.5) que utiliza la técnica knn para explotar la matriz de disimilaridad-RRP (Δ). Una de las aplicaciones hot-deck más conocidas es la imputación de los ingresos de la población realizada en el CPS. En dicha implementación los casos con variables perdidas son comparadas con casos que son similares en todas las demás covariables que si tuvieron respuesta y entonces los datos perdidos son imputados con el valor correspondiente al registro con todas las variables contestadas. Si no se encuentra un cuestionario similar, la estrategia de recuperación reduce el conjunto de covariables y reinicia la búsqueda (6.4 hot-deck jerárquico)

RRP resuelve en parte los problemas a los que se enfrenta el caso CPS al permitir que observaciones con datos faltantes puedan ser utilizados para generar una medida de proximidad, situación que no es posible resolver con distancia Euclidiana o de Mahalanobis.

El algoritmo de imputación es el siguiente:

1. Ejecuta el algoritmo RRP sobre todo el conjunto de datos incluyendo los casos con variables pérdidas.

2. Para cada observación *i* con variables pérdidas, identifica los *k* vecinos más cercanos en términos de la matriz de disimilaridad-RRP δ_{ij}.

3. Para cada covariable pérdida x_{im} del registro *i*, $m = 1, \dots, p$:
   a. Si X_m es continua, asigna a x_{im} el promedio del valor de la variable X_m de los k vecinos más cercanos
   b. Si X_m es categórica, asigna a x_{im} la moda de todos los valores de la variable X_m de los k vecinos más cercanos

Cuadro 5.6 Algoritmo de imputación hot-deck RRP

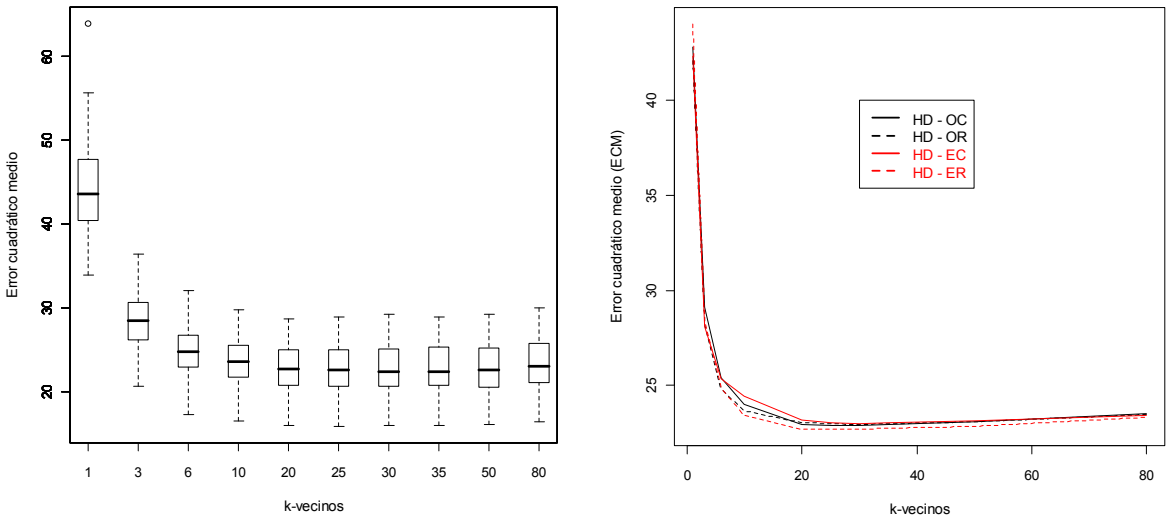


Figura 5.12 Error Cuadrático Medio del método Hot-deck para imputar el ingreso, a la izquierda se observan los diagramas de caja para la corrida con el conjunto ER y a la derecha se presenta el ECM promedio de las cuatro corridas Hot-deck.

Se puede notar que los resultados obtenidos por las cuatro configuraciones son muy similares tanto en el comportamiento medio de sus ECM como su variabilidad (en el script `Ajuste-Hot-deck.r` se calcula la variabilidad), la configuración seleccionada de este método es k=20, conjunto de datos ER, su ECM promedio es 22.7 con desviación estándar de 2.8.

5.4.5. Regresión lineal

Es un método ampliamente utilizado para la imputación de variables numéricas. La ecuación general (3) se reduce para esta técnica fijando en 0 los $\hat{e}_{mi}$.

$$\hat{y}_{mi} = \hat{\beta}_{r0} + \sum_j \hat{\beta}_{rj} z_{mij}$$

Denotaremos a $x_j = (x_{j1}, \ldots, x_{jp})$ como vector de las p variables independientes para el j-ésimo caso y a $X_r^t = (x_1^t, x_2^t, \ldots, x_n^t)$ como la matrix de covariables para los registros que respondieron el ingreso, de manera análoga X_{mi} corresponde a la matriz de variables independientes de los casos que carecen de la variable en cuestión, $W = D(w_1, w_2, \ldots, w_n)$ es la matriz diagonal de los ponderadores de la regresión de los registros con respuesta, teóricamente[6] $w_j = 1/\pi_j$, donde π_j es la probabilidad de inclusión de primer orden de la j-ésima vivienda seleccionada (ver estimador Horvitz-Thompson, Särdal [16]), y es el vector

[6] Ya que una vez concluido el levantamiento los factores se ajustan con base en la no respuesta total y en las proyecciones de población.

de respuestas efectivamente recuperadas, finalmente el vector $\hat{\beta}_r = (\hat{\beta}_{r0}, \hat{\beta}_{rj})$, entonces estimamos los coeficientes de regresión con la siguiente ecuación matricial.

$$\hat{\beta}_r = (X_r^t W X_r)^{-1} X_r^t W y$$

En consecuencia la expresión para realizar la estimación de los ingresos no reportados es:

$$\hat{y}_{mi} = X_{mi}\hat{\beta}_r$$

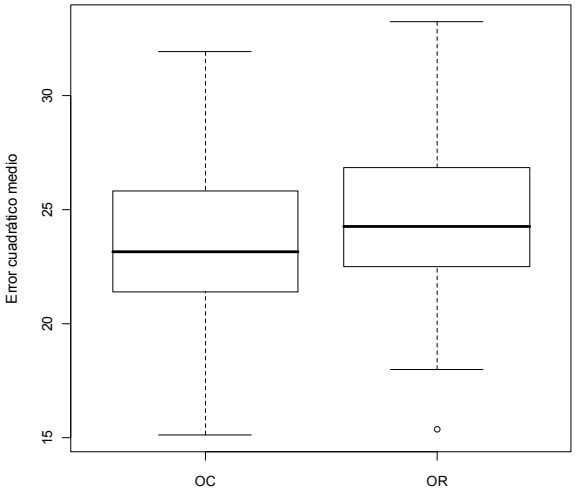


Figura 5.13 Error Cuadrático Medio de la simulación realizada con el método de regresión lineal para los conjuntos de datos OC y OR.

Observamos que los resultados para las dos corridas efectuadas sobre los conjuntos completo y reducido (OC y OR) son parecidos aunque ligeramente mejores para los datos completos. La media de los ECM es el conjunto de de las dos configuraciones 23.5 y su desviación estándar es 3.1.

5.4.6. Redes Neuronales

Es una técnica de aprendizaje propiamente llamada red de neuronas artificiales (ANN's por su siglas en inglés). El impulso que ha tenido esta técnica ha sido intermitente en el tiempo e influenciado por la investigación de sistemas inteligentes y expertos, los cuales buscan entender el funcionamiento del cerebro humano y la naturaleza de su inteligencia. Una de las teorías que busca dar respuesta a estas interrogantes es la llamada *connectionism* y utiliza analogías de neuronas y sus conexiones para construir redes neuronales [17].

Perceptrón.

El perceptrón (*single-layer perceptron*) fue propuesto por Frank Rosenblatt (1958, 1962) adecuando las teorías que Donal O. Hebb publicó en su libro *The Organization of Behavior* en 1949. El perceptrón es esencialmente la neurona McCulloch-Pitts (1943), sólo que las variables de entrada X_i, están asociadas a un ponderador $\beta_i,\ i = 1,2,\dots,r$ que las conecta con el núcleo. Las covariables $X_1, X_2, \dots, X_r$ pueden ser reales o binarias. Los ponderadores positivos ($\beta_i > 0$) reflejan sinapsis excitatorias, negativas indican sinapsis inhibitorias y la magnitud es una medida de la fuerza de la conexión.

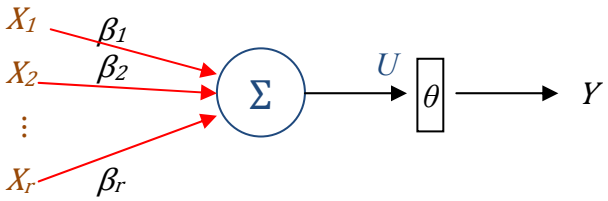


Figura 5.14 Perceptrón Rosenblatt con *r* entradas, ponderadores de conexión $\{\beta_i\}$, umbral de salida θ y salida binaria *Y*.

El funcionamiento del perceptrón que se presenta en la figura 5.14 es el siguiente: De izquierda a derecha podemos observar que se realiza la suma ponderada de las *r* entradas, si la suma supera el valor θ entonces $Y = 1$, en cualquier otro caso $Y = 0$. Gráficamente representa un hiperplano separador que divide el espacio *r*-dimensional en dos, R_1 y R_0, donde R_1 está dado por los puntos en que Y=1 y R_0 para aquellos en donde Y=0. Podemos expresar la combinación lineal *U* con la siguiente ecuación:

$$U = \beta_0 + \sum_{j=1}^{r} \beta_j X_j = \beta_0 + X^t \beta$$

Donde $X = (X_1, \dots, X_r)^t, \beta = (\beta_1, \dots, \beta_r)^t$ y β_0 corresponde a una constante relacionada con el umbral de salida θ, sin por ejemplo fijamos $\beta_0 = -\theta$ tendríamos la posibilidad de comparar la combinación lineal *U* contra el valor cero para establecer los valores de salida. Si $U \geq 0$ entonces Y=1; Y=0 en otro caso. Entonces a la asignación que se realiza a la variable salida *Y* dependiendo del valor que toma *U* se le denomina función de activación entonces $Y = f(U)$.

Función de activación	$f(u)$	Rango de valores
Identidad, lineal	u	$\Re$
Signo	$signo(u)$	$\{-1, +1\}$
Umbral	$I_{[u \geq 0]}$	$\{0, 1\}$
Gaussiana base radial	$(2\pi)^{-1/2} e^{-u^2/2}$	$\Re$
Gaussiana acumulativa (sigmoid)	$\sqrt{2\pi} \int_0^u e^{-z^2/2ti\theta}\, dz$	$(0, 1)$
Logística (sigmoid)	$(1 + e^{-u})^{-1}$	$(0, 1)$
Tangente hiperbólica (sigmoid)	$(e^u - e^{-u})/(e^u + e^{-u})$	$(-1, +1)$

Cuadro 5.7 Funciones de activación del perceptrón.

Perceptrón multicapa

A mediados de los 80's comenzó una generación de investigadores con renovados impulsos en el desarrollo de las NN junto con la disponibilidad de equipos de cómputo con mayores capacidades, permitieron retomar las inquietudes expresadas por Minsky y Papert (1969). El descubrimiento del algoritmo *backpropagation* (ver [17] Izenman, pág. 336) otorgó la capacidad de ajustar redes más complejas que son aptas para reconocer patrones en altas dimensiones.

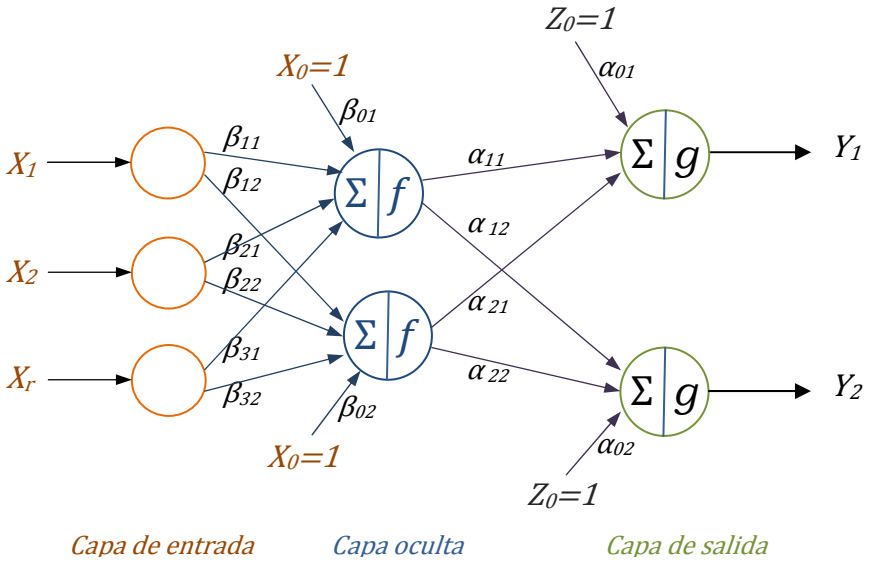


Figura 5.15.Perceptron multicapa con una capa oculta, *r=3* nodos de entrada, *s=2* nodos salida y *t=2* neuronas en la capa oculta. Las α's y β's son los ponderadores vinculados a las conexiones entre los nodos, f y g son funciones de activación.

La red neuronal multicapa *feedforward* como la que se presenta en el diagrama anterior es una técnica que direcciona no linealmente un vector de entrada $X = (X_1, \ldots, X_r)^t$ a un espacio de salida definido por $Y = (Y_1, \ldots, Y_s)^t$. Las dos capas que realizan los cálculos son la oculta y de salida, observe que los nodos entre capas sucesivas están completamente interconectados.

Las NN son empleadas para resolver problemas de regresión y clasificación. La imputación del ingreso necesita que la red neuronal se configure en modo regresión, entonces se debe disponer de un nodo de salida (*s=1*) y como función de activación se utilizará la identidad.

Aplicación de redes neuronales

El ejercicio se programó utilizando el paquete `nnet` disponible para R. Al igual que lo hecho en las otras técnicas se emplearon los cuatro conjuntos de datos (OC, OR, EC y ER). Los valores asignados a *t* iban de 1 a 10 y para cada valor de *t* se ajustó y evaluó una red para cada una de las 100 muestras disponibles. Cabe señalar que este fue uno de los métodos que se requirió más tiempo para completarse, 48 horas de ejecución continua en una PC de características recientes. El total NN ajustadas en este ensayo fue de 40 mil = 4 conjuntos de datos X 10 escenarios (diferentes valores de *t*) X 100 muestras.

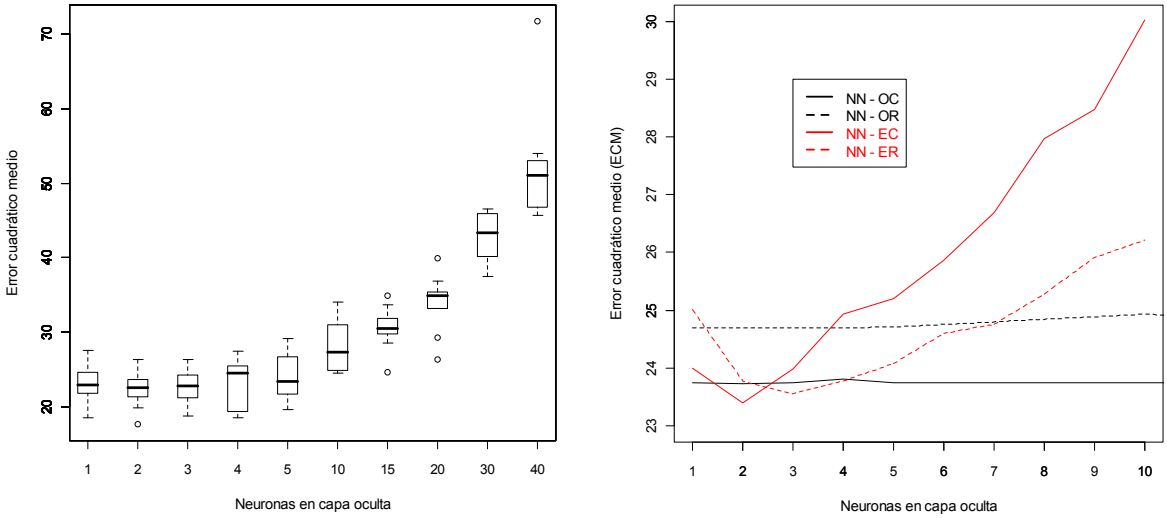


Figura 5.16 A la izquierda el diagrama de caja de los ECM para *t = (1, 2, 3, 4, 5, 10, 15, 20, 30, 40)* del conjunto EC, en la derecha el promedio de los ECM para *t = 1,...,10*.

Se puede notar que el resultado para los conjuntos en escala original OC y OR (líneas en negro) es prácticamente lineal y coincide con lo obtenido en el experimento de regresión lineal, cabe aclarar que en estos dos ejercicios (datos en la escala original) se empleo un grupo adicional de ponderadores que permiten conectar la capa de entrada directamente con la de salida.

Tibshirani [18] recomienda utilizar los datos estandarizados cuando se ajustan NN, lo cual coincide con el optimo encontrado en este ejercicio. La configuración seleccionada fue *t=2* para el conjunto EC ya que obtuvo el menor ECM promedio con 23.39 y su desviación estándar fue 3.1.

5.4.7. Regresión - knn

Los métodos de imputación pueden ser clasificados como determinísticos o estocásticos dependiendo de la manera en que se asignan los residuales $\hat{e}_{mi}$, si se fijan en cero o no. Si partimos de (3) podemos decir que puede tener dos componentes, los dos primeros términos y el último estocástico.

Elegir entre los dos tipos de imputación depende del análisis posterior que se pretenda realizar con la variable en cuestión. Por ejemplo si se busca reportar el promedio, entonces un método estocástico causara pérdida en la precisión de la media muestral [1], aunque está pérdida puede ser controlada a través de algún método especial de muestreo de residuales (Kalton y Kish 1984), aún así la pérdida de precisión ocurre. Por esta razón el esquema determinístico es preferible para la estimación de la media poblacional.
La propuesta que se hace es asignar los residuales a partir de k-vecinos más cercanos.

$$\hat{y}_{mi} = \hat{y}_{rmi} + \hat{e}_{mi}$$
$$\hat{y}_{rmi} = \beta_{r0} + \sum_j \beta_{rj} x_{mij}$$
$$\hat{e}_{mi} = \frac{\lambda}{k} \sum_{l \in v_k} (y_{rl} - \hat{y}_{rl})$$

Donde: λ: Factor de asimilación $\{0, 1\}$
k: Número de vecinos más cercanos
v_k: Conjunto de k vecinos más cercanos calculados con la distancia de Mahalanobis
y_{rl}: Ingreso reportado del l-ésimo caso
$\hat{y}_{rl} = \beta_{r0} + \sum_j \beta_{rj} x_{rlj}$: Ingreso estimado por la ecuación de regresión para los que respondieron el ingreso.

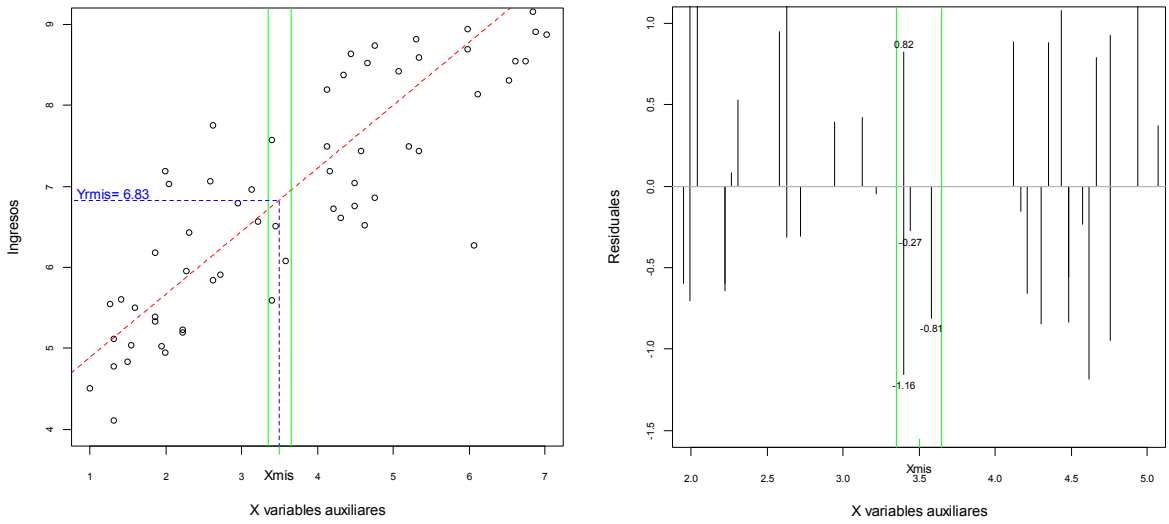


Figura 5.17 A la izquierda se observa el componente determinístico: la regresión lineal a partir de un grupo de datos simulados *X,* y respuestas de ingreso ficticias. Los coeficientes estimados por mínimos cuadrados ponderados aparece con línea punteada en color rojo, el caso con el ingreso no especificado *Xmis* genera una estimación $\hat{y}_{rmi} = 6.83$. La banda

vertical que rodea *Xmis* identifica la zona en donde se encuentran los $k = 4$ vecinos más próximos. En el recuadro de la derecha se muestra el elemento estocástico: Residuales contra el espacio *X*, en el centro del gráfico y limitados por las dos líneas verticales, en verde se ubican los 4 residuales que se consideraran para proponer un residual para $\hat{y}_{mi}$.

Para ejemplificar el procedimiento consideremos los datos ficticios desplegados en la figura 5.17, fijemos $k = 4$ y $\lambda = 1/2$ y supongamos que el espacio de las X variables auxiliares se logró colapsar en una dimensión, la predicción para un cuestionario con el ingreso perdido *Xmis* lo denotamos por $\hat{y}_{mi} = \hat{y}_{rmi} + \hat{e}_{mi}$. Podemos ver que $\hat{y}_{rmi} = 6.83$ y calcular $\hat{e}_{mi} = ((1/2)/4)(0.82 - 1.16 - 0.27 - 0.81) = -0.18$, entonces $\hat{y}_{mi} = 6.83 - 0.18 = 6.65$.

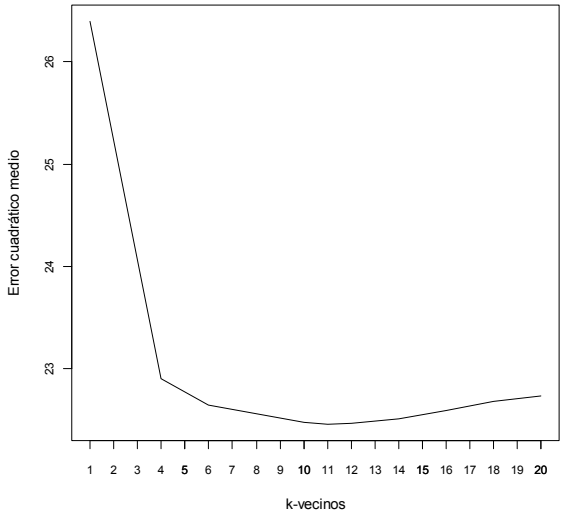


Figura 5.18 Promedio del Error Cuadrático Medio (ECM) que resulta del método Regresión-knn sobre las 100 muestras para un conjunto de valores *k* empleados en la técnica knn.

En el gráfico anterior se observa el ECM promedio, con menor magnitud se encuentra cuando k=11 alcanzando 22.5 con una desviación estándar de 3.0.

6. Resultados

En el capítulo anterior se ajustó una serie de técnicas para recuperar el ingreso que no fue declarado durante la recolección de datos de la ciudad de León, Guanajuato. Observamos de manera particular los resultados que arrojaba cada técnica para una cierta gama de configuraciones. El siguiente paso es comparar las mejores configuraciones de cada técnica. En esta confronta final incorporaremos al ECM las estimaciones del Ingreso promedio y del Coeficiente de Gini sobre los conjuntos de datos imputados para observar los efectos y considerarlos en la evaluación.

En las secciones 6.4 y 6.5 esbozaremos el efecto que la imputación tendría sobre la no respuesta real en la variable de interés empleando la mejor técnica detectada en sección 6.3.

6.1. Ingreso promedio

El ingreso laboral se emplea para generar varios de los indicadores que se presentan de la ENOE, no obstante uno de los más importantes es el ingreso promedio, la expresión para calcularlo es la siguiente:

$$\hat{\bar{y}} = \frac{1}{\widehat{N}} \sum_{e \epsilon E} \sum_{u \epsilon U} \sum_{v \epsilon V} F_{euv} \sum_{i \epsilon v} y_{euvi}$$

Donde: $\hat{\bar{y}}$ es el ingreso promedio estimado, $\widehat{N} = \sum_{e \epsilon E} \sum_{u \epsilon U} \sum_{v \epsilon V} \sum_{i \epsilon v} F_{euv}$ es el total de la población ocupada, *E* es una lista con los estratos socioeconómicos existentes. *U* incluye las unidades primarias de muestreo (UPM) del estrato e. *V* es el conjunto de viviendas seleccionadas o unidades secundarias de muestreo (USM), *i* denota al i-ésimo residente que trabaja de la vivienda (conglomerado). F_{euv} es el factor de expansión (ajustado por no respuesta y la proyección de población) de la vivienda *euv*. Finalmente y_{euvi} contiene el ingreso reportado por la persona *euvi*.

6.2. Coeficiente Gini

El coeficiente de Gini es una medida de disperción estadística, principalmente utilizado en el estudio de la desigualdad en la distribución del ingreso y formalmente es llamado Coeficiente de Concentración de Gini (CG) [19]. Existen diversas formas de derivar la expresión que se usa para su cálculo, también es posible deducirlo desarrollando un procedimiento geométfico a partir de la curva de Lorenz. Su primer aparición fue en 1912, cuando el estadístico italiano Corrado Gini definio su medida de desigualdad en los siguientes términos:

$$CG = \frac{1}{2\mu}\left[\frac{\sum_{i=1}^{n}\sum_{j=1}^{n}|y_i - y_j|}{n(n-1)}\right] = \frac{1}{2\mu}\Delta$$

Donde y es la variable observada, generalmente es el ingreso obtenido en el hogar. Δ representa la media aritmética de las *n(n-1)* diferencias absolutas de las observaciones y 2μ es su valor máximo y ocurre cuando un individuo concentra todo el ingreso.

Dos años más tarde Gini propuso un nuevo indicador que se define como 1 menos dos veces el área de la curva de Lorenz y mostró que era equivalente a la que había presentado en 1912. La nueva expresión fue:

$$CG = 1 - 2F(y)$$

Donde $F(y)$ representa el área bajo la curva de Lorenz, es decir, la proporción de individuos que tienen ingresos acumulados menores o iguales a y Si suponemos que P es la distribución empírica de probabilidad entonces: $F(y) =$ P (Y $\leq$ y)]

Otra caracterización interesante es:

$$CG = 1 - \frac{E\ \min\ (Y_{(1)}, Y_{(2)})}{E(Y)}$$

Donde $\min\ (Y_{(1)}, Y_{(2)})$ es el mínimo de una muestra de tamaño dos de la variable Y.

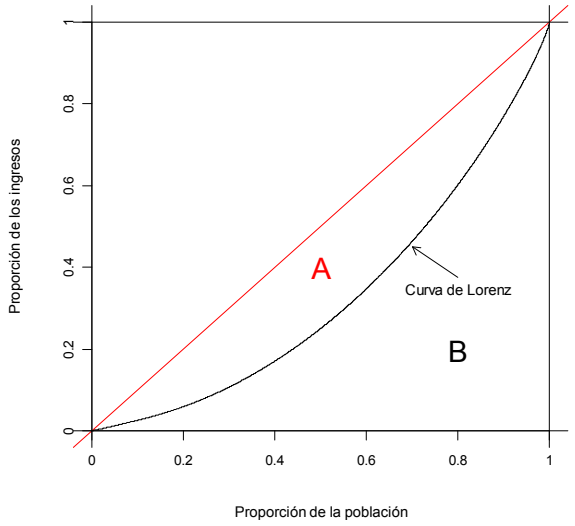


Figura 6.1 Cálculo del Coeficiente de Gini (CG).

El cálculo se puede expresar gráficamente a partir de la figura 6.1, en el eje de las abscisas se presenta la proporción de la población ordenada según el ingreso obtenido y en el eje vertical la proporción de los ingresos totales. La línea en color negro representa la curva de Lorenz y la diagonal muestra la situación de *distribución perfecta*. El CG lo podemos obtener del cociente del área A entre la suma de A y B. Existen dos casos extremos, el

primero se da cuando el área A es cero, entonces la curva de Lorenz empata con la diagonal, lo que sucede es que toda la población tiene el mismo ingreso (distribución perfecta), y la segunda, cuando el área B es cero, entonces todos los ingresos los recibe una sola persona.

En la literatura se pueden encontrar una gran variedad de expresiones que permiten obtener el coeficiente de Gini, para este estudio utilizaremos la siguiente ecuación para estimar una medida de desigualdad individual (per cápita) considerando los factores de expansión.

$$CG = \frac{1}{2\hat{N}^2\hat{\bar{y}}}\left[\sum_{i=1}^{n}\sum_{j=1}^{n} F_i F_j |y_i - y_j|\right]$$

En los textos de análisis de la desigualdad de la distribución del ingreso existe un debate importante sobre la variable que se debe analizar *ingreso total del hogar* o el *ingreso per cápita*. Por supuesto que no es posible responder de manera general a ese dilema, la decisión dependerá básicamente de los objetivos del análisis. En ciertos casos parece justificable el uso del ingreso total de la familia, por el hecho de que el hogar se considera la unidad de consumo en la cual se concentran las percepciones de sus miembros y se decide sobre el destino de tales recursos, no obstante en recientes fechas diversos investigadores han favorecido el estudio hacia el bienestar individual de los ciudadanos, por supuesto en esos casos la variable considerada es el ingreso per cápita.

6.3. Resumen de resultados

	Método	Parámetros	Ingreso promedio (IP)			Coeficiente Gini		Error cuadrático medio (ECM)	
			Promedio de las estimación	D.E.	Dif. porcent.	Promedio de las estimación	D.E.	Promedio de las estimación	D.E.
	Población total [1]		**4,236.7**			**0.321**			
A.	Media general		4,133.5	14.7	2.4	0.308	0.0018	59.7	7.6
B.	Media por clases	ER; Clases=160 (k-medias)	4,194.5	14.6	1.0	0.314	0.0016	28.8	4.1
C.	K-Vecinos más cercanos	ER; k=20; Promedio	4,194.5	14.8	1.0	0.314	0.0016	24.8	3.0
D.	Hot-deck	ER; k=20; Promedio	4,196.6	13.9	0.9	0.314	0.0016	23.2	3.0
E.	Regresión lineal	OC	4,207.7	14.0	0.7	0.316	0.0015	23.5	3.1
F.	Redes Neuronales (NN)	EC, t=2	4,205.0	14.8	0.7	0.315	0.0016	23.3	3.1
G.	Regresión-knn	OC,ER(10)[2], k=11, λ=1/2	4,208.1	13.9	0.7	0.316	0.0016	22.5	3.0

[1] Se refiere a la población ocupada de la Ciudad de León, Gto. que reportaron los ingresos laborales (truncada 1% en las colas)

[2] ER(10)[2]: knn predijo residuales a partir de un espacio reducido, tomó en cuenta exclusivamente las 10 mejores variables detectadas por regresión lasso.

Cuadro 6.1 Resultados de la comparación de métodos de imputación. La primer columna se refiere al nombre del método, la segunda da cuenta de los parámetros empleados para el ajuste de cada técnica en particular, en la siguiente se reporta la media del $\hat{\bar{y}}$ obtenido de las 100 replicas, se indica la desviación estándar (D.E.) y la diferencia porcentual con respecto al parámetro real (4,236.7), enseguida se muestra el CG y finalmente el ECM cada uno con promedio D.E.

Se resaltaron las celdas que en cada columna (excepto la diferencia porcentual) obtuvieron el mínimo. Los métodos que sobresalieron fueron D, E y G. Hot-deck generó las estimaciones de ingreso promedio más estables, pero no las más precisas, la regresión lineal

simple mejoró $\hat{\bar{y}}$ y obtuvo los CG más cercanos a la población de referencia, finalmente Regresión lineal-knn registro el $\hat{\bar{y}}$ más cercano al valor real y el ECM más bajo (ligeramente mejor que regresión lineal simple).

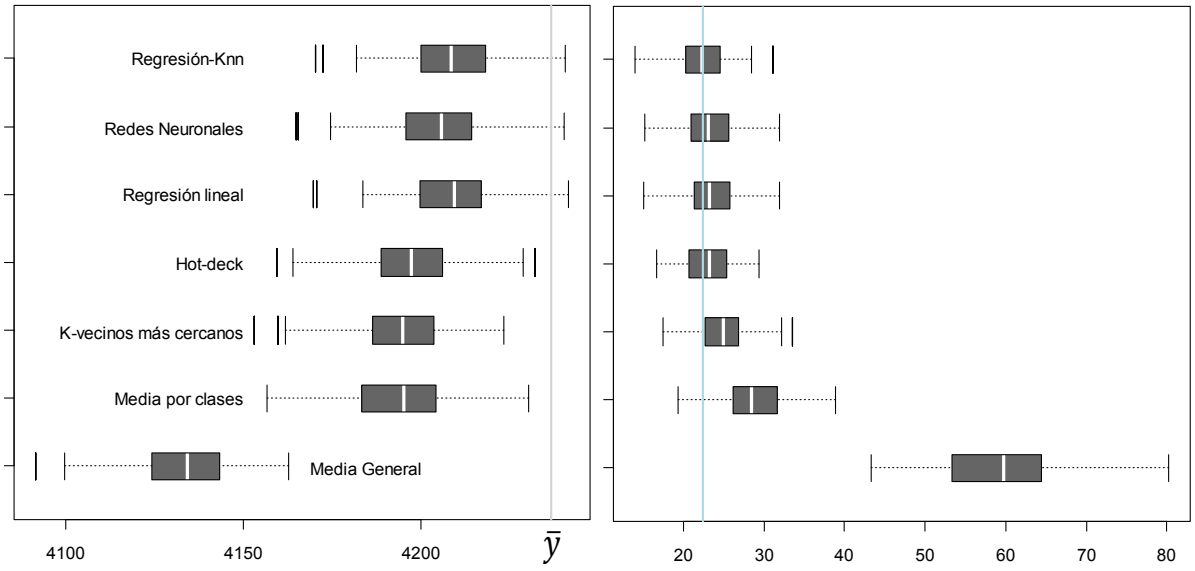


Figura 6.2 Diagramas de caja de la estimación del ingreso promedio y ECM de las 100 replicas ejecutadas a los métodos propuestos. A la izquierda $\hat{\bar{y}}$ con línea gris se observa $\bar{y}$, a la derecha la línea en azul claro marca el ECM mínimo que corresponde a regresión-knn.

Podemos observar de la figura 6.2 que los siete métodos se pueden agrupar en tres grandes bloques: i) constituido únicamente por la media general cuya ventaja es la sencillez para implementarlo pero es el que produce los peores resultados en todos renglones, ii) media por clases, k-vecinos más cercanos y hot-deck tienen como elemento en común que los tres utilizan una función de distancia, en el primer caso para agrupar observaciones y en los restantes para obtener una serie de casos semejantes (donadores) los cuales servirán para generar una predicción promediando aritméticamente sus respuestas, vemos que sus CG e $\bar{y}$ son muy parecidos y mejorando únicamente el ECM en función de la complejidad de la técnica, por último iii) redes neuronales, regresión lineal simple (RL) y RL+knn en donde la configuración final de los tres muestran cierta convergencia, por un lado NN sugiere un número pequeño de neuronas en la capa oculta t=2, pudiendo reducirse a una utilizando el conjunto OC lo que la haría equivalente a regresión lineal y por otro lado tenemos a RL+knn que por construcción tiene por columna vertebral a la regresión lineal y mejorando un poco los resultados en ECM e IP pero con alejándose un poco del CG real.

Las evidencias ($\bar{y}$, ECM y CG) indican que hot-deck produce resultados similares a RL+knn y si consideramos los aspectos generales que rodean a ambas técnicas se podría decretar un empate en el primer lugar entre hot-deck y RL+knn. El primero tiene la ventaja de que su implementación es más sencilla, su modelo no requiere tantos supuestos, funciono bien con el conjunto reducido de variables auxiliares y además las aplicaciones más importantes sobre imputación de ingresos están hechas empleando hot-deck (CPS-

BLS). El segundo en general logró un mejor desempeño, se acercó más al parámetro real $\bar{y}$ y generó el ECM más pequeño. El modelado de la regresión lineal tiene beneficios ya que puede brindar información sobre la bondad de ajuste y los coeficientes pueden generar interpretaciones y abonar en el conocimiento del fenómeno que se estudia. Finalmente RL+knn puede proveer los insumos (selección de variables y residuales) que hot-deck requiere para formar las clases, ordenar los casos o encontrar el más cercano durante su ejecución.

6.4. Aplicación

Para realizar este ejercicio se decidió tomar el mismo dominio geográfico que fue utilizado durante la simulación (capítulo 5. Métodos de predicción), como lo podemos ver en el cuadro 6.2 la tasa de no respuesta muestral (TNR) es 5.9%, en términos absolutos significan 284 casos, a los que agregaremos 21 registros que reportaron el ingreso a través de intervalos de salarios mínimos (pregunta 6.c) haciendo que el porcentaje total de casos que requieren imputación se situé en 6.4%. Los de registros disponibles para ajustar el método y predecir los datos faltantes son 4 284 una vez excluidos los 197 ocupados que no reciben ingresos.

Población ocupada	Estimación		Conteo de registros	
	Absoluto	Relativo	Absoluto	Relativo
Total	**509,630**	**100.0**	**4,786**	**100.0**
1. Hasta un salario mínimo	31,642	6.2	304	6.4
2. Más de 1 hasta 2 salarios mínimos	102,000	20.0	962	20.1
3. Más de 2 hasta 3 salarios mínimos	125,742	24.7	1,190	24.9
4. Más de 3 hasta 5 salarios mínimos	142,763	28.0	1,347	28.1
5. Más de 5 salarios mínimos	56,119	11.0	502	10.5
6. No recibe ingresos	20,771	4.1	197	4.1
7. No especificado	30,593	6.0	284	5.9

Cuadro 6.2 Población ocupada de la Ciudad de León, Gto. por estratos de ingreso perteneciente al primer trimestre de 2005.

El primer paso fue incorporar al conjunto de datos los 305 registros que no contaban con la declaración directa de ingresos y cinco que fueron excluidos en su momento por pertenecer a los extremos de las colas de la distribución. El segundo fue imputar sus variables auxiliares como lo hicimos en el capítulo 4 (ver tabla 4.6 columna Tratamiento a los no especificados), el siguiente paso fue ejecutar la imputación de la variable de interés empleando el método Regresión-knn (ver sección 5.3.7.) y por último, se generaron las estimaciones considerando el diseño muestral de promedios, medianas y curvas suavizadas de distribución para observar el efecto en los resultados antes y después de imputación.

6.5. La estimación

Se generó un conjunto de parámetros para evaluar las alteraciones en la variable ingresos antes y después de la imputación entre los cuales están: promedios, cuantiles, histogramas y de la función de densidad de probabilidad los cuales corresponden a los estudios exploratorios, adicionalmente se incorporaron el ingreso por hora trabajada (promedio y media) y coeficiente de Gini (CG) como una referencia del efecto que podría tener la recuperación de datos en los indicadores que se publican de la encuesta.

Parámetros del ingreso (pesos)	Antes		Después	
	Estimación	E.E. / (I.C.)	Estimación	E.E. / (I.C.)
Ingreso promedio	4,451	77.2	4,496	75.0
Exclusivamente casos imputados			5,126	198.9
Cuantiles				
0.25	2,580	(2,580-2,580)	2,580	(2,580-2,580)
0.50	3,600	(3,440-3,768)	3,655	(3,440-3,870)
0.75	5,160	(5,160-5,160)	5,160	(5,160-5,160)
Ingreso por hora trabajada				
Promedio	25.6	0.5	25.9	0.5
Mediana	19.6	(19.3-20.0)	20.0	(19.4-20.0)
Coeficiente Gini	0.347		0.345	

Nota: E.E. = Error estándar; I.C. = Intervalo de confianza (alfa=0.05)

Cuadro 6.3 Resumen de la variable ingresos antes y después de la imputación de la Ciudad de León, Gto. primer trimestre de 2005.

Podemos observar que los cambios en los resultados después de imputación no son muy notables. La media general $\hat{\bar{y}}$ tiene una modificación del 1%, la mediana 1.5%, el ingreso promedio por hora trabajada 0.8% y su mediana 2.0%, finalmente el CG decrece 0.6%. No obstante se observa que el 6.4% de los casos imputados tienen un ingreso promedio estimado $5 126, 15% superior al estimado inicialmente $4 451, si observamos los E.E. y calculamos los intervalos de confianza nos daremos cuenta que la diferencia es estadísticamente significativa, de manera que este resultados da elementos que apoyan la idea de que la población que se niega a contestar tiene ingresos superiores a los que si lo reportan.

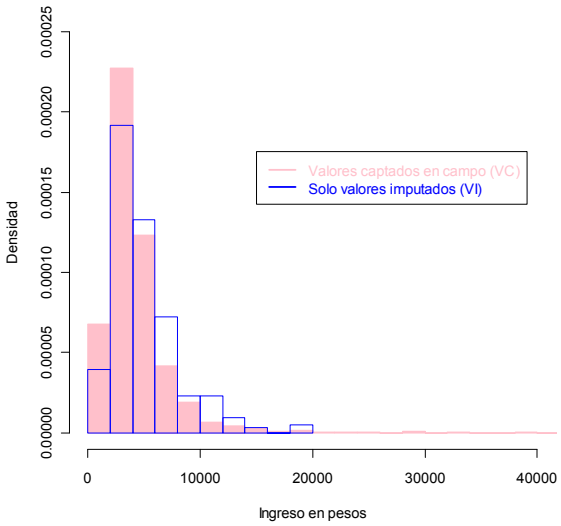


Figura 6.3 Histograma de los ingresos captados en campo (rosa) y los imputados (contornos azules).

En el histograma de la fig. 6.3 podemos observar que la distribución de los ingresos imputados se limita a valores iguales o inferiores a $20 000 mientras que los ingresos observados se mantienen con una frecuencia marginal de los $16 000 en adelante. También se nota que la probabilidad de ocurrencia de los ingresos imputados es menor para los valores de $4 000 o menos y mayor para el intervalo $4 000 a $20 000, si la comparamos con las cantidades captadas en el operativo de recolección. En la imputación no se observan valores a primera vista extremos ni deformaciones en su estructura de distribución.

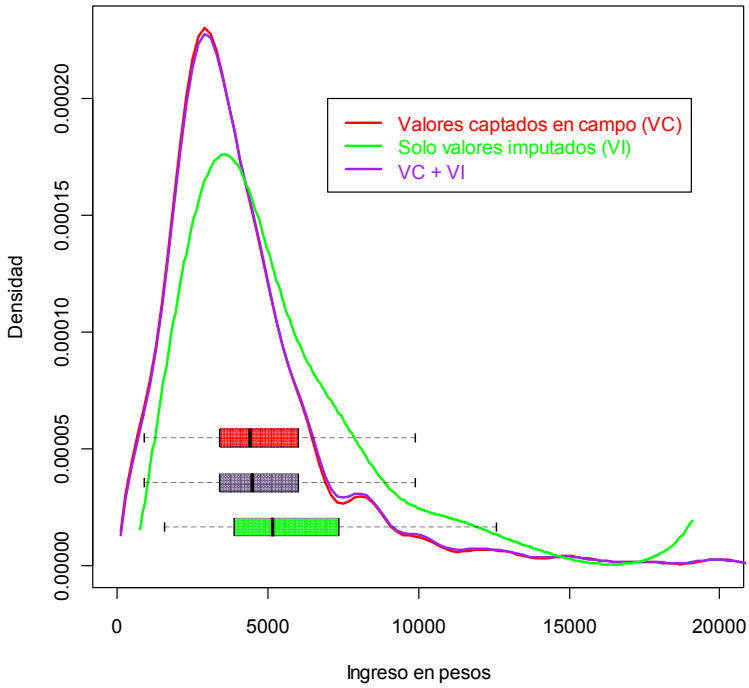


Figura 6.4 Histograma de los ingresos captados en campo (rosa) y los imputados (contornos azules).

La estimación de la función de densidad de la variable de interés se estimó empleando la función `svysmooth` incluida en el paquete `survey` y que considera el diseño muestral en sus cálculos se apoya en las técnicas *KDE* (*Kernel density estimation*). Podemos notar que las fdp de la variable antes y después de imputar -en rojo y morado respectivamente- están prácticamente sobrepuestas lo que nos confirma que no existen alteraciones importantes en el comportamiento del ingreso. La línea en verde es la fdp de los valores imputados, claramente sobresale el mayor suavizamiento, una cúspide más baja y su cola derecha con una caída menos pronunciada, lo que genera un promedio 15% mayor, al extremo derecho la línea se eleva y trunca para indicar que ya no existen casos con ingresos mayores, y denota una aglomeración en este punto, sin embargo su baja proporción (TNR=6%) impide que se muestren cambios significativos en la fdp agregada, habrá entonces que considerar en futuros modelos mayor cuidado con la cola derecha.

Se adicionaron al gráfico diagramas de caja para las tres variables (antes, después y exclusivamente valores imputados) se observan largos bigotes que reafirman el sesgo varias veces identificado en fenómenos económicos y muestra nuevamente que los datos imputados están corridos a la diestra.

7. Conclusiones

Aunque el estudio no concluye con un método definitivo para imputar el ingreso en la Encuesta Nacional de Ocupación y Empleo (ENOE), si aporta un primer ejercicio que permite bajo un cierto dominio geográfico modelar la variable y predecir el ingreso tratando de conservar características como: distribución de probabilidad y la relación con el resto de las variables del cuestionario.

Los resultados coinciden con trabajos realizados con la misma variable en otras encuestas [4] en el sentido de que las estimaciones del ingreso promedio post-imputación se ven afectadas a la alza, lo que junto con otros elementos pone en la mesa la posibilidad de la existencia de sesgo en la respuesta de los informantes que no reportan su ingreso, [20] y que al igual que la experiencia mexicana la tasa de no respuesta en esta variable alcanza el 30% en los estratos socioeconómicos altos.

Logramos identificar que la mayoría de los signos en los coeficientes de regresión son intuitivamente congruentes, por ejemplo, positivos para: años de estudio, horas y periodos del año trabajado, la posición en la empresa y el hogar (patrón/jefe de familia) y tamaño del negocio, y negativos para: personas que realizan quehaceres domésticos, trabajadores en el sector informal y por cuenta propia. No obstante coeficientes negativos para antigüedad en el trabajo y sexo reflejan una falta de capacitación en el trabajo para que las personas con mayor experiencia asuman puestos con mayor responsabilidad y una discriminación de género en el trabajo respectivamente, ambas cualidades negativas de nuestro mercado laboral. Se ensayaron algunas interacciones como en la literatura se sugiere, sin embargo, resultaron no significativas. El ejercicio para reducir la dimensionalidad de los datos trajo resultados parciales al reducir solo en un tercio (de 45 a 29).

Las siguientes investigaciones podrían aplicar métricas para evaluar la conservación de la distribución, profundizar sobre la reducción de la dimensionalidad y la construcción/transformación de variables. Verificar el comportamiento de los modelos en otros dominios (Rurales-dispersos, Ciudades altamente pobladas que lleva consigo desigualdad y múltiples escenarios, etcétera). Será muy importante contar con análisis más detallados sobre la no respuesta que permitan verificar los supuestos que se realizan (MCAR, MAR, IM) y que sirvan también para retroalimentar al método sugerido.

La experiencia internacional en el desarrollo de modelos de imputación –en particular para la variable ingresos- nos indica que se tiene que realizar mucho trabajo y resultados que evaluar antes de proponer un modelo definitivo, *el siguiente paso es aplicar las lecciones aprendidas* en nuevas investigaciones que permitan mejorar los resultados. Estamos seguros que el trabajo aquí desarrollado será un gran apoyo para los estudios venideros.

A. Anexos

1.1. Cuestionario Sociodemográfico (CS)

El archivo lo puede obtener en:
http://www.inegi.org.mx/

1.2. Cuestionario de Ocupación y Empleo (COE)

El archivo lo puede obtener en:
http://www.inegi.org.mx/

3.1 Sidiget: El script de R

En este apartado se expone un ejemplo del código que genera SIDIGET para realizar las estimaciones de un tabulado. La siguiente tabla corresponde a un cruce de dos variables, en filas el estado conyugal de las mujeres unidas (variable D5_10).y en las columnas la condición de violencia por parte de su última relación de pareja (variable D5_17).

Mujeres de 15 años y más, casadas o unidas por estado conyugal según condición de violencia hacia ellas a lo largo de la relación con su última pareja

Estado conyugal	Total	Condición de violencia de las mujeres casadas o unidas		
		Sin incidentes	Con incidentes	No especificado
Total	**21 631 993**	**11 523 757**	**10 088 340**	**19 896**
Vive en unión libre	4 896 390	2 282 199	2 610 973	3 218
Esta casada sólo por lo civil	4 990 702	2 587 136	2 397 401	6 165
Esta casada sólo por la iglesia	727 027	391 184	334 700	1 143
Esta casada civil y religiosamente	11 012 350	6 260 845	4 742 135	9 370
No especificado	5 524	2 393	3 131	0

Sidiget asiste en el diseño del tabulado a través de su interface gráfica como se describe en el capítulo 3.

Cuando usuario oprime el botón “Generar tabulado” sidiget genera un archivo de texto con las instrucciones necesarias para generar las estimaciones (script). Para solicitar que R ejecute el script se instala en cada cliente el programa R/Scilab (D)COM Server 2.50.

Los comandos que se requieren para generar las estimaciones se han dividido en dos programas el primero que contiene las especificaciones del tabulado en cuestión y se genera en tiempo de ejecución, para este ejemplo se denomina `Script_04t.r` y el segundo llamado `funciones.r` que contiene la lista de los paquetes requeridos y una serie de funciones que se desarrollaron para simplificar el código de expuesto en el primer script. En seguida se muestra el script correspondiente al tabulado de ejemplo.

Archivo: `Script_04t.r`

```
###############################################################################
# Encuesta Nacional sobre la Dinámica de las Relaciones en los Hogares
#                         ENDIREH 2006
#
# T4. Mujeres casadas que sufrieron violencia en su última relación de pareja
#
###############################################################################

# ABRE LA CONEXIÓN CON EL SERVIDOR SQL
# El proceso de instalación de sidiget, agrega el Data Source Name (dsn) en el
# administrador del ODBC.
 conn <- odbcConnect(dsn="encesp", uid="SSPI", case="tolower")

 #! Extrae los datos del servidor y los prepara para el proceso de estimación.
 # *1.- Variable que marca los registros que entran al tabulado (universo)
 # *2.- Variable que se desagrega en las columnas, cambia los blancos en 0's
 # *3.- Crea una serie de variables tipo dummy que descomponen la variable D5_17
 # *4.- Incluye los campos del diseño muestral y el ponderador.
 dat.origen <- as.data.frame(sqlQuery(conn, "
    SELECT case when (d0_3=1) then 1 else 0 end uT,                    # *1
        case when not(D5_10_C=' ') then D5_10_C else 0 end D5_10_C,    # *2
        case when D5_17>=1 and d0_3=1 then 1 else 0 end D5_17x0 ,      # *3
        case when (d0_3=1) and D5_17=1 then 1 else 0 end D5_17x01,     # *3
        case when (d0_3=1) and D5_17=2 then 1 else 0 end D5_17x02,     # *3
        case when (d0_3=1) and D5_17=3 then 1 else 0 end D5_17x03,     # *3
        zupm,zest, FAC_PER                                             # *4
        FROM TH_Mujeres"))
 attach (dat.origen)

 #! Define el diseño de muestreo
 dise <- svydesign (id=~zupm, strata=~zest, weights=~FAC_PER,
                    data=dat.origen, nest=TRUE)

#! Inicia variables
 #! cota.cv es el nivel contra el que se comparan los cv de cada celda para
 #!         considerar si una celda contiene una estimación no confiable
 cota.cv <- .20
 #! La variable formula lista las estimaciones que realiza en una corrida
 formula <- "~D5_17x0 +D5_17x01+D5_17x02+D5_17x03"
 #! nPor Es el numero de procesos (o pasadas) necesarios para construir el cuadro
 #!      El cuadro de ejemplo requiere dos procesos, el primero estima el
 #!      renglón uno (totales) y el segundo calcula el cruce de las
 #!      variables D5_17 y D5_10_C  renglones 2 a 6.
 nPor <- 2
 nFor <- 4
 por <- rep(" ",2)
 por [1] <- " "
 por [2] <- "~D5_10_C"

 m.Total <- NULL;  m.Muestral <- NULL; m.SE <- NULL; m.CV <- NULL

# El siguiente ciclo ejecuta los procesos necesarios para construir el cuadro
```

```
for (i in 1:2)
{
 #! La función estimacion (ver archivo funciones.r) realiza la estimación de los
 #!     totales listados en "formula", cruzado por el contenido de la variable
 #!     "por[i]" con el diseño de muestreo definido en "dise"
 proceso <- estimacion (formula,por[i],dise)

 #! Vacia las estimaciones en matrices separadas
 m.To <- as.matrix(proceso$total)
 m.Mu <- as.matrix(proceso$muestral)
 m.S <- as.matrix(proceso$se)
 m.C <- as.matrix(proceso$cv)

  #! Asigna nombres de las filas para guiar al proceso que vacia las cifras
  #!        a los formatos de la hoja de cálculo
 if (por[i]==" ") {
   if  (nPor>1) {
     por[i] <- paste(por[i+1]," ") }
   rownames(m.To) <- paste(sub(" *([^ ]+) *", "\\1",
paste(por[i],"x0")),rownames(m.To))
   rownames(m.Mu) <- paste(sub(" *([^ ]+) *", "\\1",
paste(por[i],"x0")),rownames(m.Mu))
   rownames(m.S) <- paste(sub(" *([^ ]+) *", "\\1",
paste(por[i],"x0")),rownames(m.S))
   rownames(m.C) <- paste(sub(" *([^ ]+) *", "\\1",
paste(por[i],"x0")),rownames(m.C))
 }
 else
 {
   rownames(m.To) <- sub(" *([^ ]+) *", "\\1", paste(sub(" *([^ ]+) *", "\\1",
                         paste(por[i],"x0")),rownames(m.To)))
   rownames(m.Mu) <- sub(" *([^ ]+) *", "\\1", paste(sub(" *([^ ]+) *", "\\1",
                         paste(por[i],"x0")),rownames(m.Mu)))
   rownames(m.S) <- sub(" *([^ ]+) *", "\\1", paste(sub(" *([^ ]+) *", "\\1",
                        paste(por[i],"x0")),rownames(m.S)))
   rownames(m.C) <- sub(" *([^ ]+) *", "\\1", paste(sub(" *([^ ]+) *", "\\1",
                        paste(por[i],"x0")),rownames(m.C)))
 }

 # Agrega las cifras generadas por el i-ésimo proceso a matrices independietes
 m.Total <- rbind(m.Total,m.To)
 m.Muestral <- rbind(m.Muestral,m.Mu)
 m.SE <- rbind(m.SE,m.S)
 m.CV <- rbind(m.CV,m.C)
}


 #! Envia a archivos csv
 write.table(m.Total, file =
"K:/Organiza/Saree/SIDIGET/ENDIREH2006/Cifras/Total_4.txt",
             sep = ",", col.names = NA)
 write.table(m.CV, file =
"K:/Organiza/Saree/SIDIGET/ENDIREH2006/Cifras/CV_4.txt",
             sep = ",", col.names = NA)
 write.table(m.SE, file =
"K:/Organiza/Saree/SIDIGET/ENDIREH2006/Cifras/SE_4.txt",
             sep = ",", col.names = NA)
  odbcCloseAll()
```

Archivo: `funciones.r`

```
#########################################################################
#             SIDIGET
#
#  FUNCIONES PARA LA ESTIMACIÓN Y EVALUACIÓN DE CIFRAS
#
#
#########################################################################

# LIBRERIRAS
  library(survey)
  library(RODBC)

# FUNCIONES

  #! Calculo de razón
  calcula.razon <- function(f0,f1,dise,tCol) {
    resultado <- NULL
    if (f1==" ") {
      for (n in 1:length(f0)) {
        out <- svymean (as.formula(paste("~",f0[n],sep="")), dise, na.rm=T,
vartype = c("cv","se"))
                  if (n==1) {
          resultado <- list (round(t(as.data.frame(out)[,1])*100,digit=1),
                             t(as.data.frame(cv(out))),
                             t(as.data.frame(out)[,2]),
                             NULL)
        } else {
        resultado <-
list(cbind(resultado[[1]],round(t(as.data.frame(out)[,1])*100,digit=1)),
                           cbind(resultado[[2]],t(as.data.frame(cv(out)))),
                           cbind(resultado[[3]],t(as.data.frame(out)[,2])),
                           NULL))
        }
      }
          } else {
        for (n in 1:length(f0)) {
          out <- svyby (as.formula(paste("~",f0[n],sep="")), as.formula(f1),
dise, svymean, na.rm=T, vartype = c("cv","se"))
                    if (n==1) {
            resultado <- list(t(t(round((as.data.frame(out)[,2])*100,digit=1))),
                              t(t(as.data.frame(cv(out)))),
                              t(t(as.data.frame(out)[,3])),
                              t(t(as.data.frame(out)[,1])))
          } else {
            resultado <-
list(cbind(resultado[[1]],t(t(round(as.data.frame(out)[,2]*100,digit=1)))),
                              cbind(resultado[[2]],t(t(as.data.frame(cv(out))))),
                              cbind(resultado[[3]],t(t(as.data.frame(out)[,3]))),
                              out[[1]])
          }
        }
      }

      names(resultado) <- c("razon","cv","se","variables")
      return (resultado)
  }

  #! Calculo  Total
  estimacion<- function (f0,f1,dise,tCol,calculo) {
    if (f1==" ") {
        out <- svytotal (as.formula(f0), dise, vartype = c("cv","se"), na.rm = T)
        mue <- svytotal (as.formula(f0), dise.M, vartype = c("cv","se"), na.rm =
T)
```

```
        resultado <- list (t(as.data.frame(out)[,1]),t(as.data.frame(mue)[,1]),
                           t(cv(out)),t(as.data.frame(out)[,2]),NULL)
        names(resultado) <- c("total","muestral","cv","se","variables")
    }
    else
    {
        out<-as.data.frame(svyby(as.formula(f0),as.formula(f1),
                          dise,svytotal,vartype=c("cv","se"),keep.var=TRUE))
        mue <- as.data.frame(svyby(as.formula(f0),as.formula(f1),dise.M,
                                   svytotal,deff=FALSE,keep.var=FALSE,
keep.names=TRUE,
                                   vartype=c("se")))

        col.by <- sum(strsplit(f1,NULL)[[1]]=="+")+1
        n_col <- (dim (out)[2] - col.by ) / 3
        varCruce <- as.data.frame(out[,1:col.by])
        colnames(varCruce) <- colnames(as.data.frame(out))[1:col.by]

        resultado <- list (as.data.frame(out[((rownames(out)!= "NA") &
                           (rownames(out)!= "0") & (rownames(out)!= "1.0") &
(rownames(out)!= "2.0")),
                           (col.by+1):(col.by+n_col)]),
                           as.data.frame(mue[((rownames(mue)!= "NA") &
                           (rownames(mue)!= "0") & (rownames(out)!= "1.0") &
(rownames(out)!= "2.0")),
                           (col.by+1):(col.by+n_col)]),
                           as.data.frame(out[((rownames(out)!= "NA") &
                           (rownames(out)!= "0") & (rownames(out)!= "1.0") &
(rownames(out)!= "2.0")),
                           (col.by+n_col+1):(col.by+(n_col*2))]),
                           as.data.frame(out[((rownames(out)!= "NA") &
                           (rownames(out)!= "0") & (rownames(out)!= "1.0") &
(rownames(out)!= "2.0")),

(col.by+(n_col*2)+1):(col.by+(n_col*3))]),varCruce[((rownames(out)!= "NA") &
                           (rownames(out)!= "0") & (rownames(out)!= "1.0") &
(rownames(out)!= "2.0")),])

       names(resultado) <- c("total","muestral","cv","se","variables")
    }
    return (resultado)
  }

  #! Pinta el cuadro con los coeficientes de variación con la escala de colores
  imagen <- function (m.CV,ren.NE,col.NE,titulo){
      m.CV.sNE <- m.CV
      if (ren.NE != 0 ) {
          m.CV.sNE <- m.CV.sNE [-ren.NE,]
      }
      if (col.NE != 0 ) {
          m.CV.sNE <- m.CV.sNE [,-col.NE]
        }
      n.ren <- dim(m.CV.sNE)[1]   # Número de renglones
      n.col <- dim(m.CV.sNE)[2]   # Número de columnas
      m.CV.sNE[(is.na(m.CV.sNE)) | (m.CV.sNE>1)] <- 1   # No se logró estimar el
CV para la celda o por error númerico el CV es > de 1
      print (m.CV.sNE)
      image (t(m.CV.sNE[n.ren:1,]),col= heat.colors(5)[5:1],
             breaks =seq(from=0,to=1, by=.2),yaxt="n")
      axis (3, at= cruce.eje (n.col),labels = colnames(m.CV.sNE),cex.axis=.6)
      axis (2, at= cruce.eje (n.ren),labels =
rownames(m.CV.sNE)[n.ren:1],cex.axis=.6,las=2)
      title (main = titulo, font.main = 4)
```

```
    }
    #! Esta función se utiliza en imagen()
    cruce.eje <- function (n) {
        if (n > 0  ) resultado <- 0
        if (n == 1 ) resultado <- .5
        if (n > 1  ) resultado <- seq (from=0,to=1,length.out=n)
    return (resultado)}


      #! Evaluación
       evalua <- function ( m.CV,ren.NE,col.NE,cota){
       m.CV.sNE <- m.CV
       m.CV.sNE <- m.CV.sNE [,-1]
       if (ren.NE != 0 ) {
           m.CV.sNE <- m.CV.sNE [-ren.NE,]
       }
       if (col.NE != 0 ) {
           m.CV.sNE <- m.CV.sNE [,-col.NE]
         }
       v.dim <- dim(m.CV.sNE)

       celdas <- prod(v.dim)
       n.menor.20 <- sum(m.CV.sNE <= cota.cv)
       n.mayor.20 <- sum(m.CV.sNE > cota.cv)
       resultado <- list
(celdas,n.menor.20,n.mayor.20,((n.menor.20/celdas)>=.80),(n.menor.20/celdas*100))
       cNaN <- sum(m.CV.sNE == NaN)
       names(resultado) <-
c("Total.celdas","T.celdas.cv.cota","T.celdas.no.cv.cota","Result.Eval","%.correc
to")
        return(resultado)}


    #! genera agrupación de celdas
    resumen <- function (m.tse,m.tcv,m.Tot,m.Mue,tab,var.Fac,var.Tot,Tabla,cond){
       query <- paste("select sum(",var.Fac,") from ",Tabla," ",cond)
       t.uni <- as.character(sqlQuery(conn,query))

       query <- paste("select count(*) from ",Tabla," ",cond)
       m.uni <- as.character(sqlQuery(conn,query))

       if (dim(m.Tot)[2]==1)
       {
       agrupa <-
cbind(rownames(m.Tot),tab,as.list(row(m.Tot)),as.list(col(m.Tot)),

t.uni,as.list(m.Tot),as.list(m.Tot/as.integer(t.uni)*100),m.uni,as.list(m.Mue),
         as.list(m.Mue/as.integer(m.uni)*100),as.list(m.tcv),as.list(m.tse))
       }
       else {
       agrupa <- cbind(tab,as.list(row(m.Tot[,-1])),as.list(col(m.Tot[,-1])),
         t.uni,as.list(m.Tot[,-1]),as.list(m.Tot[,-
1]/as.integer(t.uni)*100),m.uni,as.list(m.Mue[,-1]),
         as.list(m.Mue[,-1]/as.integer(m.uni)*100),as.list(m.tcv[,-
1]),as.list(m.tse[,-1]))
       }
       colnames(agrupa) <-
c("Renglon","Tabulado","Renglon","Columna","UniversoE","Total","Rel Abs",
         "UniversoM","Muestral" ,"Rel Mue", "CV", "SE")
       setwd(sub(" *([^ ]+) *", "\\1",paste(ruta,"Cifras/Agrupados")))
       write.table(agrupa, file = sub(" *([^ ]+) *", "\\1",paste("c",tab,".txt")),
           sep =",", col.names = NA)
  }
```

4.1 Población ocupada por posición según condición de acceso a instituciones de salud.

Población ocupada por posición en la ocupación según condición acceso instituciones de salud

Posición en la ocupación	Acceso a instituciones de salud			
	Total	Con acceso	Sin acceso	No especificado
Total	40 575 874	14 476 634	25 853 525	245 715
Trabajadores independientes	11 401 699	182 820	11 202 650	N.S.
Empleadores	1 890 071	80 576	1 802 177	N.S.
Trabajadores por cuenta propia	9 511 628	102 244	9 400 473	N.S.
Trabajadores subordinados	29 174 175	14 293 814	14 650 875	229 486
Trabajadores subordinados y remunerados	26 165 241	14 276 767	11 660 847	227 627
Asalariados	23 928 538	13 969 646	9 898 497	60 395
Con percepciones no salariales	2 236 703	307 121	1 762 350	167 232
Trabajadores no remunerados	3 008 934	N.S.	2 990 028	N.S.

FUENTE: INEGI. Encuesta nacional de ocupación y empleo. Primer trimestre del 2005

Población ocupada por posición en la ocupación según condición acceso instituciones de salud

Posición en la ocupación	Acceso a instituciones de salud			
	Total	Con acceso	Sin acceso	No especificado
Total	100.0	35.7	63.7	0.6
Trabajadores independientes	100.0	1.7	98.3	N.S.
Empleadores	99.7	4.3	95.4	N.S.
Trabajadores por cuenta propia	100.0	1.1	98.9	N.S.
Trabajadores subordinados	100.0	49.0	50.2	0.8
Trabajadores subordinados y remunerados	100.0	54.6	44.6	0.8
Asalariados	100.0	58.4	41.4	0.2
Con percepciones no salariales	100.0	13.7	78.8	7.5
Trabajadores no remunerados	99.4	N.S.	99.4	N.S.

FUENTE: INEGI. Encuesta nacional de ocupación y empleo. Primer trimestre del 2005

Trabajadores subordinados y remunerados por tipo de remuneración según prestaciones laborales

	Total	Con prestaciones	Sin prestaciones	No especificado
Total	26 165 241	15 844 721	10 197 836	122 684
Asalariados	23 928 538	15 423 943	8 395 498	109 097
Con percepciones no salariales	2 236 703	420 778	1 802 338	N.S.

FUENTE: INEGI. Encuesta nacional de ocupación y empleo. Primer trimestre del 2005

Trabajadores subordinados y remunerados por tipo de remuneración según prestaciones laborales

	Total	Con prestaciones	Sin prestaciones	No especificado
Total	100.0	60.6	39.0	0.4
Asalariados	100.0	64.5	35.1	0.4
Con percepciones no salariales	99.5	18.9	80.6	N.S.

FUENTE: INEGI. Encuesta nacional de ocupación y empleo. Primer trimestre del 2005

4.2 Diagrama de dispersión ingreso contra edad.

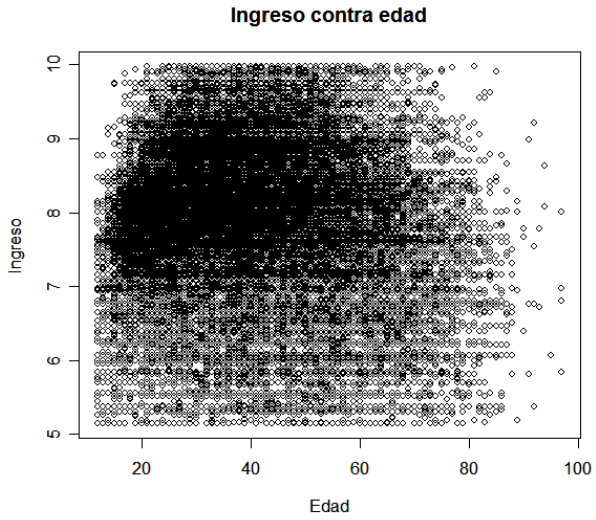


Diagrama de dispersión de los ingresos (logaritmo) contra la edad de la población ocupada. Cifras muestrales de la ENOE 105.

4.3 Calculo de la Tasa de Condiciones Críticas de Ocupación

TASA DE CONDICIONES CRÍTICAS DE OCUPACIÓN (TCCO)

$$TCCO = \frac{PO35HRS.RM + PO35HRS.1SM + PO48HRS.2SM}{PO} x100$$

Donde:

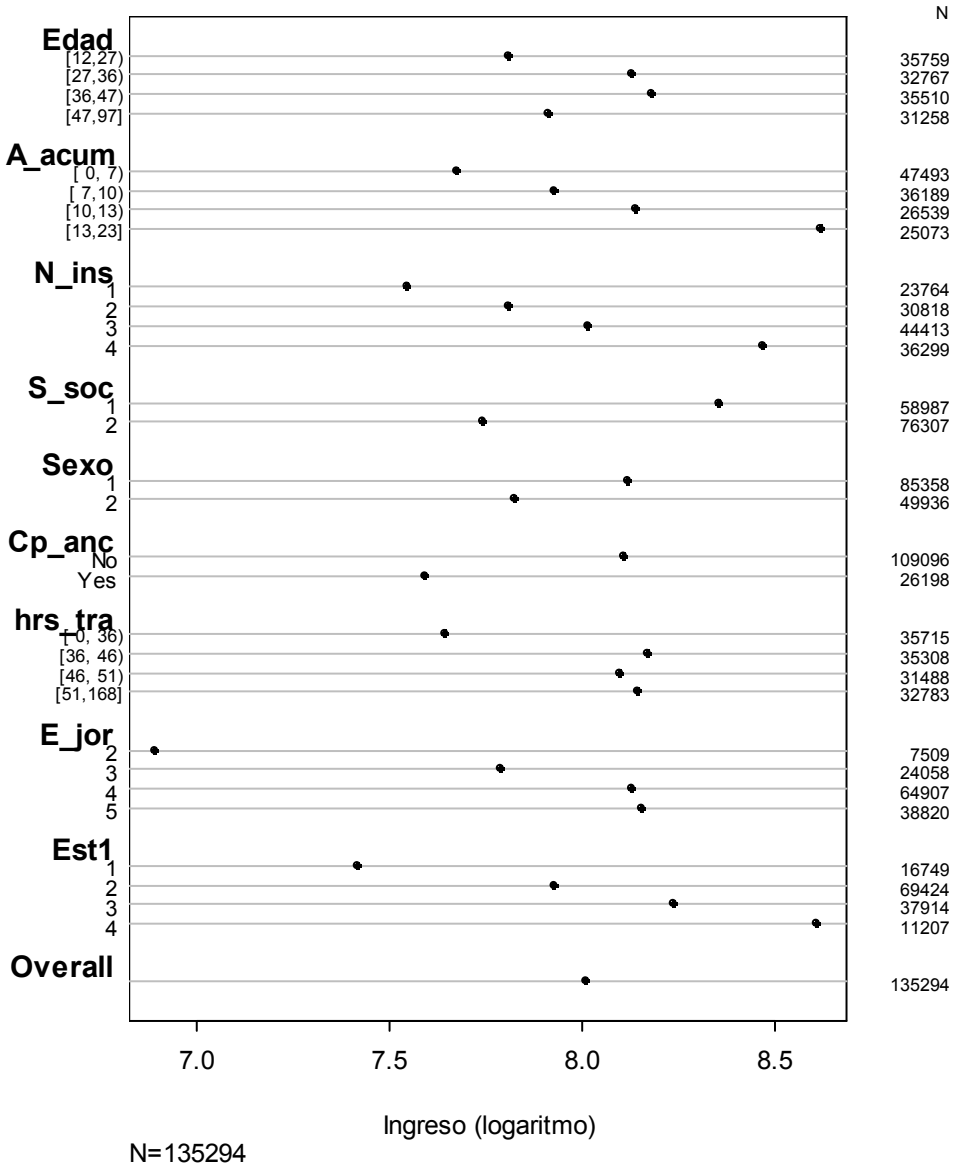
PO35HRS.RM = Población ocupada que trabaja a la semana 35 o menos horas por razones de mercado.

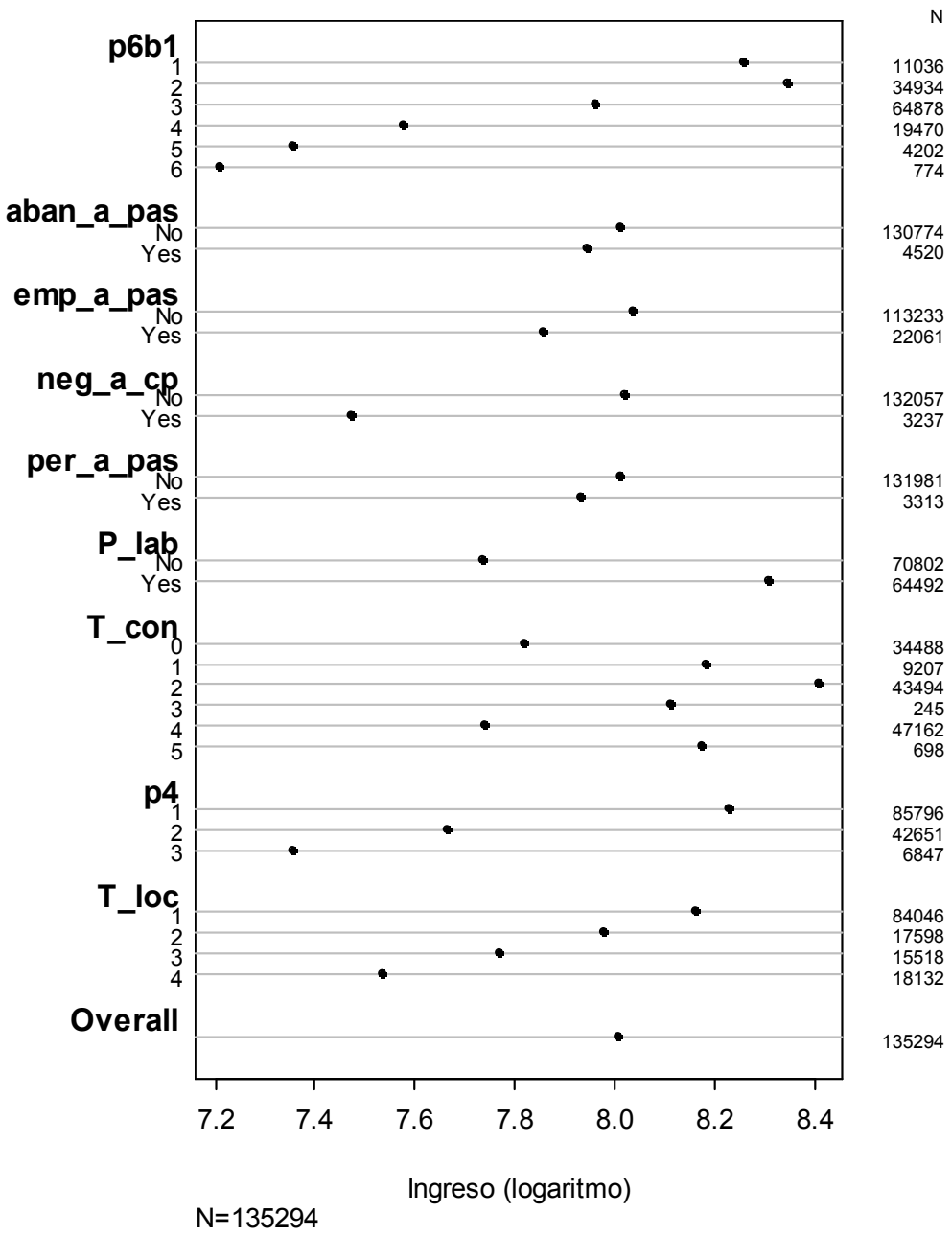
PO35HRS.1SM = Población ocupada que trabaja más de 35 horas a la semana y que gana menos de 1 salario mínimo.

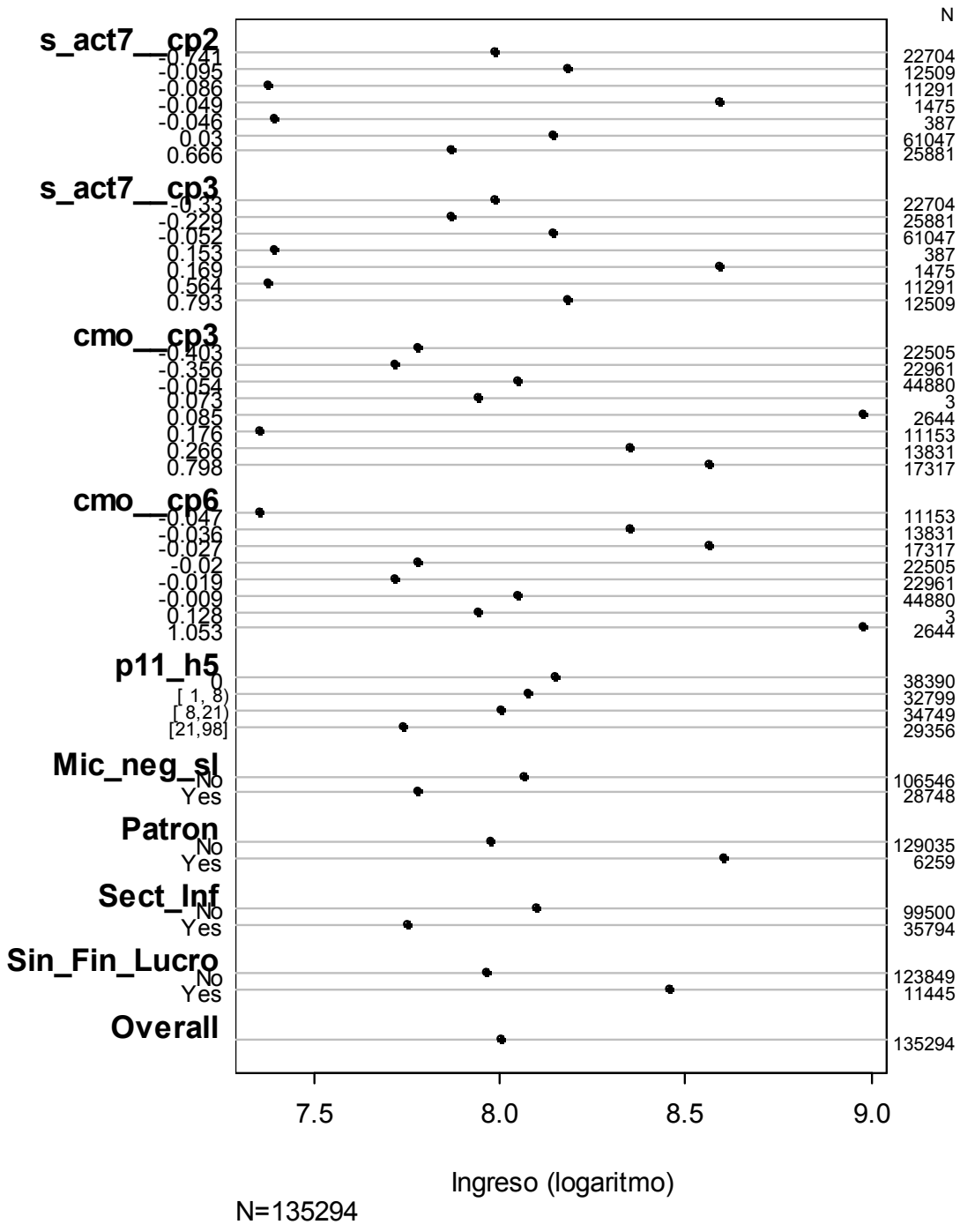
PO48HRS.2SM = Población ocupada que trabaja más de 48 horas a la semana y gana más de 1 y hasta 2 salarios mínimos.

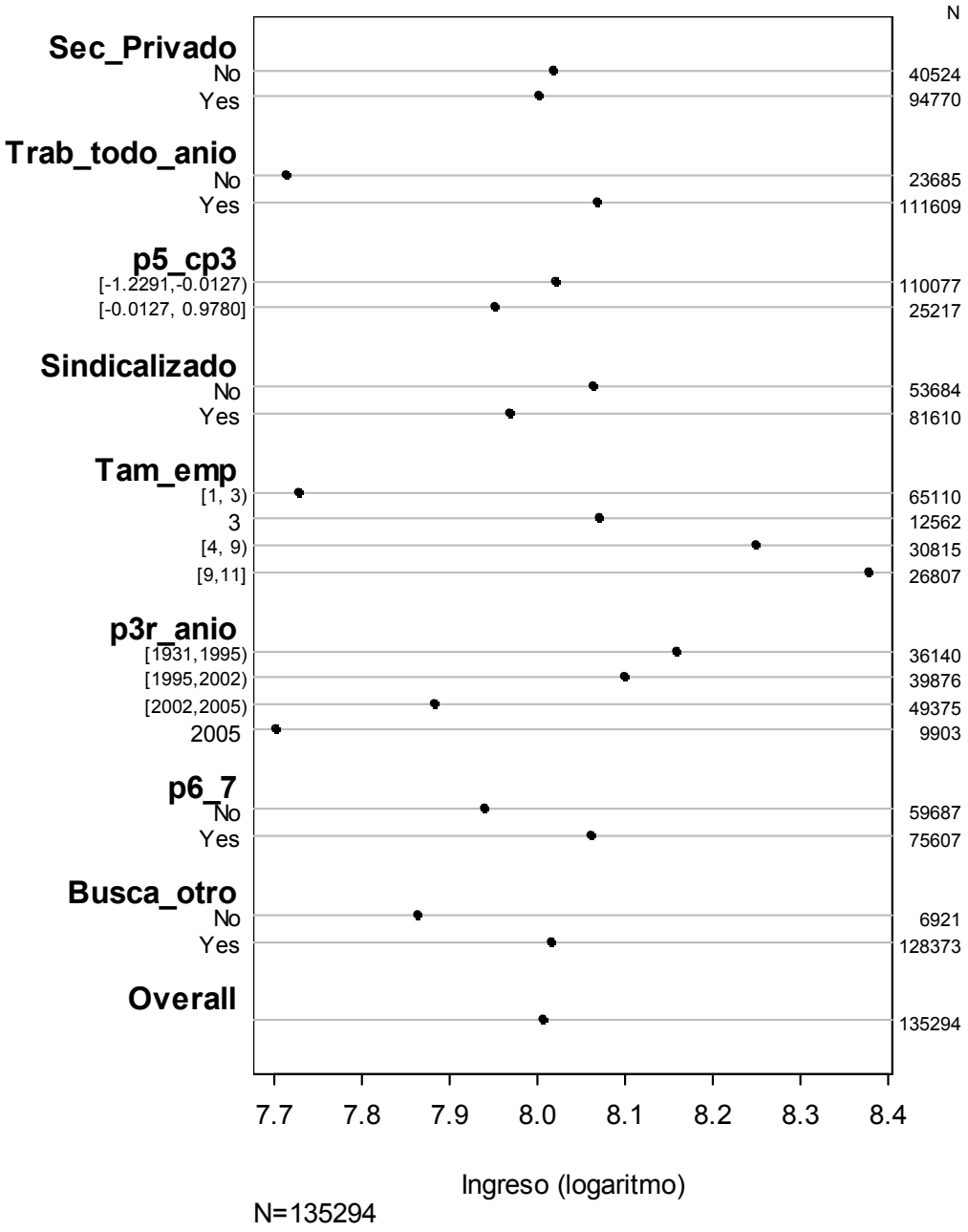
PO = Población ocupada

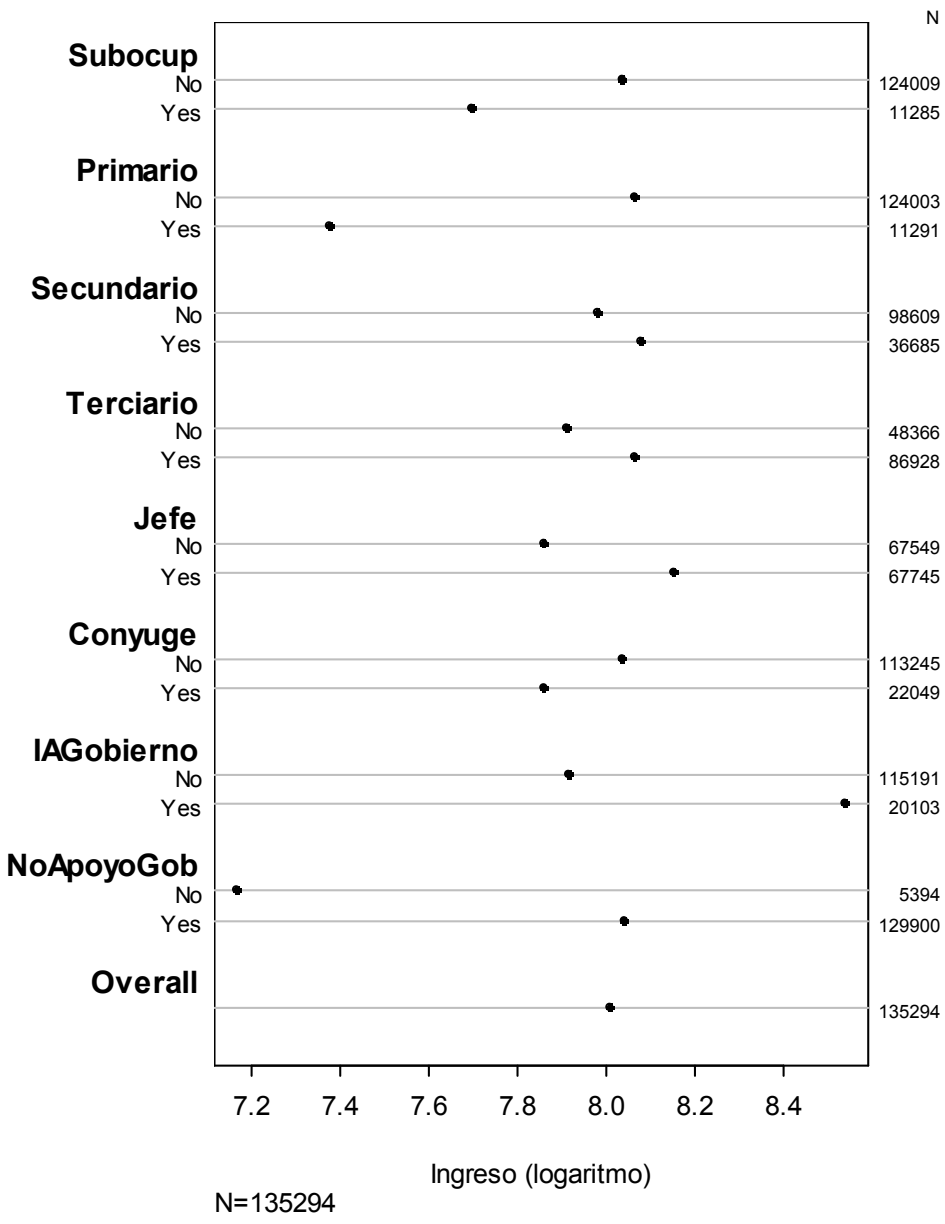
4.4 Media muestral del ingreso para una selección de variables.




N
p6b1
1 11036
2 34934
3 64878
4 19470
5 4202
6 774
aban_a_pas
No 130774
Yes 4520
emp_a_pas
No 113233
Yes 22061
neg_a_cp
No 132057
Yes 3237
per_a_pas
No 131981
Yes 3313
P_lab
No 70802
Yes 64492
T_con
0 34488
1 9207
2 43494
3 245
4 47162
5 698
p4
1 85796
2 42651
3 6847
T_loc
1 84046
2 17598
3 15518
4 18132
Overall
135294
7.2
7.4
7.6
7.8
8.0
8.2
8.4
lngreso (logaritmo)
N=135294


N
s_act7__cp2
-0.741
-0.095
-0.086
-0.049
-0.046
0.03
0.666
22704
12509
11291
1475
387
61047
25881
s_act7__cp3
-0.33
-0.229
-0.052
0.153
0.169
0.564
0.793
22704
25881
61047
387
1475
11291
12509
cmo__cp3
-0.403
-0.356
-0.054
0.073
0.085
0.176
0.266
0.798
22505
22961
44880
3
2644
11153
13831
17317
cmo__cp6
-0.047
-0.036
-0.027
-0.02
-0.019
-0.009
0.128
1.053
11153
13831
17317
22505
22961
44880
3
2644
p11_h5
0
[1, 8)
[8,21)
[21,98]
38390
32799
34749
29356
Mic_neg_sl
No
Yes
106546
28748
Patron
No
Yes
129035
6259
Sect_Inf
No
Yes
99500
35794
Sin_Fin_Lucro
No
Yes
123849
11445
Overall
135294
7.5
8.0
8.5
9.0
Ingreso (logaritmo)
N=135294


N
Sec_Privado
No
40524
Yes
94770
Trab_todo_anio
No
23685
Yes
111609
p5_cp3
[-1.2291,-0.0127)
110077
[-0.0127, 0.9780]
25217
Sindicalizado
No
53684
Yes
81610
Tam_emp
[1, 3)
65110
3
12562
[4, 9)
30815
[9,11]
26807
p3r_anio
[1931,1995)
36140
[1995,2002)
39876
[2002,2005)
49375
2005
9903
p6_7
No
59687
Yes
75607
Busca_otro
No
6921
Yes
128373
Overall
135294
7.7
7.8
7.9
8.0
8.1
8.2
8.3
8.4
lngreso (logaritmo)
N=135294


N
Subocup
No
Yes
124009
11285
Primario
No
Yes
124003
11291
Secundario
No
Yes
98609
36685
Terciario
No
Yes
48366
86928
Jefe
No
Yes
67549
67745
Conyuge
No
Yes
113245
22049
lAGobierno
No
Yes
115191
20103
NoApoyoGob
No
Yes
5394
129900
Overall
135294
7.2
7.4
7.6
7.8
8.0
8.2
8.4
lngreso (logaritmo)
N=135294

5.1 Función de predicción knn con promedio ponderado

```
knn.predict.weighted <- function (train, test, y, dist.matrix, k = 1,
    agg.meth = if (is.factor(y)) "majority" else "weighted.mean",
ties.meth = "min")
{
    n <- length(test)
    if (is.unsorted(train))
        train <- sort(train)
    if (is.unsorted(test))
        test <- sort(test)
    d <- dist.matrix[test, train]
    if (length(y) > length(train))
        y <- y[train]
    if (n == 1) {
        print ("Solo para conjuntos con de dos o más casos")
        return(NULL)
    }
    else {
        dd <- t(apply(d, 1, function(x) rank(x, ties.method =
ties.meth)))
        apply(cbind(dd,d), 1, function(x) { xm <- matrix(x,ncol=2)
                                            ww <- xm[xm[,1] <= k,2]
                                            ww <- ww /sum(ww)
                                            weighted.mean(y[xm[,1] <=
k],w=ww)})
    }
}
```

Bibliografía

[1] Graham Kalton and Daniel Kasprzyk. The Treatment of Missing Survey Data. Survey Methodology, Statistics . - 1985.

[2] Harrell Jr. Frank E. Regression Modeling Strategies With applications to linear models, logistic regression and survival analysis. - New York : Springer - Verlang, 2001. - Vol. Springer series in statistics.

[3] Porro Stefano M. Iacus y Giuseppe. Missing data imputation, classification, prediction and average treatment effect estimacion via Random Recursive Partitioning .

[4] Paulin Geoffrey D. y Sweet, Elizabeth M. Modeling Income in the U.S. Consumer Expediture Survey : Journal of Official Statistics, 1996. - Vol. 12 : pp 403-419. - No. 4.

[5] Kleiber Chirstian The Lorenz curve in economics and econometrics . - Dortmund,Germany, 2005.

[6] Encuesta Nacional de Ocupación y Empleo 50 preguntas y respuestas . – Instituto Nacional de Estadística Geografía e Informática. 2005. www.inegi.org.mx

[7] R Data Import/Export // R Development Core Team. - junio 27, 2007. - 2.5.1.

[8] Wikipedia. Open Database Connectivity (**ODBC**). Categories': Acronym's of informatics /APIs de Microsoft. http://es.wikipedia.org/wiki/ODBC

[9] Peng Roger. Interacting with data using the filehash package for R . - Johns Hopkins University, Dept. of Biostatistics, 2006. - http://www.bepress.com/jhubiostat/paper108.

[10] D. Adler1 O. Nenadic, W Zucchini, C. Gläser Georg-August The ff package: Handling Large Data Sets in R with Memory Mapped Pages of Binary Flat File . - 37073 Göttingen, Germany : Universität Göttingen Institut für Statistik und Ökonometrie, Platz der Göttingen Sieben 5.

[11] Little Roderick J. A., Donald B. Rubin Statistics Analysis with Missing Data. - New York : John Wiley & Sons, 1986.

[12] Documento metodológico del diseño de muestra del Marco Nacional de Viviendas para encuestas en hogares. Instituto Nacional de Estadística Geografía e Informática. Documento interno. DGE Dirección de Marcos y Marcos Estadísticos. Marzo 2004.

[13] Lohr Sharon L. de Muestreo: Diseño y Análisis . - México : Thomson Paraninfo. 2000.

[14] Rawlings John O., Pantula Sastry G. y Dickey David A. Applied Regression Analysis: A Research Tool (Springer Texts in Statistics)

[15] Tibshirani, R. (1996). Regression shrinkage and selection via the lasso. J. Royal. Statist. Soc B., Vol. 58, No. 1, pages 267-288). A simple explanation of the Lasso and Least Angle Regression. http://www-stat.stanford.edu/~tibs/lasso/simple.html, 2008.

[16] Särndal, Carl-Erik, Swensson, Bengt y Wretman Jan. Model Assisted Survey Sampling. Springer series in statistics. 2003.

[17] Izenman, Alan Julian. Modern Multivariate Statistical Techniques. Springer texts in statistics. 2008. pp 336-423.

[18] Hastie Trevor, Robert Tibshirani y Jerome Friedman. The elements of statistical learning. Second Edition. Springer-Verlang.

[19] Medina, Fernando. Consideraciones sobre el índice de Gini para medir la concentración del ingreso. Serie de estudios prospectivos. División de estadística y proyecciones económicas, CEPAL, ONU. Santiago de Chile 2001.

[20] Rubin, Donald B. Imputación Múltiple for Noresponse in Surveys. Department of Statistic. John Wiley & Sons. 1987.

[21] Tibshirani, Robert. . A simple explanation of the Lasso and Least Angle Regression. Stanford University. Consultado en marzo de 2009. http://www-stat.stanford.edu/~tibs/lasso/simple.html

[22] Martin David, Roderick J.A. Little, Michael E. Samuhel, Robert K. Triest. Alternative Methods for CPS Income Imputation. Journal of the American Statistical Association. Vol 81. 1986. Pp. 29-41. Fecha de consulta 25/01/2009. Liga estable: http://www.jstor.org/stable/2287965

[23] Murrell Paul, Hothorn Torsten, Fox John, Venables Bill y Ligges Uwe. RNews Magazzine, Octubre de 2006 Volumen 6/4, pag 19 a 24. Fecha de consulta 04/12/2006. Liga estable : http://CRAN.R-project.org/doc/Rnews/

[24] Adler Daniel, Gläser Christian, Nenadic Oleg, Oehlschlägel Jens y Zucchini Walter. The ff package: Handling Large Data Sets in R with Memory Mapped Pages of Binary Flat Files. Georg-August-Universität Göttingen, Institut für Statistik und Ökonometrie. 2007. Alemania.

[25] Puerta Goicoechea, Aitor. Imputación Basada en Árboles de Clasificación. Instituto Vasco de Estadística (Eustat). 2002. Liga: www.eustat.es/document/datos/ct_04_c.pdf

[26] Särndal Carl-Erik y Lundström Sixten. Estimation in Surveys with Nonresponse. Willey series in survey methodology. 2005.