Taller Análisis Exploratorio Estadístico del ICFES

Minería de Datos

Jonatan Camilo Igua Contreras
ID: 808919 | NRC: 73466

2025-04-0

Análisis Exploratorio resultados de la prueba saber 11

En el siguiente taller se realizo un análisis estadístico completo de los resultados del ICFES 2012.

1. Carga y exploración de los datos

Codificación dataset de ICFES: Para poder importar los datos del dataset del ICFES primero se tiene que mirar que tipo de codificación tiene el archivo .csv para saber como importarlo. Para esto se usa la funcion guess_encoding() del paquete readr, permite detectar la codificación de caracteres de un archivo CSV.

```
library(knitr)
library(dplyr)
library(readr)
guess_encoding("SB11-20121-RGSTRO-CLFCCN-V1-0-txt-csv.csv") %>%
  kable(col.names = c("Codificación", "Confianza"), caption = "Codificaciones dete
```

Table 1: Codificaciones detectadas por `guess_encoding`

Codificación	Confianza
UTF-8	1.00
windows-1252	0.48

Codificación	Confianza
windows-1250	0.27

Como se puede apreciar se obtuvo las codificaciones sugeridas, la confidence indica la confianza del tipo de codificación en este caso el valor es 1.00 el cual es la máxima confianza, por ende el archivo se debe leer con la codificación UTF-8

Importar dataset de ICFES: Para importar los datos del archivo csv al proyecto de R se uso read_csv del paquete readr, para leer los datos y convertirlos en un data frame, se declaro que el tipo de codificación de los caracteres es UTF-8.

```
library(readr)
datos_icfes <- read_csv("SB11-20121-RGSTRO-CLFCCN-V1-0-txt-csv.csv", locale = loca
```

Despues de visualizar los pimeros datos importados en R se obtuvo un error Delimiter: "," indicando que si bien los datos se importaron bien algunos caracteres se importaron un un formato erroneo por ese motivo se creo una segunda variable con el delimitador sep y se elimino el error.Para los analisis se tomo datos_icfes_delimiter para evitar errores con los datos.

```
datos_icfes_delimiter <- read.csv("SB11-20121-RGSTRO-CLFCCN-V1-0-txt-csv.csv", sep
```

¿Cuantas observaciones y variables hay? Para poder ver el numero de observaciones las filas de los datos se usa la función nrow().

Table 2: Cantidad total de filas en el dataset

Número de filas
45390

Para poder ver el numero de variables las columnas se usa la función ncol().

Table 3: Cantidad total de columnas en el dataset

Número de columnas
112

¿Que tipo de variables predominan: Numéricas o categóricas? Para poder determinar que tipos de variables predominan en el conjunto de datos de las notas

del ICFES, se uso la función str() para poder ver la estructura del data frame, esto permite ver el tipo de cada variable y su contenido. Se uso Kable para poder organizar los datos en formato de tabla.

```
#Primeros 20 elementos de los datos
estructura <- capture.output(str(datos_icfes_delimiter[ , 1:20]))
#Se guarda en la variable
estructura_df <- data.frame(estructura)

# Mostrar los datos
estructura_df %>%
  kable(col.names = c("Estructura de Datos"), caption = "Estructura del Dataset (p
```

Table 4: Estructura del Dataset (primeras 20 columnas)

Estructura de Datos
'data.frame': 45390 obs. of 20 variables:
$ PERIODO : int 20121 20121 20121 20121 20121 20121 20121 20121 20121 20121 ...
$ ESTU_TIPO_DOCUMENTO : chr "C" "C" "R" "C" ...
$ ESTU_PAIS_RESIDE : chr "CO" "CO" "CO" "CO" ...
$ ESTU_GENERO : chr "M" "M" "F" "M" ...
$ ESTU_NACIMIENTO_DIA : int 25 5 3 5 9 21 30 22 29 21 ...
$ ESTU_NACIMIENTO_MES : int 8 7 1 9 5 10 10 4 10 12 ...
$ ESTU_NACIMIENTO_ANNO : int 1992 1970 1994 1974 1972 1990 1984 1986 1994 1987 ...
$ ESTU_LIMITA_BAJAVISION : logi NA NA NA NA NA NA ...
$ ESTU_LIMITA_SORDOCEGUERA : chr " " " " " " " " " " ...
$ ESTU_LIMITA_COGNITIVA : chr " " " " " " " " " " ...
$ ESTU_LIMITA_INVIDENTE : chr " " " " " " " " " " ...
$ ESTU_LIMITA_MOTRIZ : chr " " " " " " " " " " ...
$ ESTU_LIMITA_SORDOINTERPRETE : chr " " " " " " " " " " ...
$ ESTU_LIMITA_SORDONOINTERPRETE: chr " " " " " " " " " " ...
$ ESTU_ETNIA : int NA NA 1 NA NA NA NA NA NA NA ...
$ ESTU_COD_RESIDE_MCPIO : int 5837 5837 5837 5837 5837 23001 5837 5837 5837 5837 ...
$ ESTU_RESIDE_MCPIO : chr "TURBO" "TURBO" "TURBO" "TURBO" ...

Estructura de Datos
$ ESTU_RESIDE_DEPTO : chr “ANTIOQUIA” “ANTIOQUIA” “ANTIOQUIA” “ANTIOQUIA” . . .
$ ESTU_ZONA_RESIDE : int 10 10 10 10 10 10 10 10 10 10 . . .
$ ESTU_AREA_RESIDE : int 1 1 1 1 1 1 1 2 1 1 . . .

Utilizando la funcion str() sin limitar la cantidad de datos a mostrar seobtuvieron 45390 filas y 112 columnas .

Ver nombres y tipo de datos de las variables Con la funcion sapply podemos visualizar el nombre de las variables y su tipo de dato, se crea una tabla con dos columnas una con los nombres de las variables y otra con su tipo de dato.

```
library(knitr)

tabla_tipos <- tibble::tibble(
  Variable = names(datos_icfes_delimiter),
  Tipo = sapply(datos_icfes_delimiter, class)
)

kable(tabla_tipos, caption = "Tipos de Variables en el DataFrame")
```

Table 5: Tipos de Variables en el DataFrame

Variable	Tipo
PERIODO	integer
ESTU_TIPO_DOCUMENTO	character
ESTU_PAIS_RESIDE	character
ESTU_GENERO	character
ESTU_NACIMIENTO_DIA	integer
ESTU_NACIMIENTO_MES	integer
ESTU_NACIMIENTO_ANNO	integer
ESTU_LIMITA_BAJAVISION	logical
ESTU_LIMITA_SORDOCEGUERA	character
ESTU_LIMITA_COGNITIVA	character
ESTU_LIMITA_INVIDENTE	character
ESTU_LIMITA_MOTRIZ	character
ESTU_LIMITA_SORDOINTERPRETE	character

Variable	Tipo
ESTU_LIMITA_SORDONOINTERPRETE	character
ESTU_ETNIA	integer
ESTU_COD_RESIDE_MCPIO	integer
ESTU_RESIDE_MCPIO	character
ESTU_RESIDE_DEPTO	character
ESTU_ZONA_RESIDE	integer
ESTU_AREA_RESIDE	integer
IND_COD_ICFES_TERMINO	integer
COLE_COD_ICFES	integer
COLE_COD_DANE_INSTITUCION	numeric
COLE_NOMBRE_SEDE	character
COLE_CALENDARIO	character
COLE_GENERO	character
COLE_NATURALEZA	character
COLE_BILINGUE	character
COLE_JORNADA	character
COLE_CARACTER	character
COLE_VALOR_PENSION	character
ESTU_VECES_ESTADO	character
ESTU_CARRDESEADA_TIPO	integer
ESTU_IES_COD_DESEADA	integer
ESTU_IES_COD_MPIO_DESEADA	integer
ESTU_IES_DEPT_DESEADA	character
ESTU_IES_DESEADA_NOMBRE	character
ESTU_IES_MPIO_DESEADA	character
ESTU_TOTAL_ALUMNOS_CURSO	character
ESTU_ANO_MATRICULA_PRIMERO	integer
ESTU_ANO_TERMINO_QUINTO	integer
ESTU_ANOS_COLEGIO_ACTUAL	integer
ESTU_ANO_MATRICULA_SEXTO	integer
ESTU_ANOS_PREESCOLAR	integer
ESTU_CUANTOS_COLE_ESTUDIO	integer
ESTU_REPROBO_CUARTO	integer
ESTU_REPROBO_DECIMO	integer
ESTU_REPROBO_NOVENO	integer
ESTU_REPROBO_OCTAVO	integer
ESTU_REPROBO_PRIMERO	integer
ESTU_REPROBO_QUINTO	integer
ESTU_REPROBO_SEGUNDO	integer

Variable	Tipo
ESTU_REPROBO_SEPTIMO	integer
ESTU_REPROBO_SEXTO	integer
ESTU_REPROBO_TERCERO	integer
ESTU_REPROBO_ONCE_MAS	integer
ESTU_POR_MEJORARPOSICIONSOCIAL	character
ESTU_POR_COLOMBIAAPRENDE	character
ESTU_POR_INFLUENCIAALGUIEN	character
ESTU_POR_INTERESPERSONAL	character
ESTU_POR_BUSCANDOCARRERA	character
ESTU_POR_TRADICIONFAMILIAR	character
ESTU_POR_ORIENTACIONVOCACIONAL	character
ESTU_RAZON_RETIRO	integer
ESTU_POR_AMIGOSESTUDIANDO	integer
ESTU_POR_COSTOMATRICULA	character
ESTU_POR_OPORTUNIDADES	character
ESTU_POR_OTRARAZON	character
ESTU_PRESTIGIOINSTITUCION	character
ESTU_POR_UBICACION	integer
ESTU_POR_UNICAQUEOFRECE	integer
ESTU_RETIRARSE_COLEGIO	character
ESTU_COD_MCPIO_PRESENTACION	character
ESTU_MCPIO_PRESENTACION	character
ESTU_DEPTO_PRESENTACION	character
ESTU_EXAM_NOMBREEXAMEN	character
FAMI_EDUCA_PADRE	character
FAMI_EDUCA_MADRE	character
FAMI_OCUPA_PADRE	integer
FAMI_OCUPA_MADRE	integer
FAMI_ESTRATO_VIVIENDA	integer
FAMI_NIVEL_SISBEN	integer
FAMI_PERSONAS_HOGAR	integer
FAMI_CUARTOS_HOGAR	integer
FAMI_PISOSHOGAR	integer
FAMI_TELEFONO_FIJO	integer
FAMI_CELULAR	integer
FAMI_INTERNET	integer
FAMI_SERVICIO_TELEVISION	integer
FAMI_COMPUTADOR	integer
FAMI_LAVADORA	integer

Variable	Tipo
FAMI_NEVERA	integer
FAMI_HORNO	integer
FAMI_DVD	integer
FAMI_MICROONDAS	integer
FAMI_AUTOMOVIL	integer
FAMI_INGRESO_FMILIAR_MENSUAL	integer
ESTU_TRABAJA	integer
ESTU_HORAS_TRABAJA	integer
PUNT_LENGUAJE	integer
PUNT_MATEMATICAS	integer
PUNT_C_SOCIALES	integer
PUNT_FILOSOFIA	integer
PUNT_BIOLOGIA	integer
PUNT_QUIMICA	integer
PUNT_FISICA	integer
PUNT_INGLES	integer
DESEMP_INGLES	character
NOMBRE_COMP_FLEXIBLE	character
PUNT_COMP_FLEXIBLE	character
DESEMP_COMP_FLEXIBLE	character
ESTU_PUESTO	character

Análisis de las variables del dataset de ICFES Saber 11 Los tipos de datos encontrados en el dataset de las notas del ICFES son:

Character: Es el tipo de dato usado para texto o cadenas de caracteres.

Integer: Se usa para los números enteros sin decimales.

numeric: Se usa para los números decimales se puede usan para números enteros también.

logical: Se usa para valores booleanos TRUE o FALSE.

Los variables del conjunto de datos son sobre datos del estudiante, residencia del estudiante, información sobre el colegio, información sobre los grados cursados, información familiar e ingresos.

Cantidad de cada tipo de variable Para poder contar cada tipo de variable se uso la función sapply() la cual aplica una función a cada columna del data frame, se le aplico la función class para que tome el tipo de datos de cada variable, lo que

genera es un vector con los datos, con la función table() se cuenta cuantas veces aparece un valor del vector creado con sapply().

```
tabla_tipos <- table(sapply(datos_icfes_delimiter, class))
tabla_tipos
```

```
##
## character   integer   logical   numeric
##        47        63         1         1
```

Los datos obtenidos son con el dataset original, sin haberle hecho ninguna técnica de transformación y limpieza de datos, por ende se van a tomar como datos cualitativos el tipo character y como datos cuantitativos los tipos: integer y numeric. Con este análisis en hay en total 47 variables categóricas o cualitativas y 64 variables cuantitativas o numéricas. Pero en el conjunto hay una variable con un tipo de dato logical.

```
variables_logicas <- names(datos_icfes_delimiter)[sapply(datos_icfes_delimiter, is
variables_logicas
```

```
## [1] "ESTU_LIMITA_BAJAVISION"
```

La variable contiene los siguientes datos:

```
unique(datos_icfes_delimiter$ESTU_LIMITA_BAJAVISION)
```

```
## [1] NA
```

```
all(is.na(datos_icfes_delimiter$ESTU_LIMITA_BAJAVISION))
```

```
## [1] TRUE
```

Con la función unique permite ver cuantos valores distintos hay el resultado fue NA osea esta variable no contiene datos y se uso all para verificar si en toda la variable soy hay datos NA y arrojo TRUE quiere decir que esta variable este campo nunca tuvo datos.

Conclusión ¿Que tipo de variables predominan? Después de conocer y analizar cada variable se puede concluir que hay mas variables de tipo numéricas con 63 variables y 47 de tipo categóricas por ahora se toman como categóricas, la variable logical se descarta ya que al no contar con información no es importante en este análisis de tipo de variables.

2. Limpieza y preparación de datos**

Verificar valores faltantes En el anterior análisis se descubrió que la variable ESTU_LIMITA_BAJAVISION tiene valores faltantes, si se usa la función any con is.na se debe obtener un valor True.

```
any(is.na(datos_icfes_delimiter))
```

```
## [1] TRUE
```

Para contar cuantos Na hay en todo el conjunto de datos se usa la función sum().

```
sum(is.na(datos_icfes_delimiter))
```

```
## [1] 1318206
```

Se encontro un valor considerable de valores NA

Variables con valores NA Para ver los valores faltantes por columna se usa colSums() para poder ver de manera sencilla la cantidad de valores NA de cada variable.

```
#Por columna
na_por_columna <- colSums(is.na(datos_icfes_delimiter))

# columnas con NA
na_filtrados <- na_por_columna[na_por_columna > 0]

# Convertir a data frame
naKable_df <- data.frame(
  Variable = names(na_filtrados),
  Cantidad_NA = as.numeric(na_filtrados)
)

#tabla
knitr::kable(naKable_df, caption = "Cantidad de valores faltantes (NA) por columna
```

Table 6: Cantidad de valores faltantes (NA) por columna

Variable	Cantidad_NA
ESTU_NACIMIENTO_DIA	436
ESTU_NACIMIENTO_MES	436
ESTU_NACIMIENTO_ANNO	436
ESTU_LIMITA_BAJAVISION	45390
ESTU_ETNIA	42344
ESTU_COD_RESIDE_MCPIO	336
ESTU_ZONA_RESIDE	318
ESTU_AREA_RESIDE	39
IND_COD_ICFES_TERMINO	2660
COLE_COD_ICFES	4
COLE_COD_DANE_INSTITUCION	9526
ESTU_CARRDESEADA_TIPO	45268
ESTU_IES_COD_DESEADA	45281
ESTU_IES_COD_MPIO_DESEADA	45285
ESTU_ANO_MATRICULA_PRIMERO	45349
ESTU_ANO_TERMINO_QUINTO	45349
ESTU_ANOS_COLEGIO_ACTUAL	45349
ESTU_ANO_MATRICULA_SEXTO	45349
ESTU_ANOS_PREESCOLAR	45349
ESTU_CUANTOS_COLE_ESTUDIO	45355
ESTU_REPROBO_CUARTO	45349
ESTU_REPROBO_DECIMO	45349
ESTU_REPROBO_NOVENO	45349
ESTU_REPROBO_OCTAVO	45349
ESTU_REPROBO_PRIMERO	45349
ESTU_REPROBO_QUINTO	45349
ESTU_REPROBO_SEGUNDO	45349
ESTU_REPROBO_SEPTIMO	45349
ESTU_REPROBO_SEXTO	45349
ESTU_REPROBO_TERCERO	45349
ESTU_REPROBO_ONCE_MAS	45349
ESTU_RAZON_RETIRO	45380
ESTU_POR_AMIGOSESTUDIANDO	45350
ESTU_POR_UBICACION	45350
ESTU_POR_UNICAQUEOFRECE	45350
FAMI_OCUPA_PADRE	283
FAMI_OCUPA_MADRE	283

Variable	Cantidad_NA
FAMI_ESTRATO_VIVIENDA	2380
FAMI_NIVEL_SISBEN	289
FAMI_PERSONAS_HOGAR	297
FAMI_CUARTOS_HOGAR	41423
FAMI_PISOSHOGAR	354
FAMI_TELEFONO_FIJO	286
FAMI_CELULAR	45
FAMI_INTERNET	280
FAMI_SERVICIO_TELEVISION	282
FAMI_COMPUTADOR	282
FAMI_LAVADORA	282
FAMI_NEVERA	45
FAMI_HORNO	280
FAMI_DVD	282
FAMI_MICROONDAS	282
FAMI_AUTOMOVIL	282
FAMI_INGRESO_FMILIAR_MENSUAL	283
ESTU_TRABAJA	38100
ESTU_HORAS_TRABAJA	41676
PUNT_LENGUAJE	78
PUNT_MATEMATICAS	4

Análisis de variables irrelevantes o vacías Después de mirar que variables contienen valores de tipo Na se analizo un conjunto de variables que para los siguientes análisis del taller no afectan a los resultados del análisis, estas variables son: Las variables de reprobación de grados: Se eliminan porque es una variable que no es relevante para los próximos análisis. Variables a eliminar: ESTU_REPROBO_CUARTO, ESTU_REPROBO_DECIMO, ESTU_REPROBO_NOVENO,ESTU_REPROBO_OCTAVO,ESTU_REPROBO_PRIMERO,ES
Se crea una nueva variable llamada datos_icfes_reprobo que guardara los datos sin tener en cuenta los que se van a descartar para los análisis.

```
library(dplyr)

datos_icfes_reprobo <- datos_icfes_delimiter %>% select(-c(
  ESTU_REPROBO_CUARTO,
    ESTU_REPROBO_DECIMO,
    ESTU_REPROBO_NOVENO,
```

```
    ESTU_REPROBO_OCTAVO,
    ESTU_REPROBO_PRIMERO,
    ESTU_REPROBO_QUINTO,
    ESTU_REPROBO_SEGUNDO,
    ESTU_REPROBO_SEPTIMO,
    ESTU_REPROBO_SEXTO,
    ESTU_REPROBO_TERCERO,
    ESTU_REPROBO_ONCE_MAS
))
```

Variables de Incapacidad: En el conjunto de datos se elimino la variable ESTU_LIMITA_BAJAVISION ya que no es relevante para el análisis del taller se eliminara de la variable datos_icfes_reprobo

```
datos_icfes_incapa <- datos_icfes_reprobo %>% select(-c(
  ESTU_LIMITA_BAJAVISION))
```

Valores inconsistentes o faltantes

ESTU_CUANTOS_COLE_ESTUDIO: Se analizo si hay datos inconsistentes en la cantidad de colegios de un estudiante se valida que si un estudiante tiene registrado mas de 10 colegios ya es un dato dudoso.Se encontraron ceros inconsistencias.

```
library(dplyr)

datos_icfes_incapa %>%
  filter(ESTU_CUANTOS_COLE_ESTUDIO > 10) %>%
  select(ESTU_CUANTOS_COLE_ESTUDIO) %>%
  distinct() %>%
  arrange(desc(ESTU_CUANTOS_COLE_ESTUDIO))
```

```
## [1] ESTU_CUANTOS_COLE_ESTUDIO
## <0 rows> (or 0-length row.names)
```

FAMI_PERSONAS_HOGAR = Se analiza esta variable en donde si hay 0 personas del hogar o mas de 20 seria un dato inconsistente.Se encontraron ceros inconsistencias.

```r
datos_icfes_incapa %>%
  filter(FAMI_PERSONAS_HOGAR == 0 | FAMI_PERSONAS_HOGAR > 20) %>%
  select(FAMI_PERSONAS_HOGAR) %>%
  distinct() %>%
  arrange(FAMI_PERSONAS_HOGAR)
```

```
## [1] FAMI_PERSONAS_HOGAR
## <0 rows> (or 0-length row.names)
```

FAMI_ESTRATO_VIVIENDA: A la hora de analizar el estrado se encontraron inconsistencias con los datos ya que se encontraron estratos mayores del estrato 6

```r
library(dplyr)
library(knitr)

# Tabla de frecuencias del estrato de vivienda (incluye NA)
datos_icfes_incapa %>%
  group_by(FAMI_ESTRATO_VIVIENDA) %>%
  summarise(Frecuencia = n()) %>%
  arrange(desc(Frecuencia)) %>%
  kable(caption = "Distribución de la variable FAMI_ESTRATO_VIVIENDA")
```

Table 7: Distribución de la variable FAMI_ESTRATO_VIVIENDA

FAMI_ESTRATO_VIVIENDA	Frecuencia
2	17500
1	12225
3	10337
NA	2380
4	2040
5	548
6	237
22	46
8	42
26	7
14	5
10	3

FAMI_ESTRATO_VIVIENDA	Frecuencia
17	3
18	3
19	3
21	3
16	2
20	2
7	1
12	1
15	1
23	1

Como solución se decidió por filtrar y limpiar los valores mayores del estrato 6.

```
#  Nuevo dataset, dejando solo estratos válidos
datos_icfes_estrato <- datos_icfes_incapa %>%
  mutate(FAMI_ESTRATO_VIVIENDA = as.character(FAMI_ESTRATO_VIVIENDA)) %>%
  filter(FAMI_ESTRATO_VIVIENDA %in% c("1", "2", "3", "4", "5", "6") | is.na(FAMI_E
  mutate(FAMI_ESTRATO_VIVIENDA = as.numeric(FAMI_ESTRATO_VIVIENDA))

#Ver los estratos
library(dplyr)
library(knitr)

datos_icfes_estrato %>%
  group_by(FAMI_ESTRATO_VIVIENDA) %>%
  summarise(Frecuencia = n()) %>%
  arrange(desc(Frecuencia)) %>%
  kable(caption = "Distribución nueva de la variable FAMI_ESTRATO_VIVIENDA")
```

Table 8: Distribución nueva de la variable FAMI_ESTRATO_VIVIENDA

FAMI_ESTRATO_VIVIENDA	Frecuencia
2	17500
1	12225
3	10337
NA	2380

FAMI_ESTRATO_VIVIENDA	Frecuencia
4	2040
5	548
6	237

Estandarizar columnas con errores estructurales

Imputar datos faltantes

2.1 Nuevas transformaciones

Calcular edad estudiante

Nueva columna edad categorizada

3. Analisis Exploratorio y Univariado