

Práctica 2: Limpieza y Validación de los Datos

Tipología y Ciclo de Vida de los Datos, Universitat Oberta de Catalunya

Abel Romero Búrdalo y Paula Muñoz Lago

03 enero 2021

Contents

1. Descripción del Dataset	2
1.1. Importancia y objetivo del análisis	4
2. Integración y selección de los datos de interés a analizar	4
3. Limpieza de los datos	5
3.1. Tipos de Variables	5
3.2. Gestión de datos inválidos	6
3.3. Identificación y tratamiento de valores extremos	9
3.4. Exportación de los datos preprocesados	10
4. Análisis de los datos	10
4.1. Selección de los grupos de datos a analizar	11
4.2. Comprobación de la normalidad y homogeneidad de la varianza	11
4.3. Aplicación de pruebas estadísticas	15
5. Representación de los resultados a partir de tablas y gráficas	20
6. Conclusiones	23
7. Código	24
COMENTARIO: Para obtener todo el código(https://felixfan.github.io/extract-r-code/)	

1. Descripción del Dataset

Los datasets con los que vamos a realizar la práctica están relacionados con el número de votos por estado de EEUU en las recientes elecciones. Se han obtenido de Kaggle:

- Participación por Estado
- Partido ganador por Estado

Cabe destacar que los datos presentes en el primer dataset tratan únicamente la población con derecho a voto, es decir, únicamente los estadounidenses mayores de 18 años. La primera fila del dataset incluye información de la totalidad del país.

En cuanto a los datos del segundo dataset, el número total de votos totales de republicanos y demócratas no coincide con el total de votos por estado del primer juego de datos. Esto es debido a que no se han contabilizado el número de votos a terceros partidos. Por ese motivo dicho dataset se va a utilizar para sacar el partido ganador en cada estado y se va a añadir al primer juego de datos. Asimismo, este dataset ya ha sido tratado y no es necesario realizar tareas de limpieza, por lo que únicamente se va a utilizar para complementar el primer juego de datos.

El dataset con los datos referentes a las votaciones dispone de 15 columnas:

- **State:** Indica el estado del que trata la fila de datos.
- **Source:** Fuente de datos (url).
- **Official/Unofficial:** Esta columna indica si los datos reportados son una vez el conteo ha alcanzado el 100% (oficial), o si aún no se ha terminado el conteo (unoficial).
- **Total Ballots Counted (Estimate):** número total de votos en dicho estado.
- **Vote for Highest Office:** Votos válidos.
- **VEP Turnout Rate:** Porcentaje de votantes. VEP, en inglés: Voting Eligible Population
- **Voting-Eligible Population:** Población con derecho a voto.
- **Voting-Age Population (VAP):** Población total de estados unidos con 18 años o más, incluyendo a personas sin derecho a voto por razones diferentes a la edad, como personas sin la nacionalidad o criminales de ciertos estados, donde la ley se lo prohíbe. fuente
- **% Non-citizen:** Porcentaje de personas con derecho a voto que no son ciudadanos estadounidenses.
- **Prision:** Número de votantes desde la cárcel.
- **Probation:** Número de criminales con el tercer grado. Es decir, disfrutan de un periodo fuera de la cárcel bajo supervisión.
- **Parole:** Personas con permiso de permanencia temporal en EEUU.
- **Total Ineligible Felon:** Número de personas en dicho estado que no tienen derecho a voto por criminalidad.
- **Overseas Eligible:** Número de estadounidenses viviendo fuera del país, independientemente del estado.
- **State abv:** Abreviatura del estado.

El dataset con los datos referentes a las votaciones a republicanos y demócratas contiene la siguiente información:

- **State:** Indica el estado del que trata la fila de datos.
- **DEM:** Número de votos de los demócratas.
- **REP:** Número de votos de los republicanos.
- **usa_state:** Indica el estado del que trata la fila de dato.
- **usa_state_code:** Abreviatura del estado.
- **percent_democrat:** Porcentaje de votos a los demócratas

Antes de proseguir, cargaremos los datos relativos al primer dataset y realizaremos una breve inspección sobre los mismos (excepto sobre la columna sources, que contiene urls), para estudiar los valores contenidos en cada columna.

```
fileDirectory <- getwd()
csv_usa <- file.path(fileDirectory, '2020 November General Election - Turnout Rates.csv')
usa_elections <- read.csv(csv_usa)
attach(usa_elections)
```

```
head(usa_elections[, -2])
```

```
##           State Official.Unofficial Total.Ballots.Counted..Estimate.
## 1 United States                                     158,835,004
## 2      Alabama           Unofficial           2,306,587
## 3      Alaska                                     367,000
## 4      Arizona                                     3,400,000
## 5      Arkansas           Unofficial           1,212,030
## 6      California           Unofficial          16,800,000
##  Vote.for.Highest.Office..President. VEP.Turnout.Rate
## 1                                     66.4%
## 2                2,297,295           62.6%
## 3                                     69.8%
## 4                                     65.5%
## 5                1,206,697           55.5%
## 6                                     64.7%
##  Voting.Eligible.Population..VEP. Voting.Age.Population..VAP. X..Non.citizen
## 1                239,247,182           257,605,088           7.8%
## 2                3,683,055           3,837,540           2.3%
## 3                525,568           551,117           3.4%
## 4                5,189,000           5,798,473           8.9%
## 5                2,182,375           2,331,171           3.6%
## 6                25,962,648           30,783,255          15.0%
##  Prison Probation Parole Total.Ineligible.Felon Overseas.Eligible
## 1 1,461,074 1,962,811 616,440           3,294,457           4,971,025
## 2   25,898   50,997  10,266           67,782
## 3    4,293    2,074   1,348           6,927
## 4   38,520   76,844   7,536          93,699
## 5   17,510   36,719  24,698          64,974
## 6  104,730     0 102,586          207,316
##  State.Abv
## 1
## 2      AL
## 3      AK
## 4      AZ
## 5      AR
## 6      CA
```

Cargamos el segundo dataset

```
csv_dem_rep<- file.path(fileDirectory, 'democratic_vs_republican_votes_by_usa_state_2020.csv')
usa_winner <- read.csv(csv_dem_rep)
attach(usa_winner)
```

```
head(usa_winner)
```

##	state	DEM	REP	usa_state	usa_state_code	percent_democrat
## 1	Alabama	843473	1434159	Alabama	AL	37.03289
## 2	Alaska	45758	80999	Alaska	AK	36.09899
## 3	Arizona	1643664	1626679	Arizona	AZ	50.25968
## 4	Arkansas	420985	761251	Arkansas	AR	35.60922
## 5	California	9315259	4812735	California	CA	65.93476
## 6	Colorado	1753416	1335253	Colorado	CO	56.76931

1.1. Importancia y objetivo del análisis

Gracias a este dataset podemos estudiar como ha influido el numero de votantes y el porcentaje total de votaciones para que en unos estados u otros hayan ganado los demócratas. También podremos plantear algunas conclusiones sobre las diferencias por estados en cuanto a votos republicanos o demócratas.

Estos análisis son de gran relevancia a la hora de establecer patrones de voto en grupos poblacionales en función de ciertas características, en este caso, según el estado de residencia. Además, disponiendo de conjuntos como este para cada año de elecciones durante un largo periodo de tiempo, podríamos predecir con Machine Learning cuál va a ser el comportamiento de los votantes de cada estado en base a su historia electoral.

2. Integración y selección de los datos de interés a analizar

En vistas a la descripción de las columnas observamos que disponemos de columnas repetidas, como es el caso de la abreviatura del estado y el nombre del mismo. Por ello, la columna relativa al nombre completo del estado será el primero que eliminemos, con el fin de evitar redundancia en los datos.

```
usa_abreviaturas = usa_elections[, ncol(usa_elections)]
usa_elections <- usa_elections[, -ncol(usa_elections)]
usa_elections[,1] <- usa_abreviaturas
```

Proseguiremos con la nueva última columna, “Overseas Eligible”, que se refiere al número de estadounidenses viviendo fuera del país. Ésta columna solo tiene un valor diferente a null, y está relacionado con el dato en la primera fila, correspondiente con la totalidad de estados. Es por ello, que a continuación retiraremos la primera fila y la guardaremos en una variable, para así poder estuadiar los datos por estado, pero manteniendo la información del total por si nos hiciese falta a continuación. Finalmente eliminaremos la columna “Overseas Eligible”, dado que todos sus valores son null.

```
usa_total <- usa_elections[1,]
usa_elections <- usa_elections[2:nrow(usa_elections), -ncol(usa_elections)]
```

La carencia de utilidad de las columnas relativas a la fuente de datos y si se trata de una fuente oficial o no, hacen que también procedamos a eliminarlas del dataset.

```
usa_elections <- usa_elections[, -c(grep("Source", colnames(usa_elections)),
                                     grep("Official.Unofficial", colnames(usa_elections)))]
```

Por último se va a generar la columna `party_winners` a partir de los datos del segundo juego de datos. Para ello se van a comparar las columnas `REP` y `DEM` y se va a sacar el ganador de cada estado.

```
party_winner<- ifelse(DEM>REP, "DEM","REP")
usa_elections$party_winner<-party_winner
```

Una vez añadida la nueva columna al dataset original se va a proceder a realizar el análisis.

3. Limpieza de los datos

En este apartado llevaremos a cabo un proceso de limpieza de datos, comenzando por establecer los tipos de datos correctos para cada variable (columna), y gestionando los valores nulos (casillas vacías). Finalmente, estudiaremos la presencia de valores extremos y cómo tratarlos.

3.1. Tipos de Variables

Cada columna de nuestro dataframe `usa_elections` es un Factor, conteniendo diferentes niveles. En el siguiente resumen de nuestro dataframe podemos observar el tipo de datos correspondiente con cada uno de ellos.

```
str(usa_elections)
```

```
## 'data.frame':   51 obs. of  12 variables:
## $ State                : Factor w/ 52 levels "", "AK", "AL", "AR", ...: 3 2 5 4 6 7 8 10 9
## $ Total.Ballots.Counted..Estimate. : Factor w/ 50 levels "1,212,030", "1,330,000", ...: 15 29 25 1 1
## $ Vote.for.Highest.Office..President.: Factor w/ 25 levels "", "1,206,697", ...: 8 1 1 2 1 1 1 19 1 1
## $ VEP.Turnout.Rate        : Factor w/ 48 levels "55.0%", "55.5%", ...: 10 30 20 2 16 45 33 3
## $ Voting.Eligible.Population..VEP.  : Factor w/ 52 levels "1,007,920", "1,079,434", ...: 24 38 36 13 1
## $ Voting.Age.Population..VAP.       : Factor w/ 52 levels "1,104,489", "1,114,466", ...: 24 38 36 13 1
## $ X..Non.citizen         : Factor w/ 37 levels "0.9%", "1.2%", ...: 12 18 36 20 10 25 32 2
## $ Prison                 : Factor w/ 50 levels "0", "1,461,074", ...: 21 34 33 9 4 14 6 40
## $ Probation              : Factor w/ 30 levels "0", "1,962,811", ...: 24 8 29 15 1 1 1 5 1
## $ Parole                 : Factor w/ 33 levels "0", "1,348", "1,780", ...: 5 2 30 16 7 1 22
## $ Total.Ineligible.Felon : Factor w/ 50 levels "0", "1,679", "10,781", ...: 38 36 49 37 16 1
## $ party_winner           : chr  "REP" "REP" "DEM" "REP" ...
```

Como se aprecia, los datos numéricos se han cargado como strings, por lo que podemos concluir que las únicas columnas que se encuentran en un tipo correcto son la primera y la última, `State` y `party_winner`. Esta última se deberá convertir a un tipo factor, puesto que representa la clase demócrata o republicano. Para el resto de columnas debemos aplicar una limpieza, quitando los caracteres no numéricos y así poder convertirlos al tipo de datos correcto. Además, dichas columnas no queremos que sean de tipo Factor, dado que son variables continuas, la única que mantendremos como tipo Factor será `State`, dado que es una variable discreta.

Para ello definiremos dos funciones, que aplicaremos a las columnas que lo necesiten. Estas funciones eliminarán los caracteres necesarios para a continuación poder convertir los datos a números.

```
# Definición de las funciones
remove_comma <- function(x) gsub(',', '', x)
remove_percent <- function(x) gsub('%', '', x)

# Aplicación de las mismas sobre las columnas apropiadas
usa_elections[,2] <- sapply(usa_elections[,2], remove_comma)
usa_elections[,3] <- sapply(usa_elections[,3], remove_percent)
```

```

usa_elections[,4] <- sapply(usa_elections[,4], remove_percent)
usa_elections[,5] <- sapply(usa_elections[,5], remove_comma)
usa_elections[,6] <- sapply(usa_elections[,6], remove_comma)
usa_elections[,7] <- sapply(usa_elections[,7], remove_percent)
usa_elections[,8] <- sapply(usa_elections[,8], remove_comma)
usa_elections[,9] <- sapply(usa_elections[,9], remove_comma)
usa_elections[,10] <- sapply(usa_elections[,10], remove_comma)
usa_elections[,11] <- sapply(usa_elections[,11], remove_comma)

```

Una vez obtenido el resultado necesario para poder convertir al tipo deseado, ejecutamos las siguientes líneas:

```

usa_elections[,2] <- as.numeric(usa_elections[,2])
usa_elections[,3] <- as.numeric(usa_elections[,3])
usa_elections[,4] <- as.numeric(usa_elections[,4])
usa_elections[,5] <- as.numeric(usa_elections[,5])
usa_elections[,6] <- as.numeric(usa_elections[,6])
usa_elections[,7] <- as.numeric(usa_elections[,7])
usa_elections[,8] <- as.numeric(usa_elections[,8])
usa_elections[,9] <- as.numeric(usa_elections[,9])
usa_elections[,10] <- as.numeric(usa_elections[,10])
usa_elections[,11] <- as.numeric(usa_elections[,11])

```

Acto seguido convertimo a tipo factor la variable cualitativa party_winner.

```

usa_elections[,12] <- as.factor(usa_elections[,12])

```

Finalmente, imprimiremos el resumen de las columnas de nuestro dataset para comprobar que todo se ha transformado correctamente.

```

str(usa_elections)

```

```

## 'data.frame':   51 obs. of  12 variables:
## $ State                : Factor w/ 52 levels "", "AK", "AL", "AR", ...: 3 2 5 4 6 7 8 10 9
## $ Total.Ballots.Counted..Estimate. : num  2306587 367000 3400000 1212030 16800000 ...
## $ Vote.for.Highest.Office..President.: num  2297295 NA NA 1206697 NA ...
## $ VEP.Turnout.Rate        : num  62.6 69.8 65.5 55.5 64.7 76.4 71.1 70.5 64.7 71.7 ...
## $ Voting.Eligible.Population..VEP. : num  3683055 525568 5189000 2182375 25962648 ...
## $ Voting.Age.Population..VAP.      : num  3837540 551117 5798473 2331171 30783255 ...
## $ X..Non.citizen         : num  2.3 3.4 8.9 3.6 15 5.7 7.7 5.8 7.1 10.1 ...
## $ Prison                 : num  25898 4293 38520 17510 104730 ...
## $ Probation              : num  50997 2074 76844 36719 0 ...
## $ Parole                 : num  10266 1348 7536 24698 102586 ...
## $ Total.Ineligible.Felon : num  67782 6927 93699 64974 207316 ...
## $ party_winner           : Factor w/ 2 levels "DEM", "REP": 2 2 1 2 1 1 1 1 1 2 ...

```

3.2. Gestión de datos inválidos

Para comprobar qué columnas contienen datos ‘vacíos’ y poder proceder a trabajar con ellas, utilizaremos la función colSums, que aplica una función a todas las columnas de un dataframe y después aplica una suma.

```
colSums(is.na(usa_elections))
```

```
##           State      Total.Ballots.Counted..Estimate.
##           0                                           0
## Vote.for.Highest.Office..President.      VEP.Turnout.Rate
##           27                                           0
##   Voting.Eligible.Population..VEP.      Voting.Age.Population..VAP.
##           0                                           0
##           X..Non.citizen                        Prison
##           0                                           0
##           Probation                        Parole
##           0                                           0
##           Total.Ineligible.Felon      party_winner
##           0                                           0
```

Como vemos, únicamente disponemos de una columna con datos vacíos, Vote for Highest Office President. Este campo indica el número de votos válidos para la presidencia. Al no disponer de dicha información se ha decidido calcular la media de votos totales que si disponen de la información de votos a la presidencia y la media de la columna votos a la presidencia. Una vez obtenidas ambas medias se va a calcular el porcentaje medio de votos que han sido válidos para la presidencia y se van a extrapolar al conjunto de datos vacíos. Es decir, se va a multiplicar el porcentaje de votos válidos al total de votos en aquellos estados en que dicho campo este vacío.

```
# Calculamos la media de los votos válidos para la presidencia
mean_president=mean(usa_elections$Vote.for.Highest.Office..President.,na.rm=TRUE)

# Calculamos la media de votos totales que tienen información sobre los votos a la presidencia
mean_total=mean(
  usa_elections$
    Total.Ballots.Counted..Estimate.
    [!is.na(usa_elections$Vote.for.Highest.Office..President.)]
  , na.rm=TRUE)

# Sacamos el porcentaje de la media de votos válidos
percentage_votes= mean_president/mean_total

# Aplicamos dicho porcentaje a los votos totales que no disponen dicha información
#y guardamos los votos válidos en su correspondiente estado
# Como los votos deben de ser un numero entero se va a redondear el resultado de
#multiplicar los votos totales por el porcentaje de votos válidos
usa_elections$Vote.for.Highest.Office..President.<-
  ifelse(is.na(usa_elections$Vote.for.Highest.Office..President.),
    trunc(usa_elections$Total.Ballots.Counted..Estimate. * percentage_votes),
    usa_elections$Vote.for.Highest.Office..President. )
```

Comprobamos que ya no existan valores NA con el siguiente código:

```
colSums(is.na(usa_elections))
```

```
##           State      Total.Ballots.Counted..Estimate.
##           0                                           0
## Vote.for.Highest.Office..President.      VEP.Turnout.Rate
```

```
##                                0                                0
##   Voting.Eligible.Population..VEP.   Voting.Age.Population..VAP.
##                                0                                0
##               X..Non.citizen                Prison
##                                0                                0
##               Probation                Parole
##                                0                                0
##               Total.Ineligible.Felon                party_winner
##                                0                                0
```

Pese a que ya no encontramos valores NA, destacamos algunas columnas cuantitativas en las que aparecen valores igual a 0.

```
sapply(usa_elections, function(r) any(c(0) %in% r))
```

```
##                                State   Total.Ballots.Counted..Estimate.
##                                FALSE                                FALSE
##   Vote.for.Highest.Office..President.   VEP.Turnout.Rate
##                                FALSE                                FALSE
##   Voting.Eligible.Population..VEP.   Voting.Age.Population..VAP.
##                                FALSE                                FALSE
##               X..Non.citizen                Prison
##                                FALSE                                TRUE
##               Probation                Parole
##                                TRUE                TRUE
##               Total.Ineligible.Felon                party_winner
##                                TRUE                FALSE
```

Se ha investigado si en dichos estados existe una prisión, y la información encontrada ha sido que sí existe en la mayoría de ellos. Es una excepción el caso de ‘District of Columbia’, en el que tenemos valor 0 para todos los datos relacionados con presidiarios, ya que no dispone de una cárcel, y los ciudadanos de dicho distrito se delegan a otras cárceles. Sin embargo, consideramos que pese a que sus ciudadanos cumplan condena en otro estado, debería verse reflejado el dato. Es por ello que aplicaremos la misma gestión de datos inválidos que anteriormente con las columnas Prison, Probation, Parole y Total.Ineligible.Felon. Es decir, según la media de votantes totales en otros estados donde sí que tenemos datos con respecto a cada una de estas variables carcelarias, obtendremos un porcentaje y rellenaremos así los datos con actual valor 0.

```
# Dado que vamos a ejecutar estas líneas de código en diferentes ocasiones, crearemos una función.
replace_0 <- function(column_index) {
  mean_column <- mean(usa_elections[,column_index])

  mean_total = mean(
    usa_elections$Total.Ballots.Counted..Estimate.[!is.na(usa_elections[,column_index])]
  )

  percentage_votes = mean_column/mean_total

  final_column <- ifelse(
    usa_elections[,column_index] == 0,
    trunc(usa_elections$Total.Ballots.Counted..Estimate. * percentage_votes),
    usa_elections[,column_index])

  return(final_column)
```



```

}

usa_elections$Prison <- replace_0(which(colnames(usa_elections) == "Prison"))
usa_elections$Probation <- replace_0(which(colnames(usa_elections) == "Probation"))
usa_elections$Parole <- replace_0(which(colnames(usa_elections) == "Parole"))
usa_elections$Total.Ineligible.Felon <- replace_0(which(colnames(usa_elections) == "Total.Ineligible.Fe

```

3.3. Identificación y tratamiento de valores extremos

Los valores extremos o outliers son aquellos que parecen no ser congruentes si los comparamos con el resto de los datos. Para identificarlos, podemos hacer uso de dos vías: (1) representar un diagrama de caja por cada variable y ver qué valores distan mucho del rango intercuartílico (la caja) o (2) utilizar la función `boxplots.stats()` de R, la cual se emplea a continuación. Así, se mostrarán sólo los valores atípicos para aquellas variables que los contienen:

```
boxplot.stats(usa_elections$Total.Ballots.Counted..Estimate.)$out
```

```
## [1] 16800000 11150000 8930000 11300000
```

```
boxplot.stats(usa_elections$Vote.for.Highest.Office..President.)$out
```

```
## [1] 16674275 11066557 8863171 11231799
```

```
boxplot.stats(usa_elections$VEP.Turnout.Rate)$out
```

```
## numeric(0)
```

```
boxplot.stats(usa_elections$Voting.Eligible.Population..VEP.)$out
```

```
## [1] 25962648 15551739 13670596 18784280
```

```
boxplot.stats(usa_elections$Voting.Age.Population..VAP.)$out
```

```
## [1] 30783255 17543341 15372655 22058260
```

```
boxplot.stats(usa_elections$X..Non.citizen)$out
```

```
## [1] 15
```

```
boxplot.stats(usa_elections$Prison)$out
```

```
## [1] 104730 91674 154913
```

```
boxplot.stats(usa_elections$Probation)$out
```

```
## [1] 205881 205033 416771 368167
```

```
boxplot.stats(usa_elections$Parole)$out
```

```
## [1] 102586 45192 109213
```

```
boxplot.stats(usa_elections$Total.Ineligible.Felon)$out
```

```
## [1] 207316 223139 329754 492390
```

Como se puede observar, existen entre 3 y 4 valores extremos superiores para prácticamente cada variable. Esto es debido a que dichos datos pertenecen a los 4 estados con más población de estados unidos que son California, Texas, Florida y Nueva York. Por lo tanto se tratan de valores extremos legítimos que no deben ser tratados ya que a más población se espera que hayan más número de votos, población con derecho a voto, etc. Asimismo algunos de estos datos, como el número total de votos en California, se han comprobado en las fuentes facilitadas en el juego de datos original y se han confirmado que son valores probables.

3.4. Exportación de los datos preprocesados

Una vez que hemos acometido sobre el conjunto de datos inicial los procedimientos de integración, validación y limpieza anteriores, procedemos a guardar estos en un nuevo fichero denominado 2020 November General Election - Turnout Rates_data_clean.csv:

```
write.csv(usa_elections,  
          "2020 November General Election - Turnout Rates_data_clean.csv",  
          row.names = FALSE)
```

4. Análisis de los datos

Gracias al tratamiento de los datos como numéricos en el punto 3.1, podemos ejecutar pequeños análisis estadísticos, en los que observar la distribución de los datos.

```
summary(usa_elections)
```

```
##      State      Total.Ballots.Counted..Estimate.  
## AK       : 1  Min.      : 278503  
## AL       : 1  1st Qu.: 867500  
## AR       : 1  Median : 2155000  
## AZ       : 1  Mean     : 3114412  
## CA       : 1  3rd Qu.: 3912500  
## CO       : 1  Max.      :16800000  
## (Other):45  
## Vote.for.Highest.Office..President. VEP.Turnout.Rate  
## Min.      : 276765          Min.      :55.00  
## 1st Qu.: 860411          1st Qu.:64.35  
## Median : 2147395          Median :67.60  
## Mean     : 3091105          Mean     :67.77  
## 3rd Qu.: 3883220          3rd Qu.:72.60  
## Max.      :16674275          Max.      :79.90  
##  
## Voting.Eligible.Population..VEP. Voting.Age.Population..VAP. X..Non.citizen
```

```
## Min.      : 431364           Min.      : 447915           Min.      : 0.900
## 1st Qu.: 1338126           1st Qu.: 1403390           1st Qu.: 3.000
## Median : 3312250           Median : 3479257           Median : 4.300
## Mean    : 4600645           Mean    : 5051080           Mean     : 5.443
## 3rd Qu.: 5313422           3rd Qu.: 5934260           3rd Qu.: 7.400
## Max.    :25962648           Max.    :30783255           Max.     :15.000
##
##      Prison      Probation      Parole      Total.Ineligible.Felon
## Min.      : 1679      Min.      : 2074      Min.      : 350      Min.      : 1679
## 1st Qu.: 6418      1st Qu.: 13133      1st Qu.: 2680      1st Qu.: 13097
## Median : 18099      Median : 36719      Median : 7381      Median : 33933
## Mean     : 23954      Mean     : 57286      Mean     : 13236      Mean     : 57924
## 3rd Qu.: 31208      3rd Qu.: 65619      3rd Qu.: 15136      3rd Qu.: 73060
## Max.     :154913      Max.     :416771      Max.     :109213      Max.     :492390
##
## party_winner
## DEM:26
## REP:25
##
##
##
##
```

4.1. Selección de los grupos de datos a analizar

El primer análisis que se va a realizar es comprobar la correlación entre las variables cuantitativas que han tenido mayor incidencia en el porcentaje de participación que ha habido por estado.

En segundo lugar, una vez se haya comprendido la incidencia entre estas variables, se va a analizar si la media del porcentaje de participación ha sido mayor en los estados en que han ganado los demócratas o los republicanos. Para ello se van a agrupar los datos según el partido ganador

```
# Agrupación por ganadores
usa_elections.dem=usa_elections[usa_elections$party_winner=="DEM",]
usa_elections.rep=usa_elections[usa_elections$party_winner=="REP",]
```

Finalmente se va a realizar una regresión lineal para predecir, según los datos que presenten una mayor dependencia, que partido ganaría en cada estado, cuál va a ser el porcentaje de votación en cada estado y una regresión logística para predecir el partido ganador según el estado.

4.2. Comprobación de la normalidad y homogeneidad de la varianza

Para la comprobación de que los valores que toman nuestras variables cuantitativas provienen de una población distribuida normalmente, utilizaremos las pruebas de normalidad de Anderson-Darling y Shapiro-Wilk. Así, se comprueba que para cada prueba se obtiene un p-valor superior al nivel de significación prefijado $\alpha = 0,05$. Si esto se cumple, entonces se considera que la variable en cuestión sigue una distribución normal.

```
library(nortest)
alpha = 0.05
col.names = colnames(usa_elections)
```

```

for (i in 1:ncol(usa_elections)) {
  if (i == 1) cat("Variables que no siguen una distribución normal
                  segun el test de Anderson-Darling:\n")
  if (is.integer(usa_elections[,i]) | is.numeric(usa_elections[,i])) {
    p_val = ad.test(usa_elections[,i])$p.value
    if (p_val < alpha) {
      cat(col.names[i])
      # Format output
      if (i < ncol(usa_elections) - 1) cat(", ")
      if (i %% 3 == 0) cat("\n")
    }
  }
}

```

```

## Variables que no siguen una distribución normal
##          segun el test de Anderson-Darling:
## Total.Ballots.Counted..Estimate., Vote.for.Highest.Office..President.,
## Voting.Eligible.Population..VEP., Voting.Age.Population..VAP.,
## X..Non.citizen, Prison, Probation,
## Parole, Total.Ineligible.Felon

```

```

library(nortest)
alpha = 0.05
col.names = colnames(usa_elections)
for (i in 1:ncol(usa_elections)) {
  if (i == 1) cat("Variables que no siguen una distribución normal
                  segun el test de Shapiro-Wilk:\n")
  if (is.integer(usa_elections[,i]) | is.numeric(usa_elections[,i])) {
    p_val = shapiro.test(usa_elections[,i])$p.value
    if (p_val < alpha) {
      cat(col.names[i])
      # Format output
      if (i < ncol(usa_elections) - 1) cat(", ")
      if (i %% 3 == 0) cat("\n")
    }
  }
}

```

```

## Variables que no siguen una distribución normal
##          segun el test de Shapiro-Wilk:
## Total.Ballots.Counted..Estimate., Vote.for.Highest.Office..President.,
## Voting.Eligible.Population..VEP., Voting.Age.Population..VAP.,
## X..Non.citizen, Prison, Probation,
## Parole, Total.Ineligible.Felon

```

Como se puede observar, en ambos casos se han obtenido los mismos resultados, por lo que podemos concluir que las variables devueltas en ambos casos no siguen una distribución normal con un 95% de confianza. En cualquier caso, para el contraste de hipótesis que se va a realizar en la aplicación de pruebas estadísticas nos interesa saber si la variable VEP.Turnout.Rate sigue una distribución normal según al grupo que pertenezca del partido ganador. Para averiguarlo se van a utilizar los grupos anteriormente creados.

En primer lugar vamos a mostrar el histograma para el partido demócrata y el republicano con su densidad de probabilidad:

```

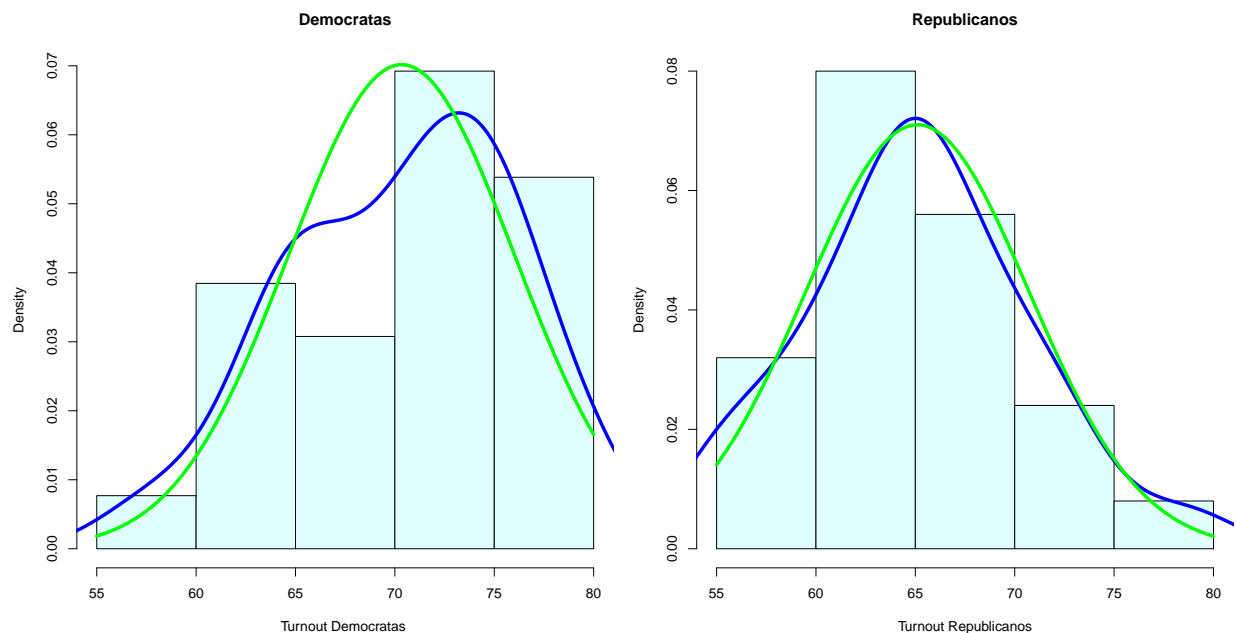
# Creamos las variables turnout en función del partido ganador
turnout_dem <- usa_elections.dem$VEP.Turnout.Rate
turnout_rep <- usa_elections.rep$VEP.Turnout.Rate

par(mfrow=c(1,2), mar=c(4,4,4,1), oma=c(0.5,0.5,0.5,0))

# histograma, densidad de probabilidad y normal calculada para turnout
hist(turnout_dem, col = 'lightcyan',
     main = 'Democratas',
     freq = FALSE,
     xlab = 'Turnout Democratas ',
     pch=16)
lines(density(turnout_dem),
     col = 'blue',
     lwd='4')
curve(dnorm(x,mean(turnout_dem), sd(turnout_dem)),col='green', lwd=4, add=T)

hist(turnout_rep, col = 'lightcyan',
     main = 'Republicanos',
     freq = FALSE,
     xlab = 'Turnout Republicanos',
     pch=16)
lines(density(turnout_rep),
     col = 'blue',
     lwd='4')
curve(dnorm(x,mean(turnout_rep), sd(turnout_rep)),col='green', lwd=4, add=T)

```



Observando ambas gráficas se puede observar que en ambos casos se sigue una distribución fiel a la curva de la normal, aunque era de esperar porque cuando se ha analizado en conjunto en los test anteriores la variable VEP.Turnout.Rate seguía una distribución normal. Para terminar se va a realizar el test de Saphiro-Wilk para confirmar su normalidad:

```
shapiro.test(turnout_dem)
```

```
##  
## Shapiro-Wilk normality test  
##  
## data: turnout_dem  
## W = 0.9684, p-value = 0.5823
```

```
shapiro.test(turnout_rep)
```

```
##  
## Shapiro-Wilk normality test  
##  
## data: turnout_rep  
## W = 0.98091, p-value = 0.9025
```

Para ambos caso el p-value es superior al nivel de significación 0.05, por lo que se puede asumir la normalidad de ambos grupos.

En cuanto a la varianza, se desea analizar la varianza relativa a los ganadores en cada estado en función del porcentaje de participación de la población. Como se ha observado anteriormente la variable VEP.Turnout.Rate sigue una distribución normal, por lo que para analizar la varianza se van a utilizar dos test, el de Fligner-Killeen y la función `var.test()` de R.

```
fligner.test(VEP.Turnout.Rate ~ party_winner, data = usa_elections)
```

```
##  
## Fligner-Killeen test of homogeneity of variances  
##  
## data: VEP.Turnout.Rate by party_winner  
## Fligner-Killeen:med chi-squared = 0.19411, df = 1, p-value = 0.6595
```

```
var.test(turnout_dem,turnout_rep)
```

```
##  
## F test to compare two variances  
##  
## data: turnout_dem and turnout_rep  
## F = 1.0242, num df = 25, denom df = 24, p-value = 0.9555  
## alternative hypothesis: true ratio of variances is not equal to 1  
## 95 percent confidence interval:  
## 0.4536965 2.2964389  
## sample estimates:  
## ratio of variances  
## 1.02418
```

En ambos caso el p-valor es superior al nivel de significación 0.05, por lo que se acepta la hipótesis nula de homocedasticidad y se concluye que la variable VEP.Turnout.Rate presenta varianzas estadísticamente iguales para los dos grupos de `party_winner`.

4.3. Aplicación de pruebas estadísticas

4.3.1. ¿Qué variables cuantitativas han influido más en el porcentaje de participación en las elecciones?

En primer lugar, procedemos a realizar un análisis de correlación entre las distintas variables para determinar cuáles de ellas ejercen una mayor influencia sobre el partido ganador en cada estado. Para ello, se utilizará el coeficiente de correlación de Spearman, puesto que hemos visto que tenemos datos que no siguen una distribución normal.

```
corr_matrix <- matrix(nc = 2, nr = 0)
colnames(corr_matrix) <- c("estimate", "p-value")
# Calcular el coeficiente de correlación para cada variable cuantitativa
# con respecto al campo "party_winner"
for (i in 2:(ncol(usa_elections) - 1)) {
  if (i!=4){
    if (is.integer(usa_elections[,i]) | is.numeric(usa_elections[,i])) {
      spearman_test = cor.test(usa_elections[,i], usa_elections[,4], method = "spearman", exact=FALSE)
      corr_coef = spearman_test$estimate
      p_val = spearman_test$p.value
      # Add row to matrix
      pair = matrix(ncol = 2, nrow = 1)
      pair[1][1] = corr_coef
      pair[2][1] = p_val
      corr_matrix <- rbind(corr_matrix, pair)
      rownames(corr_matrix)[nrow(corr_matrix)] <- colnames(usa_elections)[i]
    }
  }
  else{
    next()
  }
}
```

```
print(corr_matrix)
```

##	estimate	p-value
## Total.Ballots.Counted..Estimate.	0.200796488	0.1576969
## Vote.for.Highest.Office..President.	0.201470755	0.1562772
## Voting.Eligible.Population..VEP.	0.116526383	0.4154651
## Voting.Age.Population..VAP.	0.111322292	0.4367382
## X..Non.citizen	0.003668147	0.9796191
## Prison	-0.076070233	0.5957292
## Probation	0.128204548	0.3699466
## Parole	-0.003122525	0.9826502
## Total.Ineligible.Felon	-0.057471265	0.6887269

Como se puede observar, la correlación existente entre las distintas variables cuantitativas y el porcentaje de participación en las elecciones no es muy elevado. Asimismo, los p-valores obtenidos han sido mayores al nivel de significación, por lo que se acepta la hipótesis nula que confirma que no existe una gran correlación entre estas variables y la participación (turnout).

4.3.2. ¿Es el valor del porcentaje de participación en las elecciones superior en aquellos estados en el que han ganado los demócratas?

La segunda prueba estadística que se aplicará consistirá en un contraste de hipótesis sobre dos muestras para determinar si el valor del porcentaje de participación en las elecciones es superior dependiendo del partido que ha ganado. Para ello, tendremos dos muestras: la primera de ellas se corresponderá con los valores de la proporción de participación en aquellos estados donde han ganado los demócratas y, la segunda, con aquellos donde han ganado los republicanos. Tal y como se ha comprobado anteriormente, nuestras muestras siguen una distribución normal y presentan una desviación estándar igual, por lo que se puede aplicar un contraste de hipótesis utilizando un test paramétrico. En este caso las medias seguirán una distribución t de Student con $n_1 + n_2 - 2$ grados de libertad. Por lo tanto, el contraste de hipótesis quedaría de la siguiente forma:

Hipótesis nula:

$$H_0 : \mu_1 = \mu_2$$

Hipótesis alternativa:

$$H_1 : \mu_1 > \mu_2$$

Siendo μ_1 la media poblacional del porcentaje de participación donde han ganado los demócratas y μ_2 donde han ganado los republicanos.

Tal y como se ha comentado anteriormente se va a utilizar un test paramétrico utilizando la función `t.test()` de R, donde se le especificará que se tratan de varianzas iguales.

```
t.test(turnout_dem,turnout_rep,alternative="greater", var.equal=TRUE)
```

```
##
## Two Sample t-test
##
## data: turnout_dem and turnout_rep
## t = 3.3063, df = 49, p-value = 0.0008873
## alternative hypothesis: true difference in means is greater than 0
## 95 percent confidence interval:
##  2.58016      Inf
## sample estimates:
## mean of x mean of y
##  70.33846  65.10400
```

Como se puede observar el p-value obtenido ha sido inferior al nivel de significación $\alpha = 0.05$, por lo tanto se rechaza la hipótesis nula en favor de la alternativa y se puede afirmar con un 95% de confianza que el porcentaje de participación en las elecciones americanas ha sido superior en aquellos estados donde han ganado los demócratas, es decir que en los estados donde ganó Joe Biden hubo mayor incidencia de participación con un 95% de confianza.

4.3.3. Regresión Lineal

Tal y como se planteó en los objetivos de la actividad, resultará de mucho interés poder realizar predicciones sobre cual podría ser el porcentaje de participación en las siguientes elecciones. Así, se calculará un modelo de regresión lineal utilizando regresores tanto cuantitativos como cualitativos con el que poder realizar las predicciones de los porcentajes. Para obtener un modelo de regresión lineal considerablemente eficiente,

lo que haremos será obtener varios modelos de regresión. Como variable cualitativa únicamente se pasará el partido ganador. Así, de entre todos los modelos que tengamos, escogeremos el mejor utilizando como criterio aquel que presente un mayor coeficiente de determinación (R^2). Cabe destacar que las correlaciones obtenidas anteriormente no han sido muy elevadas, pero aun así se va intentar construir un modelo que permita predecir el valor del porcentaje de participación según un estado en concreto.

Para ello, se van a dividir los datos en datos de entrenamiento y de test. En este caso, para el test se va a separar un estado y que posteriormente va a ser utilizado para ver si con el algoritmo generado somos capaces de predecir el porcentaje de participación en dicho estado. El estado elegido ha sido el de Minnesota (MN).

```
# Generamos los datos de test y de train
test <- usa_elections[usa_elections$State == 'MN', ]
train <- usa_elections[usa_elections$State != 'MN', ]

votos = train$Total.Ballots.Counted..Estimate.
votos_validos = train$Vote.for.Highest.Office..President.
vep= train$Voting.Eligible.Population..VEP.
vap = train$Voting.Age.Population..VAP.
no_ciudadanos = train$X..Non.citizen
prision = train$Prison
probation=train$Probation
parole=train$Parole
felon = train$Total.Ineligible.Felon

# Regresores cualitativos
winner=train$party_winner

# Variable a predecir
turnout = train$VEP.Turnout.Rate

# Generación de varios modelos
modelo1 <- lm(turnout ~ votos_validos + parole + vep + felon, data = train)
#
modelo2 <- lm(turnout ~ winner + votos_validos + parole + vep + felon , data = train)
#
modelo3 <- lm(turnout ~ winner + votos + vep + probation +parole , data = train)
#
modelo4 <- lm(turnout ~ winner + votos + votos_validos + vep + prision , data = train)
#
modelo5 <- lm(turnout ~ winner + votos + parole +probation + prision+ felon , data = train)
```

Para los anteriores modelos de regresión lineal múltiple obtenidos, podemos utilizar el coeficiente de determinación para medir la bondad de los ajustes y quedarnos con aquel modelo que mejor coeficiente presente.

```
# Tabla con los coeficientes de determinación de cada modelo
tabla.coeficientes <- matrix(c(1, summary(modelo1)$r.squared,
2, summary(modelo2)$r.squared,
3, summary(modelo3)$r.squared,
4, summary(modelo4)$r.squared,
5, summary(modelo5)$r.squared),
ncol = 2, byrow = TRUE)
colnames(tabla.coeficientes) <- c("Modelo", "R^2")
tabla.coeficientes
```

```
##      Modelo      R^2
## [1,]      1 0.6127646
## [2,]      2 0.6317208
## [3,]      3 0.6356666
## [4,]      4 0.5656377
## [5,]      5 0.2919869
```

Como se puede observar, el modelo que mejor se ha ajustado ha sido el tercero puesto que su coeficiente de determinación ha sido el más elevado. Ahora, empleando este modelo, podemos proceder a realizar un predicción de participación en las elecciones en Minnesota.

```
# Guardamos los valores de los regresores del test
lineardata <- data.frame(
winner= test$party_winner,
votos = test$Total.Ballots.Counted..Estimate.,
vep = test$Voting.Eligible.Population..VEP.,
probation = test$Probation,
parole = test$Parole
)

# Predecir el turnout
turout_predicted <- predict(modelo3, lineardata)

cat("El porcentaje de participación que hubo en minessota es: ", test$VEP.Turnout.Rate,
    "\nfrente al porcetaje predecido por nuestro modelo, que ha resultado: ", turout_predicted)
```

```
## El porcentaje de participación que hubo en minessota es: 79.9
## frente al porcetaje predecido por nuestro modelo, que ha resultado: 77.40711
```

4.3.4. Regresión logística

La siguiente regresión se va a realizar con el objetivo de predecir la variable dicatómica `party_winner`. La estrategia a seguir en este caso va a ser generar distintos modelos con las mismas variables que para la regresión lineal añadiendo la variable `turnout`. En este caso, al no disponer del coeficiente de determinación, se van a evaluar las matrices de confusión de cada modelo y se va a calcular la precisión de cada modelo con el objetivo de seleccionar aquel que mejor realice la predicción de que partido ganaría.

```
# Generación de varios modelos
# Modelo 1
modelo1 <- glm(as.factor(winner) ~ votos + vep + probation +parole +turnout,
               data =train,family=binomial(link=logit))

# Modelo 2
modelo2 <- glm(as.factor(winner) ~ turnout + parole +probation,
               data = train, family=binomial(link=logit))

# Modelo 3
modelo3 <- glm(as.factor(winner) ~ votos_validos +probation + parole + vap ,
               data = train, family=binomial(link=logit))

# Modelo 4
modelo4 <- glm(as.factor(winner) ~ votos_validos + probation + felon+prision ,
               data = train, family=binomial(link=logit))

# Modelo 5
```

```
modelo5 <- glm(as.factor(winner) ~ turnout + votos + vep +vap+ felon,
              data = train, family=binomial(link=logit))
```

Definimos las funciones `calc_cmatrix` y `get_precision` para calcular la matriz de confusión y calcular la precisión del modelo

```
# Función para calcular la matriz de confusión de un modelo en concreto
calc_cmatrix<- function(model){
  pdata<-predict(model,type="response")
# Generamos un vector en el que si la predicción es superior
# a 0.5 se clasifica como REP y en caso contrario como DEM
estimatedResponses=ifelse(pdata>0.5,"REP","DEM")
# Guardamos en trueResponse, los resultados que se esperan de la variable winner
trueResponse=winner
# Generamos la matriz de confusión
return(table(estimatedResponses,trueResponse))
}

# Calculamos la precisión del modelo
get_precision <- function(table) {
  df <- as.data.frame(table)
  true_DEM <- df[1, "Freq"]
  true_REP <- df[4, "Freq"]
  return((true_DEM + true_REP) / 51)
}
```

Modelo 1:

```
# Calculamos la precisión del modelo 1
table_results <- calc_cmatrix(modelo1)
get_precision(table_results)
```

```
## [1] 0.745098
```

Modelo 2:

```
# Calculamos la precisión del modelo 2
table_results <- calc_cmatrix(modelo2)
get_precision(table_results)
```

```
## [1] 0.7058824
```

Modelo 3:

```
# Calculamos la precisión del modelo 3
table_results <- calc_cmatrix(modelo3)
get_precision(table_results)
```

```
## [1] 0.6666667
```

Modelo 4:

```
# Calculamos la precisión del modelo 5
table_results <- calc_cmatrix(modelo4)
get_precision(table_results)
```

```
## [1] 0.627451
```

Modelo 5:

```
# Calculamos la precisión del modelo 5
table_results <- calc_cmatrix(modelo5)
get_precision(table_results)
```

```
## [1] 0.7058824
```

El modelo que mejor precisión ha presentado ha sido el primero, por lo que se va a aplicar una predicción sobre los datos de test con dicho modelo para ver si es capaz de predecir que estado ganó en Minnesota.

```
logdata <- data.frame(
  votos = test$Total.Ballots.Counted..Estimate.,
  vep = test$Voting.Eligible.Population..VEP.,
  probation = test$Probation,
  parole=test$Parole,
  turnout=test$VEP.Turnout.Rate)

# Predecir el partido ganador
predicted_winner=ifelse(predict(modelo1, logdata, type="response")>0.5,"REP","DEM")

cat("Ganador en Minesotta", levels(test$party_winner)[test$party_winner],
    "\nGanador predecido en dicho estado por nuestro modelo: ", predicted_winner)
```

```
## Ganador en Minesotta DEM
## Ganador predecido en dicho estado por nuestro modelo: DEM
```

Como se puede observar, el algoritmo ha sido capaz de predecir correctamente el estado ganador en Minnesota.

5. Representación de los resultados a partir de tablas y gráficas

Para la visualización de los datos se utilizará también el lenguaje Python, dado que ofrece la representación de mapas. Este código ha sido ejecutado en un Jupyter Notebook y los resultados pueden verse a continuación.

```
import plotly.express as ex

for col in usa_elections.columns[1:]:
    vep_turnout_rate = usa_elections.loc[:,col]

    fig = ex.choropleth(locations=usa_elections.State,
                        locationmode="USA-states",
```

```

color= vep_turnout_rate,
scope="usa",
color_continuous_scale=ex.colors.diverging.Portland)

fig.update_layout(title = col)
fig.show()

```

Estudiamos primero los mapas en los que se muestra el número de votos por cada estado, así como el número de votos válidos. Se observa que es en California donde más votantes han acudido a las urnas, pero esto puede ser por la densidad poblacional. El número total de votos válidos (Vote for Highest Office President) no se mostrará, dado que al mostrarse solo, sin la relación con el número total de votos, carece de significado y el gráfico obtenido sería el mismo.

Total.Ballots.Counted..Estimate.

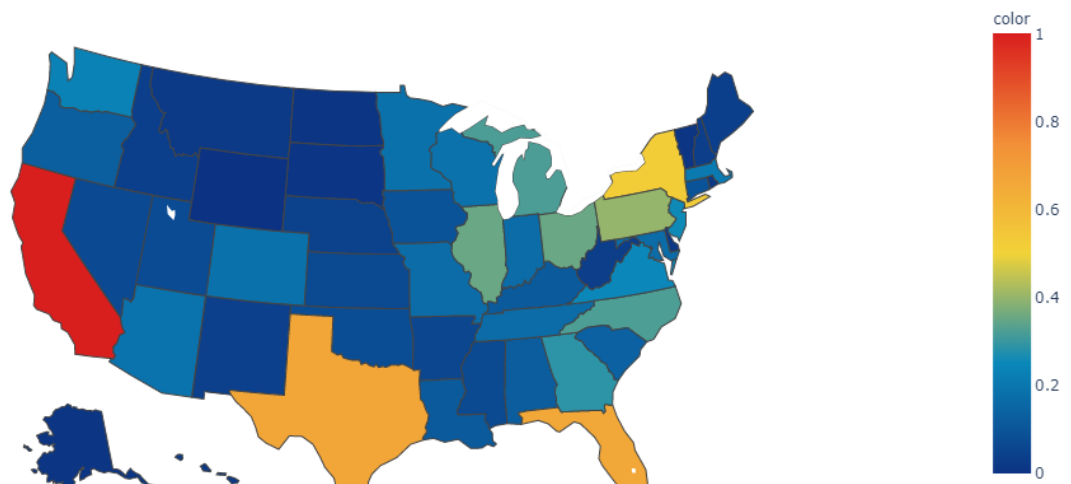


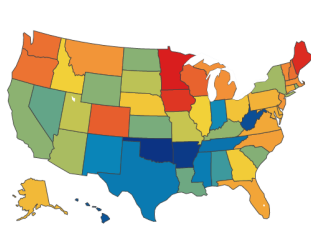
Figure 1: Número total de votos

Observamos en los gráficos siguientes que pese a que el gráfico anterior nos indicaba que era en California donde más votos se habían registrado, no todos los ciudadanos con derecho a voto de dicho estado han ejercido su derecho, dado que la participación no es la más alta, siendo mucho mayor en Maine, Minnesota e Iowa. El gráfico relacionado con las personas que tienen derecho a voto en cambio muestra un resultado muy similar al de la totalidad de los votos, por lo que la densidad poblacional vuelve a destacar en estas visualizaciones.

Las siguientes visualizaciones muestran la totalidad de personas edad para votar, incluyendo a las personas a las que se les ha revocado el derecho ya sea por criminalidad u otras causas a la izquierda, y el número de personas sin la nacionalidad estadounidense habitando en cada estado. El primer gráfico no muestra mucha diferencia con los anteriores, en los que no se incluyen las personas sin derechos, ya que este conjunto de personas es muy reducido. El segundo gráfico muestra que los estados en los que más inmigración hay son California y Texas.

El número de personas que han mandado su voto desde la cárcel muestra que han sido muchos más votantes desde Texas, seguidos de California y Florida. De la misma forma, el número de personas que han votado mientras disfrutaban del tercer grado destaca en los mismos estados, aunque se hace especialmente relevante en Georgia. Estos datos pueden darse tanto por una mayor densidad poblacional, que se convertiría en un

VEP Turnout Rate Por cada Estado



Voting.Eligible.Population..VEP.

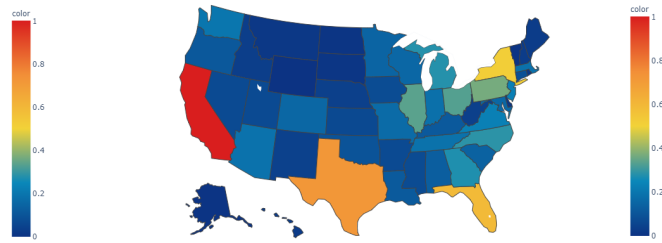
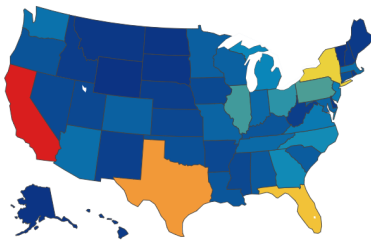


Figure 2: Porcentaje de votantes y Personas con edad para votar en todo el país y tienen derecho a voto

Voting.Age.Population..VAP.



X..Non.citizen

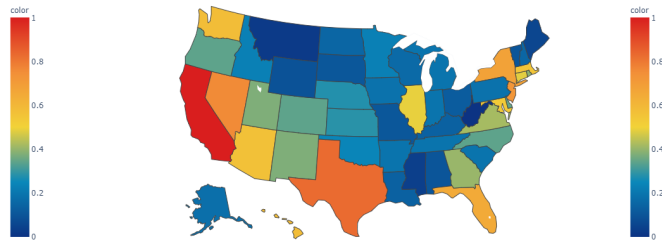
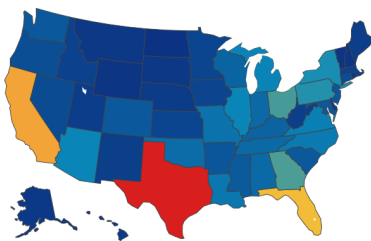


Figure 3: Personas con edad para votar en todo el país, incluyendo personas a las que se les ha revocado el derecho y Personas que no tienen la nacionalidad estadounidense

mayor número de criminales, como por unas leyes mucho más estrictas en dichos estados, que se resolvería en un mayor número de encarcelados.

Prison



Probation

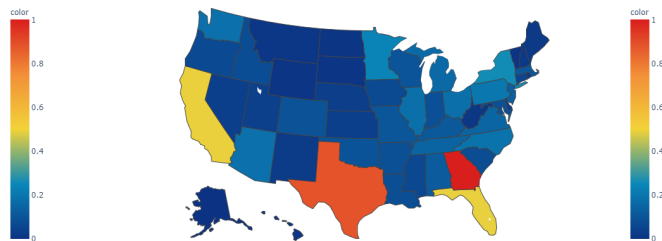


Figure 4: Personas que se encuentran internos en una prisión y votan desde dicha institución y personas con permiso de tercer grado

Los siguientes gráficos nos muestran cuantos visitantes tiene cada estado, siendo esto el número de personas con permiso de permanencia temporal, variable en la que siguen destacando California y Texas, seguidas de Nueva York. A continuación observamos que las personas que no tienen derecho a voto a causa de su criminalidad es más alta en Texas que en el resto de estados. Esta variable sería casi la totalidad (ya que podría haber otras causas) de la diferencia entre VAP y VEP, mostradas anteriormente.

Finalmente, mostramos sobre el mapa de EEUU los estados en los que ha resultado ganador cada partido. Siendo el Republicano en el que se presentaba el ex presidente Donald Trump, y el Demócrata en el que se

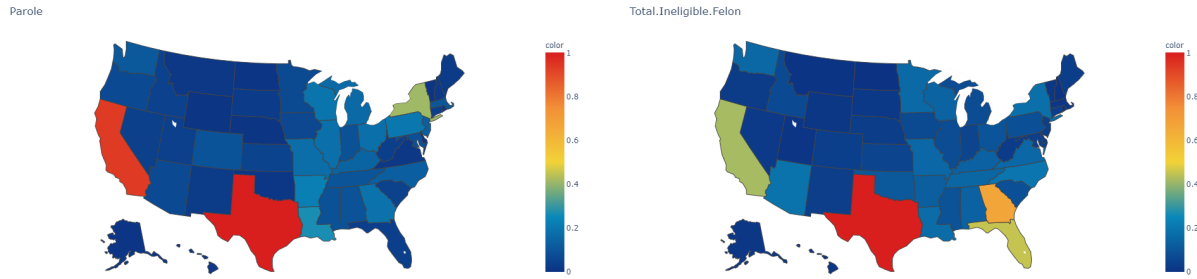


Figure 5: Personas con permiso de permanencia temporal y Personas que no tienen derecho a voto a causa del crimen que han cometido

presentaba Joe Biden.

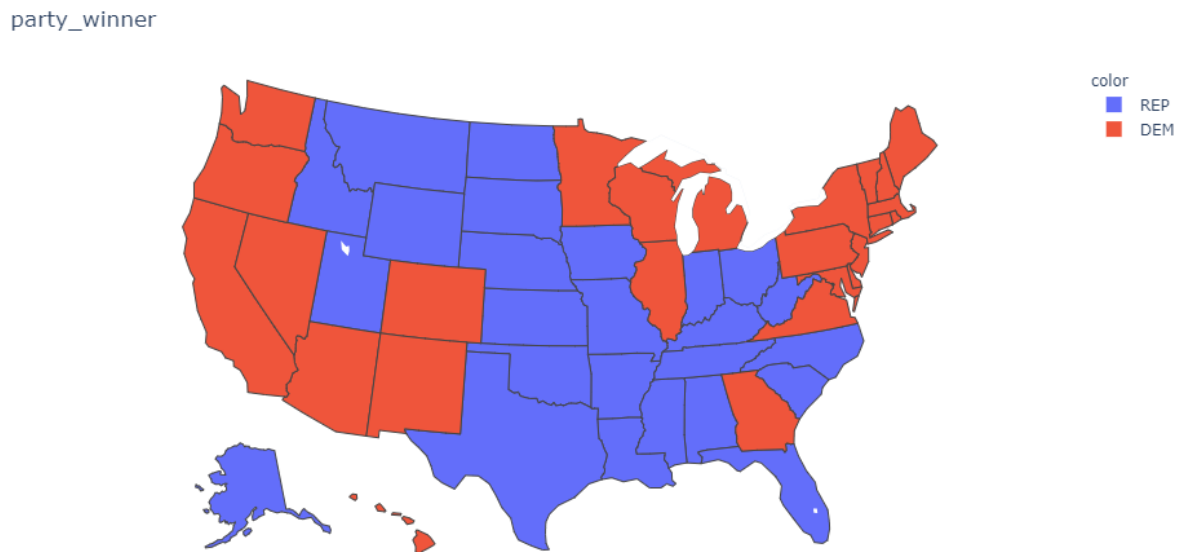


Figure 6: Partido más votado en cada estado

6. Conclusiones

Con este estudio hemos comprobado cuál es la utilidad de realizar un cuidado preprocesamiento de los datos, para así evitar diferencias en los tipos de datos, incongruencias con outliers o valores nulos (NA, cero, etc). De la misma forma, hemos visto cómo aplicar pruebas estadísticas arroja luz sobre las diferentes características de los datos con los que trabajamos, como puede ser la distribución o la correlación entre ellos. Finalmente, la realización de contraste de hipótesis nos ha llevado a sacar ciertas conclusiones sobre los datos. Hemos planteado diferentes preguntas sobre los datos, concluyendo en que no existe una gran correlación entre las variables cuantitativas estudiadas y el porcentaje de participación. Sin embargo, sí que hemos demostrado una relación entre el porcentaje de participación y la victoria demócrata. Las regresiones realizadas nos han indicado cuál es la tendencia electoral en EEUU, y dejan abierta una oportunidad de estudio de estos

datos en vistas a las próximas elecciones. La regresión lineal múltiple aplicada utilizando todas las variables cuantitativas ha resultado en un modelo con una confianza del 64%, mientras que la regresión logística ha resultado mucho mejor para esta tarea, ofreciendo una confianza del 74% a la hora de predecir el partido ganador en cada estado.

7. Código

El código se encuentra disponible en el siguiente repositorio de GitHub (<https://github.com/paulamlago/USA2020GeneralElection-DataCleaning>)