Instituto Tecnológico Autónomo de México

Propedéutico

Análisis estadístico de los decesos de COVID19 en México

Marco Antonio Ramos Juárez

142244

Rodrigo

142244

Ciudad de México 2020

Contents

Introducción

Hasta el siglo XXI, los seres humanos han sido testigos de tres pandemias mortales: SARS síndrome respiratorio de Oriente Medio (MERS) y COVID-19. Todos estos virus, que son responsables de causar infecciones agudas del tracto respiratorio (IRA), son de naturaleza altamente contagiosa y / o han causado una alta mortalidad. COVID-19 apareció por primera vez en Wuhan, China, en diciembre de 2019 y se extendió rápidamente por todo el mundo.(1)

Al 02 de julio de 2021, México es el cuarto país con mayor número de defunciones acumuladas, estando por debajo de Estados Unidos, Brasil e India. De acuerdo a un informe de Statista: Estados Unidos encabeza la clasificación al superar los 620,600 decesos, seguido de Brasil con alrededor de 520.200. Para el 2 julio de 2021, había más de 183.5millones de casos confirmados de COVID-19 en todo el mundo.(2) Es importante recalcar que día con día el número de personas fallecidas en el mundo van aumentando, por lo cual, resulta imperativo seguir investigando más sobre las causas de muerte y tomar acciones preventivas para evitar defunciones alrededor del mundo.

En el contexto de una crisis sanitaria como la que estamos viviendo, es fundamental que los análisis relacionados se hagan con rigor de tal manera que los hallazgos sean veraces y precisos. Esto es de especial importancia en el contexto de la divulgación de información falsa o de desinformación en redes sociales y medios de comunicación. Por ello, el proposito de esta practica final es el anlizar las bases de datos oficiales de contagios y decesos de COVID19 y aplicar los métodos vistos durante la clase con el fin de obtener hallazgos robustos, interesantes y útiles relacionados con el desarrollo de la pandemia en México.

Datos

Extracción

Los datos con los que trabajaremos provienen de la base de datos agregada oficial del Gobierno Federal. Esta base de datos incluye todos los casos diarios asociados a COVID-19 que el gobierno federal ha registrado. Cuenta con contenido desagregado por sexo, edad, nacionalidad, padecimientos asociados entre otras variables.

```
data <- read.csv("data.csv")
```

Limpieza

En primer lugar, analizamos la clase de cada variable:

```
lapply(data, class)
```

```
## $X.1
## [1] "integer"
##
## $X
## [1] "integer"
##
## $fecha_actualizacion
## [1] "character"
##
## $id_registro
## [1] "character"
##
## $origen
## [1] "character"
##
## $sector
## [1] "character"
##
## $entidad_um
## [1] "character"
##
## $sexo
## [1] "character"
##
## $entidad_nac
## [1] "character"
```

```
##
## $entidad_res
## [1] "character"
##
## $municipio_res
## [1] "character"
##
## $tipo_paciente
## [1] "character"
##
## $fecha_ingreso
## [1] "character"
##
## $fecha_sintomas
## [1] "character"
##
## $fecha_def
## [1] "character"
##
## $intubado
## [1] "character"
##
## $neumonia
## [1] "character"
##
## $edad
## [1] "integer"
##
## $nacionalidad
## [1] "character"
##
## $embarazo
## [1] "character"
##
## $habla_lengua_indig
## [1] "character"
##
## $indigena
## [1] "character"
##
```

```
## $diabetes
## [1] "character"
##
## $epoc
## [1] "character"
##
## $asma
## [1] "character"
##
## $inmusupr
## [1] "character"
##
## $hipertension
## [1] "character"
##
## $otra_com
## [1] "character"
##
## $cardiovascular
## [1] "character"
##
## $obesidad
## [1] "character"
##
## $renal_cronica
## [1] "character"
##
## $tabaquismo
## [1] "character"
##
## $otro_caso
## [1] "character"
##
## $toma_muestra_lab
## [1] "character"
##
## $resultado_lab
## [1] "character"
##
## $toma_muestra_antigeno
```

```
## [1] "character"
##
## $resultado_antigeno
## [1] "character"
##
## $clasificacion_final
## [1] "character"
##
## $migrante
## [1] "character"
##
## $pais_nacionalidad
## [1] "character"
##
## $pais_origen
## [1] "character"
##
## $uci
## [1] "character"
```

Notamos que todas las variables pertenecen a la clase *character*, lo cual habla de que debemos clasificarlas de manera adecuada para aprovechar la información por lo que si queremos aprovechar esas variables, el siguiente paso es asignarles la clase correcta.

```
# notamos que el formato de las fechas es AAAA/MM/DD
data$fecha_def <- as.Date(data$fecha_def)
data$fecha_sintomas <- as.Date(data$fecha_sintomas)
data$fecha_ingreso <- as.Date(data$fecha_ingreso)

# creamos un dataframe con las variables relacionadas con
# caracteristicas de las personas
df8 <- data %>% select(sexo, tipo_paciente, intubado, fecha_def,
    neumonia, edad, diabetes, epoc, asma, inmusupr, hipertension,
    obesidad, renal_cronica, tabaquismo)
# df8$sexo<-ifelse(df8$sexo=='MUJER',1,0) #Si es mujer tomará
# valor de 1, 0 en caso de ser hombre
df8$diabetes <- ifelse(df8$diabetes == "NO", 0, 1)
df8$epoc <- ifelse(df8$epoc == "NO", 0, 1)
df8$asma <- ifelse(df8$asma == "NO", 0, 1)
df8$inmusupr <- ifelse(df8$inmusupr == "NO", 0, 1)
df8$hipertension <- ifelse(df8$hipertension == "NO", 0, 1)
```

Table 1. Valores faltantes

Variable	Porcentaje
sector	0.0012109
entidad_res	85.4328979
municipio_res	85.4329316
fecha_def	97.9585041
pais_origen	0.1057210

```
df8$obesidad <- ifelse(df8$obesidad == "NO", 0, 1)
df8$renal_cronica <- ifelse(df8$renal_cronica == "NO", 0, 1)
df8$tabaquismo <- ifelse(df8$tabaquismo == "NO", 0, 1)
```

Missing values

En segundo lugar, checamos la cantidad de valores faltantes de cada variable con almenos un NA.

```
missing_data <- sapply(data, function(x) sum(length(which(is.na(x)))))

options(scipen = 999)  #quito notación científica
missing_data <- data.frame(missing_data)
missing_data <- add_rownames(missing_data, var = "variable")
```

```
## Warning: `add_rownames()` was deprecated in dplyr 1.0.0.
## Please use `tibble::rownames_to_column()` instead.
```

```
# imprimo solo las variables con nas en porcentaje
output_missing_data <- missing_data %>% filter(missing_data >
    0) %>% mutate(missing_data = missing_data * 100/nrow(data))

kable(output_missing_data, booktabs = T, align = "c", col.names = c("Variable",
    "Porcentaje"), caption = "Valores faltantes") %>% kable_styling(position = "center",
    latex_options = "repeat_header")
```

Lo primero que notamos es que las variables *entidad_res* y *municipio_res*, referentes a la entidad y municpio de residencia cuentan con una alta proporción de valores faltantes (la mayoría de sus valores está faltante). De igual manera, la fecha de defunción también cuenta con una alta proporción, auqnue en este caso será menos problemático pues podenmos hacer un supuesto bastante creible de que solo tienen valor aquellas personas que han perecido. En cuanto a sector y país de origen, los valores faltantes realmente son muy pocos y podemos prescindir de dichas observaciones.

Creación de variables de interés

```
## time to hospital
data <- data %>% mutate(tiempo_enf_to_hosp = fecha_ingreso -
    fecha_sintomas)
data$tiempo_enf_to_hosp <- as.numeric(data$tiempo_enf_to_hosp)

## time to hospital and death
data <- data %>% mutate(tiempo_to_death = fecha_def - fecha_ingreso)
data$tiempo_to_death <- as.numeric(data$tiempo_to_death)

## time from infection to death
data <- data %>% mutate(tiempo_inf_death = fecha_def - fecha_sintomas)
data$tiempo_inf_death <- as.numeric(data$tiempo_inf_death)
```

Conclusiones

De la extracción y limpieza podemos concluir que:

- No podemos aprovechar de la mejor manera los datos referentes a la localización geográfica pues son muchísimos *missing values* y una estrategia de imputación correcta va más allá de los objetivos y alcances del curso.
- La mayoría de las variables con categóricas y las observaciones son a nivel individuo.
- Convertimos las variables de fecha y categóricas al formato correcto

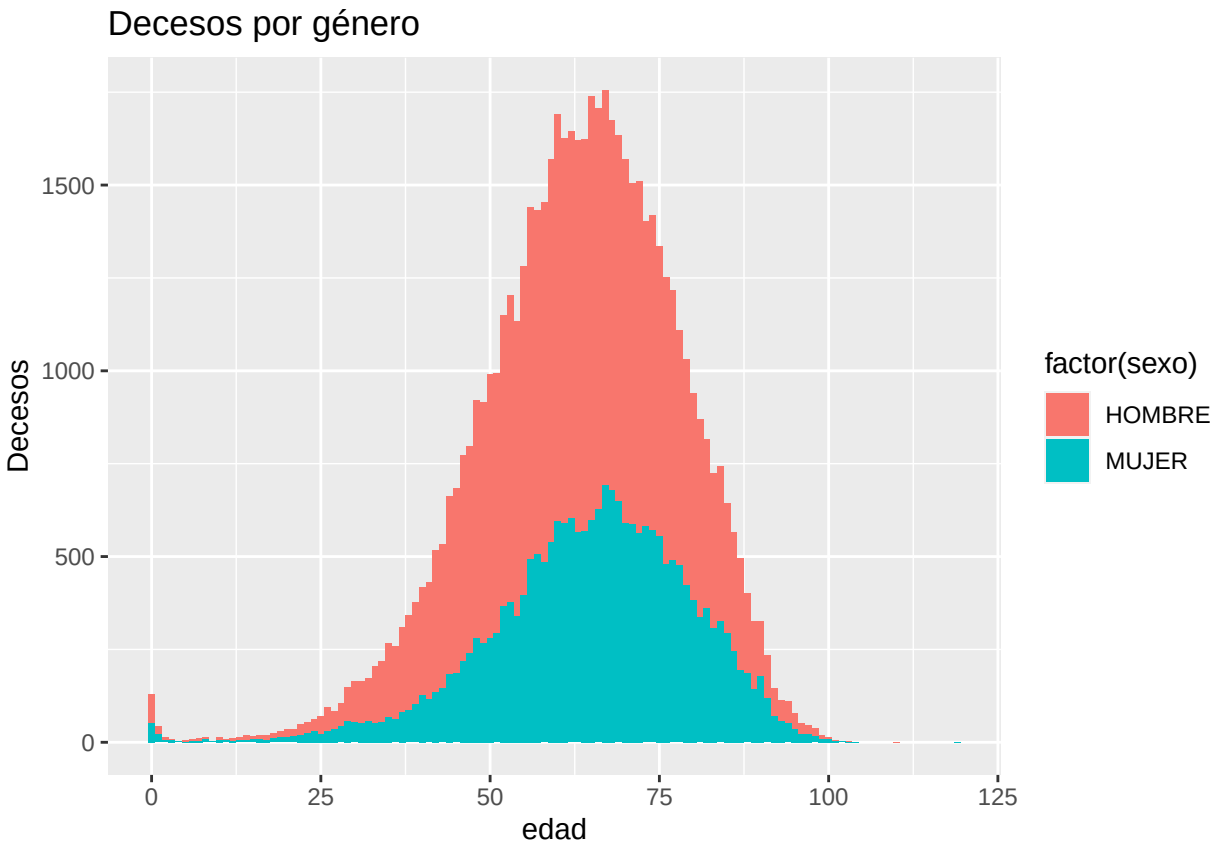
EDA

En esta parte realizaremos un análisis exploratorio de las principales variables de interés.

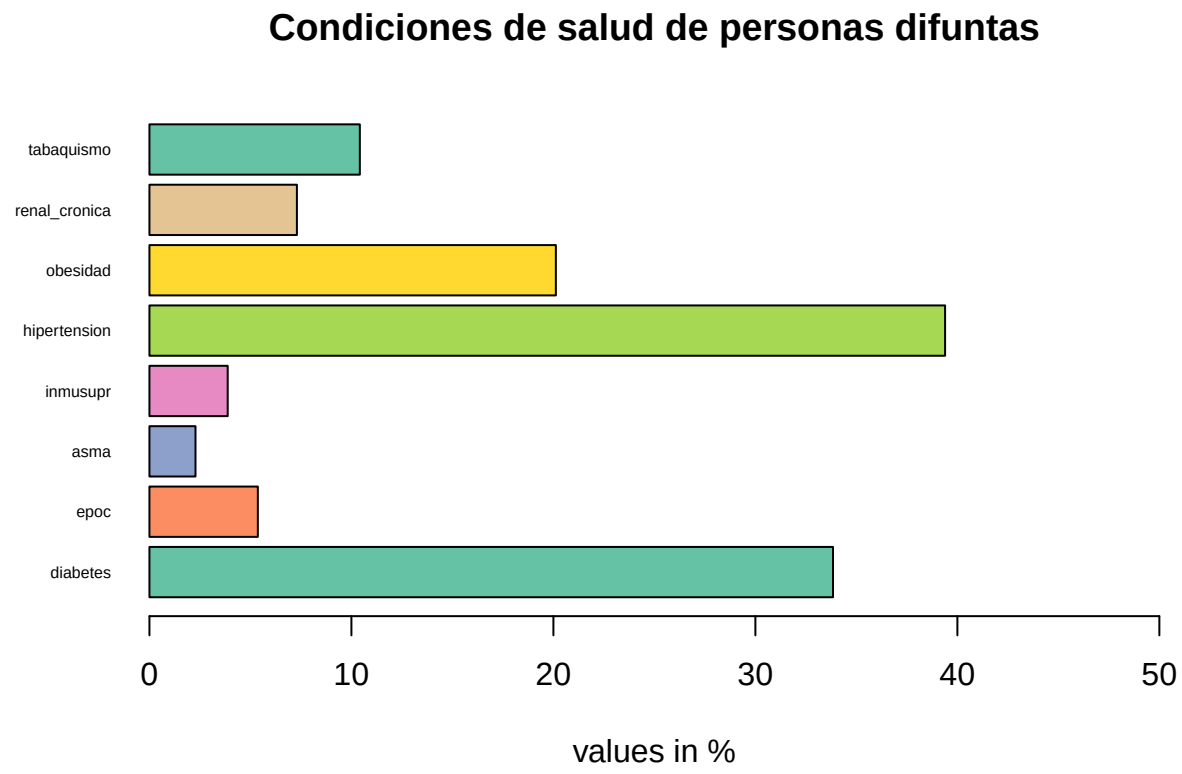
```
df8$fecha_def <- format(df8$fecha_def, format = "%V")
df8["def"] <- 1
df9 <- subset(df8, fecha_def != "NA")
df10 <- df9 %>% group_by(sexo, edad) %>% summarise(sum_def = sum(def,
    na.rm = TRUE))
```

```
## `summarise()` has grouped output by 'sexo'. You can override using the `.groups` argument.
```

```
ggplot(df10, aes(x = edad, y = sum_def, fill = factor(sexo))) +
    geom_bar(stat = "identity") + ggtitle("Decesos por género") +
    ylab("Decesos")
```




```
condiciones<-c(sum(df9$diabetes),sum(df9$epoc),sum(df9$asma),sum(df9$inmusupr),sum(df9$hiperte
porc_condiciones<-condiciones*100/sum(df9$def)
nom_cond<-c('diabetes','epoc','asma','inmusupr','hipertension','obesidad','renal_cronica','tab

data2 <- data.frame(
  name=nom_cond ,
  value=porc_condiciones
  )
coul <- brewer.pal(7, "Set2")
barplot(height=data2$value, names=data2$name,
        col=coul,
        xlab="values in %",
        main="Condiciones de salud de personas difuntas",
        xlim=c(0,50),
        horiz=TRUE,las=1,cex.names=.5

        )
```

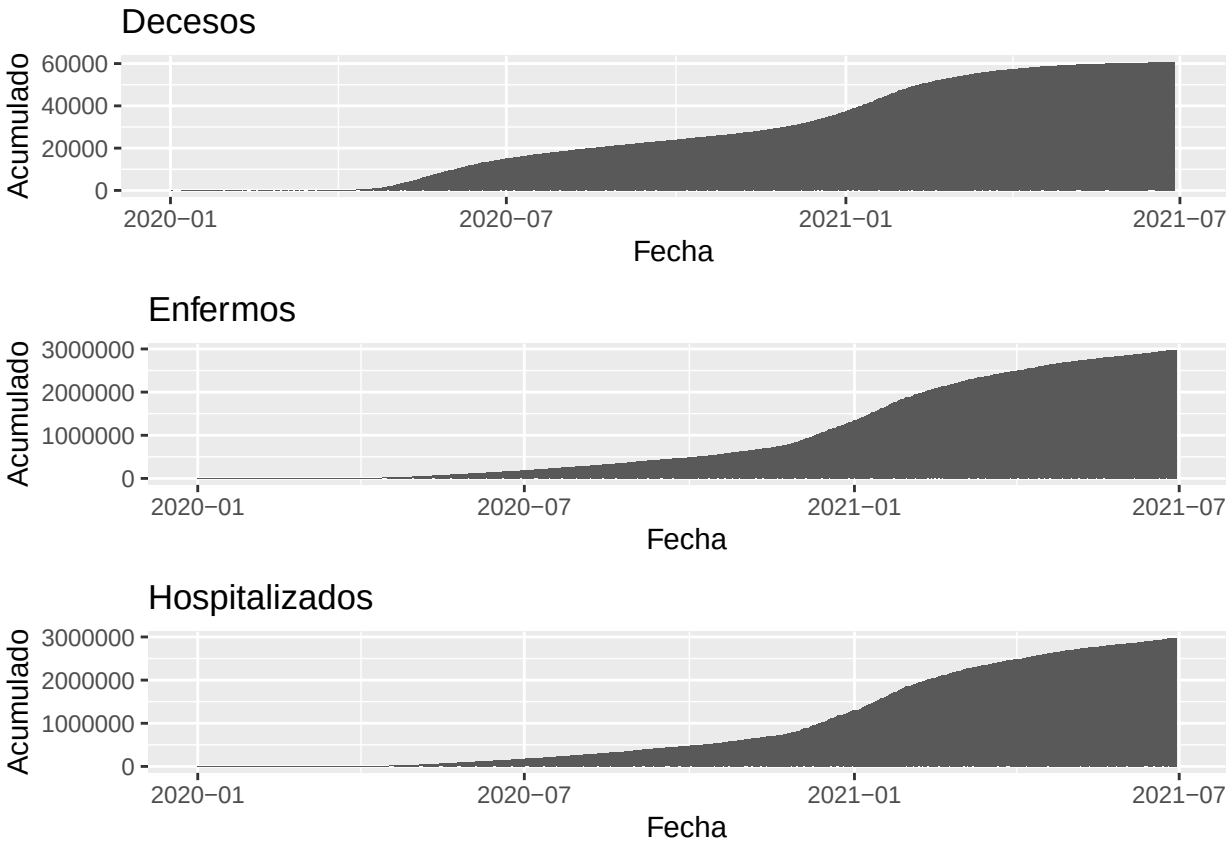


Estadística descriptiva

```
sintomas <- data %>% group_by(fecha_sintomas) %>% summarise(cantidad = n()) %>%
    mutate(cum = cumsum(cantidad))

ingreso <- data %>% group_by(fecha_ingreso) %>% summarise(cantidad = n()) %>%
    mutate(cum = cumsum(cantidad))

muerte <- data %>% group_by(fecha_def) %>% summarise(cantidad = n()) %>%
    mutate(cum = cumsum(cantidad))

a <- ggplot(data = muerte, aes(x = fecha_def, y = cum)) + geom_bar(stat = "identity") +
    ggtitle("Decesos") + xlab("Fecha") + ylab("Acumulado")

b <- ggplot(data = sintomas, aes(x = fecha_sintomas, y = cum)) +
    geom_bar(stat = "identity") + ggtitle("Enfermos") + xlab("Fecha") +
    ylab("Acumulado")

c <- ggplot(data = ingreso, aes(x = fecha_ingreso, y = cum)) +
```

```
    geom_bar(stat = "identity") + ggtitle("Hospitalizados") +
    xlab("Fecha") + ylab("Acumulado")

grid.arrange(a, b, c, ncol = 1)
```

```
## Warning: Removed 1 rows containing missing values (position_stack).
```




```
year_def <- format(data$fecha_def, format = "%Y")  #Obtenemos los años de las fechas
week_def <- format(data$fecha_def, format = "%V")  #Obtenemos las semanas de las fechas
defun <- data.frame(year_def, week_def, def = data$fecha_def)  #creamos un data frame con los
df2 <- subset(defun, year_def != "NA")  # eliminamos los datos en los cuales no hubo defuncion
df2["def"] <- 1  #Agregamos una columna de 1's para hacer la suma de defunciones por semana
tibble_df2 <- as_tibble(df2)
df3 <- df2 %>% group_by(year_def, week_def) %>% summarize(sum_def = sum(def,
    na.rm = TRUE))  #df3 contiene la suma de defunciones por semana
```

```
## `summarise()` has grouped output by 'year_def'. You can override using the `.groups` argumen
```

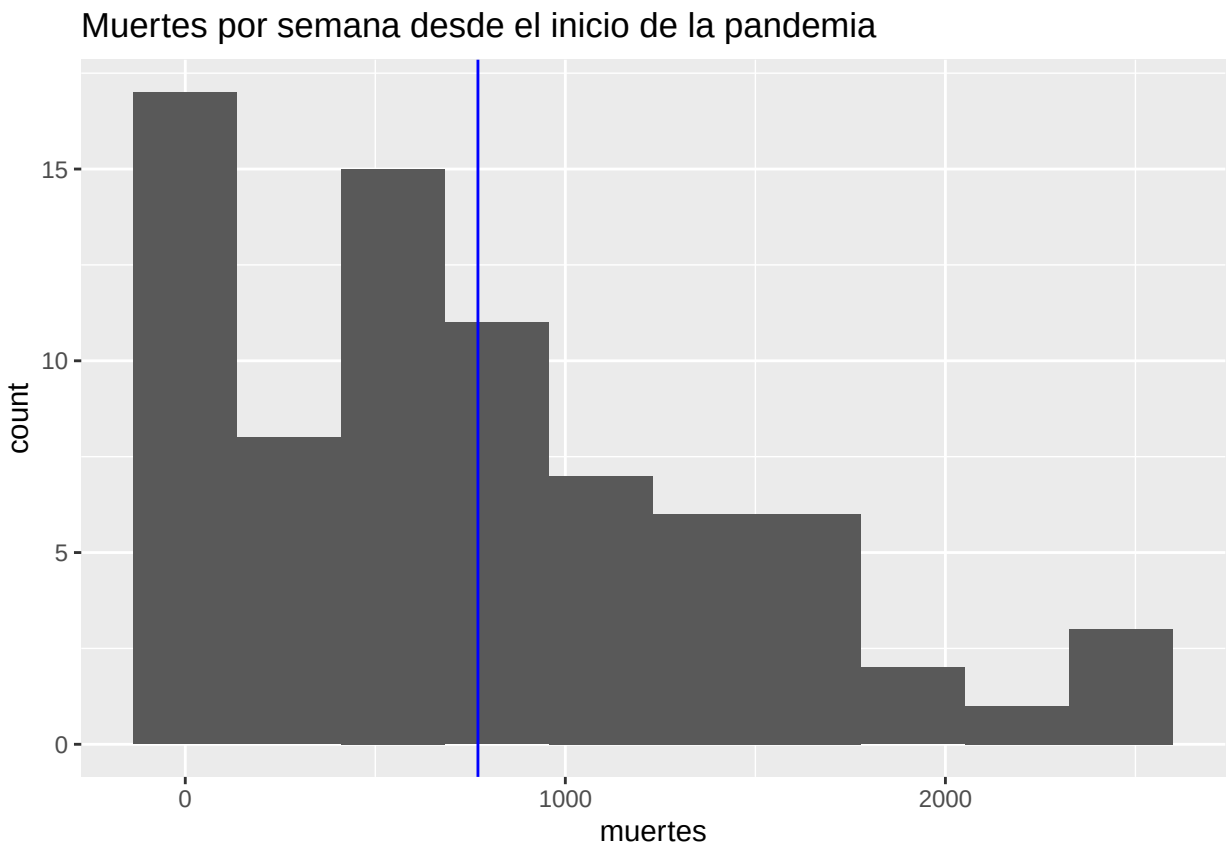
```
df4 <- subset(df3, week_def != "53")  #eliminamos datos que no son lógicos, como semana 53 del
df5 <- df4[1:(76), ]  #eliminamos última semana para después hacer una estimación de la misma,
sem <- df5$week_def
```

```r
muertes <- df5$sum_def
semana <- as.double(sem)
anio <- df5$year_def

df6 <- data.frame(anio, semana, muertes)  #A las semanas del año 2021, le agregamos 52 para di
a <- 1:24
for (i in seq_along(a)) {
    df6[52 + i, 2] <- df6[i, 2] + 52
}
```

```r
summary(df6)
```

```
##      anio               semana         muertes
##  Length:76          Min.   : 1.00   Min.   :   1.0
##  Class :character   1st Qu.:19.75   1st Qu.: 199.2
##  Mode  :character   Median :38.50   Median : 641.0
##                     Mean   :38.50   Mean   : 770.1
##                     3rd Qu.:57.25   3rd Qu.:1139.8
##                     Max.   :76.00   Max.   :2464.0
```

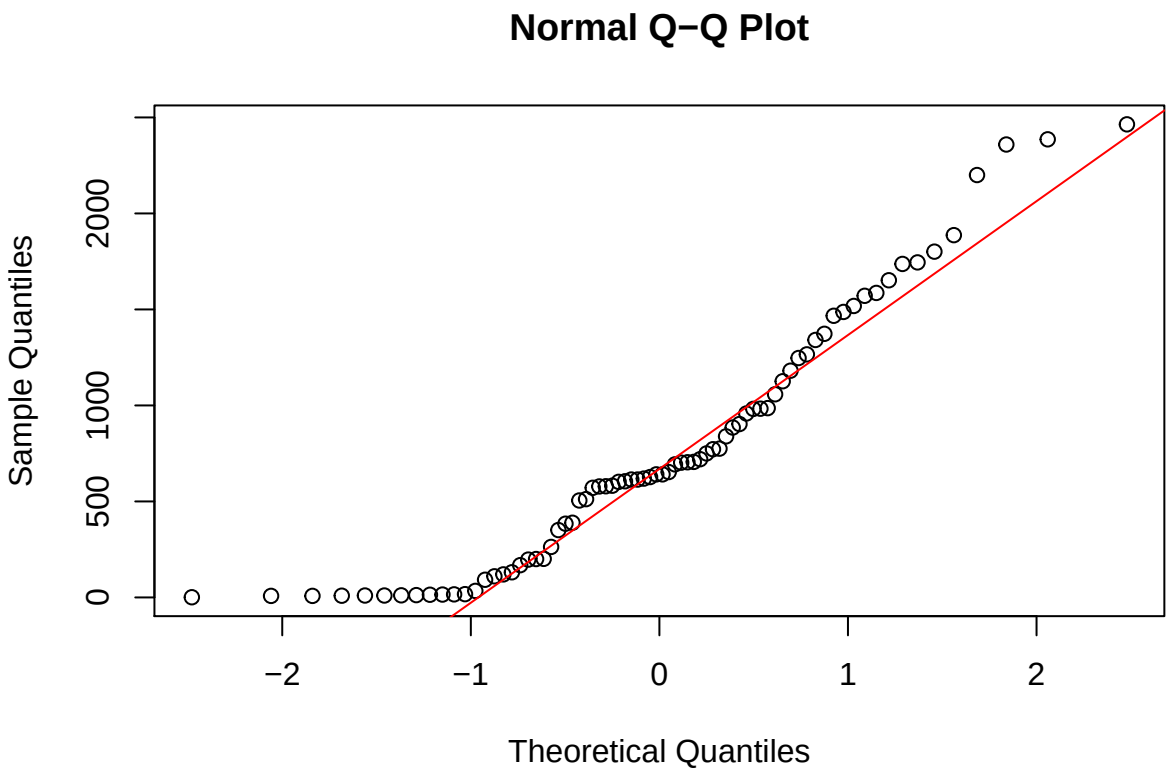
```r
ggplot(df6, aes(x = muertes)) + geom_histogram(bins = 10) + ggtitle("Muertes por semana desde
    geom_vline(xintercept = mean(muertes), color = "blue")
```



Lo que notamos principalmente esque la distribución de las muertes por semana está sesgada hacia la derecha de la media, marcada con una línea azul con valor de 770 muertes. Recordando las graficas anteriores sobre decesos, contagios y hospitalizaciones acumuladas es fácil entender que este histograma tenga esta forma, marcando muchas semanas por debajo de la media, y algunas menos semanas pero con una cantidad de muertes muy elevada a la izquierda.

Esto es más evidente si usamos un qqplot contrastando contra la distribución normal.

```
qqnorm(df6$muertes)
qqline(df6$muertes, col = "red")
```

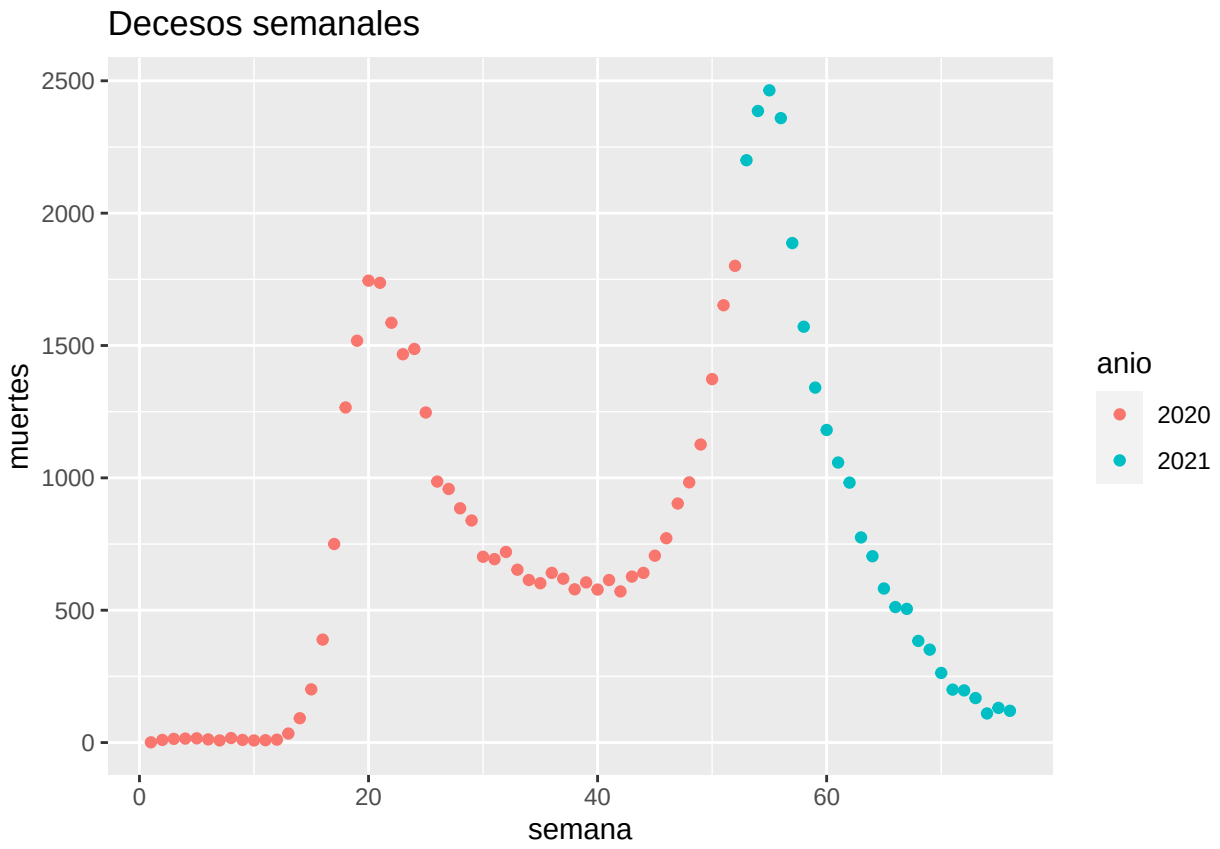


Aplicación de herramientas de clase

Interpolación

La primeras herramientas que usaremos serán aquellas de interpolación. En esta sección probaremos distintos modelos con polinomios de distinto grado con el fin de encontrar aquel que logre la predicción de los decesos de la última semana con el menor error cuadrático medio.

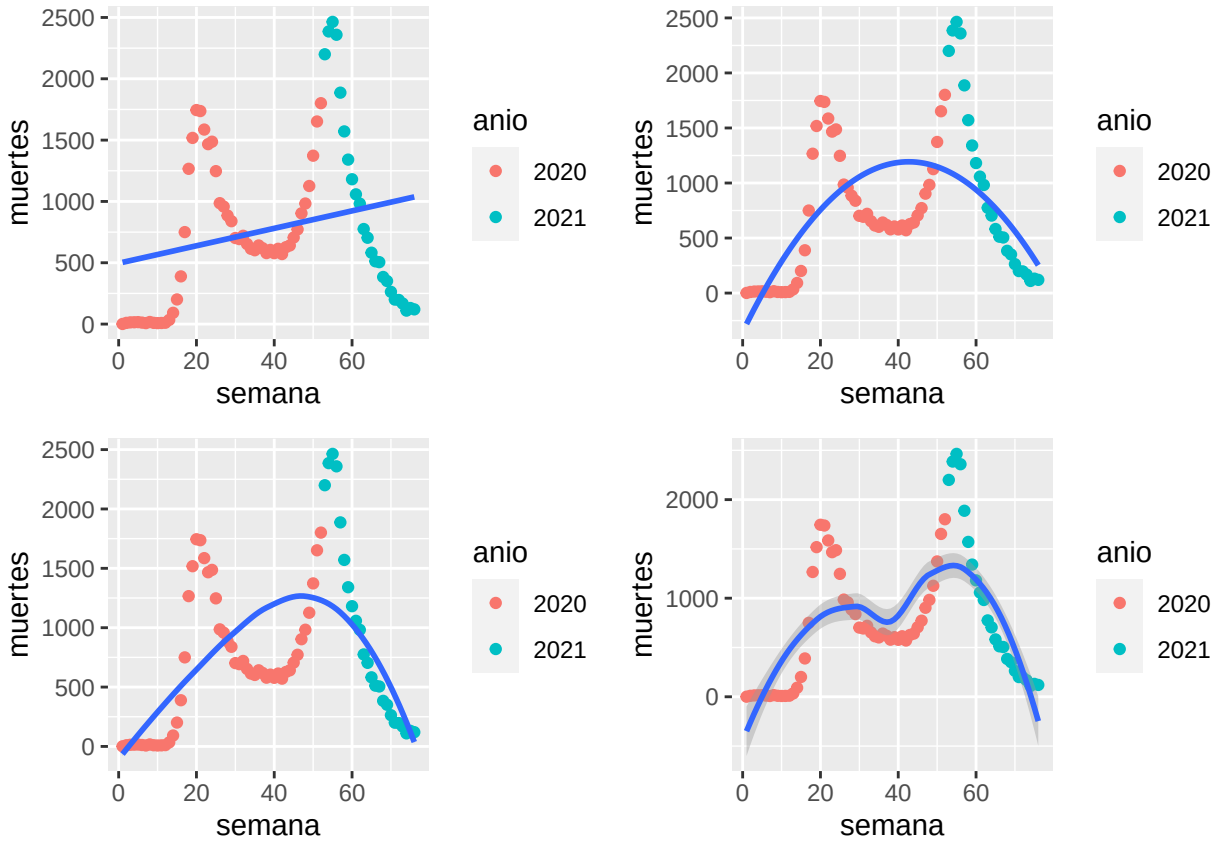
```
ggplot(df6) + geom_point(aes(x = semana, y = muertes, color = anio)) +
    ggtitle("Decesos semanales")
```




```
x1 <- df6$semana
y1 <- df6$muertes
p1 <- polyfit(x1, y1, 1)  #polinomio grado 1
p2 <- polyfit(x1, y1, 2)  #polinomio grado 2
p3 <- polyfit(x1, y1, 3)  #polinomio grado 3
p6 <- polyfit(x1, y1, 6)  #polinomio grado 6

# interpolación grado 1
df6$approx1 <- polyval(p1, x1)
a <- ggplot(df6) + geom_point(aes(x = semana, y = muertes, color = anio)) +
    geom_smooth(aes(x = semana, y = approx1))

# interpolación grado 2
df6$approx2 <- polyval(p2, x1)
b <- ggplot(df6) + geom_point(aes(x = semana, y = muertes, color = anio)) +
    geom_smooth(aes(x = semana, y = approx2))

# interpolación grado 3
df6$approx3 <- polyval(p3, x1)
```

```
c <- ggplot(df6) + geom_point(aes(x = semana, y = muertes, color = anio)) +
    geom_smooth(aes(x = semana, y = approx3))

# interpolación grado 6
df6$approx6 <- polyval(p6, x1)
d <- ggplot(df6) + geom_point(aes(x = semana, y = muertes, color = anio)) +
    geom_smooth(aes(x = semana, y = approx6))


grid.arrange(a, b, c, d, ncol = 2)
```

```
## `geom_smooth()` using method = 'loess' and formula 'y ~ x'
## `geom_smooth()` using method = 'loess' and formula 'y ~ x'
## `geom_smooth()` using method = 'loess' and formula 'y ~ x'
## `geom_smooth()` using method = 'loess' and formula 'y ~ x'
```



Habiendo visto los modelos graficamente, el siguiente paso es realizar la predicción y medir el error. De manera concreta, vamos a hacer una extrapolación del número de muertes en la semana 77. Es decir, queremos, con base en nuestros modelos, predecir el número de muertos en la semana 77.

```
Est_p1 <- polyval(p1, 77)
Est_p2 <- polyval(p2, 77)
Est_p3 <- polyval(p3, 77)
Est_p6 <- polyval(p6, 77)

abs_errors <- c(Est_p1, Est_p2, Est_p3, Est_p6)  #error en termino de decesos

# definimos la funcion error cuadratico medio
M_S_E <- function(est, obj) {
    sqsum = 0
    for (i in 1:length(est)) {
        sqsum <- (obj[i] - est[i])^2 + sqsum
        ecm <- sqsum/length(est)
    }
    print(ecm)
}

ECM_p1 <- M_S_E(df6$approx1, df6$muertes)  # ECM PARA POLINOMIO G1
```

```
## [1] 388691
```

```
ECM_p2 <- M_S_E(df6$approx2, df6$muertes)  # ECM PARA POLINOMIO G2
```

```
## [1] 255668.8
```

```
ECM_p3 <- M_S_E(df6$approx3, df6$muertes)  # ECM PARA POLINOMIO G3
```

```
## [1] 244051.9
```

```
ECM_p6 <- M_S_E(df6$approx6, df6$muertes)  # ECM PARA POLINOMIO G6
```

```
## [1] 97094.37
```

```
mse <- c(ECM_p1, ECM_p2, ECM_p3, ECM_p6)

# creo dataframe con errores
errors <- data.frame(abs_errors, mse)
# output de tabla de errores
kable(errors, booktabs = T, align = "c", col.names = c("Error en decesos para semana 77",
    "Error cuadrático medio general"), caption = "Comparación de modelos") %>%
    kable_styling(position = "center", latex_options = "repeat_header")
```

De la tabla anterior concluimos que el modelo que más se acerco a las defunciones de la semana 77, que tuvo 118 defunciones fue el modelo del polinomio grado 2. Sin embargo, el modelo con el

Table 2. Comparación de modelos

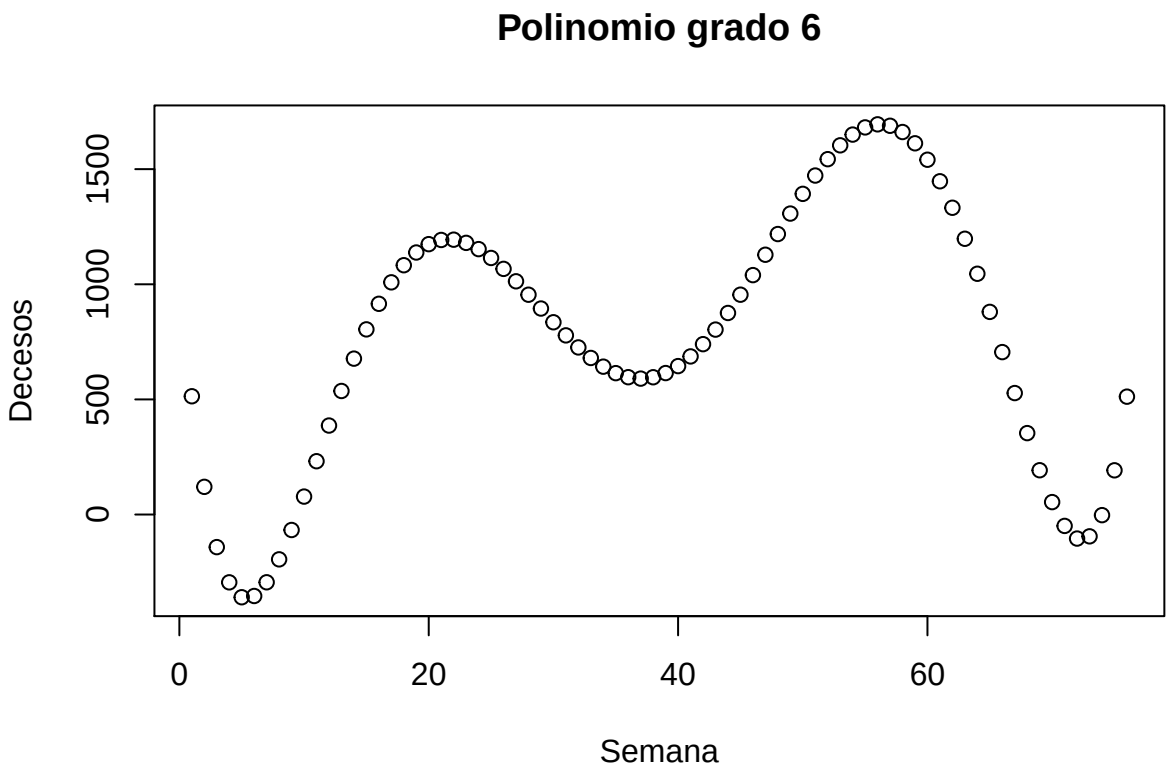
Error en decesos para semana 77	Error cuadrático medio general
1043.9653	388691.03
195.5798	255668.83
-113.0184	244051.89
982.5259	97094.37

menor error cuadrático medio fue el *spline*.

Area bajo la curva

Necesitamos conocer el número de defunciones de la semana 1 a la 76. Supongamos que no conocemos el número de defunciones por semana, por lo cual, debemos de estimarlo a partir de la función obtenida de la interpolación del polinomio grado 6.

```
eq1 = function(x) {
    (0.00000606019424051342 * x^6) - (0.00140040473599997 * x^5) +
        (0.121907555490365 * x^4) - (4.94516796308746 * x^3) +
        (92.5593257205104 * x^2) - (638.201246371754 * x) + (1064.09435543837)
}
# la eq1 tiene los coeficientes del polinomio de grado 6 que
# se utilizaron para la interpolación.
plot(1:76, eq1(1:76), main = "Polinomio grado 6", xlab = "Semana",
    ylab = "Decesos")
```




```
n <- 6
a <- 1
b <- 76
f <- function(x) {
    (0.00000606019424051342 * x^6) - (0.00140040473599997 * x^5) +
        (0.121907555490365 * x^4) - (4.94516796308746 * x^3) +
        (92.5593257205104 * x^2) - (638.201246371754 * x) + (1064.09435543837)
}
cc <- gaussLegendre(n, a, b)
Q <- sum(cc$w * f(cc$x))
Q
```

```
## [1] 57939.74
```

```
integral <- integrate(f, a, b)
integral
```

```
## 57939.74 with absolute error < 0.00000000069
```

Si comparamos el valor de nuestra estimación de la integral, con el valor de la integral real, tenemos que el error relativo de nuestra estimación de la integral es:

```
error_reltivo_estimacion <- abs((Q - 57939.74)/57939.74) * (100)
error_reltivo_estimacion
```

```
## [1] 0.00000839447
```

El error de estimación es muy bajo, es del 0.0000084%

Por otro lado, si comparamos el valor de nuestra estimación, contra el valor real del total de suma de defunciones de nuestra base de datos, podemos obtener error relativo de nuestra aproximación, dado que

```
num_def <- sum(df2$def)  # Número total de defunciones
error_reltivo_totalDef <- abs((Q - num_def)/num_def) * (100)
error_reltivo_totalDef
```

```
## [1] 4.534791
```

El error relativo de nuestra estimación de defunciones totales es del 4.5347%

Prueba de hipotesis

Contingency Analysis

Existe la hipótesis que una persona es intubada si presenta cierto tipo de condiciones: asma, diabetes, epoc, etc. Para un nivel de significancia de .05, vamos a rechazar o aceptar la hipótesis de independencia de estas variables

```
df10 <- subset(df8, intubado != "NO APLICA")
df11 <- subset(df10, intubado != "NO ESPECIFICADO")
df_intubado <- select(df11, sexo, intubado, diabetes, epoc, asma,
    inmusupr, hipertension, obesidad, renal_cronica, tabaquismo)
df_chi <- df_intubado %>% group_by(intubado) %>% summarise(sum_diab = sum(diabetes,
    na.rm = TRUE), sum_EPOC = sum(epoc, na.rm = TRUE), sum_asma = sum(asma,
    na.rm = TRUE), sum_inmusupr = sum(inmusupr, na.rm = TRUE),
    sum_hipertension = sum(hipertension, na.rm = TRUE), sum_obesidad = sum(obesidad,
        na.rm = TRUE), sum_renal_cronica = sum(renal_cronica,
        na.rm = TRUE), tabaquismo = sum(tabaquismo, na.rm = TRUE))
df_chi
```

```
## # A tibble: 2 x 9
##   intubado sum_diab sum_EPOC sum_asma sum_inmusupr sum_hipertension sum_obesidad
##   <chr>       <dbl>    <dbl>    <dbl>        <dbl>            <dbl>        <dbl>
## 1 NO          34596     4946     3146         5778            40376        23436
## 2 SI           7613     1016      514          942             8593         5613
## # ... with 2 more variables: sum_renal_cronica <dbl>, tabaquismo <dbl>
```

```
observed_table <- matrix(c(34596, 4946, 3146, 5778, 40376, 23436,
    7618, 12620, 7613, 1016, 514, 942, 8593, 5613, 1439, 2735),
    nrow = 2, ncol = 8, byrow = T)
rownames(observed_table) <- c("NO INTUBADO", "INTUBADO")
colnames(observed_table) <- c("Diabetes", "EPOC", "Asma", "Inmunosuprimido",
    "Hipertension", "Obesidad", "Renal cronico", "Tabaquismo")
observed_table
```

```
##             Diabetes EPOC Asma Inmunosuprimido Hipertension Obesidad
## NO INTUBADO    34596 4946 3146            5778        40376    23436
## INTUBADO        7613 1016  514             942         8593     5613
##             Renal cronico Tabaquismo
## NO INTUBADO          7618      12620
## INTUBADO             1439       2735
```

```
chi_test <- chisq.test(observed_table)
chi_test
```

```
##
##  Pearson's Chi-squared test
##
## data:  observed_table
## X-squared = 175.11, df = 7, p-value < 0.00000000000000022
```

El valor p es menor que el nivel de significancia (0.05). Por tanto, podemos rechazar la hipótesis nula y concluir que las dos variables (estar intubado y condición médica) no son independientes.

Letalidad por tipo de hospital

¿Existe un efecto diferenciado de la mortalidad frente al imss, isste e institución privada?

```
source <- data %>% select(resultado_lab, fecha_def, sector) %>%
    filter(resultado_lab == "POSITIVO A SARS-COV-2")  #

# aqui cuento la cantidad de personas atendidas en cada
# sector
people_by_hospital <- source %>% group_by(sector) %>% summarize(count_pplo = n())
# filter(fecha_def != 'NA' )

# Aqui asumo que la persona murio si solo si hay fecha de
# defuncion, con base en eso calculo la cantidad de
# fallecidos
```

```
death_by_hospital <- source %>% filter(!is.na(fecha_def)) %>%
    group_by(sector) %>% summarize(count_dths = n())

# match
(letalidad <- merge(death_by_hospital, people_by_hospital, by = "sector") %>%
    mutate(letalidad = count_dths/count_pplo))
```

```
##      sector count_dths count_pplo  letalidad
## 1 CRUZ ROJA          1         87 0.01149425
## 2   ESTATAL         41        246 0.16666667
## 3      IMSS      19553      84291 0.23197020
## 4    ISSSTE       2335       7660 0.30483029
## 5     PEMEX        994       5574 0.17832795
## 6   PRIVADA        446       8520 0.05234742
## 7    SEDENA        536       2842 0.18859958
## 8     SEMAR        230       3042 0.07560815
## 9       SSA       8218     243092 0.03380613
```

```
## letalidad global
(global_letality <- sum(letalidad$count_dths)/sum(letalidad$count_pplo))
```

```
## [1] 0.09104724
```

Para averiguar probaremos la siguiente hipotesis

¿Es letalidad en el sector j mayor a la letalidad del país?

$H_n : \mu_j = .091$

$H_a : \mu_j > .091$

Para la prueba de hipótesis asumiremos que la distribución de deceso es binomial y partiendo de eso aprovecharemos el teorema del limite central poder realizar las pruebas de hipotesis con base en el estadístico Z.

$Z = \frac{p - n\pi}{\sqrt{n\pi(1-\pi)}}$

donde $p =$ letalidad $\cdot n$

```
letalidad <- letalidad %>% mutate(esperanza = count_pplo * letalidad,
    st_let = sqrt(count_pplo * (1 - letalidad))) %>% mutate(z = (letalidad *
    count_pplo - global_letality * count_pplo)/sqrt(global_letality *
    count_pplo * (1 - global_letality)))

limit <- qnorm(0.995, 0, 1)  # for alpha=.01 one talied
```

```
letalidad <- letalidad %>% mutate(prueba = ifelse(z > limit,
    "Se rechaza H0", "No se rechaza H0"))

letalidad %>% select(sector, z, prueba)
```

```
##      sector          z           prueba
## 1 CRUZ ROJA  -2.579361 No se rechaza H0
## 2   ESTATAL   4.122844    Se rechaza H0
## 3      IMSS 142.222504    Se rechaza H0
## 4    ISSSTE  65.040461    Se rechaza H0
## 5     PEMEX  22.651540    Se rechaza H0
## 6   PRIVADA -12.417229 No se rechaza H0
## 7    SEDENA  18.077805    Se rechaza H0
## 8     SEMAR  -2.960037 No se rechaza H0
## 9       SSA -98.104586 No se rechaza H0
```

Conclusiones

Respecto a la mortalidad de la enfermedad respecto al sexo de las personas, nos indica que los hombres tienen una mayor defunción absoluta en el rango de edad de 58 a 73, mientras que las mujeres es en el rango de edad de 60 a 75 años. De acuerdo con los resultados, de las personas fallecidas por COVID-19, el 39.4% sufría de hipertensión, el 33.8% de diabetes y el 20.12% tenía obesidad. El 7.3% de las personas fallecidas tenían una condición renal crónica. Además, de acuerdo a nuestros hallazgos, el estar intubado o no estar intubado tienen una dependencia respect a la condición de salud de las personas hospitalizadas. Esto es consistente con la información proporcionada por la OMS: las personas que padecen afecciones médicas subyacentes, como hipertensión arterial, problemas cardíacos o pulmonares, diabetes, obesidad o cáncer, corren un mayor riesgo de presentar cuadros graves.

Fuentes

- Khan M, Adil SF, Alkhathlan HZ, Tahir MN, Saif S, Khan M, Khan ST. COVID-19: A Global Chal
- Statista, 2021. Número de personas fallecidas a causa del coronavirus en el mundo al de 2
- Organización Mundial de la Salud. 2021. Información básica sobre la COVID-19 Recuperado el