

```

1  ### SCRIPT 11 - R ###
2  # Contraste de hipotesis y analisis de datos de las modificaciones hasta 2021
3  #
4  #####
5  ## Directorio de trabajo
6
7  setwd("C:/Users/guillermo.alonso/Desktop/Tesis Guillermo/1er articulo/Scripts")
8
9  ## Paquetes
10 library(corrplot)
11 library(nortest)
12 library(coin)
13 library(ggplot2)
14
15 ## Inicializo el archivo en el que se guardan los mensajes de consola
16 if (file.exists("C:/Users/guillermo.alonso/Desktop/Tesis Guillermo/1er
articulo/Scripts/Resultados Estadistica.txt")) { #Inicializo el archivo de cero
17   file.remove("C:/Users/guillermo.alonso/Desktop/Tesis Guillermo/1er
articulo/Scripts/Resultados Estadistica.txt")
18 } ## Elimino el anterior
19 sink("C:/Users/guillermo.alonso/Desktop/Tesis Guillermo/1er
articulo/Scripts/Resultados Estadistica.txt")
20
21 ## Cargo matriz de datos
22
23 data<-read.csv2(file="Graficas Modis last update Civil.csv", dec=".", sep=";")
24 # Seleccionar las primeras 794 filas y las primeras 43 columnas
25 data <- data[1:794, 1:43]
26
27 ## Estudio de hipotesis de correlacion entre Precio.licitacion y Precio.final ----
28
29 #Selecciono las variables
30 Precio.licitacion<-data$Presupuesto.base.sin.impuestos.licitación.lote
31 Precio.final<-data$FinalLegalMonetaryTotal
32 df<-data.frame(Precio.licitacion,Precio.final)
33
34 ## Estudio de la normalidad Anderson-Darling de las variables ----
35 cat("---- Estudio de la normalidad Anderson-Darling de las variables ---- \n")
36 ad_test_lici <- ad.test(Precio.licitacion)
37 ad_test_final<- ad.test(Precio.final)
38
39 # Función para devolver "Si" o "No" según el resultado del test
40 obtener_respuesta <- function(resultado) {
41   if (resultado$p.value < 0.05) {
42     return("No")
43   } else {
44     return("Si")
45   }
46 }
47 # Imprimir el resultado para Precio.licitacion
48 cat("¿Precio.licitacion supera la prueba de normalidad de Anderson-Darling?",
obtener_respuesta(ad_test_lici), "\n")
49
50 # Imprimir el resultado para Precio.final
51 cat("¿Precio.final supera la prueba de normalidad de Anderson-Darling?",
obtener_respuesta(ad_test_final), "\n")
52
53 ## Test para variables no normales y continuas, con tendencia monotonica --> Spearman
----
54 cat("---- Estudio de la correlacion entre los precios con tendencia monotonica
mediante el test de Spearman ---- \n")
55 # Calcular el coeficiente de correlación de Spearman
56 cor_spearman <- cor.test(Precio.licitacion, Precio.final, method = "spearman",exact=F)
57
58 ## Usamos exact=F porque se produce cuando hay empates (ties) en los datos, lo que
puede dificultar el cálculo exacto del p-valor para la correlación de Spearman.
59
60 # Obtener el p-valor
61 p_valor <- cor_spearman$p.value
62
63 # Definir una función para obtener la conclusión
64 obtener_conclusion <- function(p_valor) {
  if (p_valor < 0.05) {

```

```

65     return("Si")
66 } else {
67     return("No")
68 }
69 }
70 # Imprimir el p-valor y la conclusión
71 cat("El p-valor en el test de Spearman es:", p_valor, "\n")
72 cat("¿Hay una correlación significativa mediante la correlación de Spearman? ",
obtener_conclusion(p_valor), "\n")
73
74
75 ## Prueba de Wilcoxon para comparar medianas de muestras relacionadas -----
76 cat("---- Prueba de Wilcoxon para comparar medianas de muestras relacionadas ---- \n")
77 # Realizar la prueba de Wilcoxon
78 prueba_wilcoxon <- wilcox.test(Precio.licitacion, Precio.final, paired=T)
79
80 # Obtener el valor p
81 p_valor <- prueba_wilcoxon$p.value
82
83 # Definir una función para obtener la conclusión
84 obtener_conclusion <- function(p_valor) {
85     if (p_valor < 0.05) {
86         return("Si")
87     } else {
88         return("No")
89     }
90 }
91
92 # Imprimir el valor p y la conclusión
93 cat("El valor p de la prueba de Wilcoxon es:", p_valor, "\n")
94 cat("¿Hay una diferencia significativa mediante la prueba de Wilcoxon? ",
obtener_conclusion(p_valor), "\n")
95
96
97
98 ## Test de Kolmogorov-Smirnov para caracterizar la magnitud de la diferencia entre
variables -----
99 cat("---- Test de Kolmogorov-Smirnov para caracterizar la magnitud de la diferencia
entre los precios ----\n")
100 # Realizar el test de Kolmogorov-Smirnov
101 test_ks <- ks.test(Precio.licitacion, Precio.final, exact=F)
102
103 # Obtener el estadístico D y el valor p
104 estadistico_D <- test_ks$statistic
105 p_valor <- test_ks$p.value
106
107 # Definir una función para obtener la conclusión
108 obtener_conclusion <- function(p_valor) {
109     if (p_valor < 0.05) {
110         return("Hay una diferencia significativa entre los precios mediante el test KS.\n"
)
111     } else {
112         return("No hay evidencia suficiente para afirmar una diferencia significativa
entre los precios mediante el test KS.\n")
113     }
114 }
115
116 # Imprimir el estadístico D, el valor p y la conclusión
117 cat("Valor Estadístico D en el test K-S:", estadistico_D, "\n")
118 cat("Valor p del test K-S:", p_valor, "\n")
119 cat(obtener_conclusion(p_valor), "\n")
120
121
122 sink()
123
124
125 ## Representación gráfica de Precio de Licitación y Precio Final ----
126 # Calcular la línea de tendencia lineal
127 x<-Precio.licitacion
128 y<-Precio.final
129 modelo <- lm(y ~ x, data = df)
130 pendiente <- coef(modelo)[2]
131 intercepto <- coef(modelo)[1]

```

```

132 r_cuadrado <- summary(modelo)$r.squared
133
134 # Crear el gráfico con ggplot2
135 grafico <- ggplot(df, aes(x = x, y = y)) +
136   geom_point() + # Puntos
137   geom_smooth(method = "lm", se = FALSE) # Línea de tendencia lineal
138
139 # Ajustar el formato de los ejes
140 #grafico <- grafico +
141 #   scale_x_continuous(limits = range(df$x), breaks = df$x, labels = df$x) +
142 #   scale_y_continuous(limits = range(df$y), breaks = df$y, labels = df$y)
143
144 # Añadir el valor de R cuadrado al gráfico
145 grafico <- grafico +
146   labs(title = paste("Línea de tendencia (R² =", round(r_cuadrado, 2), ")"),
147        x = "Base Bid Price", y = "Final Price")
148
149 # Mostrar el gráfico
150 print(grafico)
151
152 ## Directorio de trabajo
153
154 setwd("C:/Users/guillermo.alonso/Desktop/Tesis Guillermo/1er articulo/Scripts")
155
156 ## Paquetes
157 library(corrplot)
158 library(nortest)
159 library(coin)
160 library(ggplot2)
161 library(lubridate)
162 library(dplyr)
163
164 ## Cargo matriz de datos resultados inge civil -----
165
166 data_Rcivil<-read.csv2(file="Graficas Modis last update_Civil.csv", dec=".", sep=";")
167 # Seleccionar las primeras 794 filas y las primeras 43 columnas (los casos totales
168 # que hay)
169 data_Rcivil <- data_Rcivil[1:794, 1:43]
170
171 ## Cargo matriz de datos licitaciones constru, interseco con Resultados de Civil y
172 # lo filtro. -----
173 data_LConstru<-read.csv2(file="Licitaciones_Constru.csv", dec=".", sep=";",header=T,
174 fill=T,blank.lines.skip = T)
175 data_LConstru$Fecha.actualizacion<-dmy(as.character(data_LConstru$Fecha.actualizacion
176 ))
177 # Obtener los identificadores en común entre ambos data frames
178 identificadores_comunes <- intersect(data_LConstru$Identificador, data_Rcivil$
179 Identificador)
180
181 # Filtrar los casos de data_LConstru con los identificadores en común
182 data_LCivil<- data_LConstru[data_LConstru$Identificador %in% identificadores_comunes,
183 ]
184
185 # Filtrar los casos según los criterios establecidos
186 data_LCivil_filtrado <- data_LCivil %>%
187   group_by(Identificador) %>%
188   filter(n() == 1 | Fecha.actualizacion == max(Fecha.actualizacion)) %>%
189   ungroup()
190
191 ## Junto ambos data frames en uno unico, unifico duraciones de modificados -----
192
193 ## Combinar los data frames por el valor numérico de la variable "Identificador"
194 data_Civil_todo <- merge(data_Rcivil, data_LCivil_filtrado, by = "Identificador")
195
196 ## Calcular el modificado en tiempo, pasarlo todo a dias
197 data_Civil_todo$ContractModificationDurationMeasure <- as.double(data_Civil_todo$
198 ContractModificationDurationMeasure)
199 data_Civil_todo$FinalDurationMeasure <- as.double(data_Civil_todo$FinalDurationMeasure
200 )
201 data_Civil_todo <- data_Civil_todo %>% ##Cambio un error concreto de ANN a DAY
202   mutate(
203     FinalDurationMeasure_Uc = if_else(Identificador == 934332 &

```

```

197     FinalDurationMeasure_Uc == "ANN", "DAY", as.character(FinalDurationMeasure_Uc)))
198 data_Civil_todo <- data_Civil_todo %>% ##Duracion de la modificacion
199     mutate(
200         ContractModificationDurationMeasure = case_when(
201             ContractModificationDurationMeasure_Uc == "MON" ~
202             ContractModificationDurationMeasure * 30,
203             ContractModificationDurationMeasure_Uc == "ANN" ~
204             ContractModificationDurationMeasure * 360,
205             TRUE ~ ContractModificationDurationMeasure
206         ),
207         ContractModificationDurationMeasure_Uc = "DAY"
208     )
209 data_Civil_todo <- data_Civil_todo %>% ##Duracion final del proyecto
210     mutate(
211         FinalDurationMeasure = case_when(
212             FinalDurationMeasure_Uc == "MON" ~ FinalDurationMeasure * 30,
213             FinalDurationMeasure_Uc == "ANN" ~ FinalDurationMeasure * 360,
214             TRUE ~ FinalDurationMeasure
215         ),
216         FinalDurationMeasure_Uc = "DAY"
217     )
218 ##Calcular la variable de duracion del contrato como resta de las otras
219 data_Civil_todo <- data_Civil_todo %>%
220     mutate(InitialDurationMeasure = FinalDurationMeasure -
221             ContractModificationDurationMeasure)
222 data_Civil_todo <- data_Civil_todo %>%
223     mutate(InitialDurationMeasure_Uc = "DAY")
224
225 write.table(data_Civil_todo, file="data_Civil_todo.csv", sep=";", dec=".", na="NA",
226             row.names = F)
227
228 ### Seleccion de las variables para representar en corrplot -----
229 # No meto el CPV porque ya se tratan todos casos de ingenieria civil
230
231 data_corrplot<-data_Civil_todo[c("Presupuesto.base.sin.impuestos.licitación.lote"
232                                , "Fecha.del.acuerdo.licitación.lote"
233                                , "Número.de.ofertas.recibidas.por.licitación.lote"
234                                , "Importe.adjudicación.sin.impuestos.licitación.lote"
235                                , "ContractModificationDurationMeasure"
236                                , "InitialDurationMeasure"
237                                , "ContractModificationLegalMonetaryTotal"
238                                , "FinalLegalMonetaryTotal"
239                                , "Tipo.de.contrato" #Demasiados de tipo Obra que no
240                                da pie a correlacion
241                                , "Tipo.de.Administración"
242                                , "Tipo.de.procedimiento"
243                                , "Tramitación"
244                                )]
245
246 #Cambio la fecha por un valor numérico del año en el que se hizo
247
248 data_corrplot$Fecha.del.acuerdo.licitación.lote<-as.Date(data_corrplot$
249 Fecha.del.acuerdo.licitación.lote,"%d/%m/%Y")
250 data_corrplot$Fecha.del.acuerdo.licitación.lote<-format(data_corrplot$
251 Fecha.del.acuerdo.licitación.lote,"%Y")
252 data_corrplot$Fecha.del.acuerdo.licitación.lote<-as.numeric(data_corrplot$
253 Fecha.del.acuerdo.licitación.lote)
254
255 ## Cambio las tipo factor a tipo numeric:
256
257 #data_corrplot <- data_corrplot %>% ## Demasiados de tipo Obras que dan pie a poca
258 correlacion
259 # mutate(
260 #     Tipo.de.contrato = case_when(
261 #         Tipo.de.contrato == "Obras" ~ 1L,
262 #         Tipo.de.contrato == "Servicios" ~ 2L,
263 #         Tipo.de.contrato == "Suministros" ~ 3L,
264 #         Tipo.de.contrato == "Gestión de Servicios Públicos" ~ 4L,
265 #         Tipo.de.contrato == "Privado" ~ 5L,
266 #         TRUE ~ NA_integer_
267 #     )

```

```

260 # )
261 data_corrplot <- data_corrplot %>%
262   mutate(
263     Tramitación = case_when(
264       Tramitación == "Ordinaria" ~ 1L,
265       Tramitación == "Urgente" ~ 2L,
266       Tramitación == "Emergencia" ~ 3L,
267       TRUE ~ NA_integer_
268     )
269   )
270 data_corrplot <- data_corrplot %>%
271   mutate(
272     Tipo.de.Administración = case_when(
273       Tipo.de.Administración == "Administración General del Estado" ~ 1L,
274       Tipo.de.Administración == "Comunidad Autónoma" ~ 2L,
275       Tipo.de.Administración == "Administración Local" ~ 3L,
276       Tipo.de.Administración == "Entidad de Derecho Público" ~ 4L,
277       Tipo.de.Administración == "Otras Entidades del Sector Público" ~ 5L,
278       TRUE ~ NA_integer_
279     )
280   )
281 data_corrplot <- data_corrplot %>%
282   mutate(
283     Tipo.de.procedimiento = case_when(
284       Tipo.de.procedimiento == "Abierto" ~ 1L,
285       Tipo.de.procedimiento == "Abierto simplificado" ~ 2L,
286       Tipo.de.procedimiento == "Derivado de acuerdo marco" ~ 3L,
287       Tipo.de.procedimiento == "Negociado con publicidad" ~ 4L,
288       Tipo.de.procedimiento == "Normas internas" ~ 5L,
289       Tipo.de.procedimiento == "Negociado sin publicidad" ~ 6L,
290       Tipo.de.procedimiento == "Restringido" ~ 7L,
291       TRUE ~ NA_integer_
292     )
293   )
294
295
296 ## Paso todas las variables a numeric
297
298 data_corrplot<-apply(data_corrplot,2,as.numeric)
299
300 ## Renombro variables
301
302 variables_ingles<-c("Tender price"
303                     , "Year"
304                     , "Biddings number"
305                     , "Awarding price"
306                     , "Modification delay"
307                     , "Initial duration"
308                     , "Modification cost"
309                     , "Final project cost"
310                     , "Contract type" ## Demasiados de tipo Obras que dan pie a poca
correlacion
311                     , "Administration type"
312                     , "Process type"
313                     , "Urgency"
314                     )
315 colnames(data_corrplot)<-variables_ingles
316
317 ### Calculo y representacion del mapa de calor de correlaciones
318 matriz_correlacion <- cor(data_corrplot, method = "spearman", use =
"pairwise.complete.obs")
319
320 # Configurar el dispositivo PNG con alta calidad
321 png("corrplot.png", width = 2100, height = 1700, units = "px", res = 400)
322
323 corrplot(matriz_correlacion, method = "color", type = "full", order = "hclust", tl.cex
= 0.65, tl.col = "black", addCoef.col = "black", cl.cex = 0.65, number.cex = 0.6)
324
325 # Finalizar la salida del PNG
326 dev.off()
327
328
329

```

```

330 ### Estadística descriptiva de la base de datos -----
331
332 # Selección de las variables cuantitativas continuas
333 variables_quan<-c("Tender price"
334                 , "Biddings number"
335                 , "Awarding price"
336                 , "Modification cost"
337                 , "Initial duration"
338                 , "Modification delay"
339                 , "Final project cost"
340 )
341
342
343 # Cálculo de las estadísticas descriptivas
344 stats <- apply(data_corrplot[, variables_quan], 2, function(x) {
345   n_cases <- sum(!is.na(x))
346   prop_na <- sum(is.na(x)) / length(x)
347   range_iqr <- IQR(x, na.rm = TRUE)
348   c(
349     Min = min(x, na.rm = TRUE),
350     Median = median(x, na.rm = TRUE),
351     Max = max(x, na.rm = TRUE),
352     Mean = mean(x, na.rm = TRUE),
353     SD = sd(x, na.rm = TRUE),
354     Variance = var(x, na.rm = TRUE),
355     N_Cases = n_cases,
356     Prop_NA = prop_na,
357     Range_IQR = range_iqr
358   )
359 })
360
361 # Creación de la tabla de estadísticas descriptivas
362 tabla_quan <- as.data.frame(stats)
363 tabla_quan<-t(tabla_quan)
364 write.table(tabla_quan, file="table_quan.csv", sep=";", dec=".", na="NA", row.names = T,
365            col.names = T)
366
367 # Selección de las variables cualitativas
368 data_cual<-data_Civil_todo[c("Fecha.del.acuerdo.licitación.lote"
369                             , "Tipo.de.contrato" #Demasiados de tipo Obra que no
370                             da pie a correlacion
371                             , "Tipo.de.Administración"
372                             , "Tipo.de.procedimiento"
373                             , "Tramitación"
374 )]
375
376 #Cambio la fecha por un valor numérico del año en el que se hizo
377
378 data_cual$Fecha.del.acuerdo.licitación.lote<-as.Date(data_cual$
379 Fecha.del.acuerdo.licitación.lote,"%d/%m/%Y")
380 data_cual$Fecha.del.acuerdo.licitación.lote<-format(data_cual$
381 Fecha.del.acuerdo.licitación.lote,"%Y")
382 data_cual$Fecha.del.acuerdo.licitación.lote<-as.factor(data_cual$
383 Fecha.del.acuerdo.licitación.lote)
384
385 # Traducir los elementos factor de la variable 'Tipo.de.contrato'
386 data_cual$Tipo.de.contrato <- factor(data_cual$Tipo.de.contrato, levels = c("Obras",
387 "Servicios", "Suministros", "Gestión de Servicios Públicos", "Privado"), labels = c(
388 "Works", "Servicies", "Supplies", "Public Service Managemnt", "Private"))
389
390 # Traducir los elementos factor de la variable 'Tramitación'
391 data_cual$Tramitación <- factor(data_cual$Tramitación, levels = c("Ordinaria",
392 "Urgente", "Emergencia"), labels = c("Ordinary", "Urgent", "Emergency"))
393
394 # Traducir los elementos factor de la variable 'Tipo.de.Administración'
395 data_cual$Tipo.de.Administración <- factor(data_cual$Tipo.de.Administración, levels =
396 c("Administración General del Estado", "Comunidad Autónoma", "Administración Local",
397 "Entidad de Derecho Público", "Otras Entidades del Sector Público"), labels = c(
398 "Central Administration", "Regional Administration", "Local Administration", "Public
399 Law Entity", "Other Public Sector Entities"))
400
401 # Traducir los elementos factor de la variable 'Tipo.de.procedimiento'
402 data_cual$Tipo.de.procedimiento <- factor(data_cual$Tipo.de.procedimiento, levels = c(

```

```

391 "Abierto", "Abierto simplificado", "Derivado de acuerdo marco", "Negociado con
392 publicidad", "Normas internas", "Negociado sin publicidad", "Restringido"), labels = c
393 ("Open", "Simplified Open", "Derived from Macro Agreement", "Negotiated with
    Advertising", "Internal Rules", "Negotiated without Advertising", "Restricted"))
394
395 variables_ingles<-c("Year"
396                     , "Contract type" ## Demasiados de tipo Obras que dan pie a poca
397                     correlacion
398                     , "Administration type"
399                     , "Process type"
400                     , "Urgency"
401 )
402
403 colnames(data_cual)<-variables_ingles
404
405 # Cálculo de las estadísticas descriptivas
406
407 stats2 <- lapply(data_cual[, variables_ingles], function(x) {
408   n_cases <- sum(!is.na(x))
409   prop_na <- sum(is.na(x)) / length(x)
410   table_data <- table(x, useNA = "ifany")
411   prop_data <- prop.table(table_data)
412   data.frame(Frequency = table_data, Proportion = prop_data, N_Cases = n_cases, N_As =
413     sum(is.na(x)))
414 })
415
416 # Combinar las estadísticas en una tabla final
417 tabla_cual <- do.call(rbind, stats2)
418 write.table(tabla_cual, file="table_cual.csv", sep=";", dec=".", na="NA", row.names = T,
419 col.names=T)

```