



**Escuela Universitaria
Politécnica - La Almunia**
Centro adscrito
Universidad Zaragoza

**ESCUELA UNIVERSITARIA POLITÉCNICA
DE LA ALMUNIA DE DOÑA GODINA (ZARAGOZA)**

ANEXOS

Optimización de rutas de recogida de fangos en las depuradoras

Sludge Collection Routes Optimization in Treatment Plants

608.23.38

Autor

Guillermo Aldunate Ozcariz

Director: Luis Mariano Esteban

Fecha: 29/02/2024



INDICE DE CONTENIDO BREVE

1. ANEXO 1. RELACIÓN PUNTOS DE RECOGIDA	-2-
2. ANEXO 2. MATRIZ DE DISTANCIAS	-2-
3. CODIGO DE R	-3-



INDICE DE CONTENIDO

1. ANEXO 1. RELACIÓN PUNTOS DE RECOGIDA	- 2 -
1.1. COORDENADAS DE LAS EDAR	- 2 -
2. ANEXO 2. MATRIZ DE DISTANCIAS	- 2 -
3. CODIGO DE R	- 3 -
3.1. ARCHIVO SERVER.R	- 3 -



1. ANEXO 1. RELACIÓN PUNTOS DE RECOGIDA

1.1. COORDENADAS DE LAS EDAR

Deposito	LATITUD	LONGITUD
IRAOLA	42,7877778	-1,62583333
BARASOAIN	42,6008333	-1,64944444
PUEYO	42,5636111	-1,56083333
SAN MARTIN DE UNX	42,5233333	-1,56861111
UJUE	42,5033333	-1,50055555
OLITE	42,4825	-1,64111111
PITILLAS	42,4144444	-1,62944444
SANTACARA	42,3658333	-1,5508333
CARCASTILLO	42,3825	-1,4572222
RADA	42,3236111	-1,58055555
FIGAROL	42,3483333	-1,3758333

2. ANEXO 2. MATRIZ DE DISTANCIAS

	IRAOLA	BARASOAIN	PUEYO	SAN MARTIN	UJUE	OLITE	PITILLAS	SANTACARA	CARCASTILLO	RADA	FIGAROL
IRAOLA	0	20865,41154	25474,24	29759,48624	33233,15	33949,47	41490,69	47292,17269	47107,3044	51717,21	52951,62
BARASOAIN	20865,41	0	8350,532	10861,31971	16308,46	13168,41	20778,55	27335,41082	28919,88428	31320,41	35904,31
PUEYO	25474,24	8350,532306	0	4521,247842	8320,093	11155,37	17503,01	21994,8689	21841,59295	26720,73	28315
SAN MARTIN	29759,49	10861,31971	4521,248	0	5998,223	7471,118	13085,79	17563,83744	18113,79727	22217,24	25045,67
UJUE	33233,15	16308,45737	8320,093	5998,223085	0	11736,89	14453,8	15825,3807	13889,21171	21019,34	20024,18
OLITE	33949,47	13168,41035	11155,37	7471,118032	11736,89	0	7623,283	14923,88892	18711,43904	18340,44	26342,35
PITILLAS	41490,69	20778,55187	17503,01	13085,79182	14453,8	7623,283	0	8402,274758	14538,68441	10859	22023
SANTACARA	47292,17	27335,41082	21994,87	17563,83744	15825,38	14923,89	8402,275	0	7883,93018	5285,609	14457,4
CARCASTILLO	47107,3	28919,88428	21841,59	18113,79727	13889,21	18711,44	14538,68	7883,93018	0	12031,96	7668,73
RADA	51717,21	31320,41042	26720,73	22217,24372	21019,34	18340,44	10859	5285,609496	12031,95614	0	16982,85
FIGAROL	52951,62	35904,31059	28315	25045,66547	20024,18	26342,35	22023	14457,39509	7668,730314	16982,85	0

3. CODIGO DE R

3.1. ARCHIVO SERVER.R

```
library(shiny)
library(TSP)
library(gdata)
library(readxl)
library(gor)
#library(igraph)
library(leaflet)
#library(rsconnect)
library(openrouteservice)
library(geodist)

Coordenadas<-read_excel("C:/Guillermo Aldunate/Estudios/2.
Universidad/5º Mecatrónica e IOI/Semestre 2/TFG/TFG/Programa
TFG/Programa Guillermo/CoordenadasGuillermo.xlsx",col_names =
TRUE,sheet=1)

MatrizDistancias<-geodist(Coordenadas)
Coordenadas<-as.data.frame(Coordenadas)
row.names(Coordenadas)<-Coordenadas[,1]
Coordenadas<-Coordenadas[,c(2,3)]
names(Coordenadas)<-c("LATITUD","LONGITUD")
Distancias<-as.matrix(MatrizDistancias)
rownames(Distancias)<-row.names(Coordenadas)
colnames(Distancias)<-rownames(Distancias)
Coordenadas<-as.matrix(Coordenadas)
coordenadas<-Coordenadas
ID1=rownames(Coordenadas)<-rownames(Distancias)
```

```
ui<-fluidPage(  
  titlePanel("RUTA DE RECOGIDA DE FANGOS EN LAS EDAR DE  
    NAVARRA"),  
  sidebarPanel( #entradas  
    tabsetPanel(  
      tabPanel(title="Sector zona Media Navarra",  
        checkboxGroupInput("variable1","Seleccione las EDAR de  
          recogida",  
choices=c("IRAOLA"="IRAOLA","BARASOAIN"="BARASOAIN","PUEYO  
          "="PUEYO","SAN MARTIN DE UNX"="SAN MARTIN DE  
            UNX","UJUE"="UJUE","OLITE"="OLITE","PITILLAS"=  
              "PITILLAS","SANTACARA"="SANTACARA",  
                "CARCASTILLO"="CARCASTILLO","RADA"="RADA",  
                  "FIGAROL"="FIGAROL"),  
selected=c("IRAOLA"="IRAOLA","BARASOAIN"="BARASOAIN","PUEY  
O  
          "="PUEYO","SAN MARTIN DE UNX"="SAN MARTIN DE  
            UNX","UJUE"="UJUE","OLITE"="OLITE","PITILLAS"=  
              "PITILLAS","SANTACARA"="SANTACARA",  
                "CARCASTILLO"="CARCASTILLO","RADA"="RADA",  
                  "FIGAROL"="FIGAROL"),  
      tabPanel(title="Parámetros algoritmo genético",  
        numericInput("Npop","Seleccione la población inicial",value=6,  
          min=2,max=100, step=1),  
        numericInput("Ngen","Seleccione el número de generaciones",  
          value=2,min=1,max=100, step=1),  
        numericInput("Local","Seleccione el parámetro Local",  
          value=0.05, min=0,max=1, step=0.05),  
        numericInput("Elite","Seleccione el parámetro Elite",value=2,  
          min=1,max=6, step=1))
```

```
),  
width=45 #Anchura de la ventana de la app.  
,  
submitButton("Calcular ruta"),  
mainPanel(textOutput("Distancia"), #salida de app  
# verbatimTextOutput("values"),  
tableOutput("values"),#comparacion algoritmo tsp y genetico  
leafletOutput("mapa"))#coloca el mapa  
)  
server<-function(input,output){  
  slidervalues<-reactive({  
    MatrizTrabajo<-Distancias[c("IRAOLA",c(input$variable1)),  
                                c("IRAOLA",c(input$variable1))]  
    if(length(c("IRAOLA",c(input$variable1)))<10){  
  
    MatrizTrabajo[lower.tri(MatrizTrabajo)]=t(MatrizTrabajo)[lower.tri(Ma  
trizTrabajo)]  
    isSymmetric(MatrizTrabajo)  
    MatrizTSP<-TSP(as.matrix(MatrizTrabajo), labels = NULL)  
    initialtour<-as.integer(which(labels(MatrizTrabajo)[[1]]=="  
                                IRAOLA"))  
    solucion1<-solve_TSP((MatrizTSP),method="farthest_insertion",  
                          control=list(start=initialtour))  
    solucion2<-solve_TSP((MatrizTSP),method="nearest_insertion",  
                          control=list(start=initialtour))  
    Longitud1<-tour_length(solucion1)  
    Longitud2<-tour_length(solucion2)
```

```
Ruta<-list(c(paste("Longitud (km):",round(Longitud1,digits=2)),
              labels(solucion1)),
          c(paste("Longitud(km):",round(Longitud2,digits=2)),
            labels(solucion1)))
Longitudes<-c(Longitud1,Longitud2)
Seleccion<-which.min(Longitudes)
Ruta}

else {

MatrizTrabajo[lower.tri(MatrizTrabajo)]=t(MatrizTrabajo)[lower.tri(Ma
trizTrabajo)]
isSymmetric(MatrizTrabajo)
MatrizAG<-MatrizTrabajo
colnames(MatrizAG)<-NULL
MatrizTSP<-TSP(as.matrix(MatrizTrabajo), labels = NULL)
initialtour<- as.integer(which(labels(MatrizTrabajo)
                               [[1]]=="IRAOLA"))
solucion1<- solve_TSP((MatrizTSP),method="farthest_insertion",
                      control=list(start=initialtour))
solucion2<- solve_TSP((MatrizTSP),method="nearest_insertion",
                      control=list(start=initialtour))
solucion3<- search_tour_genetic(MatrizAG,dim(MatrizAG)
                               [1],input$Npop,input$Ngen,input$Local,input$Elite)
Longitud1<-tour_length(solucion1)
Longitud2<-tour_length(solucion2)
Longitud3<-solucion3$distance
```

```
Ruta<-list(c(paste("Longitud (km):",round(Longitud1/1000,digits
=2)),labels(solucion1)),
c(paste("Longitud (km):",round(Longitud2/1000,digits
=2)),labels(solucion1)),
c(paste("Longitud (km):",round(Longitud3/1000,digits
=2)),colnames(MatrizTrabajo)[solucion3$tour]))
Longitudes<-c(Longitud1,Longitud2,Longitud3)
Seleccion<-which.min(Longitudes)
Ruta
}
})
mapaselect<-reactive({
  if(length(c("IRAOLA",c(input$variable1)))<10){
    coordenadas<-coordenadas[c("IRAOLA",c(input$variable1))]
    coordenadas<-as.data.frame(coordenadas)
    MatrizTrabajo<-Distancias[c("IRAOLA",c(input$variable1)),
c("IRAOLA",c(input$variable1))]

MatrizTrabajo[lower.tri(MatrizTrabajo)]=t(MatrizTrabajo)[lower.tri(Ma
trizTrabajo)]

isSymmetric(MatrizTrabajo)

MatrizTSP<-TSP(as.matrix(MatrizTrabajo), labels = NULL)

initialtour<-
as.integer(which(labels(MatrizTrabajo)[[1]]=="IRAOLA"))

solucion1<-solve_TSP((MatrizTSP),method="farthest_insertion",
control=list(start=initialtour))

solucion2<-solve_TSP((MatrizTSP),method="nearest_insertion",
control=list(start=initialtour))

Longitud1<-tour_length(solucion1)
Longitud2<-tour_length(solucion2)
```

```
Ruta<-list(labels(solucion1),labels(solucion2))
Longitudes<-c(Longitud1,Longitud2)
Seleccion<-which.min(Longitudes)
Rutafinal<-Ruta[[Seleccion]]
coordenadas<-coordenadas[order(Rutafinal),]
coordenadas<-as.data.frame(coordenadas)
ruta<-data.frame(coordenadas$LONGITUD,coordenadas$
                  LATITUD)
names(ruta)<-c("LONGITUD","LATITUD")
ID1=rownames(coordenadas)

ors_api_key('5b3ce3597851110001cf6248c461a41c55e84e7fac8efbd
a9e23fc4e')

route <- ors_directions(asplit(ruta,1), format="geojson",
                        radiuses=1000000)

mapa <- leaflet(data = coordenadas) %>%
  addTiles()
  setView(lng = coordenadas$LONGITUD[1], lat = coordenadas$
          LATITUD[1] , zoom = 1)
  addMarkers(~LONGITUD, ~LATITUD, popup = ~as.character
            (ID1), label = ~as.character(ID1))
  addGeoJSON(route, fill=FALSE)
  fitBBox(route$bbox)}
else{
coordenadas<-coordenadas[c("IRAOLA",c(input$variable1)),]
coordenadas<-as.data.frame(coordenadas)

MatrizTrabajo<-Distancias[c("IRAOLA",c(input$variable1)),
                           c("IRAOLA",c(input$variable1))]
```

```
MatrizTrabajo[lower.tri(MatrizTrabajo)]=t(MatrizTrabajo)[lower.tri(Ma
trizTrabajo)]

isSymmetric(MatrizTrabajo)

MatrizAG<-MatrizTrabajo

colnames(MatrizAG)<-NULL

MatrizTSP<-TSP(as.matrix(MatrizTrabajo), labels = NULL)

initialtour<-
as.integer(which(labels(MatrizTrabajo)[[1]]=="IRAOLA"))

solucion1<-solve_TSP((MatrizTSP),method="farthest_insertion",
                     control=list(start=initialtour))

solucion2<-solve_TSP((MatrizTSP),method="nearest_insertion",
                     control=list(start=initialtour))

solucion3<-search_tour_genetic(MatrizAG,dim(MatrizAG)[1],input$
                              Npop,input$Ngen,input$Local,input$Elite)

Longitud1<-tour_length(solucion1)
Longitud2<-tour_length(solucion2)
Longitud3<-solucion3$distance

Ruta<-
list(labels(solucion1),labels(solucion2),colnames(MatrizTrabajo)[soluc
ion3$tour])

Longitudes<-c(Longitud1,Longitud2,Longitud3)

Seleccion<-which.min(Longitudes)

Rutafinal<-Ruta[[Seleccion]]

coordenadas<-coordenadas[order(Rutafinal),]

coordenadas<-as.data.frame(coordenadas)

ruta<-data.frame(coordenadas$LONGITUD,coordenadas$LATITUD)

names(ruta)<-c("LONGITUD","LATITUD")

ID1=rownames(coordenadas)
```

```
ors_api_key('5b3ce3597851110001cf6248c461a41c55e84e7fac8efbd  
a9e23fc4e')  
  
route <- ors_directions(asplit(ruta,1), format="geojson",  
                        radiuses=100000)  
  
mapa <- leaflet(data = coordenadas)  
  addTiles()  
  setView(lng = coordenadas$LONGITUD[1], lat = coordenadas$  
          LATITUD[1] , zoom = 1)  
  addMarkers(~LONGITUD, ~LATITUD, popup = ~as.character  
            (ID1), label = ~as.character(ID1))  
  addGeoJSON(route, fill=FALSE)  
  fitBBox(route$bbox)  
})  
  
output$values<-renderTable({  
  Tabla<-data.frame(slidervalues())  
  if(length(c("IRAOLA",c(input$variable1)))<10){names(Tabla)<-  
    c("ALGORITMO DEL VECINO MÁS CERCANO",  
      "ALGORITMO DEL VECINO MÁS LEJANO")  
    Tabla}  
  else{names(Tabla)<-c("ALGORITMO DEL VECINO MÁS CERCANO",  
"ALGORITMO DEL VECINO MÁS LEJANO","ALGORITMO GENÉTICO")  
    Tabla}  
  })  
  output$mapa<-renderLeaflet({mapaselect()})  
}  
  
shinyApp(ui = ui, server = server
```




Relación de documentos

(X) Memoria 58 páginas

(_) Anexos 17 páginas

La Almunia, a 29 de 02 de 2024

Firmado: Guillermo Aldunate Ozcariz