



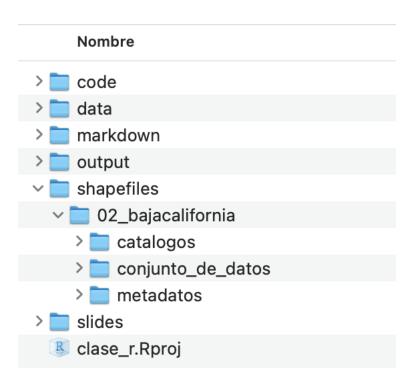
Despliegue de capas de información

José Luis Texcalac Sangrador

Procesos básicos de medición ambiental



- Trabaje en su proyecto "clase_r"
- Genere su script de la sesión
- Genere la carpeta "shapefiles"
- Copie los archivos de Baja California en la carpeta "shapefiles"





Modelo geométrico de rasgo simple (simple feature o simple feature access)

- Estándar formal (ISO 19125-1: 2004) que describe cómo los objetos en el mundo real pueden ser representados en las computadoras, con énfasis en la geometría espacial de estos objetos.
- Estándar de código abierto desarrollado y respaldado por el Open Geospatial Consortium (OGC)
- Permite representar una amplia gama de información geográfica.
- Modelo de datos jerárquico que simplifica los datos geográficos al condensar un amplio rango de formas geográficas en una única clase de geometría.
- Describe cómo los objetos se pueden almacenar y recuper de las bases de datos, y las operaciones geométricas que deberían definirse para ellos.
- Modelo de datos ampliamente respaldado que subyace a las estructuras de datos en muchas aplicaciones de SIG, pemitiendo la transferencia de datos a otras configuraciones (software, librerías, etc.).



- Se considera una cosa o un objeto en el mundo real (un edificio o un árbol)
- Como en la vida areal, los objetos, a menudo conforman otros objetos. Un conjunto de *features* puede formar un solo *feature*.
 - Un árbol puede ser un *feature*, un bosque es un *feature*.
 - Un edificio puede ser un *feature* y una ciudad es un *feature*.
 - Un pixel de una imagen satelital es un feature, y la imagen completa también es un feature.
- Tienen una geometría que describe su posición sobre la superficie terrestre.
- Tienen atributos que describen propiedades adicionales.
 - La geometría de un árbol puede ser la delineación de su corona, de su tallo o el punto que indica su centro.
 - Otras propiedades pueden incluir su altura, color, diámetro a la altura del pecho en una fecha particular, y así sucesivamente.
- El estándar dice: "La OpenGIS Abstract especifica que un *simple feature* posee ambas características, es decir, atributos espaciales y no espaciales."



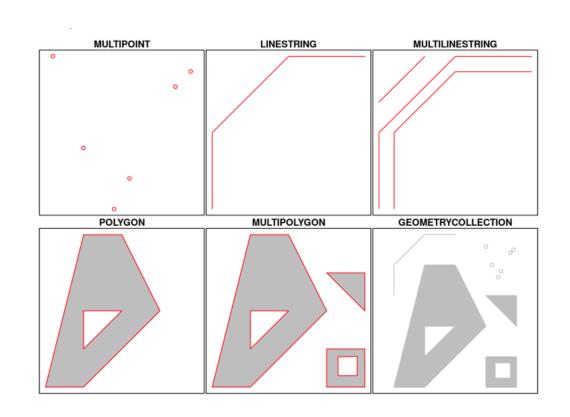
Ventajas del uso del paquete sf

- Proporciona una interfaz casi directa para las funciones GDAL y GEOS C++
- La lectura y escritura de datos son rápidas.
- El rendimiento de despliegue.
- Compatible con los tipos de simple feature utilizados en la mayoría de las operaciones de análisis espacial: puntos, líneas, polígonos y sus respectivas versiones "múltiples".
- Los nombres de las funciones de sf son relativamente consistentes e intuitivas (todos comienzan con st_).
- Los objetos *sf* pueden procesarse como un *tibble* o *data frame*.
- Las funciones pueden combinarse usando el operador pipe %>%
- Funcionan bien con la colección de librerías de tidyverse.



Objetos espaciales en el paquete sf

- Existen diferentes clases de objetos espaciales dependiendo del tipo de información que almacenan
- Permiten gestionar coberturas de puntos, líneas y polígonos.
 - Point y Multipoint,
 - Linestring y Multilinestring
 - Polygon y Multipolygon
- La clase *Geometrycollection* permite juntar, en un mismo objeto, diferentes geometrías.









- 02mun.cpg
- 02mun.dbf
- 02mun.prj
- 02mun.shp
- 02mun.shx

Shapefile



Un shapefile se compone de varios Archivos.

Los mínimo son tres:

Almacena las entidades geométricas. El .shp

El .shx Almacena el índice de las entidades geométricas.

El .dbf Es la base de datos, en formato dBASE

También pueden tener un .prj, .sbn, .sbx, .fbn, .fbx .ain, .aih, .shp.xml.



TIPO DE DATOS **QUE ALMACENA.**

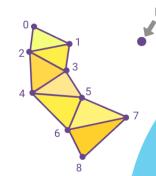
Linea

Punto



Polígono





MultiPach

Es el formato más extendido y popular entre la comunidad GIS.

MultiPunto

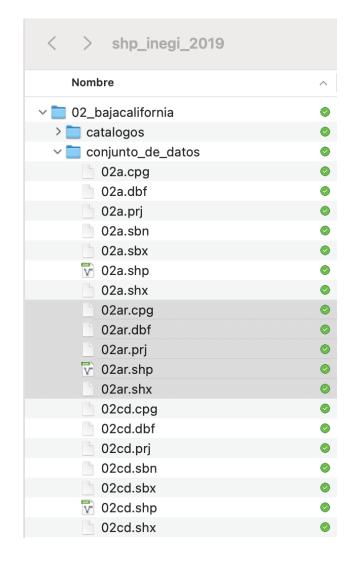


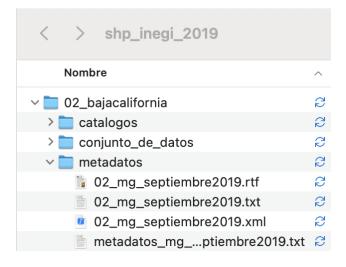




Shapefiles del marco geoestadístico nacional

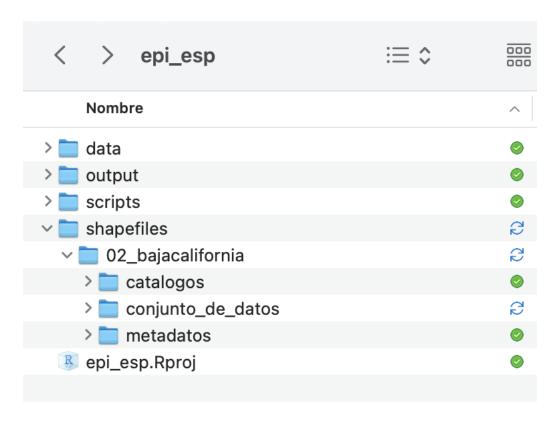
| < > shp_inegi_2019 | | | | | |
|----------------------------------|---|--|--|--|--|
| Nombre | ^ | | | | |
| ∨ 📄 02_bajacalifornia | | | | | |
| √ | | | | | |
| contenido.pdf | 2 | | | | |
| contenido.txt | 2 | | | | |
| entidades_federativas.csv | S | | | | |
| entidades_federativas.pdf | 2 | | | | |
| leme.pdf | 2 | | | | |
| leme.txt | 2 | | | | |
| localidades_islas.csv | 2 | | | | |
| localidades_islas.pdf | 2 | | | | |
| localidades_rurabasan_AGEB.csv | 2 | | | | |
| localidades_rurabasan_AGEB.pdf | 2 | | | | |
| localidades_rurasu_municipio.csv | 2 | | | | |
| localidades_rurasu_municipio.pdf | 2 | | | | |
| localidades_urbsu_municipio.csv | 2 | | | | |
| localidades_urbsu_municipio.pdf | 2 | | | | |
| localidades_urbamanzanadas.csv | 2 | | | | |
| localidades_urbamanzanadas.pdf | 2 | | | | |
| municipios.csv | 8 | | | | |
| municipios.pdf | 2 | | | | |
| > 📄 conjunto_de_datos | 3 | | | | |
| > 🛅 metadatos | 2 | | | | |







Shapefiles en nuestro proyecto





Importando datos con atributos espaciales en

library(sf)

```
objeto <-
  st_read("ruta_archivo.shp",
     stringsAsFactors = FALSE,
     options = "ENCODING = WINDOWS-1252")
```

Alias:

```
read_sf("ruta_archivo.shp")
```



Estructura de un simple feature

> bc_mun

Simple feature collection with 5 features and 4 fields

geometry type: MULTIPOLYGON

dimension: XY

bbox: xmin: 911292 ymin: 1810279 xmax: 1493532 ymax: 2349615

projected CRS: MEXICO_ITRF_2008_LCC

| | CVEGE0 | CVE_ENT | CVE_MUN | NOMGEO | | g€ | eometry |
|---|--------|---------|--------------|--------------------|--------------|------------|---------|
| 1 | 02001 | 02 | 001 | Ensenada | MULTIPOLYGON | (((1493197 | 184 |
| 2 | 02002 | 02 | ↑ 002 | Mexicali | MULTIPOLYGON | (((1437921 | 188 |
| 3 | 02003 | 02 | 003 | Tecate | MULTIPOLYGON | (((1113633 | 234 |
| 4 | 02004 | 02 | 004 | Tijuana | MULTIPOLYGON | (((1058854 | 233.\. |
| 5 | 02005 | 02 | 005 | Playas de Rosarito | MULTIPOLYGON | (((1077326 | 233 |
| | | | | | <u> </u> | | |
| | | | ' | | I | | |

Simple feature

Simple feature geometry list-column (sfc)

Simple feature geomtery (sfg)



Polígonos

- Cargue la capa "O2mun"
- Nombre al nuevo objeto como bc_mun

```
library(tidyverse)
library(sf)

objeto <-
   st_read("ruta_archivo.shp") %>%
   print()
```

Leyendo datos de Baja California

```
bc_mun <-
    st_read("./shapefiles/02_bajacalifornia/conjunto_de_datos/02mun.shp") %>%
    clean_names() %>%
    print()
```

```
> bc_mun <-
    st_read("./shapefiles/02_bajacalifornia/conjunto_de_datos/02mun.shp") %>%
    clean_names() %>%
   print( )
Reading layer `02mun' from data source
  `/Users/tex/OneDrive/Docencia/PASPE_R_EPI_Espacial/epi_esp/shapefiles/02_bajacalifornia/conjunto_de_dato
s/02mun.shp'
  using driver `ESRI Shapefile'
Simple feature collection with 5 features and 4 fields
Geometry type: MULTIPOLYGON
Dimension:
              XY
Bounding box: xmin: 911292 ymin: 1810279 xmax: 1493532 ymax: 2349615
Projected CRS: MEXICO_ITRF_2008_LCC
Simple feature collection with 5 features and 4 fields
Geometry type: MULTIPOLYGON
Dimension:
               XΥ
Bounding box: xmin: 911292 ymin: 1810279 xmax: 1493532 ymax: 2349615
Projected CRS: MEXICO_ITRF_2008_LCC
  cvegeo cve_ent cve_mun
                                                                  geometry
                                     nomgeo
1 02001
                                   Ensenada MULTIPOLYGON (((1493197 184...
              02
                     001
                                   Mexicali MULTIPOLYGON (((1437921 188...
2 02002
              02
                    002
              02
                    003
  02003
                                     Tecate MULTIPOLYGON (((1113633 234...
                                    Tijuana MULTIPOLYGON (((1058854 233...
4 02004
             02
                    004
              02
5 02005
                    005 Playas de Rosarito MULTIPOLYGON (((1077326 233...
```



plot(st_geometry(bc_mun))

- st_geometry nos permite visualizar sólo la geometría de la capa.
- Intente el plot omitiendo st_geometry plot(bc_mun)



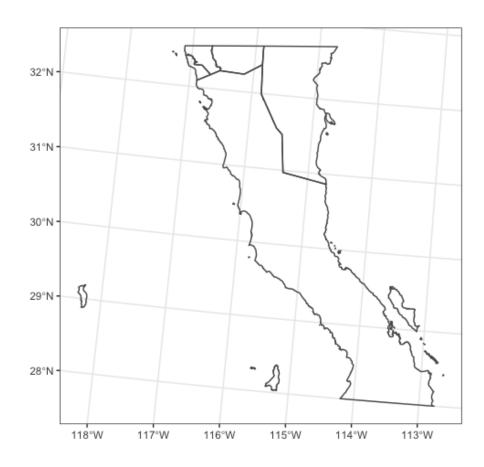
Mapa básico con ggplot

```
ggplot( ) +
  geom_sf( ) +
  theme( )
```



```
ggplot() +
  geom_sf(data = bc_mun, fill = "transparent" ) +
  theme_bw()
```



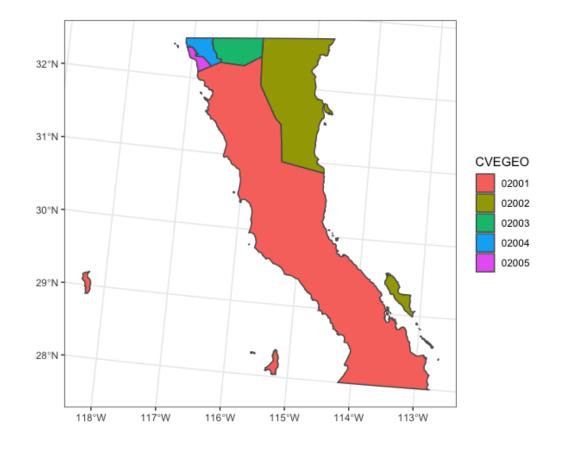


```
INSP/ESPM
ESCUELA DE SALUD
PÚBLICA DE MÉXICO
```

```
ggplot() +
  geom_sf(data = bc_mun, aes(fill = cvegeo)) +
  theme_bw()
```

Podemos colorear polígonos tomando como base los valores de sus atributos.

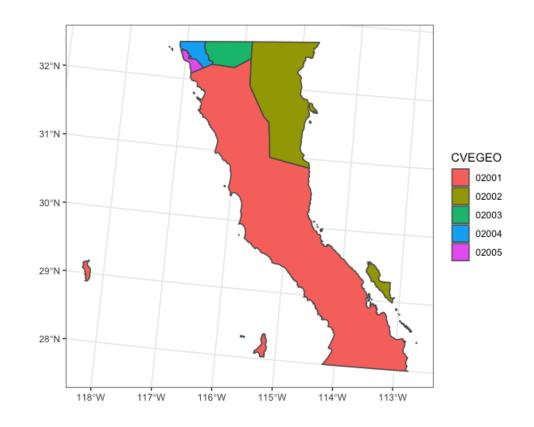
- Población
- Contaminación
- Pobreza
- Vulnerabilidad



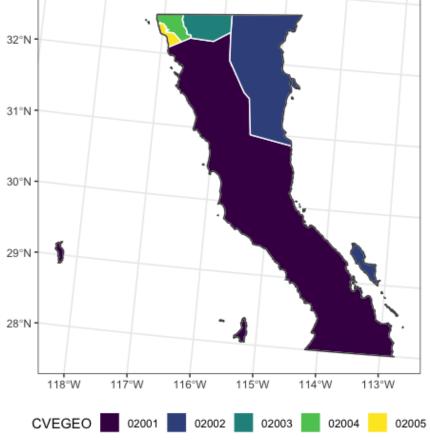


Su turno...

- Cargue la capa de entidad como bc_ent
- Contorno de municipios blanco
- Contorno de entidad con grosor de 1.5
- Paleta de colores viridis
- Aplique el tema bw
- Leyenda en la parte baja



```
ggplot() +
  geom_sf(data = bc_mun, aes(fill = cvegeo), colour = "white") +
  geom_sf(data = bc_ent, fill = "transparent", colour = "grey30", size = 1.5) +
  scale_fill_viridis(discrete = T) +
  theme_bw() +
  theme(legend.position = "bottom")
```





Líneas



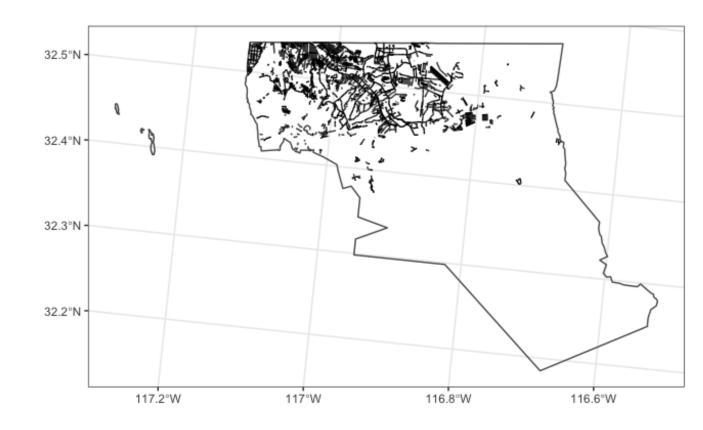
- Cargue la capa 02e.shp
- Nombre al objeto como bc_vial
 - ¿Qué valores distintos tiene la columna tipovial?
 - Mantenga en la capa a los siguientes tipos de vialidad: Avenida, Boulevard, Calzada y Carretera.
- A partir de bc_vial genere una nueva capa de nombre tij_vial
 - Mantenga en su capa sólo las vialidades del municipio de Tijuana
- Elija unicamente al municipio de Tijuana en bc_mun y llame a la nueva capa tij_mun
- Genere un mapa con ambas capas



```
ggplot()
  geom_sf data = tij_mun, fill = "transparent") +
  geom_sf(data = tij_vial) +
  theme_bw()
```

¿Qué más podemos hacer?

¿Colores de avenidas?

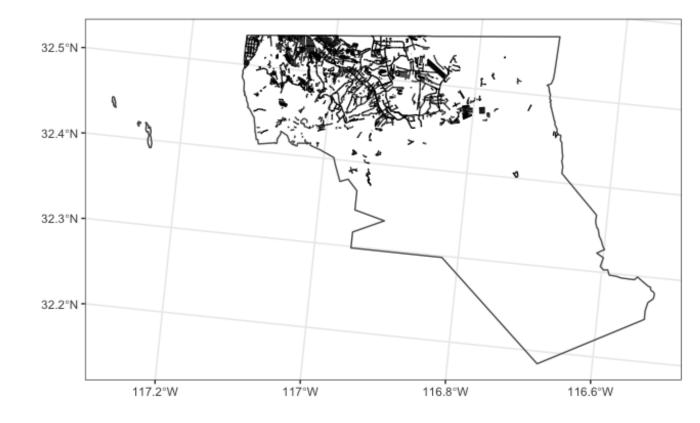




```
ggplot()
  geom_sf data = tij_mun, fill = "transparent") +
  geom_sf(data = tij_vial) +
  theme_bw()
```

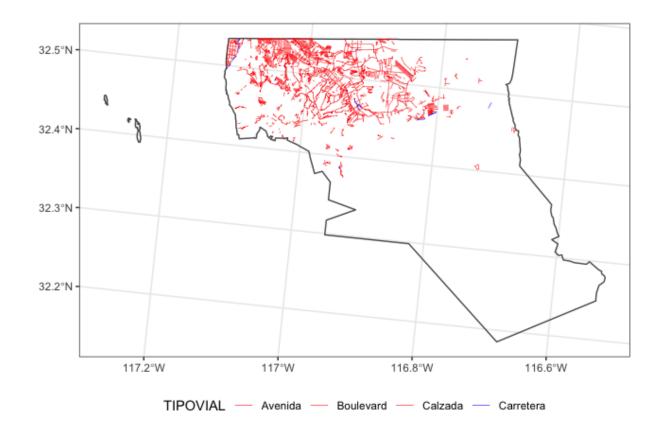
¿Qué más podemos hacer?

Boulevard, Calzada y Avenida son casi lo mismo, deberían ser un solo color que las diferencíe de Carretera.





```
ggplot()
  geom_sf data = tij_mun, fill = "transparent") +
  geom_sf(data = tij_vial, aes(colour = tipovial), size = 0.2) +
  scale_linetype_manual(values = c("solid", "solid", "solid", "dashed")) +
  scale_color_manual(values = c('red', 'red', 'red', 'blue')) +
  theme_bw()
  theme(legend.position = "bottom")
```





Puntos

Su turno

- Cargue la capa O2sip.shp y nombre al objeto como bc_serv_punt
- Cargue las capas de municipio, puntual y eje vial
 - Elija sólo los *features* del municipio de Mexicali
 - Nombre los objetos nuevos como: mxc_mun, mxc_ serv_punt, mxc_vial

```
mxc_mun <-
bc_mun %>%
filter(cve_mun == "002") %>%
print()

plot(st_geometry(mxc_mun))
```

Su turno...

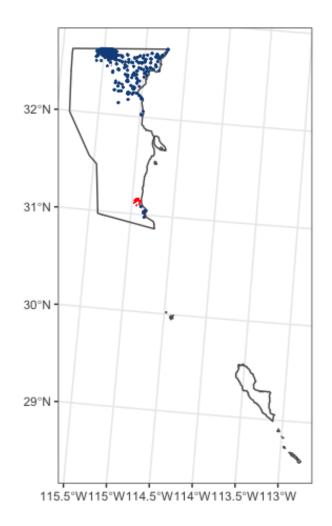
Genere un mapa con las capas de Mexicali

- Vialidades todas de color rojo y tamaño 0.2
- Puntos de color "dodgerblue4" y tamaño 0.3
- Tema: theme_bw

```
ggplot() +
  geom_sf(data = mxc_mun, fill = "transparent") +
  geom_sf(data = mxc_vial, colour = "red", show.legend = "line", size = 0.2) +
  geom_sf(data = mxc_serv_punt, size = 0.3, color = "dodgerblue4") +
  theme_bw()
```

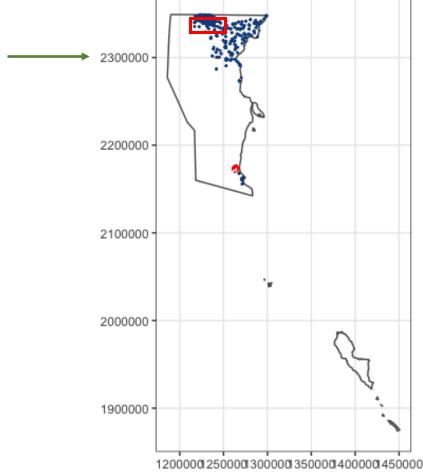
```
INSP/ESPM
ESCUELA DE SALUD
PÚBLICA DE MÉXICO
```

```
ggplot() +
  geom_sf(data = mxc_mun, fill = "transparent") +
  geom_sf(data = mxc_vial, colour = "red", size = 0.2) +
  geom_sf(data = mxc_serv_punt, size = 0.3, color = "dodgerblue4") +
  theme_bw()
```

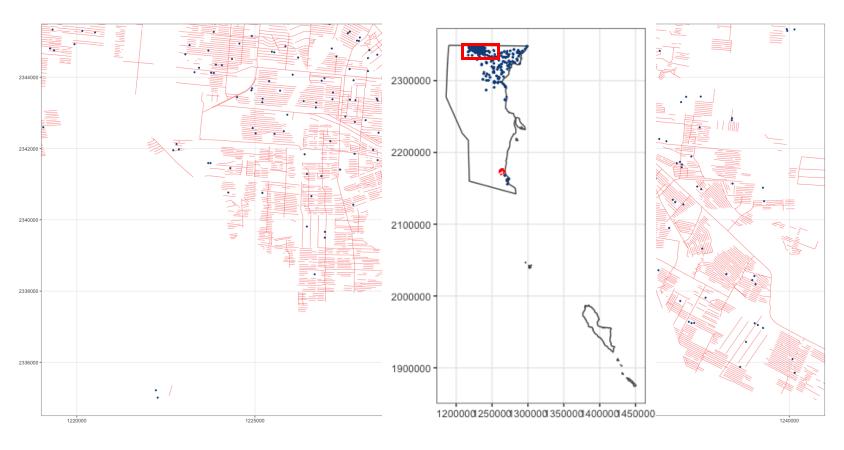




```
ggplot() +
  geom_sf(data = mxc_mun, fill = "transparent") +
  geom_sf(data = mxc_vial, colour = "red", size = 0.2) +
  geom_sf(data = mxc_serv_punt, size = 0.3, color = "dodgerblue4") +
  coord_sf(datum = st_crs(mxc_mun)) +
  theme_bw()
```









```
ggplot() +
    geom_sf(data = mxc_mun, fill = "transparent") +
    geom_sf(data = mxc_vial, colour = "red", size = 0.2) +
    geom_sf(data = mxc_serv_punt, size = 0.3, color = "dodgerblue4") +
    coord_sf(xlim = c(1220000, 1240000),
        ylim = c(2335000, 2345000)) +
    theme_bw()
```



Exportar shapefile

```
st_write(objeto, "ruta/nombre_archivo.shp")
st_write(tij_mun, "./capas/tij_mun.shp")
```