Utah Car Accidents Data and Code

Kenneth Pomeyie, Scout Jarman, Paul Gaona-Partida

Load Libraries and Data file

```
# For Data Cleaning and Visualization
library("dplyr")
```

```
## Warning: package 'dplyr' was built under R version 4.1.1
```

```
##
## Attaching package: 'dplyr'
```

```
## The following objects are masked from 'package:stats':
##
##     filter, lag
```

```
## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```
library("tigris")
```

```
## Warning: package 'tigris' was built under R version 4.1.1
```

```
## To enable
## caching of data, set `options(tigris_use_cache = TRUE)` in your R script or .Rprofile.
```

```
library("ggplot2")
```

```
## Warning: package 'ggplot2' was built under R version 4.1.1
```

```
library("sf")
```

```
## Warning: package 'sf' was built under R version 4.1.1
```

```
## Linking to GEOS 3.9.1, GDAL 3.2.3, PROJ 7.2.1; sf_use_s2() is TRUE
```

```
crash <- read.csv("../data-raw/RawCrashData2020.csv")
stations <- read.csv("../data-raw/i_15_Flow_Data_2020.csv")
```

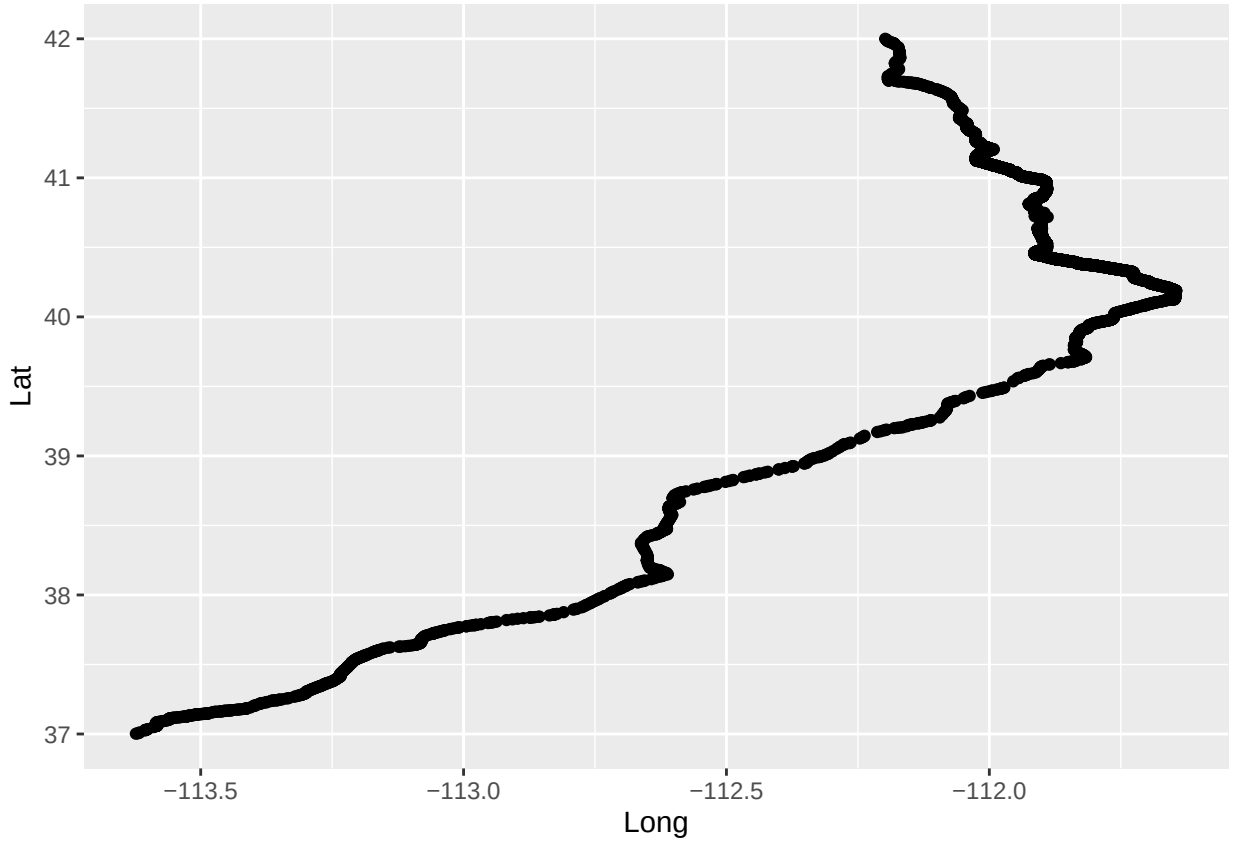
Data Prepping

Car Accidents

```
# subset to retain only potentially useful information
# Route: I-15 Highway route
# milepoint: Milepoint of accident
# Coords: Lat and Long

crash_i15 <- crash %>%
```

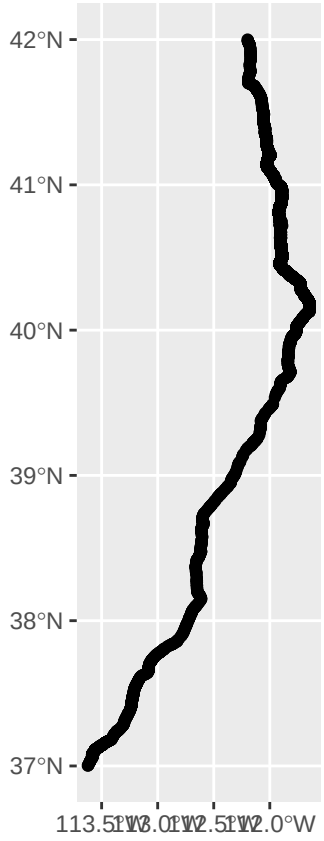
```
  dplyr::filter(Route == "0015") %>%
  dplyr::select(.,Lat, Long)
```

```
# Basic plot of points of accidents
ggplot() +
  geom_point(
    data = crash_i15,
    aes(x = Long, y = Lat)
  )
```




```
# Sf with NAD83 matches CRS of Utah map and Roads
i15_spat <- sf::st_as_sf(
  x = crash_i15,
  coords = c("Long", "Lat"),
  crs = 4269
)
```

```
# Plot of spatial object crashes
crash_plot <- ggplot(i15_spat) +
  geom_sf()
crash_plot
```



Roads

https://www2.census.gov/geo/pdfs/maps-data/data/tiger/tgrshp2020/TGRSHP2020_TechDoc.pdf

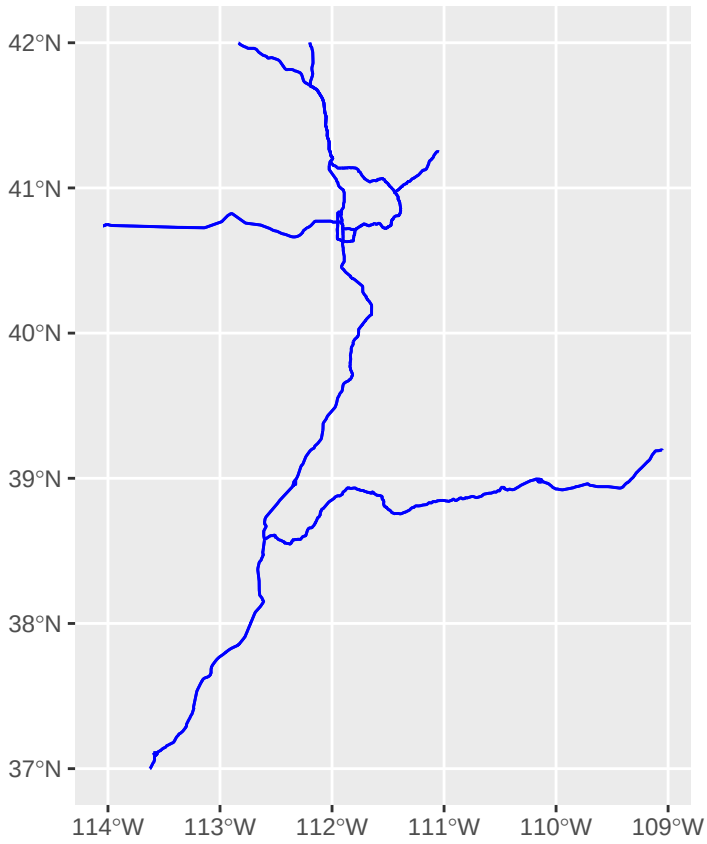
Link: Census.gob information on Tigris shapefiles

Important: Original CRS = NAD83

```
# subset of Interstates in Utah
roads <- tigris::primary_secondary_roads("Utah") %>%
  dplyr::filter(RTTYP %in% c("I"))
```

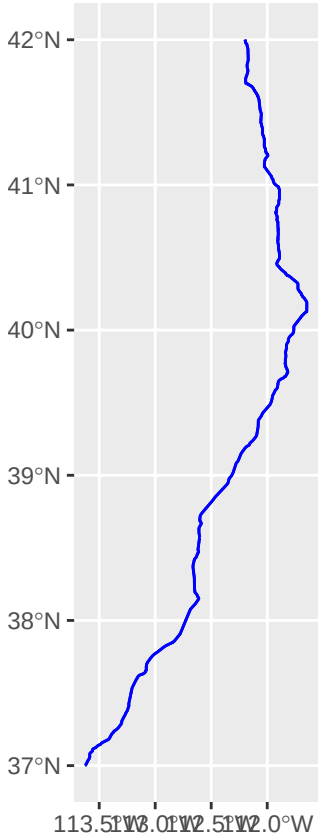
```
##   |                                                                      |
```

```
# plot of ALL Interstates in Utah (Utah not fitted)
ggplot() +
  geom_sf(
    data = roads,
    color = "blue",
    aes(geometry = geometry)
  )
```




```
# subset of I-15 and convert to MultiLineString
roads_i15 <- roads[which(roads$FULLNAME == "I- 15"), ] %>%
  sf::st_cast(., "MULTILINESTRING")
```

```
# plot of i-15 in Utah (Interstate only)
road_plot <- ggplot() +
  geom_sf(
    data = roads_i15,
    color = "blue",
    aes(geometry = geometry)
  )
road_plot
```



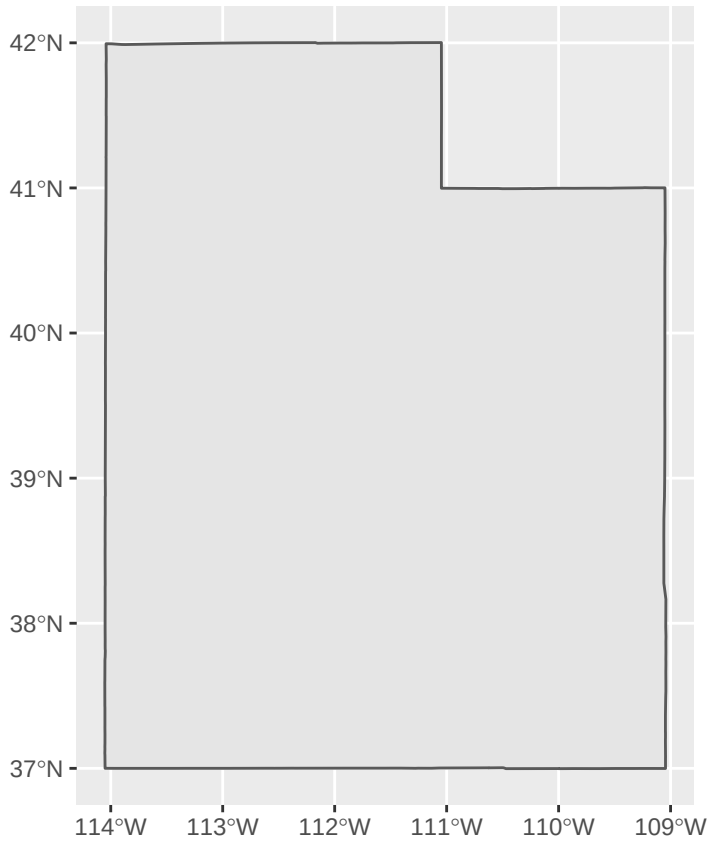
Utah

```
# sf of USA
us_geo <- tigris::states(class = "sf", cb = TRUE) %>%
  shift_geometry()
```

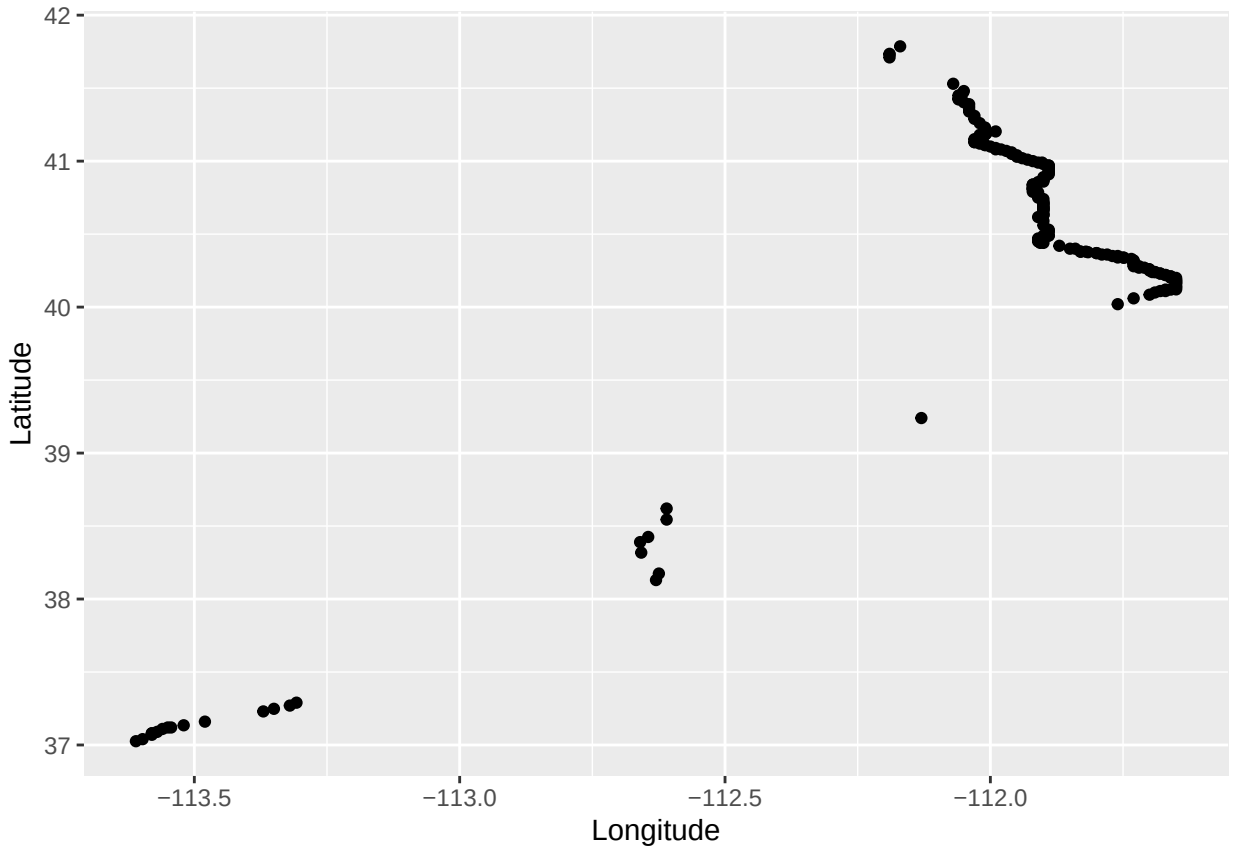
```
##   |                                                                              |
```

```
# Subsetting to Utah and changing crs to match roads in Utah
ut_map <- us_geo[which(us_geo$NAME == "Utah"), ] %>%
  st_transform(., crs = st_crs(roads_i15))
```

```
# plot of Utah
ut_plot <- ggplot(ut_map) +
  geom_sf()
ut_plot
```

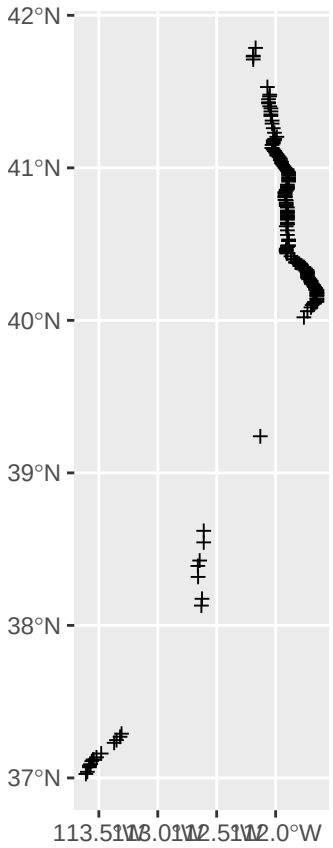



```
# plot of Stations
ggplot() +
  geom_point(
    data = stations,
    aes(x = Longitude, y = Latitude)
  )
```

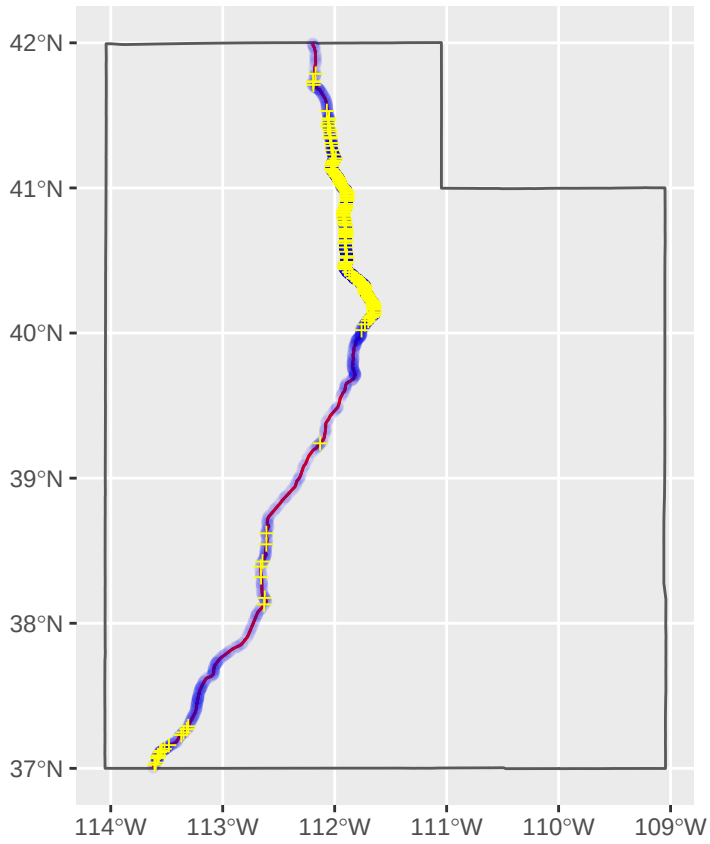



```
# Sf with NAD83 matches CRS of Utah map and Roads
stat_spat <- sf::st_as_sf(
  x = stations,
  coords = c("Longitude", "Latitude"),
  crs = 4269
)
```

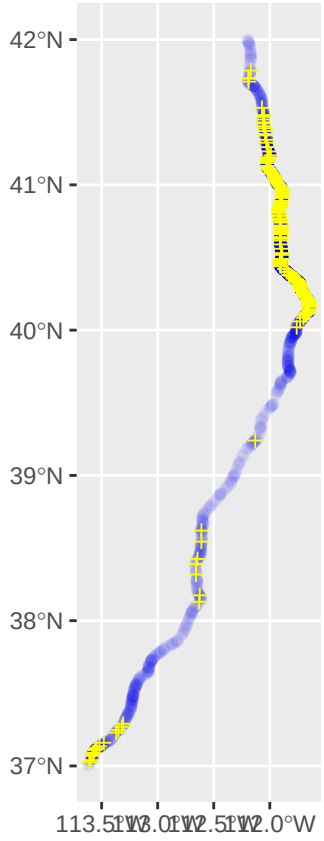
```
# Plot of spatial object crashes
station_plot <- ggplot(stat_spat) +
  geom_sf(shape = 3)
station_plot
```




```
# plot of Utah, I-15, and Car Accidents
all_plot <- ggplot(ut_map) +
  geom_sf(fill = NA) +
  geom_sf(
    data = roads_i15,
    color = "red"
  ) +
  geom_sf(
    data = i15_spat,
    color = "blue",
    alpha = 0.025
  ) +
  geom_sf(
    data = stat_spat,
          shape = 3,
    color = "yellow"
    )
all_plot
```




```
# car crashes and stations
ggplot()+
  geom_sf(
    data = i15_spat,
    color = "blue",
    alpha = 0.025
  ) +
  geom_sf(
    data = stat_spat,
          shape = 3,
    color = "yellow"
    )
```




```
library(ggmap)
```

```
## Warning: package 'ggmap' was built under R version 4.1.1
```

```
## Google's Terms of Service: https://cloud.google.com/maps-platform/terms/.
```

```
## Please cite ggmap if you use it! See citation("ggmap") for details.
```

```
tbbox <- sf::st_bbox(ut_map)
names(tbbox) <- c("left", "bottom", "right", "top")
samp_map <- ggmap::get_map(location = tbbox,
                           maptype = "terrain",
                           source = "google"
)
```

```
## Source : http://tile.stamen.com/terrain/8/46/95.png
```

```
## Source : http://tile.stamen.com/terrain/8/47/95.png
```

```
## Source : http://tile.stamen.com/terrain/8/48/95.png
```

```
## Source : http://tile.stamen.com/terrain/8/49/95.png
```

```
## Source : http://tile.stamen.com/terrain/8/50/95.png
```

```
## Source : http://tile.stamen.com/terrain/8/46/96.png
```

```
## Source : http://tile.stamen.com/terrain/8/47/96.png
```

```
## Source : http://tile.stamen.com/terrain/8/48/96.png
```

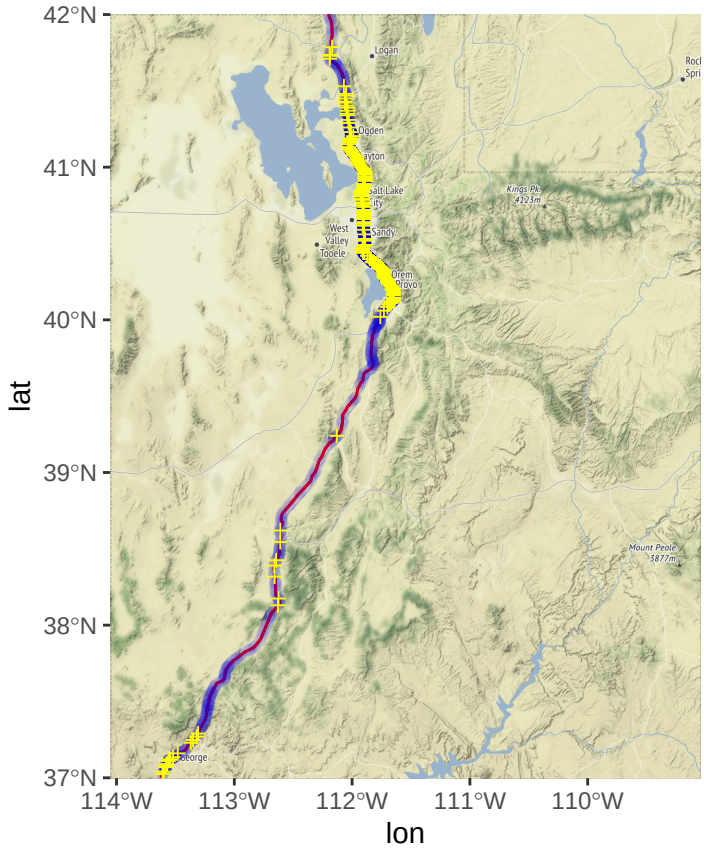
```
## Source : http://tile.stamen.com/terrain/8/49/96.png
```

```
## Source : http://tile.stamen.com/terrain/8/50/96.png
```

```
## Source : http://tile.stamen.com/terrain/8/46/97.png
## Source : http://tile.stamen.com/terrain/8/47/97.png
## Source : http://tile.stamen.com/terrain/8/48/97.png
## Source : http://tile.stamen.com/terrain/8/49/97.png
## Source : http://tile.stamen.com/terrain/8/50/97.png
## Source : http://tile.stamen.com/terrain/8/46/98.png
## Source : http://tile.stamen.com/terrain/8/47/98.png
## Source : http://tile.stamen.com/terrain/8/48/98.png
## Source : http://tile.stamen.com/terrain/8/49/98.png
## Source : http://tile.stamen.com/terrain/8/50/98.png
## Source : http://tile.stamen.com/terrain/8/46/99.png
## Source : http://tile.stamen.com/terrain/8/47/99.png
## Source : http://tile.stamen.com/terrain/8/48/99.png
## Source : http://tile.stamen.com/terrain/8/49/99.png
## Source : http://tile.stamen.com/terrain/8/50/99.png
```

```r
# map is slightly off
ggmap(samp_map) +
  geom_sf(
    data = roads_i15,
    inherit.aes = F,
    color = "red"
  ) +
  geom_sf(
    data = i15_spat,
    inherit.aes = F,
    color = "blue",
    alpha = 0.025
  ) +
  geom_sf(
    data = stat_spat,
    inherit.aes = F,
    shape = 3,
    color = "yellow",
  )
```

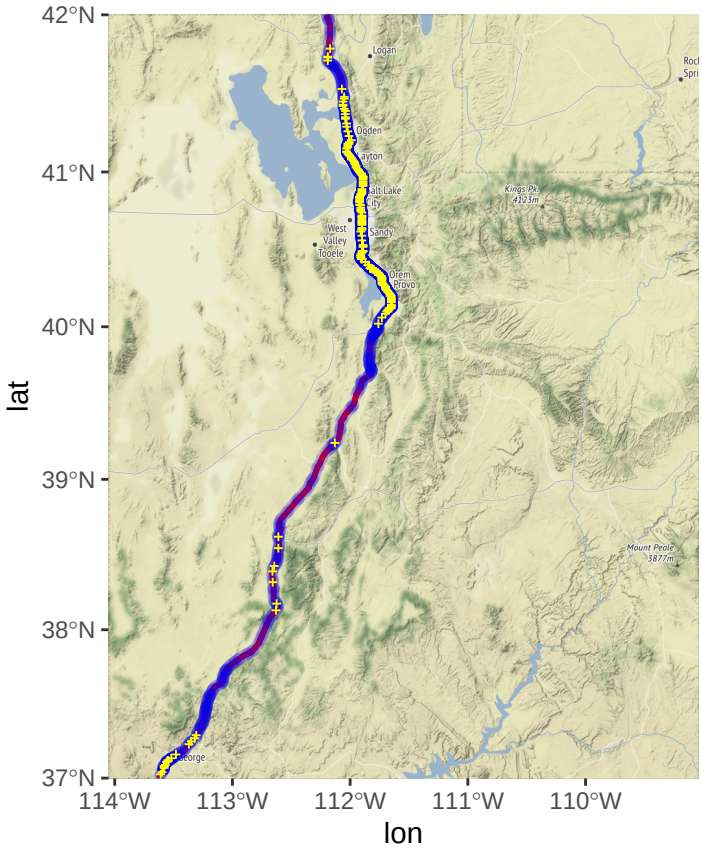
```
## Coordinate system already present. Adding new coordinate system, which will replace the existing one
```




```
roads_3857 <- sf::st_transform(roads_i15,3857)
crash_3857 <- sf::st_transform(i15_spat,3857)
station_3857 <- sf::st_transform(stat_spat,3857)

ggmapbbox <- function(map) {
  if (!inherits(map, "ggmap")) stop("map must be a ggmap object")
  # Extract the bounding box (in lat/lon) from the ggmap to a numeric vector,
  # and set the names to what sf::st_bbox expects:
  map_bbox <- setNames(unlist(attr(map, "bb")),
                       c("ymin", "xmin", "ymax", "xmax"))

  # Coonvert the bbox to an sf polygon, transform it to 3857,
  # and convert back to a bbox (convoluted, but it works)
  bbox_3857 <- st_bbox(st_transform(st_as_sfc(st_bbox(map_bbox, crs = 4326)), 3857))

  # Overwrite the bbox of the ggmap object with the transformed coordinates
  attr(map, "bb")$ll.lat <- bbox_3857["ymin"]
  attr(map, "bb")$ll.lon <- bbox_3857["xmin"]
  attr(map, "bb")$ur.lat <- bbox_3857["ymax"]
  attr(map, "bb")$ur.lon <- bbox_3857["xmax"]
  map
}

# final map
samp_map2 <- ggmapbbox(samp_map)

ggmap(samp_map2) +
```

```
  geom_sf(
    data = roads_3857,
    inherit.aes = FALSE,
    fill = NA,
    lwd = 1,
    color = "red"
  ) +
  geom_sf(
    data = crash_3857,
    inherit.aes = FALSE,
    color = "blue",
    alpha = 0.05
    ) +
  geom_sf(
    data = station_3857,
    inherit.aes = FALSE,
    shape = 3,
    color = "yellow",
    lwd = .75,
  )
```

```
## Coordinate system already present. Adding new coordinate system, which will replace the existing one
```




```
#save code

#roads_i15
#i15_spat
```

```r
#stat_spat

save(roads_i15, file = "../data-raw/RObject/roads_i15")
save(i15_spat, file = "../data-raw/RObject/i15_spat")
save(stat_spat, file = "../data-raw/RObject/stat_spat")
```