ACTION Report 1 2020

Alejandro Alija, PhD

11/10/2020

Pre-requisitos

```
#Installing dependencies
## First specify the packages of interest
packages = c("tidyverse", "dplyr",
             "ggplot2", "plotly", "readr",
             "lubridate", "tibbletime",
             "timetk", "modeltime",
             "tidymodels", "data.table")

## Now load or install&load all
package.check <- lapply(
  packages,
  FUN = function(x) {
    if (!require(x, character.only = TRUE)) {
      install.packages(x, dependencies = TRUE)
      library(x, character.only = TRUE)
    }
  }
)
```

```
#Setting the working directory
setwd("/Users/alija.alejandro/Documents/RProjects/REDES_Report_1_2020")
```

Descarga e Importación del conjunto de datos

```
#Following the pattern

if (dir.exists(".files") == FALSE)
  dir.create("./files")
```

```
## Warning in dir.create("./files"): './files' already exists
```

```
setwd("./files")

datasets <- c("https://datos.madrid.es/egob/catalogo/300228-12-accidentes-trafico-
detalle.csv",
              "https://datos.madrid.es/egob/catalogo/300228-13-accidentes-trafico-
detalle.csv",
              "https://datos.madrid.es/egob/catalogo/300228-14-accidentes-trafico-
detalle.csv",
              "https://datos.madrid.es/egob/catalogo/300228-15-accidentes-trafico-
detalle.csv",
              "https://datos.madrid.es/egob/catalogo/300228-16-accidentes-trafico-
```

```
detalle.csv",
              "https://datos.madrid.es/egob/catalogo/300228-17-accidentes-trafico-
detalle.csv",
              "https://datos.madrid.es/egob/catalogo/300228-18-accidentes-trafico-
detalle.csv",
              "https://datos.madrid.es/egob/catalogo/300228-19-accidentes-trafico-
detalle.csv",
              "https://datos.madrid.es/egob/catalogo/300228-21-accidentes-trafico-
detalle.csv"
)

dt <- list()
for (i in 1:length(datasets)){
  files <- c("traffic2012",
             "traffic2013",
             "traffic2014",
             "traffic2015",
             "trafic2016",
             "trafic2017",
             "trafic2018",
             "trafic2019",
             "trafic2020")
  #Uncomment the following line if you want donwload the files (e.g if this is the
first time you execute the notebook)

  #download.file(datasets[i], files[i])
  filelist <- list.files(".")
  print(i)
  dt[i] <- lapply(filelist[i], read_delim, ";", escape_double = FALSE,
                  locale = locale(encoding = "WINDOWS-1252"),
                  trim_ws = TRUE)
}
```

```
## [1] 1
## [1] 2
## [1] 3
## [1] 4
## [1] 5
## [1] 6
## [1] 7
## [1] 8
## [1] 9
```

```
## Warning: Missing column names filled in: 'X14' [14], 'X15' [15]
```

```
## Warning: 1 parsing failure.
##  row col           expected actual         file
## 9874 X15 1/0/T/F/TRUE/FALSE      , 'trafic2020'
```

```
traffic<-rbindlist(dt, use.names=TRUE, fill=TRUE)
traffic <- setDT(traffic)
```

Converting formats

```
#Formating the Date and some other data types

traffic$FECHA <- dmy(traffic$FECHA)
traffic$`TIPO ACCIDENTE` <- as.factor(traffic$`TIPO ACCIDENTE`)
traffic$`TIPO VEHÍCULO` <- as.factor(traffic$`TIPO VEHÍCULO`)
traffic$`TIPO PERSONA` <- as.factor(traffic$`TIPO VEHÍCULO`)
traffic$`ESTADO METEREOLÓGICO` <- as.factor(traffic$`ESTADO METEREOLÓGICO`)
traffic$`RANGO EDAD` <- as.factor(traffic$`RANGO EDAD`)
traffic$SEXO <- as.factor(traffic$SEXO)
traffic$SEXO <- toupper(traffic$SEXO )
```

Algunas figuras básicas

```
print(summary(traffic))
```

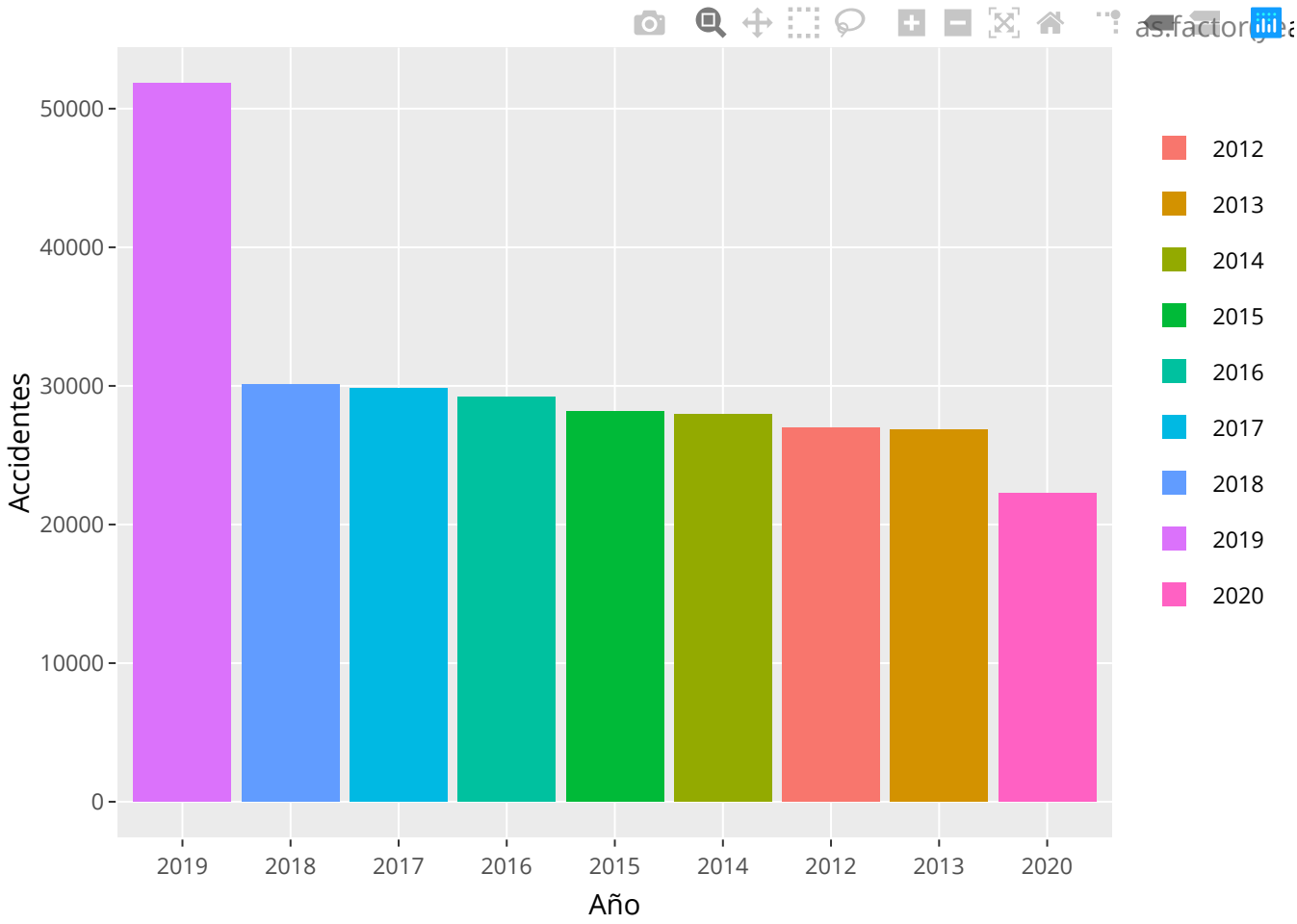
```
##      FECHA             RANGO HORARIO       DIA SEMANA          DISTRITO
##  Min.   :2012-01-01   Length:273147      Length:273147      Length:273147
##  1st Qu.:2014-07-09   Class :character   Class :character   Class :character
##  Median :2016-11-29   Mode  :character   Mode  :character   Mode  :character
##  Mean   :2016-09-20
##  3rd Qu.:2019-02-11
##  Max.   :2020-09-30
##
##  LUGAR ACCIDENTE          N°            N° PARTE         CPFA Granizo
##  Length:273147      Min.   :    0.0   Length:273147      Length:273147
##  Class :character   1st Qu.:    0.0   Class :character   Class :character
##  Mode  :character   Median :    1.0   Mode  :character   Mode  :character
##                     Mean   :  968.7
##                     3rd Qu.:   50.0
##                     Max.   :53500.0
##                     NA's   :77156
##   CPFA Hielo        CPFA Lluvia        CPFA Niebla         CPFA Seco
##  Length:273147      Length:273147      Length:273147      Length:273147
##  Class :character   Class :character   Class :character   Class :character
##  Mode  :character   Mode  :character   Mode  :character   Mode  :character
##
##
##
##
##   CPFA Nieve        CPSV Mojada        CPSV Aceite         CPSV Barro
##  Length:273147      Length:273147      Length:273147      Length:273147
##  Class :character   Class :character   Class :character   Class :character
##  Mode  :character   Mode  :character   Mode  :character   Mode  :character
##
##
##
##
##  CPSV Grava Suelta   CPSV Hielo        CPSV Seca Y Limpia N° VICTIMAS *
##  Length:273147      Length:273147      Length:273147      Min.   : 1.00
##  Class :character   Class :character   Class :character   1st Qu.: 1.00
##  Mode  :character   Mode  :character   Mode  :character   Median : 1.00
```

```
##                                                          Mean   : 1.46
##                                                          3rd Qu.: 2.00
##                                                          Max.   :19.00
##                                                          NA's   :104191
##                   TIPO ACCIDENTE   Tipo Vehiculo
##  COLISIÓN DOBLE         :114867   Length:273147
##  ATROPELLO              : 31775   Class :character
##  COLISIÓN MÚLTIPLE      : 22802   Mode  :character
##  Colisión fronto-lateral: 18365
##  Alcance                : 16750
##  (Other)                : 68563
##  NA's                   :    25
##                    TIPO PERSONA        SEXO            LESIVIDAD
##  Turismo                : 51655   Length:273147      Length:273147
##  Furgoneta              :  4663   Class :character   Class :character
##  Motocicleta > 125cc    :  4525   Mode  :character   Mode  :character
##  Motocicleta hasta 125cc:  4133
##  Autobús                :  1840
##  (Other)                :  6965
##  NA's                   :199366
##   Tramo Edad        * Nº VICTIMAS    Nº  EXPEDIENTE         HORA
##  Length:273147      Min.   : 1.00    Length:273147      Length:273147
##  Class :character   1st Qu.: 1.00    Class :character   Class1:hms
##  Mode  :character   Median : 1.00    Mode  :character   Class2:difftime
##                     Mean   : 1.46                       Mode  :numeric
##                     3rd Qu.: 2.00
##                     Max.   :19.00
##                     NA's   :243025
##     CALLE              NÚMERO              ESTADO METEREOLÓGICO
##  Length:273147      Length:273147      Despejado     : 57762
##  Class :character   Class :character   Lluvia débil  :  4349
##  Mode  :character   Mode  :character   Nublado       :  2897
##                                        Se desconoce  :   957
##                                        LLuvia intensa:   779
##                                        (Other)       :    22
##                                        NA's          :206381
##                   TIPO VEHÍCULO              RANGO EDAD      LESIVIDAD*
##  Turismo                : 51655   DE 40 A 44 AÑOS:  5736   Length:273147
##  Furgoneta              :  4663   DE 25 A 29 AÑOS:  5610   Class :character
##  Motocicleta > 125cc    :  4525   DE 35 A 39 AÑOS:  5573   Mode  :character
##  Motocicleta hasta 125cc:  4133   DE 30 A 34 AÑOS:  5463
##  Autobús                :  1840   DESCONOCIDA    :  5333
##  (Other)                :  6965   (Other)        : 24091
##  NA's                   :199366   NA's           :221341
##  * La correspondencia de los códigos se encuentra descrito en la estructura del
fichero.
##  Mode:logical
##  NA's:273147
##
##
##
##
##
##  RANGO DE EDAD        X14            X15
##  Length:273147      Mode:logical   Mode:logical
##  Class :character   NA's:273147    NA's:273147
##  Mode  :character
```

```
##
##
##
##
```

```
traffic$YEAR <- factor(year(traffic$FECHA))
trafficcount <- traffic[, .(count = .N), by= year(FECHA)]
trafficcount <- trafficcount[order(-count)]
ggplot(trafficcount[order(count)], aes(x=reorder(year, -count))) +
  geom_bar(aes(y=count, fill=as.factor(year)), stat = "identity") +
  xlab("Año") +
  ylab("Accidentes") -> baseplot

ggplotly(baseplot)
```



Ranking de accidentes por años. Varias consideraciones al respecto:

- 2020 solo contiene datos hasta el mes de Octubre. Independientemente de disponer de un menor histórico, la reducción drástica del número de accidentes es debido al confinamiento domiciliario derivados del la crisis del covid-19.
- En 2019 los datos son significativamente mayores que le resto de años debido al cambio de cuantificación desde 2019 en adelante. De 2010 a 2018 solo registran los accidentes con heridos o con daños al patrimonio municipal.

```
## Warning: `group_by_()` is deprecated as of dplyr 0.7.0.
## Please use `group_by()` instead.
## See vignette('programming') for more help
```

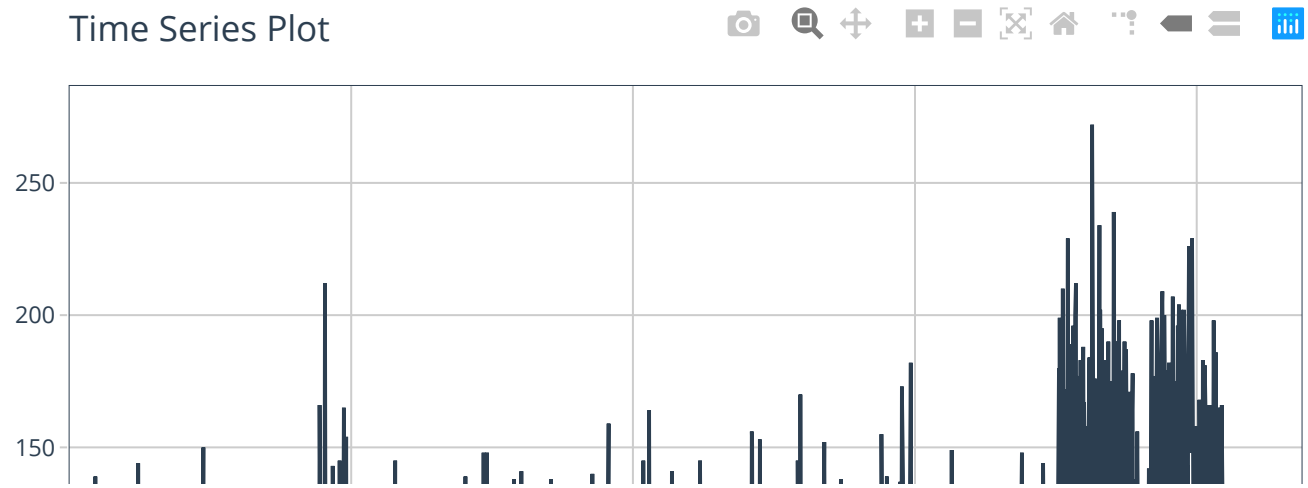
```
## This warning is displayed once every 8 hours.
## Call `lifecycle::last_warnings()` to see where this warning was generated.
```

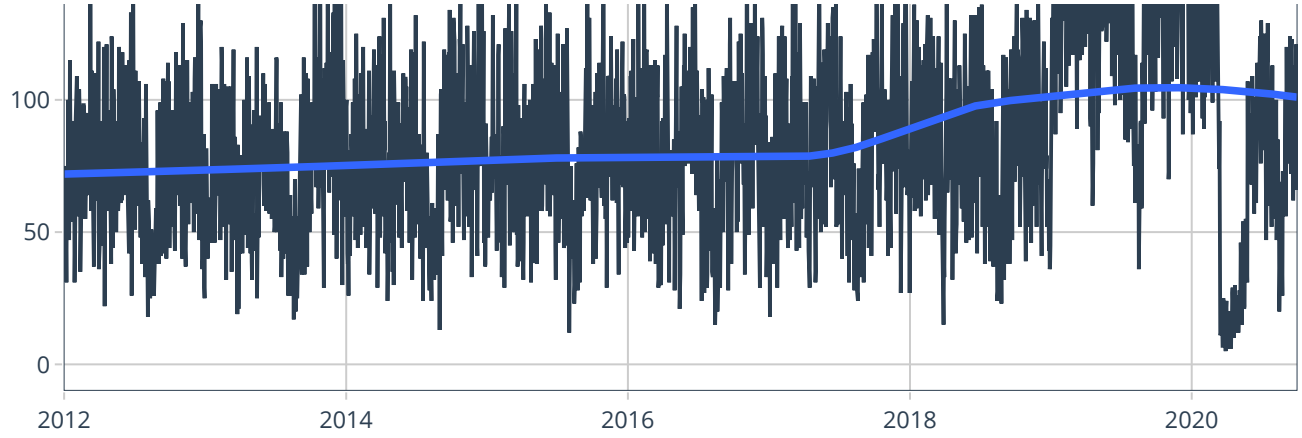
Observamos en esta figura el número de accidentes a lo largo de los años, clasificados por el distrito correspondiente y analizando el efecto del sexo del involucrado en el accidente. Como bien es sabido de todas las estadísticas facilitadas por las autoridades, los hombres cuentan con más siniestralidad de tráfico que las mujeres.

Algunas agregaciones básicas

Algunos plots básicos de series temporales

En esta figura se observa la serie temporal y el efecto del confinamiento total a partir de Marzo de 2020.

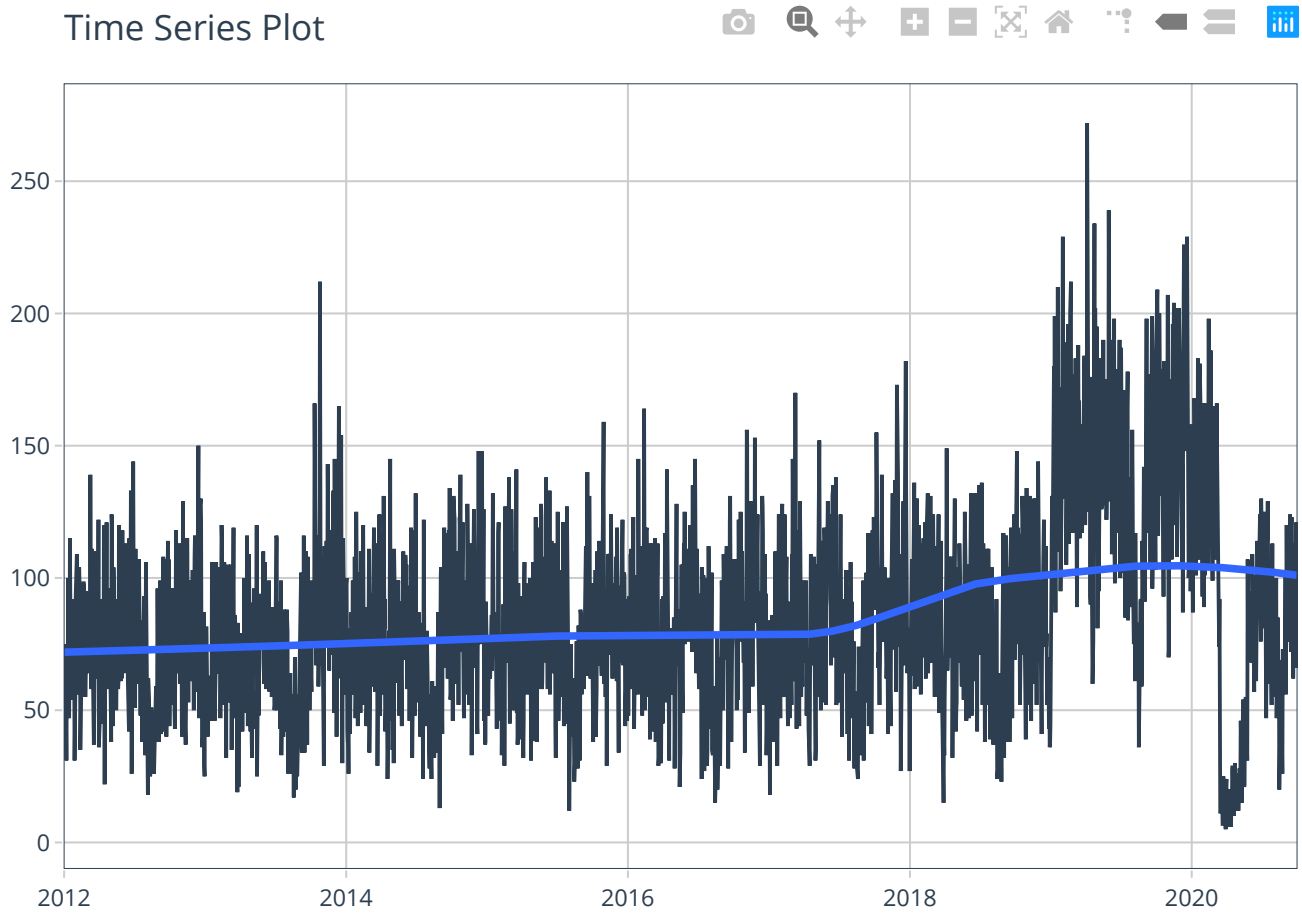




Analítica predictiva

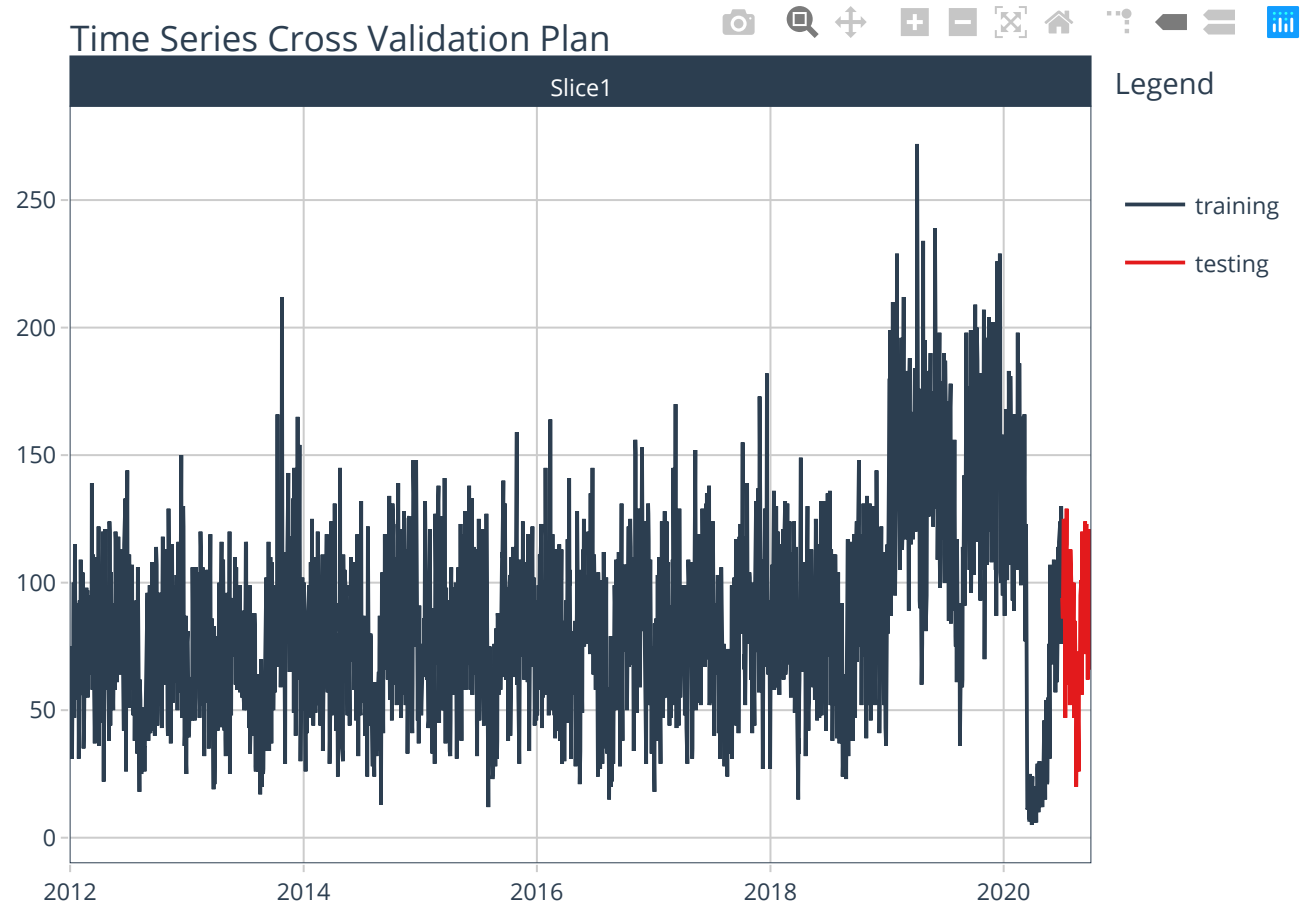
```
names(traffic_agg3) <- c("date", "value")

plot_time_series(traffic_agg3, date, value)
```




```
splits <- traffic_agg3 %>%
  time_series_split(assess = "3 months", cumulative = TRUE)

splits %>%
  tk_time_series_cv_plan() %>%
  plot_time_series_cv_plan(date, value)
```




```
# Add time series signature
recipe_spec_timeseries <- recipe(value ~ ., data = training(splits)) %>%
  step_timeseries_signature(date)

bake(prep(recipe_spec_timeseries), new_data = training(splits))
```

```
## # A tibble: 3,105 x 29
##    date       value date_index.num date_year date_year.iso date_half
##    <date>     <int>          <int>     <int>         <int>     <int>
##  1 2012-01-01    43     1325376000      2012          2011         1
##  2 2012-01-02    61     1325462400      2012          2012         1
##  3 2012-01-03    75     1325548800      2012          2012         1
##  4 2012-01-04    70     1325635200      2012          2012         1
##  5 2012-01-05    68     1325721600      2012          2012         1
##  6 2012-01-06    46     1325808000      2012          2012         1
##  7 2012-01-07    31     1325894400      2012          2012         1
##  8 2012-01-08    65     1325980800      2012          2012         1
##  9 2012-01-09    67     1326067200      2012          2012         1
## 10 2012-01-10    55     1326153600      2012          2012         1
## # … with 3,095 more rows, and 23 more variables: date_quarter <int>,
## #   date_month <int>, date_month.xts <int>, date_month.lbl <ord>,
## #   date_day <int>, date_hour <int>, date_minute <int>, date_second <int>,
## #   date_hour12 <int>, date_am.pm <int>, date_wday <int>, date_wday.xts <int>,
## #   date_wday.lbl <ord>, date_mday <int>, date_qday <int>, date_yday <int>,
## #   date_mweek <int>, date_week <int>, date_week.iso <int>, date_week2 <int>,
## #   date_week3 <int>, date_week4 <int>, date_mday7 <int>
```

```
recipe_spec_final <- recipe_spec_timeseries %>%
  step_fourier(date, period = 365, K = 5) %>%
  step_rm(date) %>%
  step_rm(contains("iso"), contains("minute"), contains("hour"),
          contains("am.pm"), contains("xts")) %>%
  step_normalize(contains("index.num"), date_year) %>%
  step_dummy(contains("lbl"), one_hot = TRUE)

juice(prep(recipe_spec_final))
```

```
## # A tibble: 3,105 x 47
##    value date_index.num date_year date_half date_quarter date_month date_day
##    <int>          <dbl>     <dbl>     <int>        <int>      <int>    <int>
##  1    43          -1.73     -1.53         1            1          1        1
##  2    61          -1.73     -1.53         1            1          1        2
##  3    75          -1.73     -1.53         1            1          1        3
##  4    70          -1.73     -1.53         1            1          1        4
##  5    68          -1.73     -1.53         1            1          1        5
##  6    46          -1.73     -1.53         1            1          1        6
##  7    31          -1.72     -1.53         1            1          1        7
##  8    65          -1.72     -1.53         1            1          1        8
##  9    67          -1.72     -1.53         1            1          1        9
## 10    55          -1.72     -1.53         1            1          1       10
## # … with 3,095 more rows, and 40 more variables: date_second <int>,
## #   date_wday <int>, date_mday <int>, date_qday <int>, date_yday <int>,
## #   date_mweek <int>, date_week <int>, date_week2 <int>, date_week3 <int>,
## #   date_week4 <int>, date_mday7 <int>, date_sin365_K1 <dbl>,
## #   date_cos365_K1 <dbl>, date_sin365_K2 <dbl>, date_cos365_K2 <dbl>,
## #   date_sin365_K3 <dbl>, date_cos365_K3 <dbl>, date_sin365_K4 <dbl>,
## #   date_cos365_K4 <dbl>, date_sin365_K5 <dbl>, date_cos365_K5 <dbl>,
## #   date_month.lbl_01 <dbl>, date_month.lbl_02 <dbl>, date_month.lbl_03 <dbl>,
## #   date_month.lbl_04 <dbl>, date_month.lbl_05 <dbl>, date_month.lbl_06 <dbl>,
## #   date_month.lbl_07 <dbl>, date_month.lbl_08 <dbl>, date_month.lbl_09 <dbl>,
## #   date_month.lbl_10 <dbl>, date_month.lbl_11 <dbl>, date_month.lbl_12 <dbl>,
## #   date_wday.lbl_1 <dbl>, date_wday.lbl_2 <dbl>, date_wday.lbl_3 <dbl>,
## #   date_wday.lbl_4 <dbl>, date_wday.lbl_5 <dbl>, date_wday.lbl_6 <dbl>,
## #   date_wday.lbl_7 <dbl>
```

```
model_spec_lm <- linear_reg(mode = "regression") %>%
  set_engine("lm")

workflow_lm <- workflow() %>%
  add_recipe(recipe_spec_final) %>%
  add_model(model_spec_lm)

workflow_lm
```

```
## == Workflow ============================================================
=
## Preprocessor: Recipe
## Model: linear_reg()
##
## -- Preprocessor ----------------------------------------------------------
-
```

```
## 6 Recipe Steps
##
## ● step_timeseries_signature()
## ● step_fourier()
## ● step_rm()
## ● step_rm()
## ● step_normalize()
## ● step_dummy()
##
## ── Model ────────────────────────────────────────────────────────────────
─
## Linear Regression Model Specification (regression)
##
## Computational engine: lm
```

```
workflow_fit_lm <- workflow_lm %>% fit(data = training(splits))

model_table <- modeltime_table(workflow_fit_lm)

model_table
```

```
## # Modeltime Table
## # A tibble: 1 x 3
##   .model_id .model     .model_desc
##       <int> <list>     <chr>
## 1         1 <workflow> LM
```

```
calibration_table <- model_table %>%
  modeltime_calibrate(testing(splits))
```

```
## Warning: Problem with `mutate()` input `.nested.col`.
## ℹ prediction from a rank-deficient fit may be misleading
## ℹ Input `.nested.col` is `purrr::map2(...)`.
```

```
## Warning in predict.lm(object = object$fit, newdata = new_data, type =
## "response"): prediction from a rank-deficient fit may be misleading
```

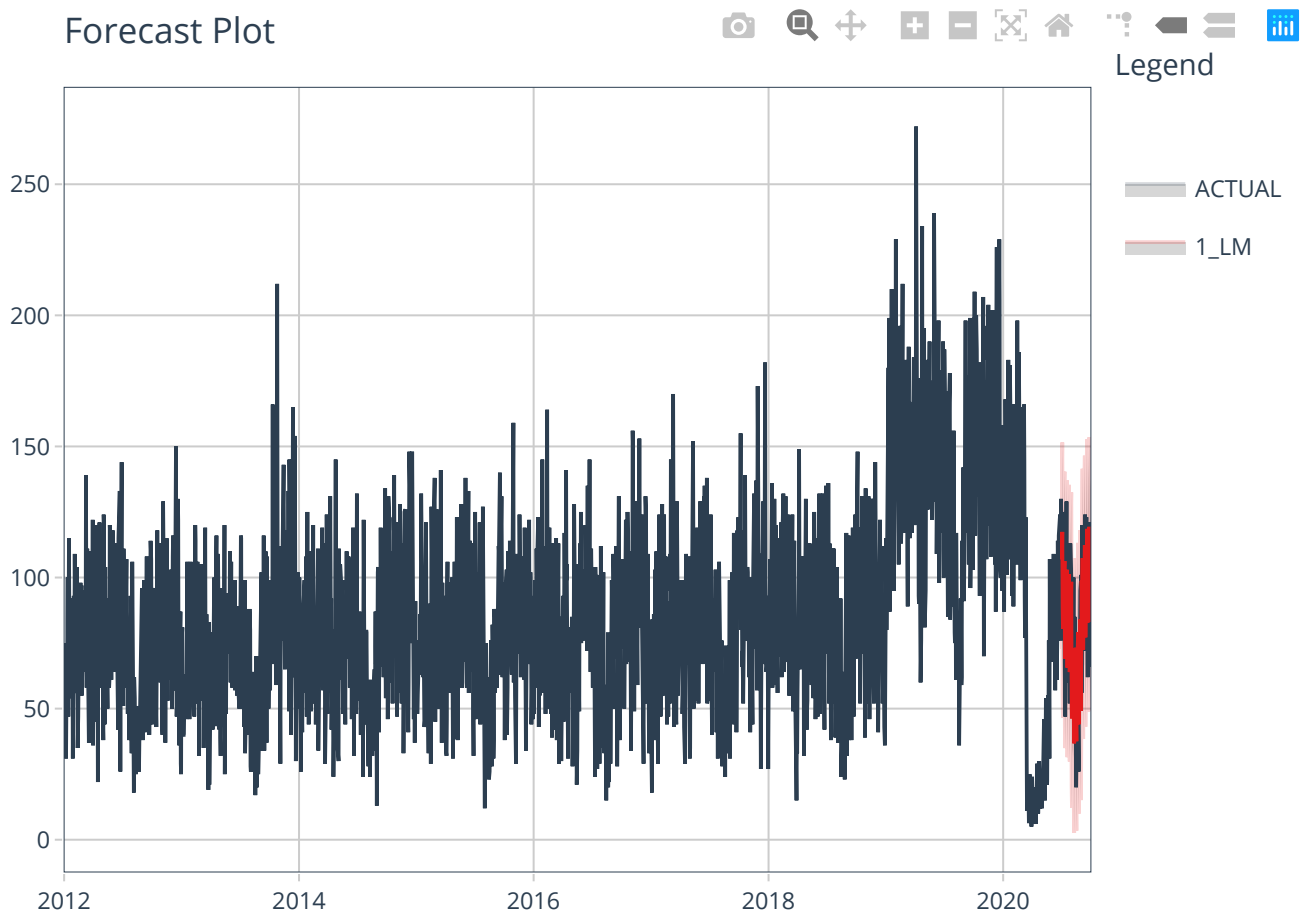
```
calibration_table
```

```
## # Modeltime Table
## # A tibble: 1 x 5
##   .model_id .model     .model_desc .type .calibration_data
##       <int> <list>     <chr>       <chr> <list>
## 1         1 <workflow> LM          Test  <tibble [91 × 4]>
```

```
calibration_table %>%
  modeltime_forecast(actual_data = traffic_agg3) %>%
  plot_modeltime_forecast()
```

```
## Warning: Problem with `mutate()` input `.nested.col`.
## ℹ prediction from a rank-deficient fit may be misleading
## ℹ Input `.nested.col` is `purrr::map2(...)`.

## Warning: prediction from a rank-deficient fit may be misleading
```




```
calibration_table %>%
  modeltime_accuracy() %>%
  table_modeltime_accuracy()
```

Search

.model_id	.model_desc	.type	mae	mape	mase	sm
1	LM	Test	14.36	19.47	0.72	18

```
calibration_table %>%
  modeltime_refit(traffic_agg3) %>%
  modeltime_forecast(h = "3 months", actual_data = traffic_agg3) %>%
  plot_modeltime_forecast()
```

```
## Warning: Problem with `mutate()` input `.nested.col`.
## ℹ prediction from a rank-deficient fit may be misleading
## ℹ Input `.nested.col` is `purrr::map2(...)`.
```

```
## Warning: prediction from a rank-deficient fit may be misleading
```