Riverside Air Quality Analysis

Anna Madison Burns

5/26/2020

Work plan:

- establish hypothesis
- subset data (whole dataset is too large)
- visualize subset (need still smaller - a few plots of time vs. pollutant level)
- specify model to test hypothesis (pollution is a function of time of the day -> day vs. night lots of pollutants are UV specific -> figure out how long the sun is out by day, merge a different dataset/create categorical variable that assigns 8 pm - 7am of night, but can have more potential for analyzing data)
- run model and diagnostics (look at assumptions, compare models w/ AIC)

Introduction

Background

Objectives

Methods

Data Processing

Source of data?

need to read the data for the Rmd file to work...

Subsetting the data

```
NO2_R <- NO2_CA[ which(NO2_CA$`County Name`=='Riverside'), ]
```

```
PM10_R <- PM10_CA[ which(PM10_CA$`County Name` == 'Riverside'),]
```

```
Riverside <- rbind(PM10_R, NO2_R)
```

```
Riverside <- Riverside[c(-1,-2, -3, -4, -5, -6, -7, -8, -12, -13, -15, -16, -17, -18, -19, -20, -21, -2
```

```r
colnames(Riverside) <- c("type","date","time","sample")
```

```r
Riverside$light <- "Day"
  #format(Riverside$`Time Local`, format="%H%M%S")
Riverside$light[Riverside$time<hms("08:00:00")|Riverside$time >hms("19:00:00")] <- "Night"
  # Riverside$`Light` = with(Riverside, if(`Time Local`< "08:00:00" & `Time Local` > "20:00:00")"Night"
# str(Riverside)
```

Other Options:

flights_dt <- flights_dt %>% mutate(overnight = arr_time < dep_time, arr_time = arr_time + days(overnight * 1), sched_arr_time = sched_arr_time + days(overnight * 1))

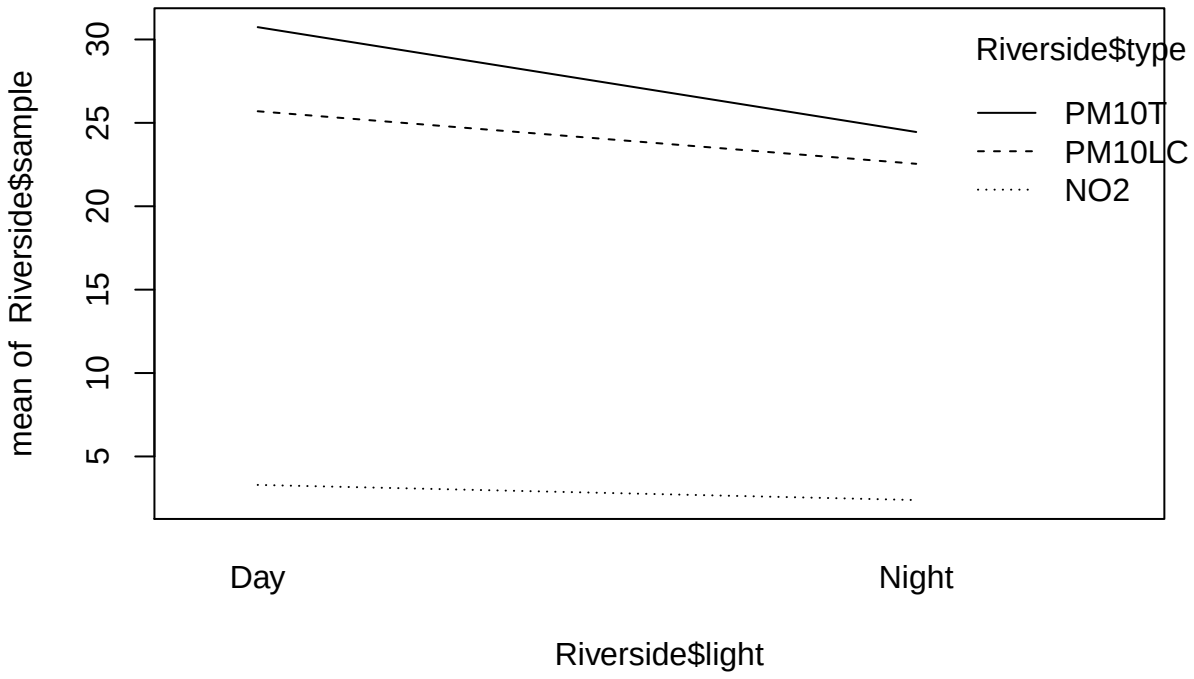
x$daynight = with(x, ifelse(TOD > "053000" & TOD < "200000", "Day", "Night"))

OR Light Function

Specify Model:

```r
Riverside$type[Riverside$type=="Nitrogen dioxide (NO2)"] <- "NO2"
Riverside$type[Riverside$type=="PM10 - LC"] <- "PM10LC"
Riverside$type[Riverside$type=="PM10 Total 0-10um STP"] <- "PM10T"
```

```r
interaction.plot(Riverside$light, Riverside$type, Riverside$sample)
```



```
ggplot(data=Riverside, mapping=aes(time, sample)) + geom_point(mapping=aes(color=type))
```