

# Code\_1

S18511

2024-01-29

```
library(tidyverse)
```

```
## Warning: package 'tidyverse' was built under R version 4.3.2
## Warning: package 'ggplot2' was built under R version 4.3.2
## Warning: package 'tibble' was built under R version 4.3.2
## Warning: package 'tidyr' was built under R version 4.3.2
## Warning: package 'readr' was built under R version 4.3.2
## Warning: package 'purrr' was built under R version 4.3.2
## Warning: package 'dplyr' was built under R version 4.3.2
## Warning: package 'stringr' was built under R version 4.3.2
## Warning: package 'forcats' was built under R version 4.3.2
## Warning: package 'lubridate' was built under R version 4.3.2
## -- Attaching core tidyverse packages ----- tidyverse 2.0.0 --
## v dplyr      1.1.4      v readr      2.1.4
## v forcats    1.0.0      v stringr    1.5.1
## v ggplot2    3.4.4      v tibble     3.2.1
## v lubridate  1.9.3      v tidyr      1.3.0
## v purrr      1.0.2
## -- Conflicts ----- tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()     masks stats::lag()
## i Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to become errors
```

```
library(janitor)
```

```
## Warning: package 'janitor' was built under R version 4.3.2
##
## Attaching package: 'janitor'
##
## The following objects are masked from 'package:stats':
##
##   chisq.test, fisher.test
```

```
library(latex2exp)
```

```
## Warning: package 'latex2exp' was built under R version 4.3.2
```

```
london_air = read_csv(file = "../Data/london_local_data_2022.csv")
```

```
## Rows: 289069 Columns: 10
## -- Column specification -----
## Delimiter: ","
## chr (2): site, code
## dbl (7): nox, no2, no, pm10, o3, pm2_5, so2
## dtm (1): date
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
london_sites = read_csv(file = "../Data/london_local_sites.csv")
```

```
## Rows: 67 Columns: 5
## -- Column specification -----
## Delimiter: ","
## chr (3): code, site, Parameter_name
## dbl (2): latitude, longitude
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
glimpse(london_air)
```

```
## Rows: 289,069
## Columns: 10
## $ site <chr> "Brent - John Keble Primary School", "Brent - John Keble Primary~
## $ code <chr> "BT6", "BT6", "BT6", "BT6", "BT6", "BT6", "BT6", "BT6", "BT6", "~
## $ date <dtm> 2022-01-01 00:00:00, 2022-01-01 01:00:00, 2022-01-01 02:00:00, ~
## $ nox <dbl> 13.4, 16.0, 11.1, 7.8, 8.6, 10.1, 16.5, 12.8, 14.6, 16.7, 17.5, ~
## $ no2 <dbl> 10.1, 11.3, 7.0, 5.3, 5.7, 6.9, 12.2, 9.2, 10.2, 11.1, 11.0, 9.1~
## $ no <dbl> 2.2, 3.0, 2.6, 1.7, 1.9, 2.1, 2.8, 2.4, 2.9, 3.7, 4.2, 4.2, 4.6, ~
## $ pm10 <dbl> 29.9, 17.5, 16.0, 16.5, 14.8, 11.3, 12.9, 11.9, 9.6, 11.6, 14.4, ~
## $ o3 <dbl> NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, ~
## $ pm2_5 <dbl> NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, ~
## $ so2 <dbl> NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, ~
glimpse(london_sites)
```

```
## Rows: 67
## Columns: 5
## $ code <chr> "BT8", "BT8", "BT6", "BT6", "CT4", "CT4", "CT3", "CT3", ~
## $ site <chr> "Brent - ARK Franklin Primary Academy", "Brent - ARK Fr~
## $ latitude <dbl> 51.53240, 51.53240, 51.53780, 51.53780, 51.52023, 51.52~
## $ longitude <dbl> -0.217719, -0.217719, -0.247793, -0.247793, -0.096106, ~
## $ Parameter_name <chr> "Nitrogen dioxide", "PM10 particulate matter (Hourly me~
london_air = london_air %>% remove_empty(c('cols', 'rows'))
london_sites = london_sites %>% remove_empty(c('cols', 'rows'))
```

```
sum(is.na(london_air))
```

```
## [1] 1084871
```

```
sum(is.na(london_sites))
```

```
## [1] 0
```

there are only NA values in London\_air data set

```
london_air_cleaned=london_air %>% replace (is.na(.),0)
sum(is.na(london_air_cleaned))
```

```
## [1] 0
```

now there are no missing values

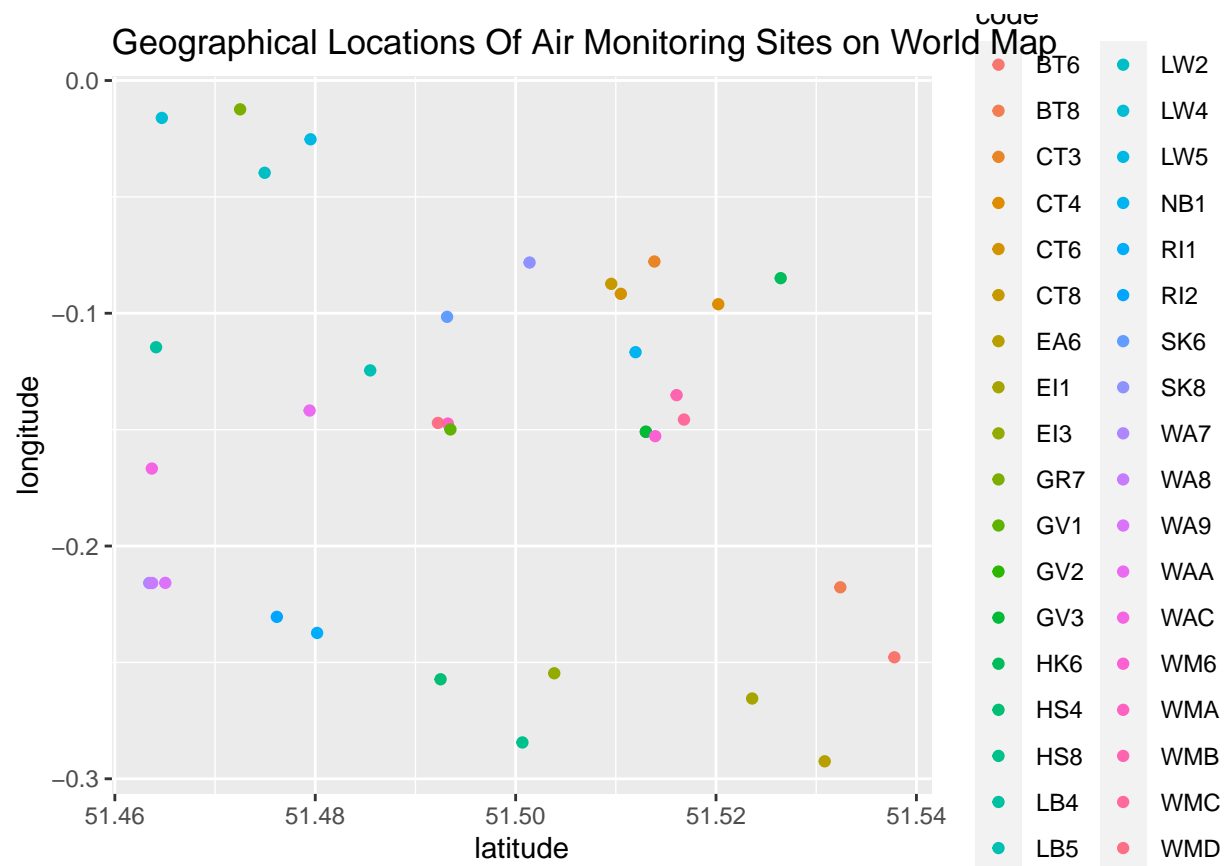
```
london_cleaned_1 <- london_air_cleaned %>%
  mutate(
    datetime = as.POSIXct(date, format = "%Y-%m-%dT%H:%M:%SZ"),
    date_only = as.Date(datetime),
    time = format(datetime, "%H:%M:%S")
  )

london_cleaned_1 = london_cleaned_1 %>% relocate(datetime,.after=date)
london_cleaned_1 = london_cleaned_1 %>% relocate(date_only,.after=datetime)
london_cleaned_1 = london_cleaned_1 %>% relocate(time,.after=date_only)
```

```
head(london_cleaned_1)
```

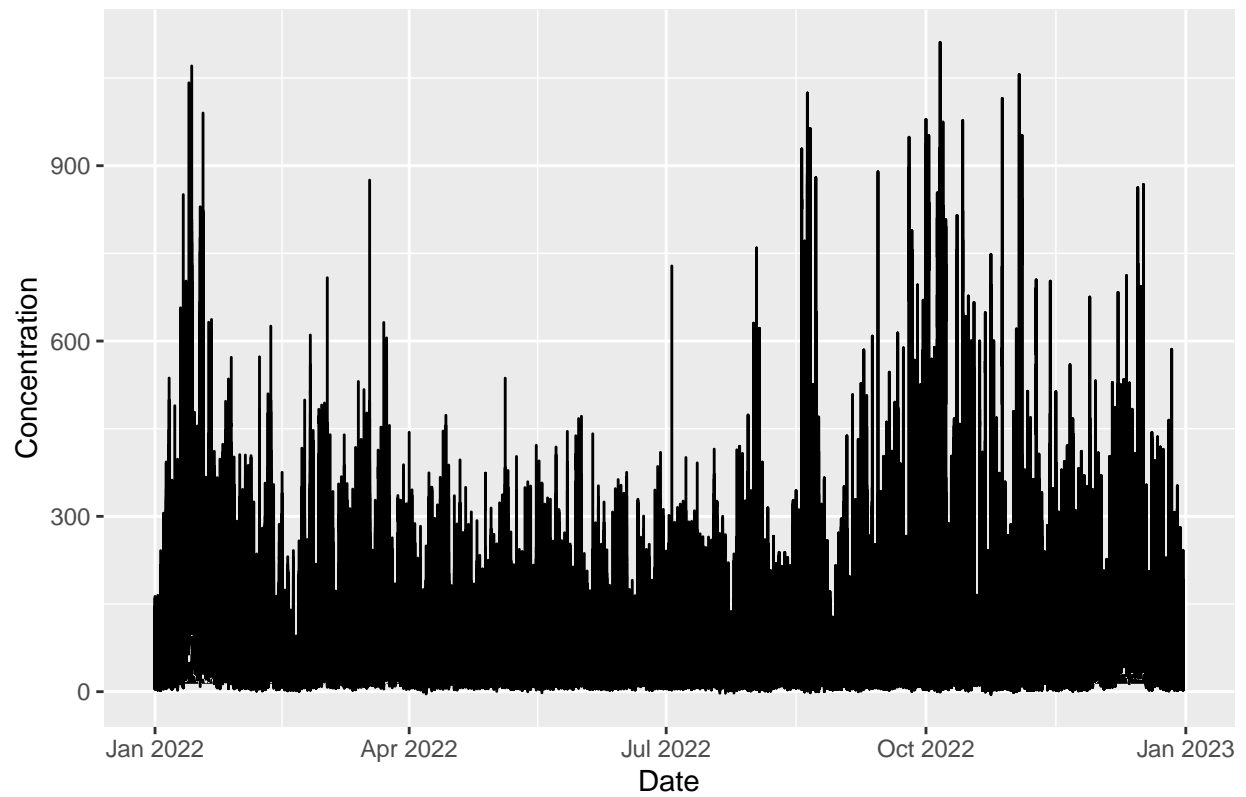
```
## # A tibble: 6 x 13
##   site      code date      datetime      date_only time    nox
##   <chr>    <chr> <dtm>      <dtm>      <date>    <chr> <dbl>
## 1 Brent - ~ BT6 2022-01-01 00:00:00 2022-01-01 00:00:00 2022-01-01 00:0~ 13.4
## 2 Brent - ~ BT6 2022-01-01 01:00:00 2022-01-01 01:00:00 2022-01-01 01:0~ 16
## 3 Brent - ~ BT6 2022-01-01 02:00:00 2022-01-01 02:00:00 2022-01-01 02:0~ 11.1
## 4 Brent - ~ BT6 2022-01-01 03:00:00 2022-01-01 03:00:00 2022-01-01 03:0~ 7.8
## 5 Brent - ~ BT6 2022-01-01 04:00:00 2022-01-01 04:00:00 2022-01-01 04:0~ 8.6
## 6 Brent - ~ BT6 2022-01-01 05:00:00 2022-01-01 05:00:00 2022-01-01 05:0~ 10.1
## # i 6 more variables: no2 <dbl>, no <dbl>, pm10 <dbl>, o3 <dbl>, pm2_5 <dbl>,
## #   so2 <dbl>
```

```
Locations_of_sites = london_sites %>% group_by(code) %>% select(code,site,latitude,longitude)%>% unique
Locations_of_sites %>% ggplot(aes(x=latitude,y=longitude)) + geom_point(aes(col= code)) + labs(title =
```

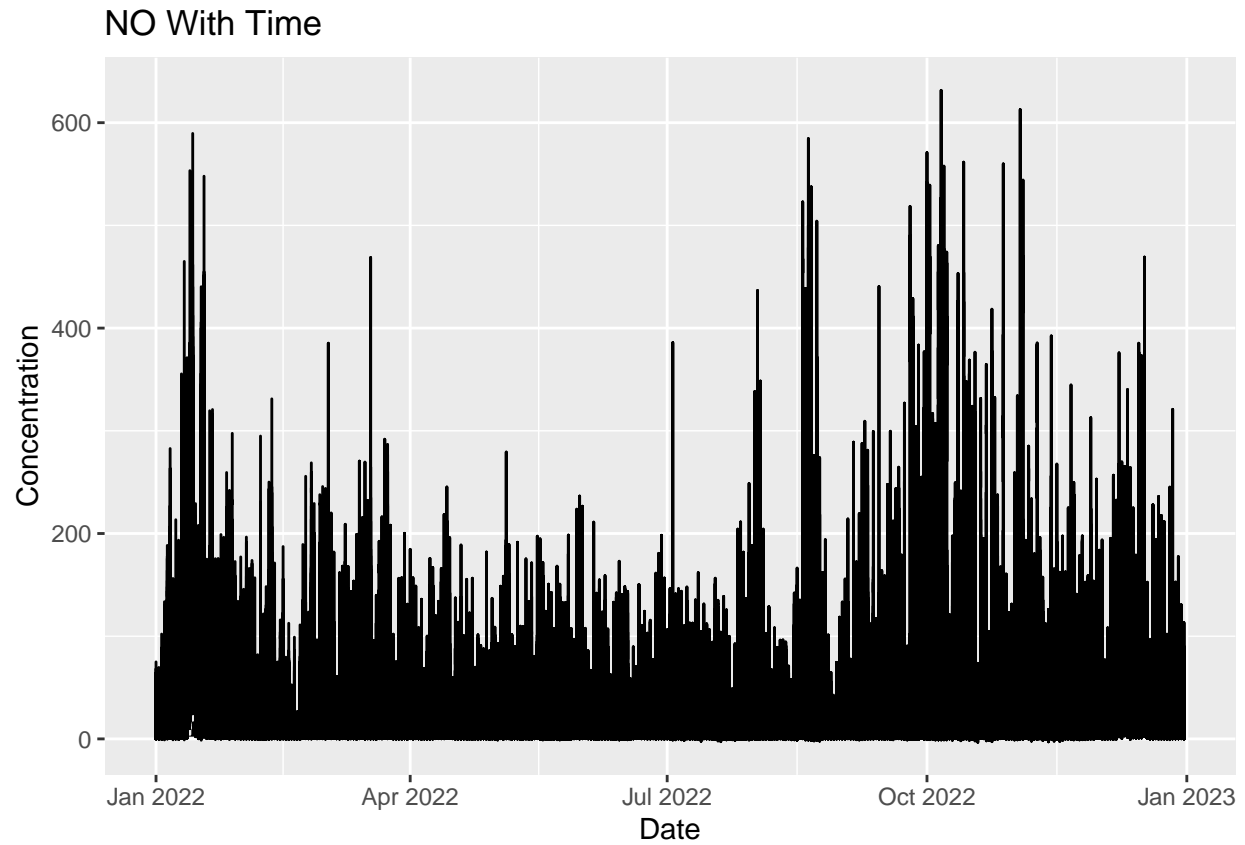


```
nox_time = london_cleaned_1 %>% filter(nox != 0) %>% select(date_only, nox)
ggplot(nox_time, aes(x = date_only, y = nox)) +
  geom_line() +
  labs(x = "Date", y = "Concentration", title = "NO_X With Time")
```

NO\_X With Time

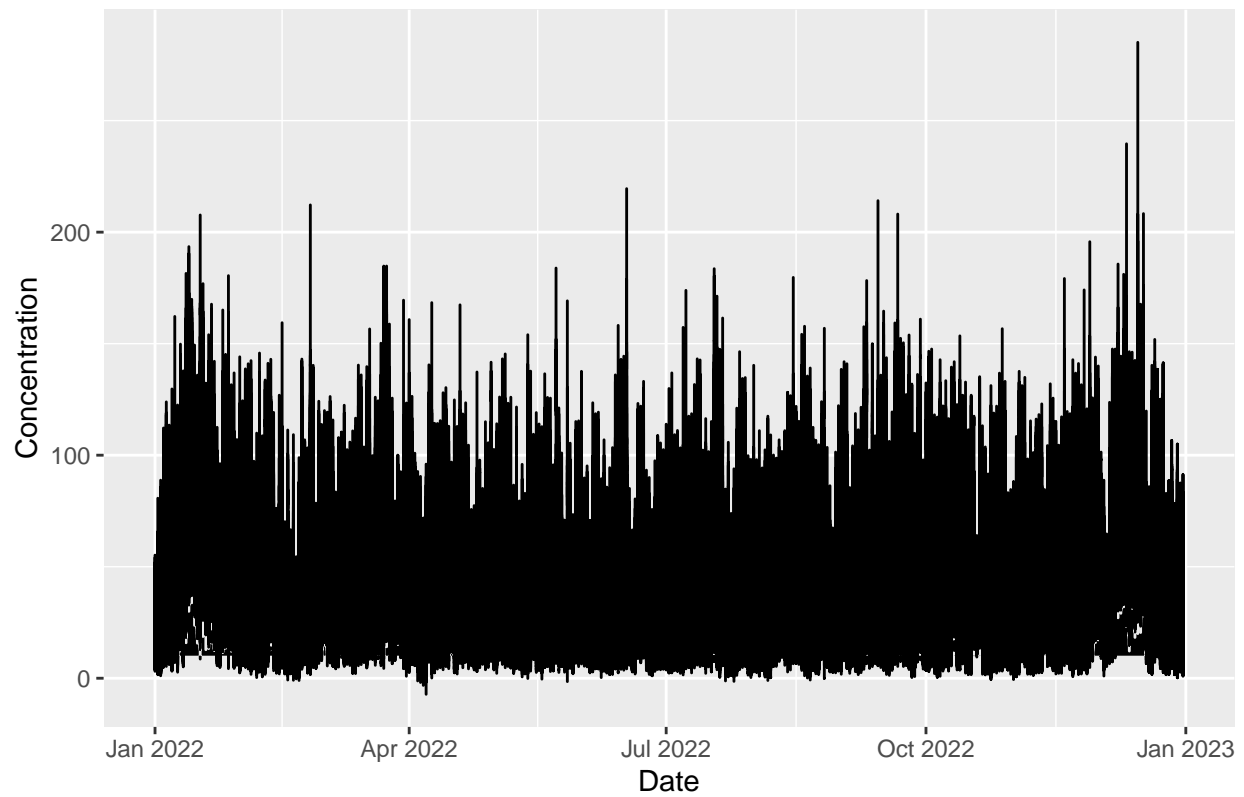


```
no_time = london_cleaned_1 %>% filter(no != 0) %>% select(date_only, no)
ggplot(no_time, aes(x = date_only, y = no)) +
  geom_line() +
  labs(x = "Date", y = "Concentration", title = "NO With Time")
```



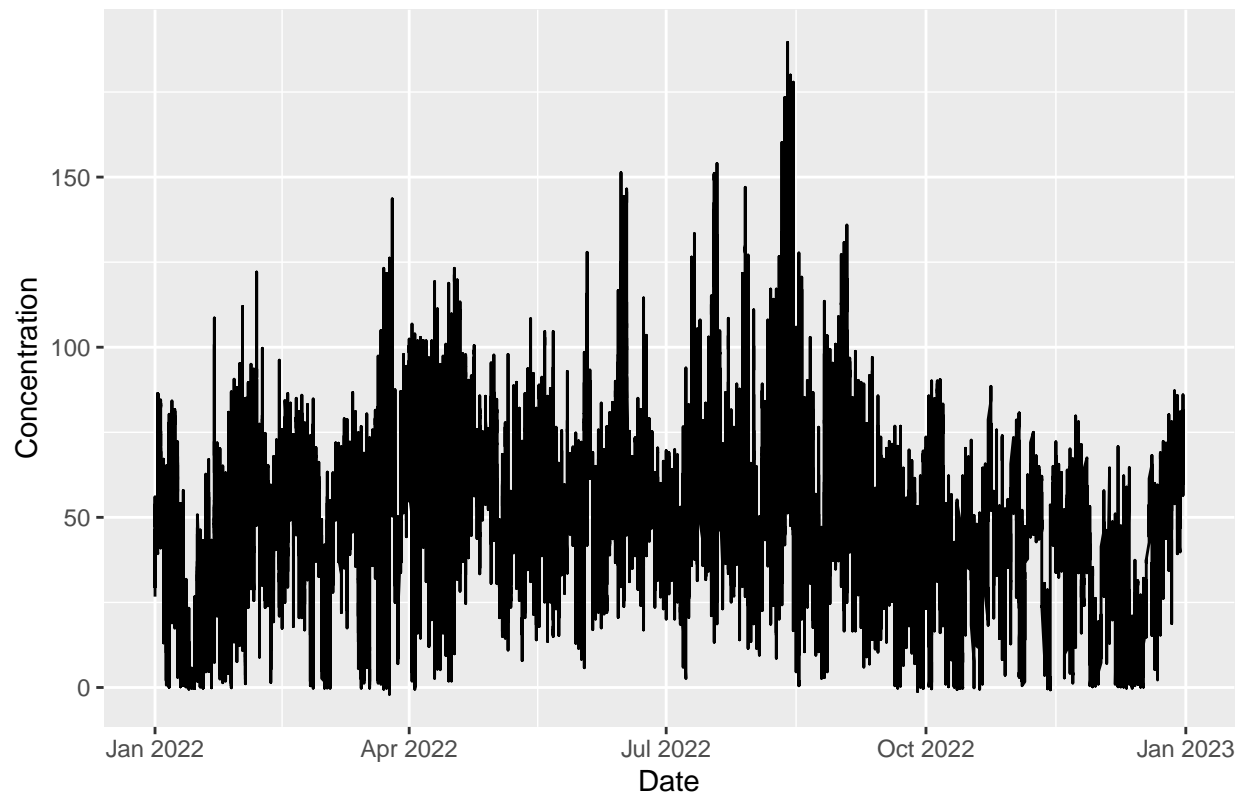
```
no2_time = london_cleaned_1%>% filter(no2 != 0)%>% select(date_only, no2)
ggplot(no2_time, aes(x = date_only, y = no2)) +
  geom_line() +
  labs(x = "Date", y = "Concentration", title = "NO2 With Time")
```

NO2 With Time



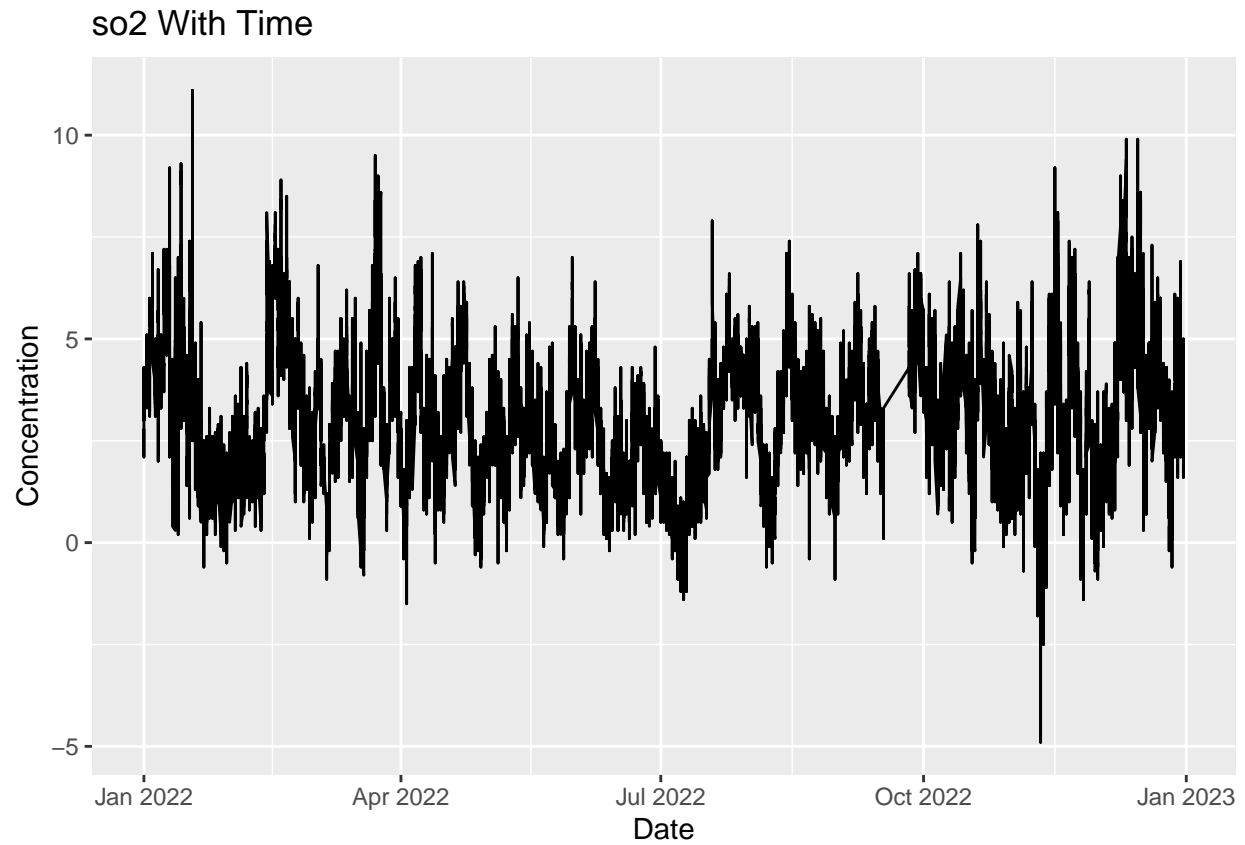
```
o3_time = london_cleaned_1 %>% filter(o3 != 0) %>% select(date_only, o3)
ggplot(o3_time, aes(x = date_only, y = o3)) +
  geom_line() +
  labs(x = "Date", y = "Concentration", title = "O3 With Time")
```

O3 With Time

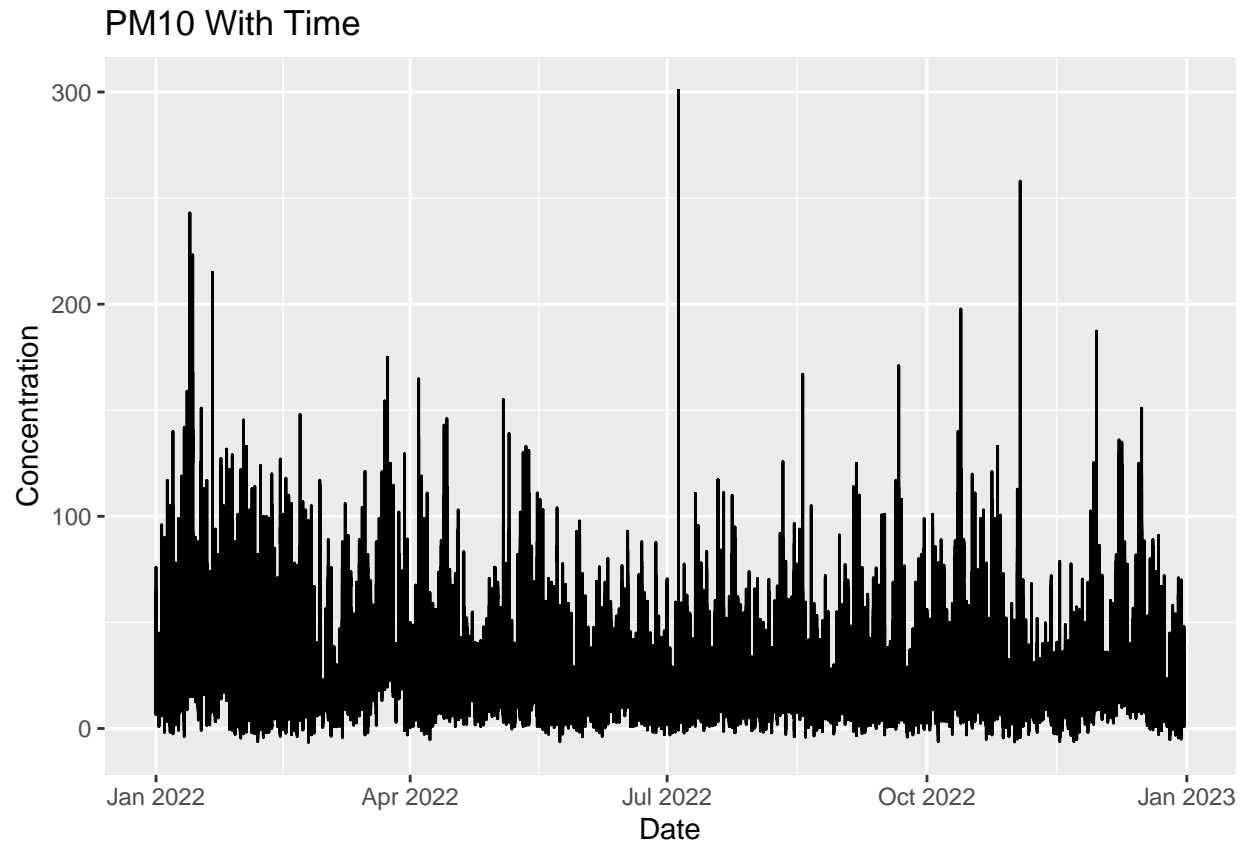


```
so2_time = london_cleaned_1 %>% filter(so2 != 0) %>% select(date_only, so2)
ggplot(so2_time, aes(x = date_only, y = so2)) +
  geom_line() +
  labs(x = "Date", y = "Concentration", title = "so2 With Time")
```



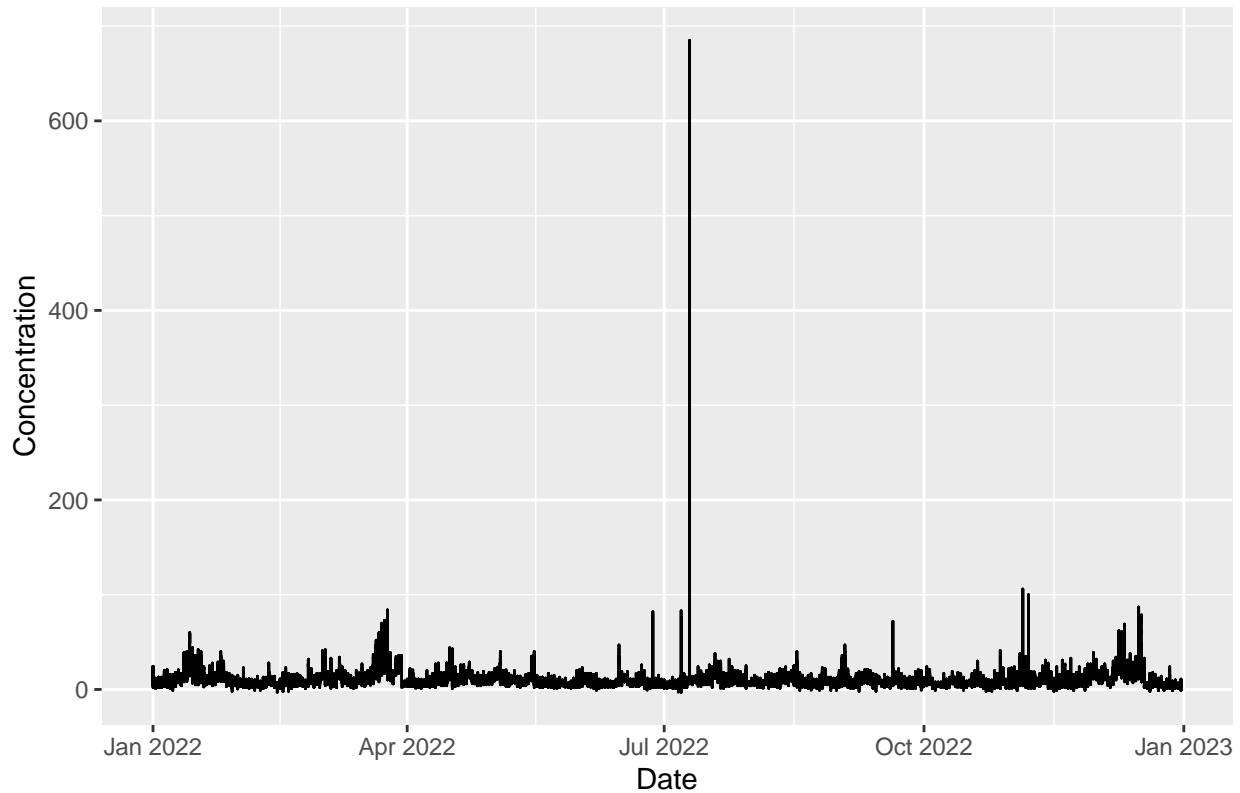


```
PM10_time =london_cleaned_1 %>% filter(pm10 != 0) %>% select(date_only,pm10)
ggplot(PM10_time, aes(x = date_only, y =pm10 )) +
  geom_line() +
  labs(x = "Date", y = "Concentration", title = "PM10 With Time")
```



```
PM2.5_time = london_cleaned_1 %>% filter(pm2_5 != 0) %>% select(date_only, pm2_5)
ggplot(PM2.5_time, aes(x = date_only, y = pm2_5)) +
  geom_line() +
  labs(x = "Date", y = "Concentration", title = "PM2.5 With Time")
```

## PM2.5 With Time



since NO2, NO ,NOx and PM\_10 are measured in many site,

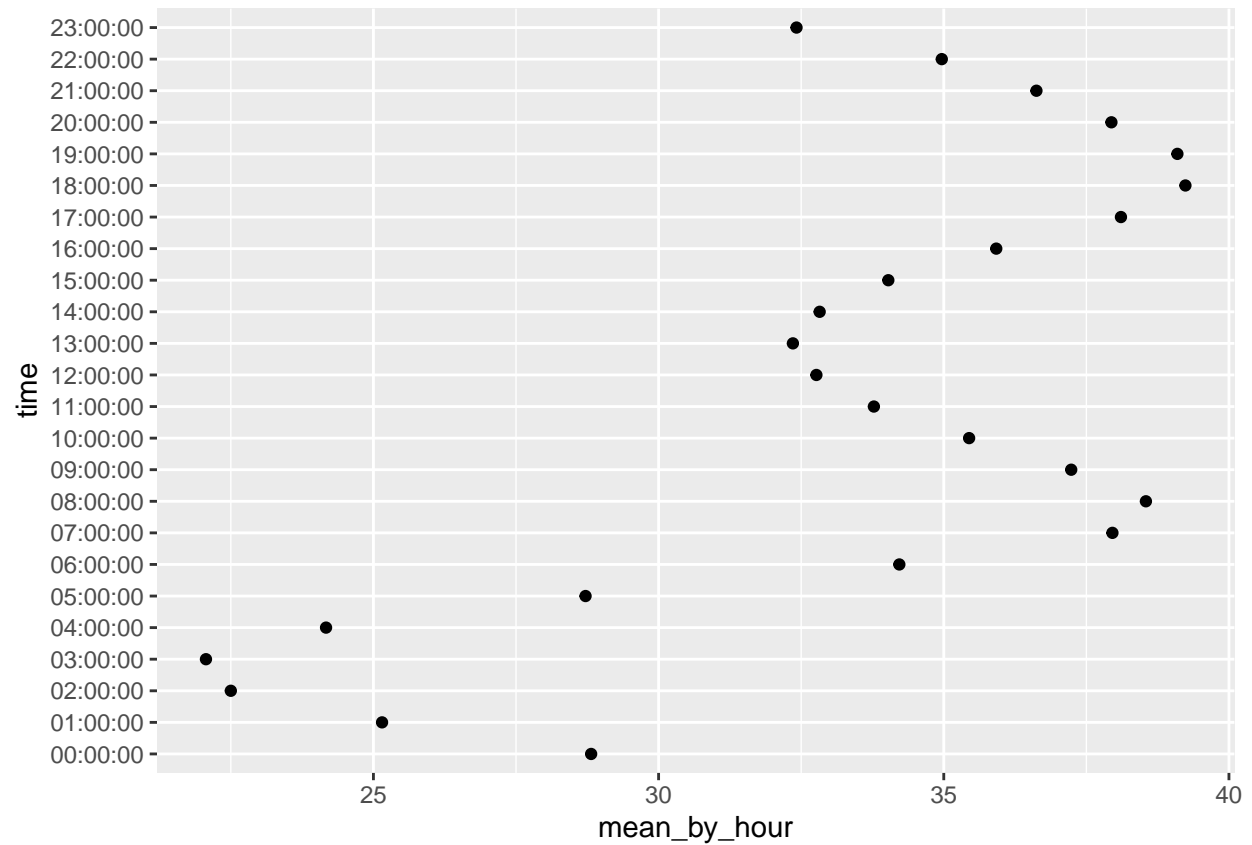
```
site_with_sum = london_cleaned_1 %>% group_by(code) %>% summarise(sum_nox = sum(nox),
                                                                    sum_no = sum(no),
                                                                    sum_no2 = sum(no2),
                                                                    sum_Pm10= sum(pm10),
                                                                    total_sum = sum(c(sum_nox,sum_no,sum_no2 ,sum_Pm10))
                                                                    ) %>% arrange(desc(total_sum))

head(site_with_sum)
```

```
## # A tibble: 6 x 6
##   code    sum_nox  sum_no sum_no2 sum_Pm10 total_sum
##   <chr>    <dbl>  <dbl>  <dbl>    <dbl>    <dbl>
## 1 LB4    1213653. 434015. 548170.  159326. 2355163.
## 2 EA6    1217362. 501426. 448518.  146535. 2313841.
## 3 CT6     969462. 344236. 441656.     0 1755354.
## 4 LW4     863372. 347381. 330725.  169640 1711119.
## 5 EI1     606755. 196366. 305669.  213899. 1322689.
## 6 WMB     708121. 224802. 363433.     0 1296355.
```

```
mean_by_hour_no2=london_cleaned_1 %>% filter (no2 !=0 ) %>% group_by(time) %>% summarise(mean_by_hour =
mean_by_hour_no2)%>% ggplot(aes(x=time,y=mean_by_hour))+ geom_point()+geom_line() +coord_flip()
```

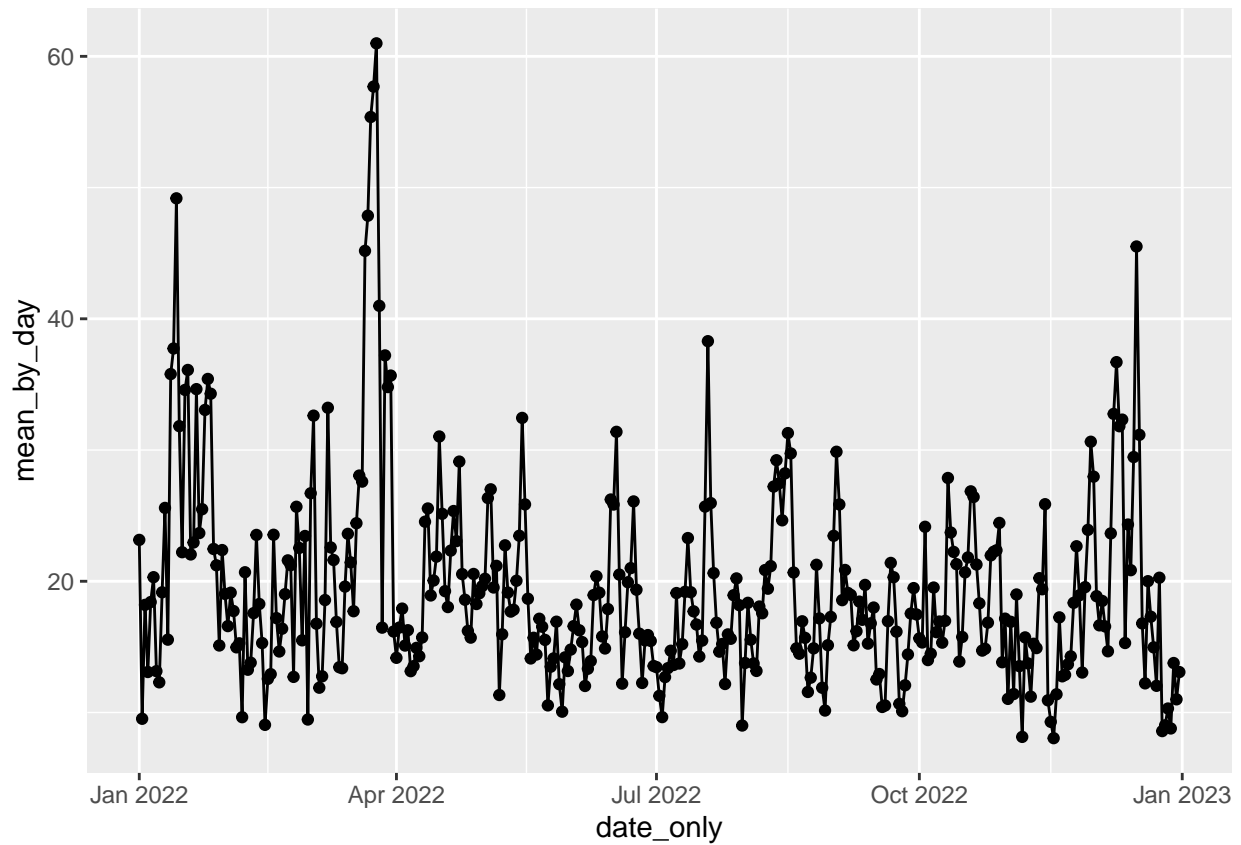
```
## `geom_line()`: Each group consists of only one observation.
## i Do you need to adjust the group aesthetic?
```



```
mean(mean_by_hour_no2$mean_by_hour)
```

```
## [1] 33.11974
```

```
mean_by_day_pm10=london_cleaned_1 %>% filter (pm10 !=0 ) %>% group_by(date_only) %>% summarise(mean_by_day_pm10=mean(pm10))
mean_by_day_pm10%>% ggplot(aes(x=date_only,y=mean_by_day_pm10))+ geom_point()+geom_line()
```

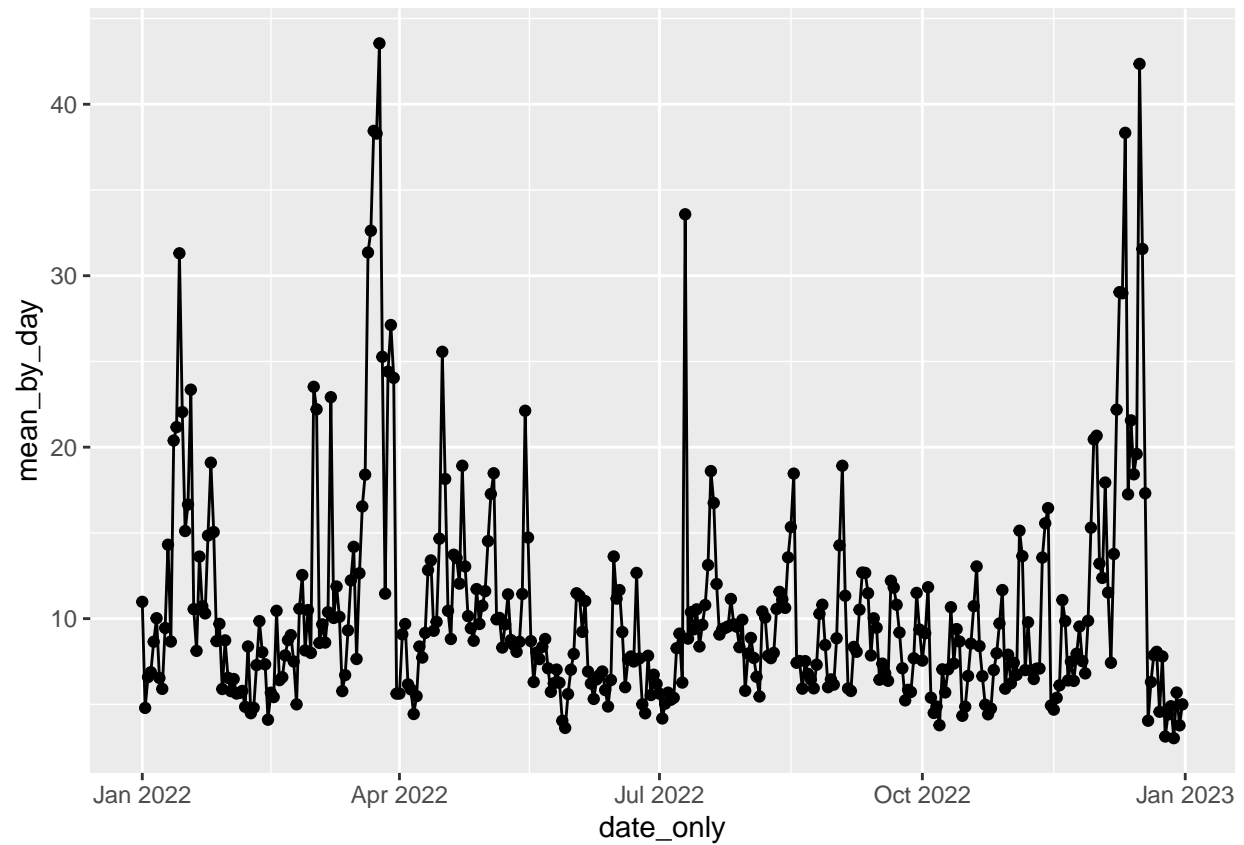


```
mean(mean_by_day_pm10$mean_by_day)
```

```
## [1] 19.49384
```

```
mean_by_day_pm2_5=london_cleaned_1 %>% filter (pm2_5 !=0 ) %>% group_by(date_only) %>% summarise(mean_by_day_pm2_5=
```

```
mean_by_day_pm2_5)%>% ggplot(aes(x=date_only,y=mean_by_day))+ geom_point()+geom_line()
```



```
mean(mean_by_day_pm2_5$mean_by_day)
```

```
## [1] 10.47
```