2021 Summary - Kingston Air Quality

Bard Center for Environmental Sciences and Humanities

1/7/2022

## **Intro**

Particulate matter (PM) is defined by the EPA as “mixture of solid particles and liquid droplets found in the air,” manifesting in a variety of shapes and sizes. PM10 consists of particles ≤10µm while PM 2.5 consists of particles ≤2.5µm. Although some particulate matter commonly exists naturally in the air, such as pollen and dust, others such as soot, smoke, and other chemical compounds pose larger risks to both the environment and human health. We measure PM levels using data collected from sensors installed on the roof of the Andy Murphy Center in Kingston, N.Y.

The purpose of the monitoring program is to provided baseline information about the state of local particulate matter pollution in our area, and to track changes over time.

In this report, we summarize the particulate matter observations from 2021.

## **Annual Mean**

One way that particulate matter is summarized is as an Annual Mean. In this case, this summary statistic is a simple arithmetic mean of all the observations taken during that year.

### EPA Standard

The U.S. Environmental Protection Agency’s (EPA) National Ambient Air Quality Standards (NAAQS) (1) provide a primary annual mean standard for PM2.5 of **12µm/m3**. To comply with EPA’s PM2.5 standard, an area’s three-year average of annual PM2.5 means must be below this level.  
The EPA does not provide an annual mean standard for PM10.

### WHO Standard

The World Health Organization (WHO) provides a similar annual mean standard (2). The WHO annual mean standard for PM2.5 is **5µm/m3**. The WHO annual mean standard for PM10 is **15µm/m3**.

### Kingston 2021 Results

According to the sensor taking particulate matter measurements every minute on top of the Andy Murphy Neighborhood Center, the PM2.5 annual mean for 2021 was:

# Calculate pm2.5 annual mean  
pm2.5mean2021 <- mean(df2021$pm2.5, na.rm = TRUE)  
cat(pm2.5mean2021)

## 5.174594

The PM10 annual mean for 2021 was:

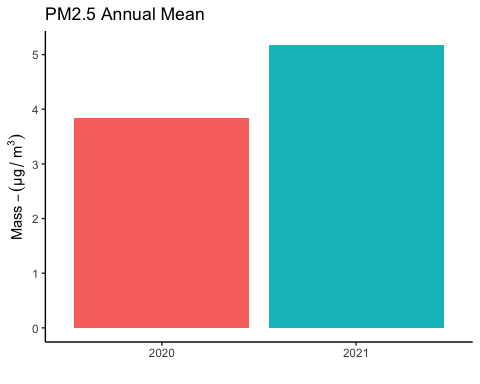
# Calculate pm10 annual mean  
pm10mean2021 <-mean(df2021$pm10,na.rm = TRUE)  
cat(pm10mean2021)

## 6.172208

### Multi-Year Comparison

This air quality monitoring program began in 2020. Below you can see the PM2.5 annual mean from both years:

# Calculate pm2.5 annual mean - 2020  
pm2.5mean2020 <- mean(df2020$pm2.5, na.rm = TRUE)  
  
  
# Bar chart showing 2020 and 2021 annual means for pm2.5  
bardf2.5 <- data.frame(year=c("2020", "2021"),  
 pm=c(pm2.5mean2020,pm2.5mean2021))  
  
  
ggplot(bardf2.5, aes(x=year, y=pm, fill = year)) +  
 geom\_bar(stat="identity", show.legend = FALSE) +  
 labs(  
 title = "PM2.5 Annual Mean",  
 y = expression(Mass - (μg/~m^3)),  
 x = ""  
 ) +  
 theme\_classic()

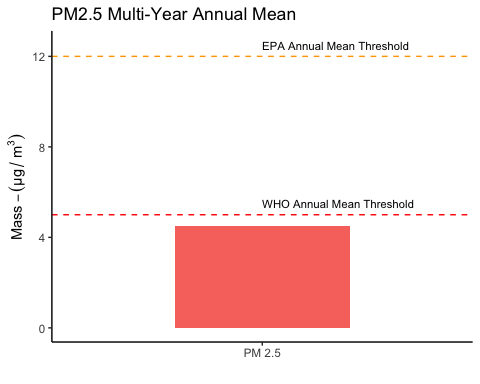
 The average of the two past years’ means for PM2.5 is:

# Calculate pm2.5 average annual mean from 2020 and 2021 together  
multiyear\_pm2.5 <- mean(c(pm2.5mean2020, pm2.5mean2021))  
cat(multiyear\_pm2.5)

## 4.504889

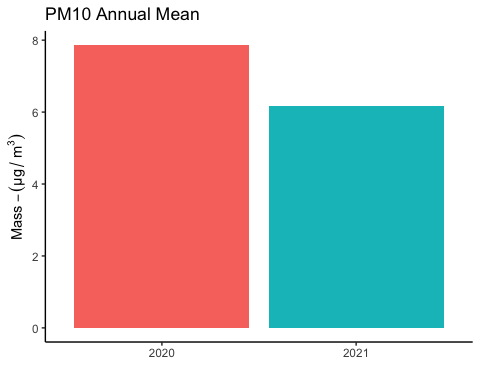
Here is the PM2.5 multi-year annual mean of the data currently available, with the EPA and WHO’s annual mean standards shown for reference:

# Bar chart with dotted lines for EPA and WHO standards  
  
annualbardf2.5 <- data.frame(AnnualMean = c("PM 2.5"),  
 PM= c(multiyear\_pm2.5))  
  
  
ggplot(annualbardf2.5, aes(x=AnnualMean, y=PM, fill = AnnualMean)) +  
 geom\_bar(stat="identity", width = .5, show.legend = FALSE) +  
 geom\_hline(yintercept=12, linetype="dashed", color = "orange") +  
 geom\_hline(yintercept=5, linetype="dashed", color = "red") +  
 annotate("text", x = "PM 2.5", y = 12.5, label = "EPA Annual Mean Threshold", size = 3, hjust=0) +  
 annotate("text", x = "PM 2.5", y = 5.5, label = "WHO Annual Mean Threshold", size = 3, hjust=0) +  
 labs(  
 title = "PM2.5 Multi-Year Annual Mean",  
 y = expression(Mass - (μg/~m^3)),  
 x = ""  
 ) +  
 theme\_classic()



PM10 annual mean from both years:

# Calculate pm10annual mean - 2020  
pm10mean2020 <- mean(df2020$pm10, na.rm = TRUE)  
  
  
# Bar chart showing 2020 and 2021 annual means for pm10  
bardf10 <- data.frame(year=c("2020", "2021"),  
 pm=c(pm10mean2020,pm10mean2021))  
  
  
ggplot(bardf10, aes(x=year, y=pm, fill = year)) +  
 geom\_bar(stat="identity", show.legend = FALSE) +  
 labs(  
 title = "PM10 Annual Mean",  
 y = expression(Mass - (μg/~m^3)),  
 x = ""  
 ) +  
 theme\_classic()

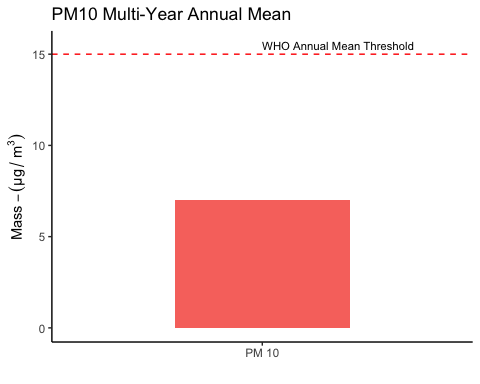
 The average of the two past years’ means for PM10 is:

# Calculate pm10average annual mean from 2020 and 2021 together  
multiyear\_pm10 <- mean(c(pm10mean2020, pm10mean2021))  
cat(multiyear\_pm10)

## 7.018028

Here is the PM10 multi-year annual mean of the data currently available, with the WHO’s annual mean standard shown for reference:

# Bar chart with dotted line for WHO standard  
annualbardf10 <- data.frame(AnnualMean = c("PM 10"),  
 PM= c(multiyear\_pm10))  
  
ggplot(annualbardf10, aes(x=AnnualMean, y=PM, fill = AnnualMean)) +  
 geom\_bar(stat="identity", width = .5, show.legend = FALSE) +  
 geom\_hline(yintercept=15, linetype="dashed", color = "red") +  
 annotate("text", x = "PM 10", y = 15.5, label = "WHO Annual Mean Threshold", size = 3, hjust=0) +  
 labs(  
 title = "PM10 Multi-Year Annual Mean",  
 y = expression(Mass - (μg/~m^3)),  
 x = ""  
 ) +  
 theme\_classic()



## **24-Hour Average**

Another important summary statics for particulate matter is the 24-hour average. The EPA and WHO define this standard slightly differently.

### EPA Standard

An area meets the EPA’s 24-hour Standard for PM2.5 if the 98th percentile of daily averages, averaged over three years, is below **35µm/m3** (3). In other words, we take all of the daily averages for the year, calculate the 98th percentile, and then average that number for three years of monitoring to get our summary statistic.

The EPA’s 24-hour Standard for PM10 is not calculated the same way. For PM10, the 24-hour Standard says that an area cannot have its 24 hour averages exceed **150µm/m3** more than once a year, on average over three years. (4)

### WHO Standard

An area meets the WHO’s 24-hour Standard for PM2.5 if the 99th percentile of daily averages is below **15µm/m3** (3). In other words, we take all of the daily averages for the year, and then calculate the 99th percentile to get our summary statistic.

An area meets the WHO’s 24-hour Standard for PM10 if the 99th percentile of daily averages is below **45µm/m3**, calculated in the same way.

### Kingston 2021 Results

According to the sensor taking particulate matter measurements every minute on top of the Andy Murphy Neighborhood Center, the PM2.5 24-hour Standard Summary Statistic for 2021 was:

EPA methodology:

# Calculate pm2.5 24-hour statistic  
#EPA Version  
dailies2021 <- df2021 %>%  
 mutate(date = as\_date(Time),  
 ) %>%  
 group\_by(date) %>% #group by the date  
 summarise(  
 sdpm2.5 = sd(pm2.5), #standard deviation of daily pm2.5  
 sdpm10 = sd(pm10),  
 pm10 = mean(pm10,na.rm = TRUE),  
 pm2.5 = mean(pm2.5,na.rm = TRUE),  
 sepm2.5 = (sdpm2.5)/sqrt(n()), #standard error of daily pm2.5  
 sepm10 = (sdpm10)/sqrt(n())  
 )   
  
pm2.5\_98th\_percentile2021 <- quantile(dailies2021$pm2.5, probs = .98)  
# calculating the standard error for the percentile estimate - needs review  
pm2.5\_98th\_percentile2021\_se <- sd(dailies2021$pm2.5) / sqrt(nrow(dailies2021))  
  
pm2.5\_98th\_percentile2021

## 98%   
## 13.41873

WHO methodology:

# Calculate pm2.5 24-hour statistic  
#WHO Version  
pm2.5\_99th\_percentile2021 <- quantile(dailies2021$pm2.5, probs = .99)  
# calculating the standard error for the percentile estimate - needs review  
pm2.5\_99th\_percentile2021\_se <- sd(dailies2021$pm2.5) / sqrt(nrow(dailies2021))  
  
pm2.5\_99th\_percentile2021

## 99%   
## 15.63331

The PM10 24-hour Standard Summary Statistic for 2021 was: EPA methodology:

# Calculate pm10 24-hour statistic  
#EPA Version  
top\_two\_pm10\_2021 <- dailies2021 %>%  
 arrange(desc(pm10))  
  
second\_highest\_pm10\_2021 <- top\_two\_pm10\_2021$pm10[2]  
cat(second\_highest\_pm10\_2021)

## 33.1415

WHO methodology:

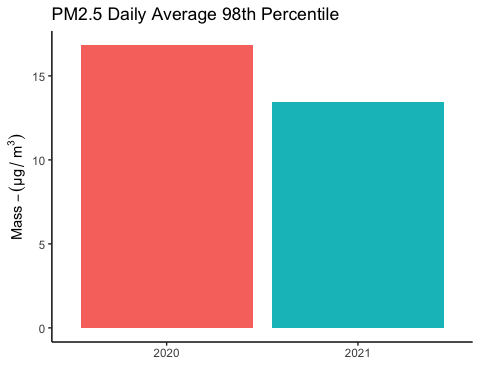
# Calculate pm10 24-hour statistic  
#WHO Version  
pm10\_99th\_percentile2021 <- quantile(dailies2021$pm10, probs = .99)  
pm10\_99th\_percentile2021

## 99%   
## 21.15395

### Multi-Year Comparison

This air quality monitoring program began in 2020. Below you can see the PM2.5 daily average 98th percentile statistic from both years:

# Bar chart showing 2020 and 2021 PM2.5 daily average 98th percentile  
dailies2020 <- df2020 %>%  
 mutate(date = as\_date(Time),  
 ) %>%  
 group\_by(date) %>% #group by the date  
 summarise(  
 sdpm2.5 = sd(pm2.5), #standard deviation of daily pm2.5  
 sdpm10 = sd(pm10),  
 pm10 = mean(pm10,na.rm = TRUE),  
 pm2.5 = mean(pm2.5,na.rm = TRUE),  
 sepm2.5 = (sdpm2.5)/sqrt(n()), #standard error of daily pm2.5  
 sepm10 = (sdpm10)/sqrt(n())  
 )   
pm2.5\_98th\_percentile2020 <- quantile(dailies2020$pm2.5, probs = .98)  
# calculating the standard error for the percentile estimate - needs review  
pm2.5\_98th\_percentile2020\_se <- sd(dailies2020$pm2.5) / sqrt(nrow(dailies2020))  
  
bardf2.5\_98th <- data.frame(year=c("2020", "2021"),  
 pm=c(pm2.5\_98th\_percentile2020,pm2.5\_98th\_percentile2021),  
 se = c(pm2.5\_98th\_percentile2020\_se, pm2.5\_98th\_percentile2021\_se)  
 )  
  
  
ggplot(bardf2.5\_98th, aes(x=year, y=pm, fill = year)) +  
 geom\_bar(stat="identity", show.legend = FALSE) +  
 labs(  
 title = "PM2.5 Daily Average 98th Percentile",  
 y = expression(Mass - (μg/~m^3)),  
 x = ""  
 ) +  
 theme\_classic()

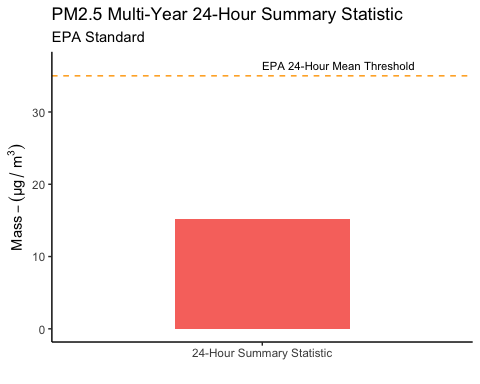
 The average of the two past years’ PM2.5 daily average 98th percentile gives us the EPA 24-hour Summary Statistic:

# Calculate pm2.5 daily average 98th percentile from 2020 and 2021 together  
multiyear\_pm2.5\_98th <- mean(c(pm2.5\_98th\_percentile2020, pm2.5\_98th\_percentile2021))  
cat(multiyear\_pm2.5\_98th)

## 15.12755

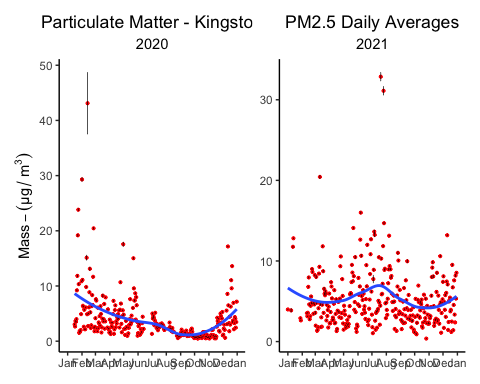
Here is the multi-year PM2.5 24-hour Summary Statistic, with the EPA’s 24-hour standard shown for reference:

# Bar chart with dotted lines for EPA standards  
  
multiyear\_pm2.5\_98thdf <- data.frame(statistic = c("24-Hour Summary Statistic"),  
 PM= c(multiyear\_pm2.5\_98th))  
  
ggplot(multiyear\_pm2.5\_98thdf, aes(x=statistic, y=PM, fill = statistic)) +  
 geom\_bar(stat="identity", width = .5, show.legend = FALSE) +  
 annotate("text", x = "24-Hour Summary Statistic", y = 36.5, label = "EPA 24-Hour Mean Threshold", size = 3, hjust=0) +  
 geom\_hline(yintercept=35, linetype="dashed", color = "orange") +  
 labs(  
 title = "PM2.5 Multi-Year 24-Hour Summary Statistic",  
 subtitle = "EPA Standard",  
 y = expression(Mass - (μg/~m^3)),  
 x = ""  
 ) +  
 theme\_classic()

 Here are the daily averages of PM2.5 as a time series for the years measured:

#2020  
yearplot2020 <- ggplot(data = dailies2020,  
 aes(x = date,y = pm2.5)) +  
 geom\_point(aes(y = pm2.5), color = "red", size= .75) +  
 geom\_errorbar(aes(ymin = pm2.5 - sepm2.5, ymax = pm2.5 + sepm2.5), size = .3, width = 1,alpha = .8) +  
 geom\_smooth(se = FALSE) +  
 #geom\_line(aes(y = daily\_mean), color = "red") +  
 scale\_x\_date(limits = as.Date(c("2020-01-01","2020-12-31")), date\_breaks = "months", date\_labels = "%b") +  
 labs(  
 y = expression(Mass - (μg/~m^3)),  
 x = NULL,  
 title = paste(  
 "Particulate Matter - Kingston"  
 ),  
 subtitle = paste(  
 "2020"  
 )) +  
 theme\_classic() +  
 theme(plot.title = element\_text(hjust = 0.5)) +  
 theme(plot.subtitle = element\_text(hjust = 0.5))  
  
#2021  
yearplot2021 <- ggplot(data = dailies2021,  
 aes(x = date,y = pm2.5)) +  
 geom\_point(aes(y = pm2.5), color = "red", size= .75) +  
 geom\_errorbar(aes(ymin = pm2.5 - sepm2.5, ymax = pm2.5 + sepm2.5), size = .3, width = 1,alpha = .8) +  
 geom\_smooth(se = FALSE) +  
 #geom\_line(aes(y = daily\_mean), color = "red") +  
 scale\_x\_date(limits = as.Date(c("2021-01-01","2021-12-31")), date\_breaks = "months", date\_labels = "%b") +  
 labs(  
 y = NULL, #expression(Mass - (μg/~m^3)),  
 x = NULL,  
 title = paste(  
 "PM2.5 Daily Averages"  
 ),  
 subtitle = paste(  
 "2021"  
 )) +  
 theme\_classic() +  
 theme(plot.title = element\_text(hjust = 0.5)) +  
 theme(plot.subtitle = element\_text(hjust = 0.5))  
  
library(patchwork)  
  
yearplot2020 + yearplot2021

## `geom\_smooth()` using method = 'loess' and formula 'y ~ x'  
## `geom\_smooth()` using method = 'loess' and formula 'y ~ x'

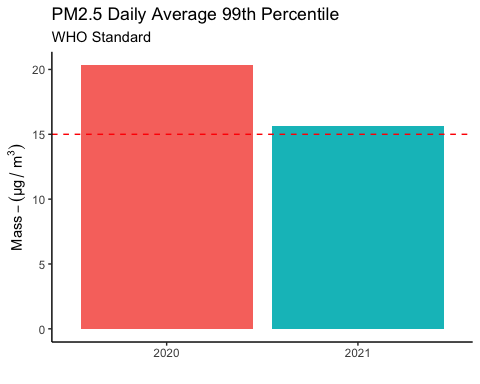


ggsave("pm2.5daily\_avg\_by\_year.png", width = 8, height = 4, units = "in")

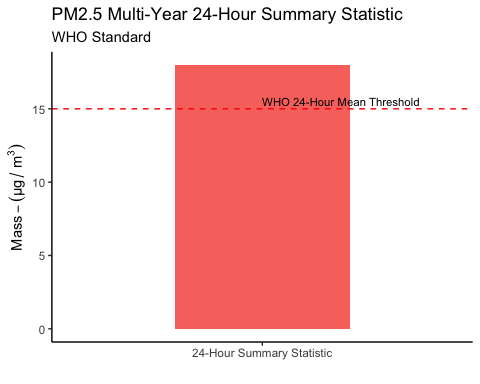
## `geom\_smooth()` using method = 'loess' and formula 'y ~ x'  
## `geom\_smooth()` using method = 'loess' and formula 'y ~ x'

Below is shown the PM2.5 daily average 99th percentile statistic from both years, with the WHO’s 24-hour standard shown for reference:

pm2.5\_99th\_percentile2020 <- quantile(dailies2020$pm2.5, probs = .99)  
# calculating the standard error for the percentile estimate - needs review  
pm2.5\_99th\_percentile2020\_se <- sd(dailies2020$pm2.5) / sqrt(nrow(dailies2020))  
  
  
multiyear\_pm2.5\_99th <- mean(c(pm2.5\_99th\_percentile2020, pm2.5\_99th\_percentile2021))  
  
# Bar chart showing 2020 and 2021 PM2.5 daily average 99th percentile w/ WHO reference line  
bardf2.5\_99th <- data.frame(year=c("2020", "2021"),  
 pm=c(pm2.5\_99th\_percentile2020,pm2.5\_99th\_percentile2021),  
 se = c(pm2.5\_99th\_percentile2020\_se, pm2.5\_99th\_percentile2021\_se)  
 )  
  
ggplot(bardf2.5\_99th, aes(x=year, y=pm, fill = year)) +  
 geom\_bar(stat="identity", show.legend = FALSE) +  
 geom\_hline(yintercept=15, linetype="dashed", color = "red") +  
 labs(  
 title = "PM2.5 Daily Average 99th Percentile",  
 subtitle = "WHO Standard",  
 y = expression(Mass - (μg/~m^3)),  
 x = ""  
 ) +  
 theme\_classic()

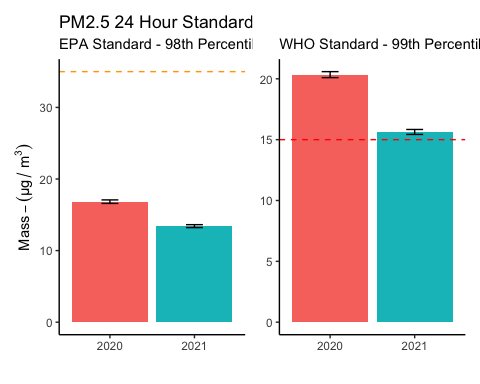


# Bar chart showing average of 2020 and 2021 PM2.5 daily average 99th percentile w/ WHO reference line  
multiyear\_pm2.5\_99thdf <- data.frame(statistic = c("24-Hour Summary Statistic"),  
 PM= c(multiyear\_pm2.5\_99th))  
  
ggplot(multiyear\_pm2.5\_99thdf, aes(x=statistic, y=PM, fill = statistic)) +  
 geom\_bar(stat="identity", width = .5, show.legend = FALSE) +  
 geom\_hline(yintercept=15, linetype="dashed", color = "red") +  
 annotate("text", x = "24-Hour Summary Statistic", y = 15.5, label = "WHO 24-Hour Mean Threshold", size = 3, hjust=0) +  
 labs(  
 title = "PM2.5 Multi-Year 24-Hour Summary Statistic",  
 subtitle = "WHO Standard",  
 y = expression(Mass - (μg/~m^3)),  
 x = ""  
 ) +  
 theme\_classic()



Combining the 24-hour standard metrics into one presentation plot:

EPAstandard\_by\_year <- ggplot(bardf2.5\_98th, aes(x=year, y=pm, fill = year)) +  
 geom\_bar(stat="identity", show.legend = FALSE) +  
 geom\_hline(yintercept=35, linetype="dashed", color = "orange") +  
 geom\_errorbar(aes(ymin = pm - se, ymax = pm + se), width = 0.2) +  
 labs(  
 title = "PM2.5 24 Hour Standard",  
 subtitle = "EPA Standard - 98th Percentile",  
 y = expression(Mass - (μg/~m^3)),  
 x = ""  
 ) +  
 theme\_classic()  
  
WHOstandard\_by\_year <- ggplot(bardf2.5\_99th, aes(x=year, y=pm, fill = year)) +  
 geom\_bar(stat="identity", show.legend = FALSE) +  
 geom\_hline(yintercept=15, linetype="dashed", color = "red") +  
 geom\_errorbar(aes(ymin = pm - se, ymax = pm + se), width = 0.2) +  
 labs(  
 title = "",  
 subtitle = "WHO Standard - 99th Percentile",  
 y = NULL,  
 x = ""  
 ) +  
 theme\_classic()  
  
EPAstandard\_by\_year + WHOstandard\_by\_year



ggsave("24hour\_standard\_by\_year.png", width = 8, height = 4, units = "in")

Recall that an area meets the EPA’s 24-hour standard for PM10 if it does not exceed 150µm/m3 more than once a year, averaged over three years. Here are the number of times that the 24-hour average exceeded 150µm/m3 over the years monitored:

# Bar chart showing number of times that 24-hour average exceeded 150µm/m^3  
num\_over150\_2020 <- dailies2020 %>%  
 filter(  
 pm10 > 150  
 ) %>%  
 nrow()  
  
num\_over150\_2021 <- dailies2021 %>%  
 filter(  
 pm10 > 150  
 ) %>%  
 nrow()  
  
cat("2020: ",num\_over150\_2020, "\n")

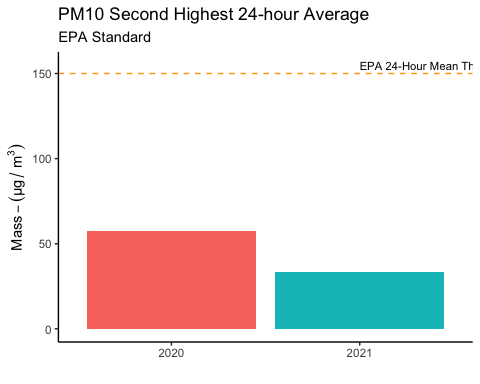
## 2020: 0

cat("2021: ",num\_over150\_2021)

## 2021: 0

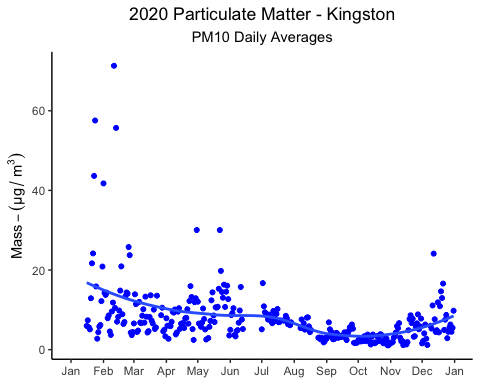
As a proxy summary statistic to compare Kingston’s performance in regards to this standard, let’s look at the 2nd highest 24-hour PM10 average for each year monitored. The EPA’s standard number is shown for reference:

top\_two\_pm10\_2020 <- dailies2020 %>%  
 arrange(desc(pm10))  
  
second\_highest\_pm10\_2020 <- top\_two\_pm10\_2020$pm10[2]  
  
  
# Bar chart showing second highest 24-hour average, with EPA 150 µm/m^3 reference line  
  
  
bardf10\_second\_highest <- data.frame(year=c("2020", "2021"),  
 pm=c(second\_highest\_pm10\_2020,second\_highest\_pm10\_2021))  
  
  
ggplot(bardf10\_second\_highest, aes(x=year, y=pm, fill = year)) +  
 geom\_bar(stat="identity", show.legend = FALSE) +  
 geom\_hline(yintercept=150, linetype="dashed", color = "orange") +  
 annotate("text", x = "2021", y = 155, label = "EPA 24-Hour Mean Threshold", size = 3, hjust=0) +  
 labs(  
 title = "PM10 Second Highest 24-hour Average",  
 subtitle = "EPA Standard",  
 y = expression(Mass - (μg/~m^3)),  
 x = ""  
 ) +  
 theme\_classic()

 Here are the daily averages of PM10 as a time series for the years measured:

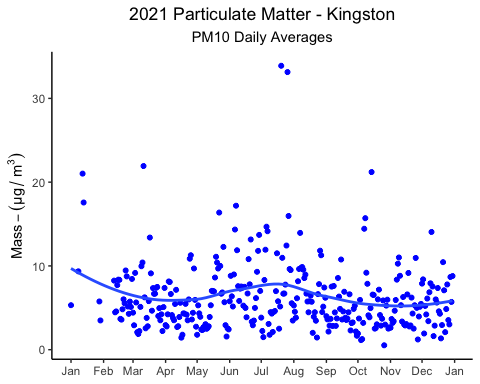
#2020  
ggplot(data = dailies2020,  
 aes(x = date,y = pm10)) +  
 geom\_point(aes(y = pm10), color = "blue", size= 1.4) +  
 #geom\_errorbar(aes(ymin = pm10 - sepm10, ymax = p10 + sepm10), size = .2, width = 1,alpha = 1) +  
 geom\_smooth(se = FALSE) +  
 #geom\_line(aes(y = daily\_mean), color = "red") +  
 scale\_x\_date(limits = as.Date(c("2020-01-01","2020-12-31")), date\_breaks = "months", date\_labels = "%b") +  
 labs(  
 y = expression(Mass - (μg/~m^3)),  
 x = NULL,  
 title = paste(  
 "2020 Particulate Matter - Kingston"  
 ),  
 subtitle = paste(  
 "PM10 Daily Averages"  
 )) +  
 theme\_classic() +  
 theme(plot.title = element\_text(hjust = 0.5)) +  
 theme(plot.subtitle = element\_text(hjust = 0.5))

## `geom\_smooth()` using method = 'loess' and formula 'y ~ x'

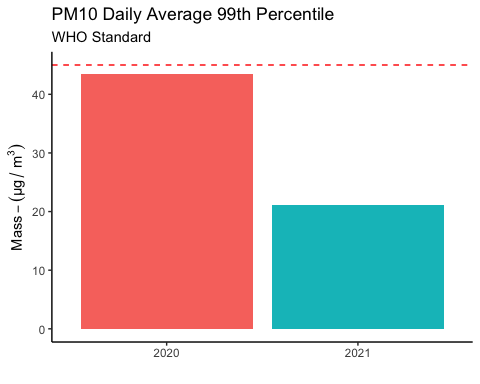


#2021  
ggplot(data = dailies2021,  
 aes(x = date,y = pm10)) +  
 geom\_point(aes(y = pm10), color = "blue", size= 1.4) +  
 #geom\_errorbar(aes(ymin = pm10 - sepm10, ymax = pm10 + sepm10), size = .2, width = 1,alpha = .3) +  
 geom\_smooth(se = FALSE) +  
 #geom\_line(aes(y = daily\_mean), color = "red") +  
 scale\_x\_date(limits = as.Date(c("2021-01-01","2021-12-31")), date\_breaks = "months", date\_labels = "%b") +  
 labs(  
 y = expression(Mass - (μg/~m^3)),  
 x = NULL,  
 title = paste(  
 "2021 Particulate Matter - Kingston"  
 ),  
 subtitle = paste(  
 "PM10 Daily Averages"  
 )) +  
 theme\_classic() +  
 theme(plot.title = element\_text(hjust = 0.5)) +  
 theme(plot.subtitle = element\_text(hjust = 0.5))

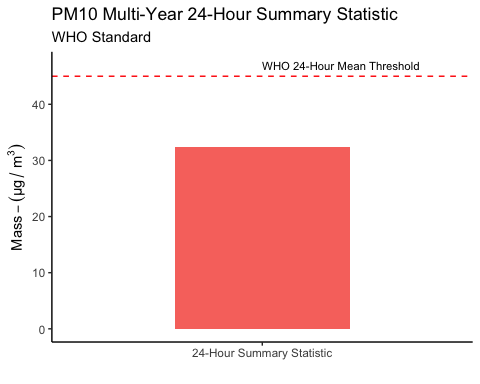
## `geom\_smooth()` using method = 'loess' and formula 'y ~ x'

 Below is shown the PM10 daily average 99th percentile statistic from both years, with the WHO’s 24-hour standard shown for reference:

pm10\_99th\_percentile2020 <- quantile(dailies2020$pm10, probs = .99)  
  
# Bar chart showing 2020 and 2021 PM10 daily average 99th percentile w/ WHO reference line  
  
bardf10\_99th <- data.frame(year=c("2020", "2021"),  
 pm=c(pm10\_99th\_percentile2020,pm10\_99th\_percentile2021))  
  
  
ggplot(bardf10\_99th, aes(x=year, y=pm, fill = year)) +  
 geom\_bar(stat="identity", show.legend = FALSE) +  
 geom\_hline(yintercept=45, linetype="dashed", color = "red") +  
 labs(  
 title = "PM10 Daily Average 99th Percentile",  
 subtitle = "WHO Standard",  
 y = expression(Mass - (μg/~m^3)),  
 x = ""  
 ) +  
 theme\_classic()



multiyear\_pm10\_99th <- mean(c(pm10\_99th\_percentile2020, pm10\_99th\_percentile2021))  
  
# Bar chart showing average of 2020 and 2021 PM10 daily average 99th percentile w/ WHO reference line  
multiyear\_pm10\_99thdf <- data.frame(statistic = c("24-Hour Summary Statistic"),  
 PM= c(multiyear\_pm10\_99th))  
  
ggplot(multiyear\_pm10\_99thdf, aes(x=statistic, y=PM, fill = statistic)) +  
 geom\_bar(stat="identity", width = .5, show.legend = FALSE) +  
 geom\_hline(yintercept=45, linetype="dashed", color = "red") +  
 annotate("text", x = "24-Hour Summary Statistic", y = 47, label = "WHO 24-Hour Mean Threshold", size = 3, hjust=0) +  
 labs(  
 title = "PM10 Multi-Year 24-Hour Summary Statistic",  
 subtitle = "WHO Standard",  
 y = expression(Mass - (μg/~m^3)),  
 x = ""  
 ) +  
 theme\_classic()



## **Nowcast**

The EPA provides a metric called the “NowCast” to quantify and communicate current conditions for criteria air pollutants, including particulate matter. (5) The NowCast formula calculates a concentration number and an associated unit-less Air Quality Index number. Essentially, the NowCast concentration is a 12 hour average that provides extra weight to recent hours if conditions are changing quickly. The EPA provides six different warning level categories for different ranges of concentrations, and provides cautionary guidance with each level.

[ NowCast table]

### Nowcast Concentrations

While the metrics that provide comparison to EPA and WHO standards can be helpful for characterizing the state of particulate matter pollution, they can gloss over the moments when air quality pollutants spike and create unhealthy conditions for area residents, especially vulnerable populations.  
Here we take a look at how many times in this past year did the NowCast number for PM2.5 reach any of the warning level categories that have accompanying cautionary guidance.

2020 PM2.5:

# Calculate PM2.5 Nowcast numbers for each hour of the year, and assign warning category factor variable   
# Count up number of different occurrences for each category besides "Good" and display  
  
nowcastdf2020 <- df2020 %>%  
 group\_by(Time = cut(Time, breaks="60 min")) %>%  
 summarize(  
 PM10 = mean(pm10,na.rm = TRUE),  
 PM2.5 = mean(pm2.5,na.rm = TRUE)  
 ) %>%  
 mutate(  
 time = ymd\_hms(Time)  
 ) %>%  
 select(-Time)  
   
nowcastdf\_computed2020 <- nowcastdf2020 %>%  
# rowwise() %>%  
 mutate(  
 row\_n = row\_number()  
 ) %>%   
 rowwise() %>%  
 mutate(  
 now\_pm2.5 = nowcast(nowcastdf2020[(ifelse(row\_n < 12, 0, row\_n-12)):row\_n,]),  
 now\_pm10 = nowcast10(nowcastdf2020[(ifelse(row\_n < 12, 0, row\_n-12)):row\_n,])  
 )  
   
nowcastdf\_computed2020 <- nowcastdf\_computed2020 %>%  
 mutate(  
 pm2.5category = ifelse(now\_pm2.5< 12.1, "good",   
 ifelse(now\_pm2.5 < 35.5, "moderate",  
 ifelse(now\_pm2.5 < 55.5, "unhealthy for sensitive groups",  
 ifelse(now\_pm2.5 < 150.5, "unhealthy",  
 ifelse(now\_pm2.5 < 250.5, "very unhealthy", "hazardous"  
 ))))),  
 pm10category = ifelse(now\_pm10 < 55, "good",   
 ifelse(now\_pm10 < 155, "moderate",  
 ifelse(now\_pm10 < 255, "unhealthy for sensitive groups",  
 ifelse(now\_pm10 < 355, "unhealthy",  
 ifelse(now\_pm10 < 425, "very unhealthy", "hazardous"  
 ))))),  
 ) %>%   
 mutate(  
 pm2.5category = factor(pm2.5category, levels = c("good","moderate","unhealthy for sensitive groups","unhealthy","very unhealthy", "hazardous")),  
 pm10category = factor(pm10category, levels = c("good","moderate","unhealthy for sensitive groups","unhealthy","very unhealthy", "hazardous"))  
 )   
  
pm2.5nowcastdf\_computed2020 <- nowcastdf\_computed2020 %>%   
 count(pm2.5category, name = "number of observations", .drop = FALSE)  
   
print(pm2.5nowcastdf\_computed2020)

## # A tibble: 6 × 2  
## # Rowwise:   
## pm2.5category `number of observations`  
## <fct> <int>  
## 1 good 6590  
## 2 moderate 376  
## 3 unhealthy for sensitive groups 6  
## 4 unhealthy 3  
## 5 very unhealthy 1  
## 6 hazardous 1

And now for 2020 PM10:

# Calculate PM10 Nowcast numbers for each hour of the year, and assign warning category factor variable   
# Count up number of different occurrences for each category besides "Good" and display  
pm10nowcastdf\_computed2020 <- nowcastdf\_computed2020 %>%   
 count(pm10category, name = "number of observations", .drop = FALSE)  
   
print(pm10nowcastdf\_computed2020)

## # A tibble: 6 × 2  
## # Rowwise:   
## pm10category `number of observations`  
## <fct> <int>  
## 1 good 6940  
## 2 moderate 32  
## 3 unhealthy for sensitive groups 2  
## 4 unhealthy 1  
## 5 very unhealthy 1  
## 6 hazardous 1

2021 PM2.5:

# Calculate PM2.5 Nowcast numbers for each hour of the year, and assign warning category factor variable   
# Count up number of different occurrences for each category besides "Good" and display  
  
  
nowcastdf2021 <- df2021 %>%  
 group\_by(Time = cut(Time, breaks="60 min")) %>%  
 summarize(  
 PM10 = mean(pm10,na.rm = TRUE),  
 PM2.5 = mean(pm2.5,na.rm = TRUE)  
 ) %>%  
 mutate(  
 time = ymd\_hms(Time)  
 ) %>%  
 select(-Time)  
  
# Commenting out this section for now, as there is some unsolved problem with it and we're not using it for now...   
  
# nowcastdf\_computed2021 <- nowcastdf2021 %>%  
# # rowwise() %>%  
# mutate(  
# row\_n = row\_number()  
# ) %>%  
# rowwise() %>%  
# mutate(  
# now\_pm2.5 = nowcast(nowcastdf2020[(ifelse(row\_n < 12, 0, row\_n-12)):row\_n,]),  
# now\_pm10 = nowcast10(nowcastdf2020[(ifelse(row\_n < 12, 0, row\_n-12)):row\_n,])  
# )  
#   
# nowcastdf\_computed2021 <- nowcastdf\_computed2021 %>%  
# mutate(  
# pm2.5category = ifelse(now\_pm2.5< 12.1, "good",  
# ifelse(now\_pm2.5 < 35.5, "moderate",  
# ifelse(now\_pm2.5 < 55.5, "unhealthy for sensitive groups",  
# ifelse(now\_pm2.5 < 150.5, "unhealthy",  
# ifelse(now\_pm2.5 < 250.5, "very unhealthy", "hazardous"  
# ))))),  
# pm10category = ifelse(now\_pm10 < 55, "good",  
# ifelse(now\_pm10 < 155, "moderate",  
# ifelse(now\_pm10 < 255, "unhealthy for sensitive groups",  
# ifelse(now\_pm10 < 355, "unhealthy",  
# ifelse(now\_pm10 < 425, "very unhealthy", "hazardous"  
# ))))),  
# ) %>%  
# mutate(  
# pm2.5category = factor(pm2.5category, levels = c("good","moderate","unhealthy for sensitive groups","unhealthy","very unhealthy", "hazardous")),  
# pm10category = factor(pm10category, levels = c("good","moderate","unhealthy for sensitive groups","unhealthy","very unhealthy", "hazardous"))  
# )  
#   
# pm2.5nowcastdf\_computed2021 <- nowcastdf\_computed2021 %>%  
# count(pm2.5category, name = "number of observations", .drop = FALSE)  
#   
# print(pm2.5nowcastdf\_computed2021)

And now for 2021 PM10:

# Calculate PM10 Nowcast numbers for each hour of the year, and assign warning category factor variable   
# Count up number of different occurrences for each category besides "Good" and display  
# pm10nowcastdf\_computed2021 <- nowcastdf\_computed2021 %>%  
# count(pm10category, name = "number of observations", .drop = FALSE)  
#   
# print(pm10nowcastdf\_computed2021)

### Raw Concentration Observations

While raw observations are not directly comparable to the EPA’s NowCast categories, it may be instructive to see how many times individual 1-minute readings fall into these different concentration ranges.

2020 PM2.5:

# For each obervation that year, assign warning category factor variable   
# Count up number of different occurrences for each category range besides "Good" and display  
  
raw\_counts2020 <- df2020 %>%  
 mutate(  
 pm2.5category = ifelse(pm2.5 < 12.1, "good",   
 ifelse(pm2.5 < 35.5, "moderate",  
 ifelse(pm2.5 < 55.5, "unhealthy for sensitive groups",  
 ifelse(pm2.5 < 150.5, "unhealthy",  
 ifelse(pm2.5 < 250.5, "very unhealthy", "hazardous"  
 ))))),  
 pm10category = ifelse(pm10 < 55, "good",   
 ifelse(pm10 < 155, "moderate",  
 ifelse(pm10 < 255, "unhealthy for sensitive groups",  
 ifelse(pm10 < 355, "unhealthy",  
 ifelse(pm10 < 425, "very unhealthy", "hazardous"  
 ))))),  
 ) %>%   
 mutate(  
 pm2.5category = factor(pm2.5category, levels = c("good","moderate","unhealthy for sensitive groups","unhealthy","very unhealthy", "hazardous")),  
 pm10category = factor(pm10category, levels = c("good","moderate","unhealthy for sensitive groups","unhealthy","very unhealthy", "hazardous"))  
 )   
  
pm2.5raw\_counts2020 <- raw\_counts2020 %>%   
 count(pm2.5category, name = "number of observations", .drop = FALSE)  
   
print(pm2.5raw\_counts2020)

## # A tibble: 6 × 2  
## pm2.5category `number of observations`  
## <fct> <int>  
## 1 good 393411  
## 2 moderate 22352  
## 3 unhealthy for sensitive groups 689  
## 4 unhealthy 167  
## 5 very unhealthy 15  
## 6 hazardous 50

2021 PM2.5:

# For each obervation that year, assign warning category factor variable   
# Count up number of different occurrences for each category range besides "Good" and display  
  
raw\_counts2021 <- df2021 %>%  
 mutate(  
 pm2.5category = ifelse(pm2.5 < 12.1, "good",  
 ifelse(pm2.5 < 35.5, "moderate",  
 ifelse(pm2.5 < 55.5, "unhealthy for sensitive groups",  
 ifelse(pm2.5 < 150.5, "unhealthy",  
 ifelse(pm2.5 < 250.5, "very unhealthy", "hazardous"  
 ))))),  
 pm10category = ifelse(pm10 < 55, "good",  
 ifelse(pm10 < 155, "moderate",  
 ifelse(pm10 < 255, "unhealthy for sensitive groups",  
 ifelse(pm10 < 355, "unhealthy",  
 ifelse(pm10 < 425, "very unhealthy", "hazardous"  
 ))))),  
 ) %>%  
 mutate(  
 pm2.5category = factor(pm2.5category, levels = c("good","moderate","unhealthy for sensitive groups","unhealthy","very unhealthy", "hazardous")),  
 pm10category = factor(pm10category, levels = c("good","moderate","unhealthy for sensitive groups","unhealthy","very unhealthy", "hazardous"))  
 )  
  
pm2.5raw\_counts2021 <- raw\_counts2021 %>%  
 count(pm2.5category, name = "number of observations", .drop = FALSE)  
  
print(pm2.5raw\_counts2021)

## # A tibble: 7 × 2  
## pm2.5category `number of observations`  
## <fct> <int>  
## 1 good 431597  
## 2 moderate 31054  
## 3 unhealthy for sensitive groups 1088  
## 4 unhealthy 492  
## 5 very unhealthy 0  
## 6 hazardous 0  
## 7 <NA> 2

2020 PM10:

pm10raw\_counts2020 <- raw\_counts2020 %>%  
 count(pm10category, name = "number of observations", .drop = FALSE)  
  
print(pm10raw\_counts2020)

## # A tibble: 6 × 2  
## pm10category `number of observations`  
## <fct> <int>  
## 1 good 414323  
## 2 moderate 2169  
## 3 unhealthy for sensitive groups 71  
## 4 unhealthy 20  
## 5 very unhealthy 9  
## 6 hazardous 92

2021 PM10:

pm10raw\_counts2021 <- raw\_counts2021 %>%  
 count(pm10category, name = "number of observations", .drop = FALSE)  
   
print(pm10raw\_counts2021)

## # A tibble: 7 × 2  
## pm10category `number of observations`  
## <fct> <int>  
## 1 good 463239  
## 2 moderate 920  
## 3 unhealthy for sensitive groups 46  
## 4 unhealthy 13  
## 5 very unhealthy 7  
## 6 hazardous 6  
## 7 <NA> 2

## **Conclusion**

[To be written after calculations are done.]

References:  
1) <https://www.epa.gov/criteria-air-pollutants/naaqs-table>  
2) <https://www.who.int/news-room/fact-sheets/detail/ambient-(outdoor)-air-quality-and-health>  
3) <https://www.epa.gov/sites/default/files/2016-04/documents/2012_aqi_factsheet.pdf>  
4) <https://www.in.gov/idem/files/factsheet_oaq_criteria_pm.pdf>