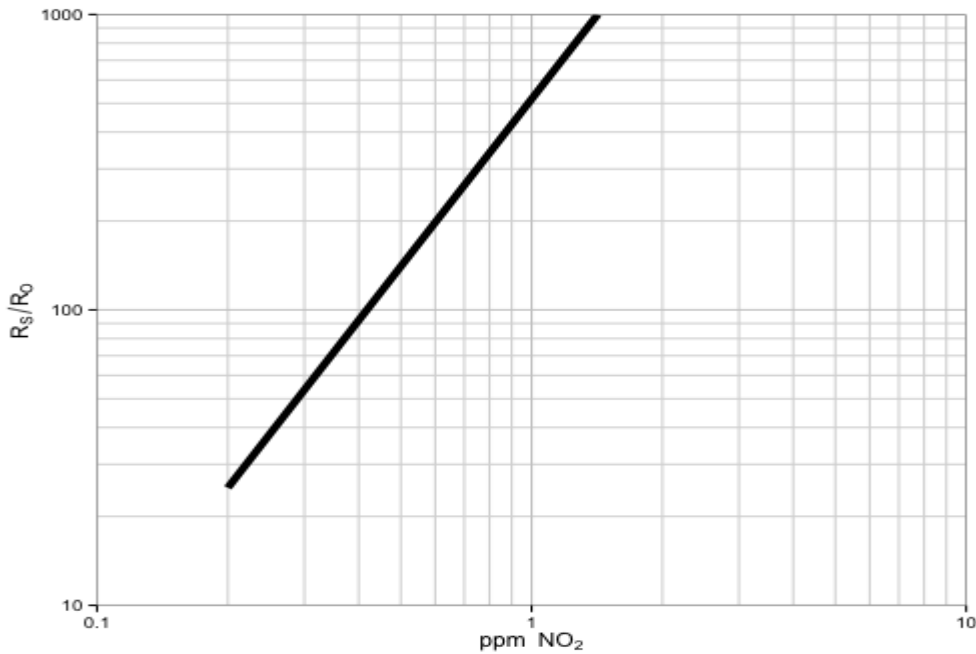


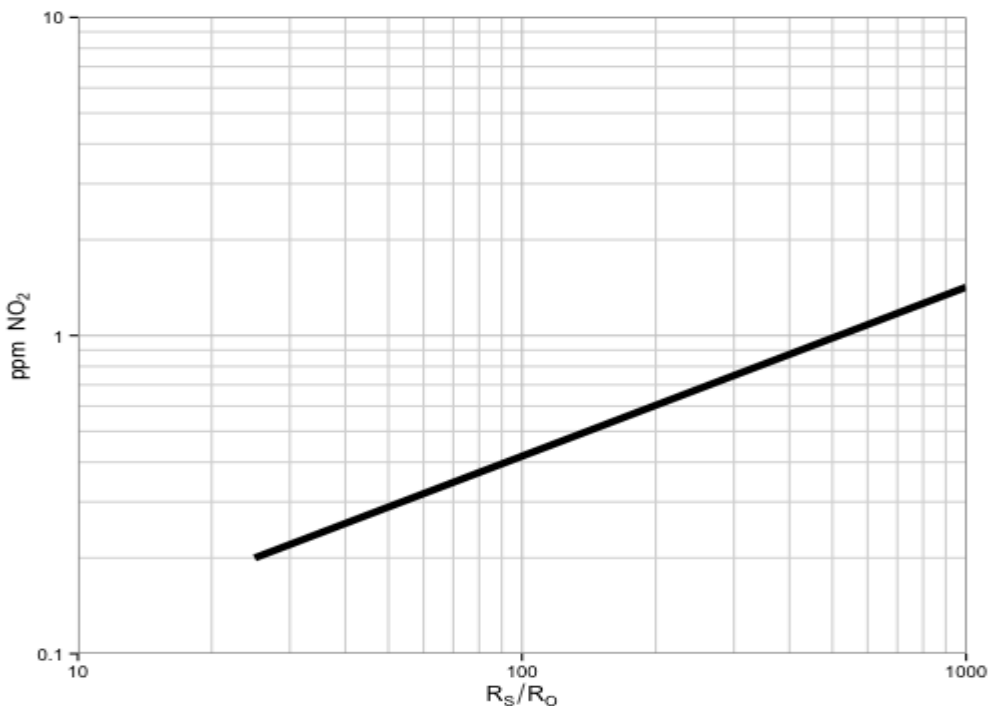
MiCS-2710 Response to NO₂: Model I

Manufacturer's specification

Here is a copy of a spec sheet for the MiCS-2710, obtained from <http://airqualityegg.wikispaces.com/file/view/MICS-2710+-+NO2pdf>. It's a line between two points on a log-log scale. R_o is resistance in "air". R_s/R_o is the ratio of the resistance of the sensor, R_s , to that reference value.

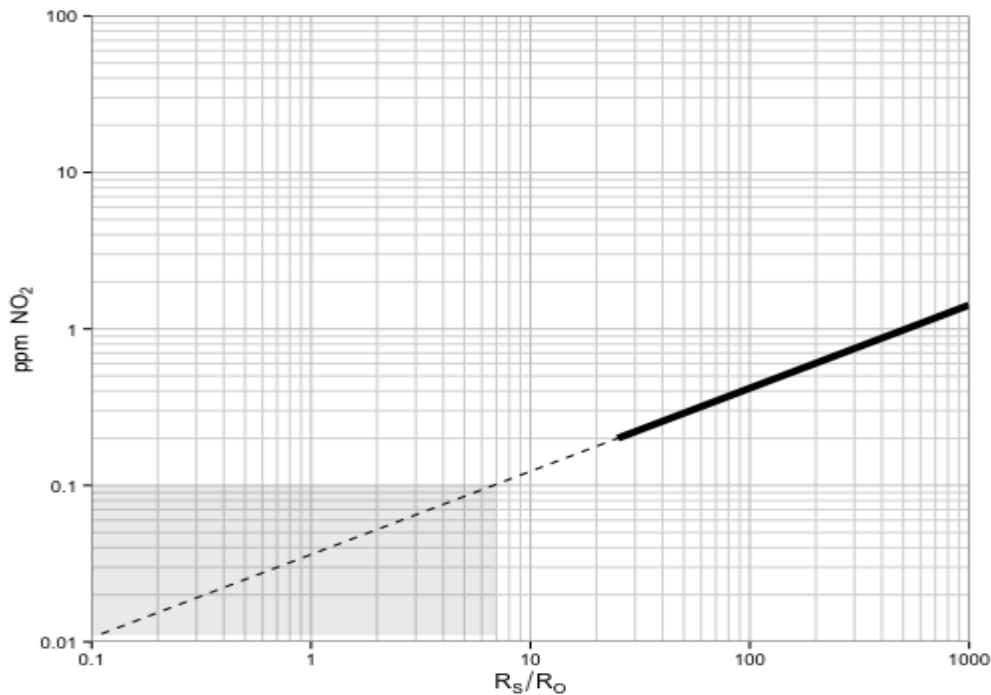


Same plot, with the axes flipped so that R_s/R_o becomes the independent variable.

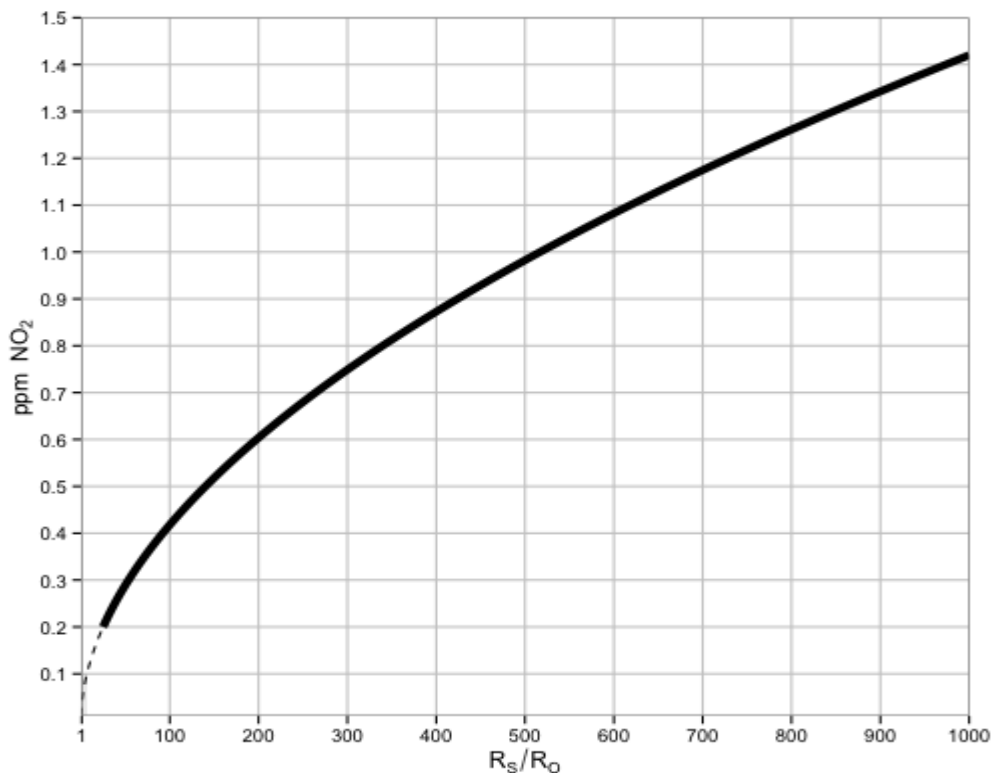


Logarithmic extrapolation

When $R_s/R_o = 1$, we are measuring “air”. Air isn't precisely defined, but on the plot above, an R_s/R_o of 1 corresponds to about 0.036 ppm NO₂. This sounds about right. In urban areas in the United States, typical hourly ambient concentrations of NO₂ are on the order of 0.010-0.100 ppm, and the standard for the maximum annual average is set at 53 ppb, or 0.053 ppm.



Same range of R_s/R_o , but now with arithmetic scales. The NO₂ scale has been seriously zoomed. The region of interest indicated by the gray box is still squashed into the lower-left corner.



Zooming in to the lower left corner, still with arithmetic scales.

Generating values

For those who want to play around, and like to use a different plotting tool, here's R code to construct a forward model (from RsRo to ppm) and an inverse model (from ppm to RsRo).

```
MiCS2710_spec <- data.frame(
  ppm = c(0.20, 1.42),
  RSR0 = c(25, 1000)
)

MiCS2710_ppm <- function(RSR0) {
  model <- lm(ppm ~ RSR0, data=log10(MiCS2710_spec))
  10 ** predict(model, data.frame(RSR0=log10(RSR0)))
}

MiCS2710_RSR0 <- function(ppm) {
  model <- lm(RSR0 ~ ppm, data=log10(MiCS2710_spec))
  10 ** predict(model, data.frame(ppm=log10(ppm)))
}
```

You can use these to generate arbitrary ranges of points, for your own curves, and save them to a CSV file.

```
# Start with either ppm or RSR0
ppm <- seq(from=0.001, to=0.100, by=0.001)

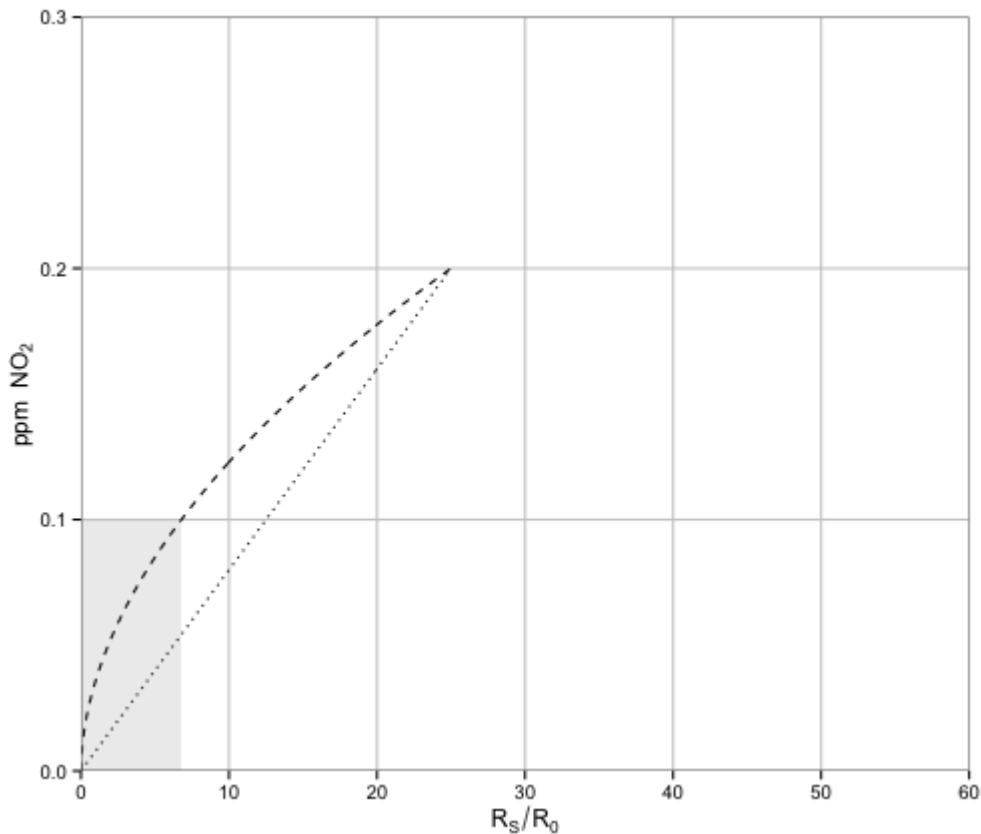
# Use the corresponding function to generate the other variable
RSR0 <- MiCS2710_RSR0(ppm)

# Bind them together and save to a CSV file
data <- data.frame(RSR0, ppm)
write.csv(data, file='mydata.csv', row.names=FALSE)
```

Linear extrapolation

Suppose our model of sensor response at low concentrations is (piecewise) linear, rather than logarithmic, and goes straight from the lower bound of the manufacturer's curve to zero.

```
x0 <- with(MiCS2710_spec, min(RSR0))
y0 <- with(MiCS2710_spec, min(ppm))
f <- function(x) (y <- x * y0 / x0)
data <- data.frame(RSR0=seq(0, x0, by=0.1))
data <- transform(data, logarithmic=MiCS2710_ppm(RSR0), linear=f(RSR0))
head(data)
##   RSR0 logarithmic linear
## 1  0.0      0.0000 0.0000
## 2  0.1      0.0106 0.0008
## 3  0.2      0.0154 0.0016
## 4  0.3      0.0191 0.0024
## 5  0.4      0.0222 0.0032
## 6  0.5      0.0250 0.0040
```



So, if the logarithmic curve is the “true” curve, then this method would result in an underestimate of the “true” concentration. We can look at the average difference, in ppm, over the region of interest.

```

difference <- function(x) MiCS2710_ppm(x) - f(x)
ratio <- function(x) MiCS2710_ppm(x) / f(x)

integrate(difference, 0.01, x0)$value / (x0 - 0.01)
## [1] 0.0306
integrate(ratio, 0.01, x0)$value / (x0 - 0.01)
## [1] 1.85

```

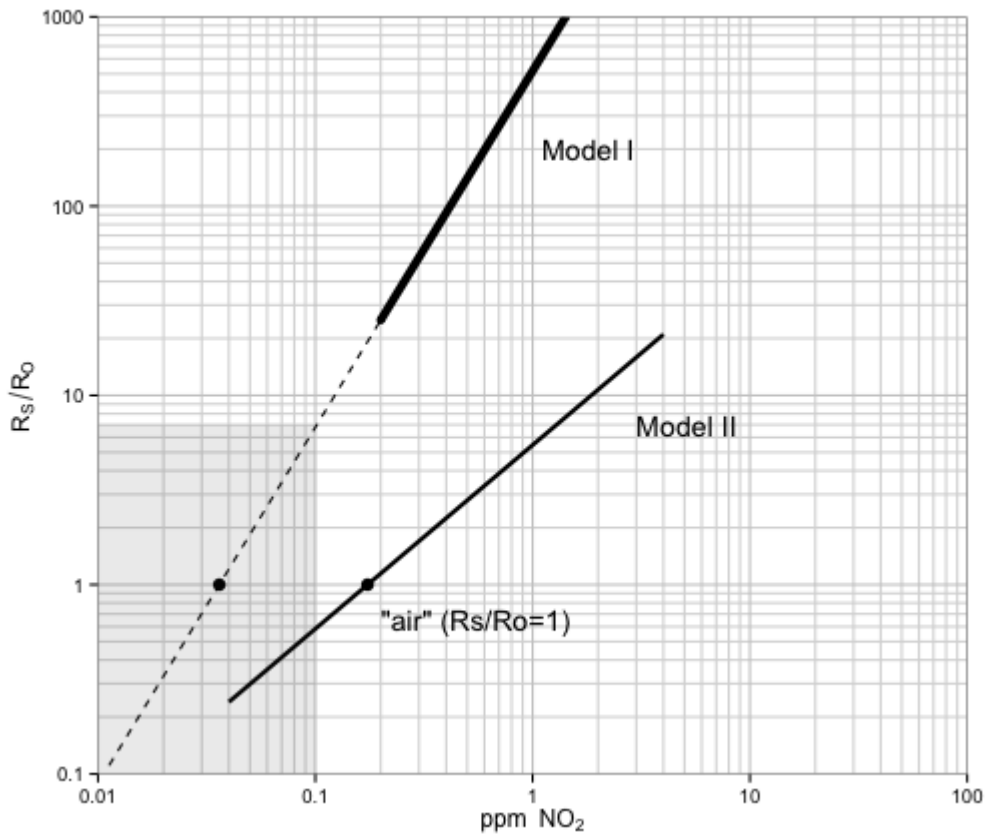
On average, there's about a 2x difference on a ratio scale, or +0.031 on an arithmetic scale. That's not entirely fair, since not every value in the region of interest is equally probable, but it's a start.

Caveat

NEITHER method is guaranteed to be appropriate. Both are extrapolations with no support from nearby data. The point of collecting calibration data in this range is to figure out what the RIGHT function looks like—as well as how much error to expect (which hasn't been discussed) from measurement to measurement, and from sensor to sensor.

MiCS-2710/2714 Sensor: Model II

For comparison, here is a different curve (thin line) for NO₂ response for the MiCS2710/2714 obtained from <http://airqualityegg.wikispaces.com/file/view/e2v+Mics+SensorCharacterizationData.pdf>. The response curve from the figures above (thick line) has also been plotted. We're back on a log-log scale with NO₂ (not Rs/Ro) as the independent variable.



Two questions:

- What accounts for the drastic difference between the two curves? Both are supposed to apply to the MiCS-2710, correct? Are the temperature and humidity conditions the same?
- Why, in model II, does $R_s/R_o=1$ (that is, "air") correspond to an NO_2 concentration of over 100 ppb? This is more than twice the average hourly concentration at most "regional" monitors in developed countries. (I'm most familiar with the US, and I'm not certain what the average is in, say, Delhi, or Beijing, though one could try to find this out. In any setting one can definitely find higher concentrations on or near highways.)