Problem 1: Consider the following function,

$f(x_1, x_2, x_3) = |2x_1 + x_2 - x_3 + 1| + |x_1 - 2x_2 + x_3 - 2| + |3x_1 - x_2 + 2|$

This function is ***not linear***, nor does it have any constraints, however, we can still find its minimum point $\mathbf{x}$ by converting this problem into a linear optimization problem. The trick here (as illustrated in class) is to construct the linear function,

$$\widehat{f}(x_1, x_2, x_3, y_1, y_2, y_3) = y_1 + y_2 + y_3$$

with the constraints (there are six of them) that,

$$\begin{cases} |2x_1 + x_2 - x_3 + 1| & \leq & y_1 \\ |x_1 - 2x_2 + x_3 - 2| & \leq & y_2 \\ |3x_1 - x_2 + 2| & \leq & y_3 \end{cases}$$

Now find the minimum point for $\widehat{f}$ and it will give you (after you disregard that y's) the minimum point for the original f.
The minimum occurs at $x_1 = \frac{1}{3}, x_2 = 0, x_3 = \frac{4}{3}$.

Now consider the function, (we call this a "***minimax*** *problem*")

$g(x_1, x_2, x_3) = \max\Big(|2x_1 + x_2 - x_3 + 1|, |x_1 - 2x_2 + x_3 - 2|, |3x_1 - x_2 + 2|\Big)$

This function is also not linear, but we can do a trick to convert it into a linear problem and then determine its minimum point. Construct the linear function,

$$\widehat{g}(x_1, x_2, x_3, z) = z$$

but with the constraints that,

$$\begin{cases} |2x_1 + x_2 - x_3 + 1| & \leq & z \\ |x_1 - 2x_2 + x_3 - 2| & \leq & z \\ |3x_1 - x_2 + 2| & \leq & z \end{cases}$$

Minimum occurs at $\mathbf{x} = (0, 1, 3)$.

Problem 2: Suppose $(x_i, y_i)_{1\leq i\leq n}$ are a collection of data points.

Write a code in R for finding the equation of the line $y = ax + b$ such that "*the maximum error is minimized*" i.e. minimize the maximum error (this is exactly just like the ***minimax*** in **Problem 1**). More specifically if $\varepsilon_i = |ax_i + b - y_i|$ then we wish to minimize the quantity,

$$\max(\varepsilon_1, ..., \varepsilon_n)$$

by choosing a and b.

The objective function,

$$f(a, b) = \max_{1\leq i\leq n} |ax_i + b - y_i|$$

is not a linear function. Therefore, extra tricks will be required to convert this problem into a linear optimization problem.

Instead consider the linear function of three variables,

$$g(a, b, \theta) = \theta$$

Minimize this linear function subject to the constraints that,

$$|ax_i + b - y_i| \leq \theta$$

(Note, in order for θ to be small as possible but to still $\geq \varepsilon_i$, it means that θ must be the maximum of all the ε's. Therefore, this new linear version of the property will achieve the minimum that we seek.)

Call your code `best.line.minmax(x,y)`. In constrast, in class we developed a code called `best.line(x,y)`. The line-of-best-fit was optimal from all lines that minimized the total error.

Problem 3: The package `datasets` is probably automatically loaded when you start R. If you are starting R Studio Cloud then it will automatically be loaded. This package is just a collection of various data that was used in research. The only purpose to these data sets is to practice.

One of the data sets present in this package is called `women`. If you run `women` in R you will get a table of 15 women listed according to their height (inches) and weight (lbs) that was used in someone's research. Type,

```
x=women$height
y=women$weight
```

You can check now that `x` and `y` are vectors consisting 15 numbers each. If you type `plot(x,y)` will get a scatter plot of women's height vs women's weight. Notice how close to a straight line these lines are positioned.

Find the line-of-best-fit which minimizes the maximum error (previous problem) and find the line-of-best-fit which minimizes the total error (from class). Plot both of these lines visually together in the same plot. You should notice that even though you used two different methods the lines are both very close to each other.

Warning: The regression line for minimizing total deviation is given by $y = 3.375x - 82.875$, however if you run our class code you would get that $y = 2.07x$. Indeed, we can visually see that $y = 2.07x$ does not have a good fit. The reason for this is because in the line $y = ax + b$ the value of b needs to be chosen as *negative.* The package `lpSolve` makes the assumption that all variables are ≥ 0. To fix this issue set up the regression line to have the form $y = ax - b$, where now $a, b \geq 0$. Now your program will correctly find the values.