Avoid Fines by Warming-Up the Machines [*]

S. Elaine Hale[†] and Martin J. Mohlenkamp[‡]

April 29, 2014

Abstract

A company produces 1000 metal cylinders a day for 10 days. The government Safety Authority will take a random sample of cylinders, take a random sample of locations within each of these cylinders, and test for percentage Nickel (Ni) and percentage Chromium (Cr). For each sample in which the percentages are outside some intervals, they fine the company €1,000,000. If the proportion of percentages outside of some tighter intervals is too high, they fine the company €1,000,000. For quality control, the company first takes their own sample of the percentage Ni and Cr at some locations on some cylinders. Based on this data, they wish to know the expected value of the fine they must pay.

Based on the data collected by the company, we determine the functional dependence of the percentages Ni and Cr on the day, cylinder number within that day, and location of the sample on that cylinder. We then used two different approaches to find the expected fine. By determining the frequencies of cells violating the specification intervals and making simplifying assumptions on the sampling method, we used properties of Binomial distributions to determine the expected value of the fine to be €430,000. By simulating the sampling and fine method of the Safety Authority, we determine a Monte Carlo estimate of the expected fine to be €429732; by performing a bootstrap analysis, we obtain a 95% confidence interval of $[427879, 431588]$.

(Note: This paper is an entry in a game.)

Keywords: game, Monte Carlo, R

[*] Produced in the class MATH 5530 Statistical Computing Spring 2014; Department of Mathematics, Morton Hall 321, 1 Ohio University, Athens OH 45701

[†] sh151509@ohio.edu

[‡] mohlenka@ohio.edu

1 Introduction

The mathematical game *Checking an Industrial Process* [1] presents the following scenerio:

A company produces 1000 metal cylinders a day for 10 days, so 10,000 total. The government Safety Authority will take a random sample of 100 of these 10,000 cylinders (uniform, without replacement). Each cylinder will be cut into 5 layers, and each layer cut into 80 cells (plus some scrap), so a total of 400 cells. From these 400 cells, the Safety authority will sample 10 (uniform, without replacement). In total, the Safety Authority tests 1000 cells.

The percentage Nickel (Ni) and percentage Chromium (Cr) is determined in each cell. For each cell that violates the specification intervals

$$\text{Ni} \in I_1 = [6.94, 9.10] \qquad \text{and} \qquad \text{Cr} \in J_1 = [16.95, 19.10]\,, \tag{1}$$

the Safety Authority fines the company €1,000,000. The number of cells that violate the (stricter) specification intervals

$$\text{Ni} \in I_2 = [7, 9] \qquad \text{and} \qquad \text{Cr} \in J_2 = [17, 19] \tag{2}$$

is counted; if there are more than 50 then the Safety Authority fines the company an additional €1,000,000.

For quality control, the company first takes their own sample of the percentage Ni and Cr at some locations on some cylinders. The system of splitting a cylinder into layers and cells is the same as the Safety Authority's, but the selection of which cylinders and cells is not. This data was provided to us in a spreadsheet.

The question presented in the game is:

What is the expected value of the fine?

The data collected by the company has values for percentages Ni and Cr, the day, cylinder number within that day, and location of the sample cell on that cylinder given in three coordinates. In Table 1 we gather these variables and set the notation we will use for them. Based on the data collected by the company, we determine the functional dependence of the percentages Ni and Cr on the day, cylinder number within that day, and location of the sample on that cylinder to be

$$\text{Ni}(d, c, x, y, z) = 8 + \frac{11}{450}\left(x + \frac{1}{2}\right) e^{-y^2}(z+1)\left(1 + e^{-(c-1)/1000}\right) \quad \text{and} \tag{3}$$

$$\text{Cr}(d, c, x, y, z) = 18 + \frac{11}{900}\left(x + \frac{1}{2}\right) e^{-y^2}(10 - z)\left(1 + e^{-(c-1)/1000}\right)\,. \tag{4}$$

These formulas fit the data extremely well, so we believe they were used to generate the data. In Section 2 we describe the analysis that led to (3) and (4).

In this paper	In data spreadsheet	Possible values	Comments
d	day	$\{1, 2, \ldots, 9, 10\}$	
c	nb of cylinder	$\{1, 2, \ldots, 999, 1000\}$	
x	coord_x of cells	$\{-5, -4, \ldots, 3, 4\}$	$(x + 1/2)^2 + (y + 1/2)^2 \leq 25$
y	coord_y of cells	$\{-5, -4, \ldots, 3, 4\}$	$(x + 1/2)^2 + (y + 1/2)^2 \leq 25$
z	coord_z of cells	$\{0, 1, 2, 3, 4\}$	
Ni	%Ni	$[0, 100]$	Concentrated near 8
Cr	%Cr	$[0, 100]$	Concentrated near 18

Table 1: Variables used.

With these formulas in hand, we took two approaches to determining the expected value of the fine. In the first approach, we simplified the Safety Authority's sampling method to testing 1000 cells, with replacement, from among the 400 × 1000 in one day's production. By determining the frequencies of cells violating the specification intervals (1) and (2), we can then use properties of Binomial distributions to determine the expected fine. With this method, we conclude the expected value of the fine to be €430,000. In Section 3 we describe this approach and its results.

In the second approach, we simulate the sampling and fining method of the Safety Authority many times and produce a Monte Carlo estimate of the expected value of the fine. The basic method is to simulate the sampling method of the Safety Authority to create 1000 values of (d, c, x, y, z), plug into (3) and (4) to get 1000 values of Ni and Cr, and then compute the fine. Replicating this process 5000 times and averaging the resulting fines yielded a Monte Carlo estimate of the expected fine of €429200. By performing a bootstrap on the Monte Carlo samples of the fine, we obtain a 95% confidence interval of $[410421, 447722]$. This basic method does a lot of unnecessary work and so became slow when attempted for large numbers of replicates. By instead precomputing the number of cells in each cylinder that violate the specifications (1) and (2), we obtain an equivalent but much faster method to generate sample fines. Using 500,000 replicates we determine another Monte Carlo estimate of the expected fine to be €429732. By performing a bootstrap on the Monte Carlo samples of this fine, we obtain a 95% confidence interval of $[427879, 431588]$. The expected value of the fine did not change much, but we now have a much tighter confidence interval. In Section 4 we describe this approach and its results.

The expected values of the fines using simplifying assumptions and using Monte Carlo are quite close. Our analysis provides other information that may be useful to the company. The formulas (3) and (4) indicate where their production process has problems. In particular, for $200 < c$ the specifications (1) and (2) are never violated. If the machinery was better warmed-up, or the process run continuously, the company would produce a better product and never face a fine.

This work was performed as part of a Statistical Computing class at Ohio University [2]. We would like to thank other students in the class for their contributions. Computations were performed using the R programming language [3]. Calculations and collaborative work was done on the SageMath Cloud (`https://cloud.sagemath.com`) using Sage worksheets [4]. Sage worksheets providing further details and enabling our results to be reproduced are available at `http://www.ohio.edu/people/mohlenka/20142/4530-5530/game`.

2 Determining the Functional Dependence of Ni and Cr on x, y, z, c, and d

An initial understanding of the dependence of Ni and Cr on the variables x, y, z, c, and d is shown in Figure 1, which is produced by the pairs() function in R. There appears to be a linear dependence of the variation on x, with smallest value near the center. There appears to be a linear dependence of the variation on z, with Ni variation growing with z and Cr variation decreasing with z. The variation is largest for y near the center and then decreases rapidly away from the center. With hindsight, we can say the variation decreases slowly with c and is independent of d, but that is not obvious in the plot. Our strategy is to remove these dependencies one by one until we have determined the functions (3) and (4).

2.1 Nickel Dependence

In Figure 2 on the left we plot x versus Ni, and can see the linear relationship. The lines seem to cross at $(8, -1/2)$. The mean of Ni is 8, and the x coordinate is of the lower-left corner of the sampled square, so that $x + 1/2$ is the physical center of the disc. To remove our dependence on x, we subtracted 8 and divided by $x + 1/2$. In Figure 2 on the right we plot x versus $(\mathrm{Ni} - 8)/(x + 1/2)$, and can see the relationship seems to have been removed.

In Figure 3 on the left we plot z versus $(\mathrm{Ni}-8)/(x+1/2)$, and can see a linear relationship, with slope and intercept determined by y. When fitting a line to this data restricting to $y = 0$, we noticed that the intercept and slope are similar, which suggests dependence proportional to $z + 1$. To remove our dependence on z, we divided by $(z + 1)$, and show the results on the right of Figure 3.

In Figure 4 on the left we plot y versus $(\mathrm{Ni} - 8)/(x + 1/2)/(z + 1)$, and can see a strong decay away from $y = 0$. Note that the way the cells are specified, $y = 0$ is not the phyical center of the disc. Trying several functional forms using the nls() function in R, we found $\exp(-y^2)$ to be an excellent fit. To remove our dependence on y, we divided by $\exp(-y^2)$, and show the results on the right of Figure 4. We remark that to create this figure we used the full precision available in the data spreadsheet and not just the digits displayed by default.

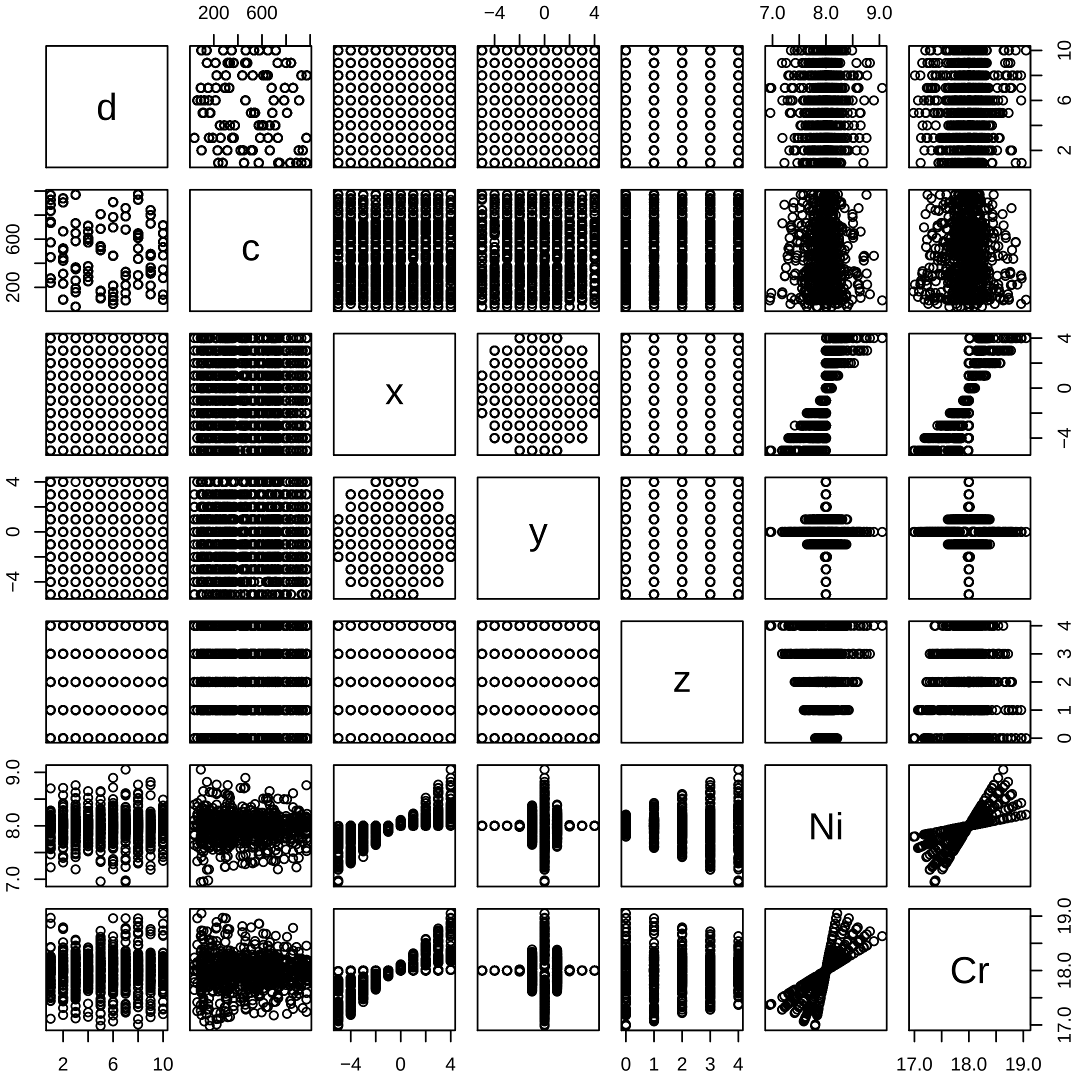

Figure 1: A pairs plot of Ni and Cr with the variables x, y, z, c, and d.

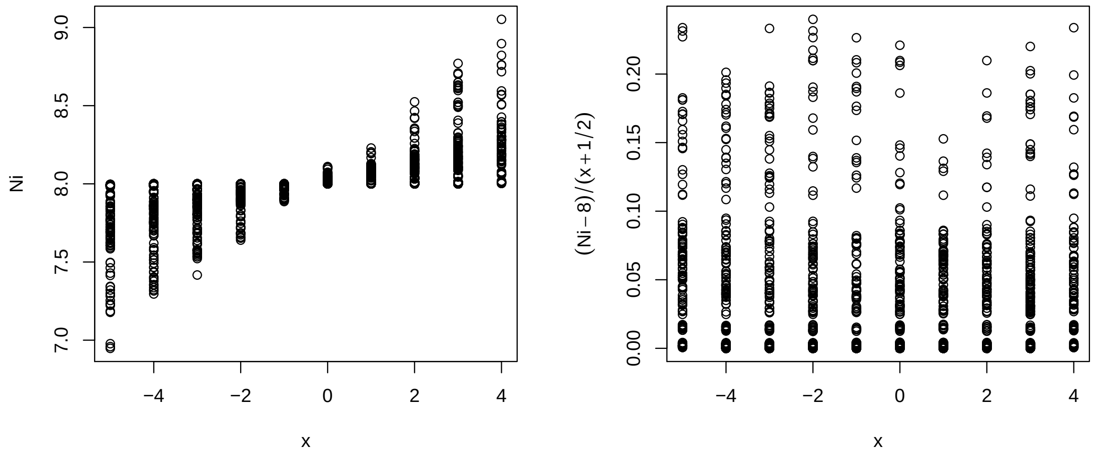

Figure 2: Left: The dependence of Ni on x. Right: The result of removing the dependence via $(\mathrm{Ni}-8)/(x+1/2)$.

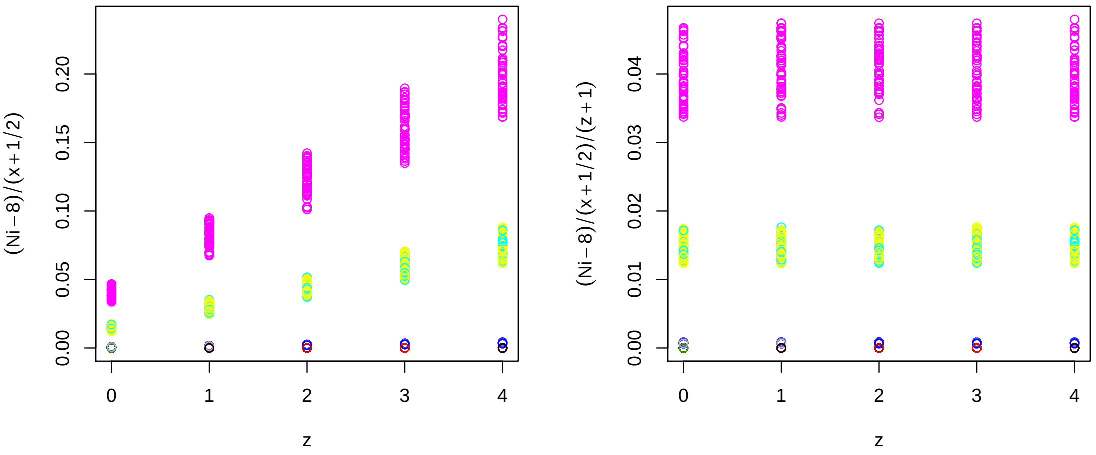

Figure 3: Left: The dependence of $(\mathrm{Ni}-8)/(x+1/2)$ on z, colored by y. Right: The result of removing the dependence via $(\mathrm{Ni}-8)/(x+1/2)/(z+1)$.

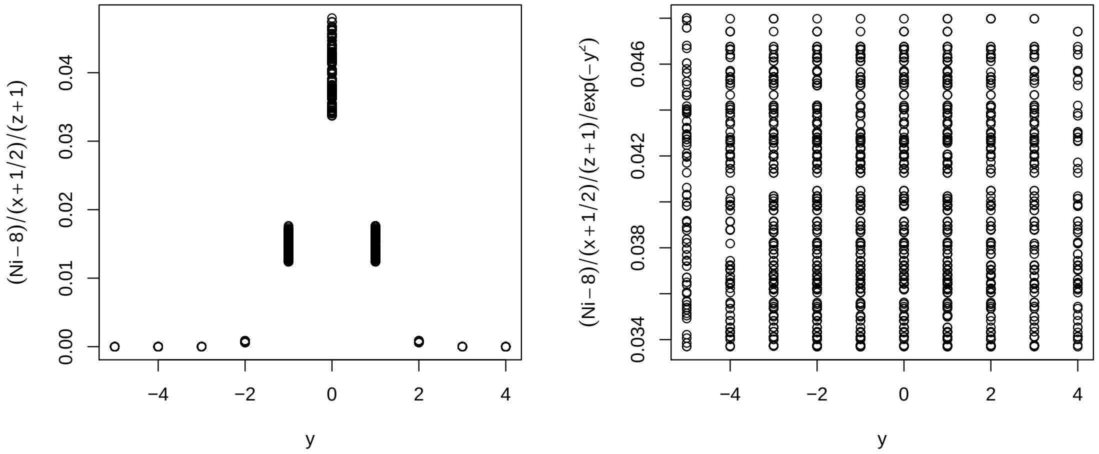

Figure 4: Left: The dependence of $(\mathrm{Ni} - 8)/(x + 1/2)/(z + 1)$ on y. Right: The result of removing the dependence via $(\mathrm{Ni} - 8)/(x + 1/2)/(z + 1)/\exp(-y^2)$.

In Figure 5 on the left we plot c versus $(\mathrm{Ni} - 8)/(x + 1/2)/(z + 1)/\exp(-y^2)$, and can see a slow non-linear decrease away from $c = 1$. Trying several functional forms using the nls() function in R, we found $(1 + \exp(-(c - 1)/1000))$ to be an excellent fit. To remove our dependence on c, we divided by $(1 + \exp(-(c - 1)/1000))$, and show the results on the right of Figure 5. The outlying points correspond to larger values of $|y|$ and are due to loss-of-precision errors.

In Figure 6 we plot d versus $(\mathrm{Ni}-8)/(x+1/2)/(z+1)/\exp(-y^2)/(1+\exp(-(c-1)/1000))$, and can see no apparent dependence. We can see there is a scalar factor of about 0.244 that we still need to include. Restricting to $y = 0$ shows the scalar to be $0.2\overline{4} = 11/450$. Reversing all our operations, we obtain the formula (3).

To test the accuracy of the formula (3), we compute the difference between the value it gives and the value of Ni in the data. The data is provided in two sets, with “Destructive” testing including cells throughout the cylinder and “Non-Destructive” testing only including cells on the top an bottom surfaces of the cylinder. For the Descructive dataset, the maximum difference is less than 7×10^{-15} in absolute value, indicating that the only error is roundoff error. (Note that we used the full precision in the data spreadsheet, not just the digits displayed by default.) In the Non-Destructive dataset, the maximum difference is also less than 7×10^{-15} in absolute value, except for a single outlier. This outlier occurs at index 802 in the given Excel data sheet. For $(d, c, x, y, z) = (5, 3, -4, 0, 0)$ the Ni value is given as 7.78021978014659, whereas (3) gives value approximately 7.82906. We have no explanation for this outlier, but note that both the given value and our prediction are well within (2).

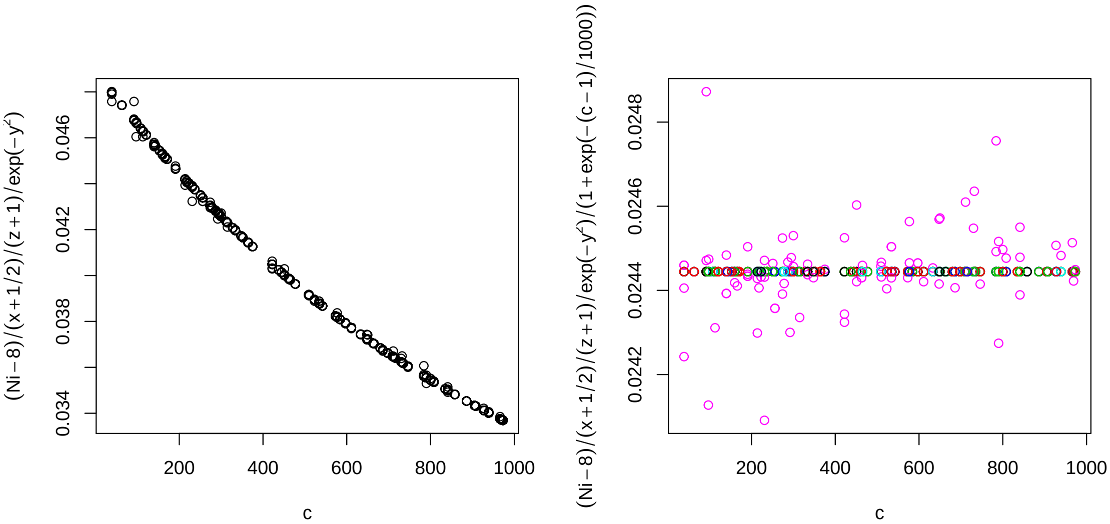

Figure 5: Left: The dependence of (Ni$-$8)/$(x$+1/2)/$(z$+1)/ exp$(-y^2)$ on c. Right: The result of removing the dependence via (Ni$-$8)/$(x$+1/2)/$(z$+1)/ exp$(-y^2)$/(1+exp$(-(c$$-$1)/1000)), colored by $|y|$.

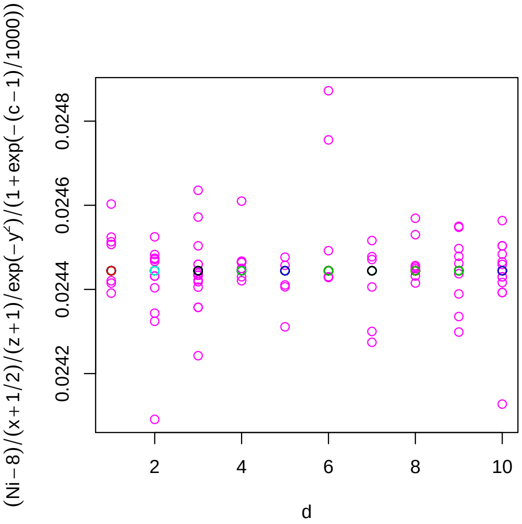

Figure 6: The dependence of $(\mathrm{Ni}-8)/(x+1/2)/(z+1)/\exp(-y^2)/(1+\exp(-(c-1)/1000))$ on d, colored by $|y|$.

2.2 Chromium Dependence

We followed a similar process to determine the formula for Cr. We have a linear relationship between Cr and x. The lines seem to cross at (18,-1/2). The mean of Cr is 18, and the x coordinate is of the lower-left corner of the square. Therefore to remove our dependence on x, we subtracted 18 and divided by $x + 1/2$. We have a linear relation between Cr and z. We noticed that the intercept and slope differ by a factor of -10, which suggests Ni is proportional to $10 - z$. To remove our dependence on z, we divided by $(10 - z)$. In Figure 7 we show the dependence on z and the result of dividing by $(10 - z)$. As with Ni, the dependence on y is $\exp(-y^2)$, the dependence on c is $(1+\exp(-(c-1)/1000))$, and their is no dependence on d. The difference between our formula (4) and the value of Cr in the data is less than 5×10^{-14} for both the Destructive and Non-Destructive datasets, with no exceptions.

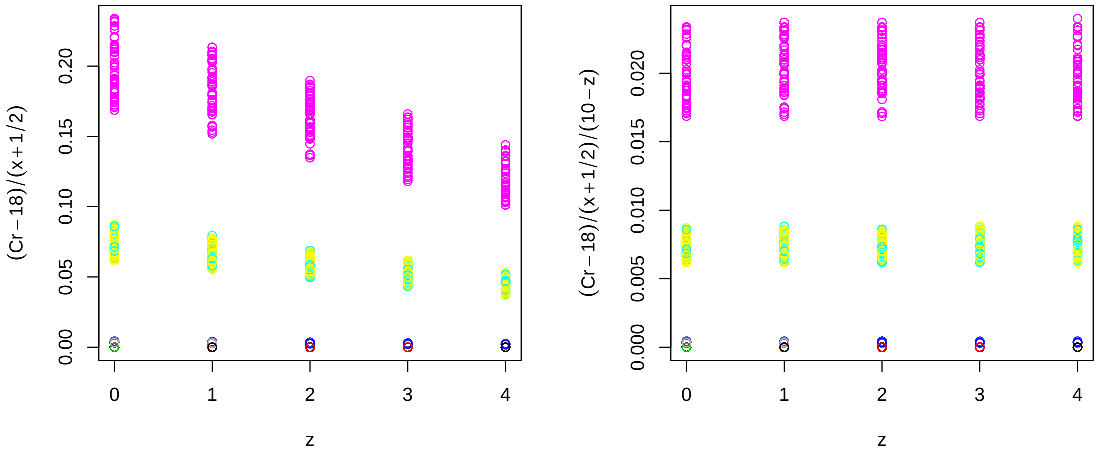

Figure 7: Left: The dependence of $(\mathrm{Cr} - 18)/(x + 1/2)$ on z, colored by y. Right: The result of removing the dependence via $(\mathrm{Cr} - 18)/(x + 1/2)/(10 - z)$.

3 Semi-Analytic Approach

We can calculate the expected fine directly if we make a simplifying assumption:

> The Safety Authority samples 1000 cells uniformly and with replacement from among the 400 × 1000 cells in one day's production.

With this assumption, the expected fine is €1,000,000 times

$$1000p_1 + \left(1 - \sum_{k=0}^{50} B(k; 1000, p_2)\right) \tag{5}$$

where p_1 is the probability of a cell violating the specification (1), p_2 is the probability of a cell violating the specification (2), and $B(k; 1000, p_2)$ is the probability of k successes in a Binomial distribution with 1000 trials and probability of success p_2. The first term in (5) is the expected number of violations of the specification (1), and the second is the probability that more than 50 cells violate the specification (2).

Given our formulas (3) and(4) for Ni and Cr, we can compute p_1 and p_2 directly, by

$$p_1 = \frac{1}{400 \times 1000} \sum_{c=1}^{1000} \sum_{(x,y,z)} \chi_1(x, y, z, c) \quad \text{and}$$

$$p_2 = \frac{1}{400 \times 1000} \sum_{c=1}^{1000} \sum_{(x,y,z)} \chi_2(x, y, z, c)$$

with

$$\chi_1(x, y, z, c) = \begin{cases} 0 & \text{if Ni}(1, c, x, y, z) \in I_1 \text{ and Cr}(1, c, x, y, z) \in J_1 \\ 1 & \text{otherwise} \end{cases} \quad \text{and}$$

$$\chi_2(x, y, z, c) = \begin{cases} 0 & \text{if Ni}(1, c, x, y, z) \in I_2 \text{ and Cr}(1, c, x, y, z) \in J_2 \\ 1 & \text{otherwise} \end{cases} .$$

We find that

- For $1 \leq c \leq 76$, 2 cells violate (1) and 4 cells violate (2).
- For $76 < c \leq 96$, 1 cell violates (1) and 4 cells violate (2).
- For $96 < c \leq 200$, no cells violate (1) and 4 cells violate (2).
- For $200 < c$ there are no violations.

These counts lead to the probabilities $p_1 = 0.00043$ and $p_2 = 0.00201$. The second term in (5) is negligible and so the total fine is €1,000,000p_1 =€430,000.

The simplifying assumption is of course incorrect, but it is reasonable to consider. We determined that the Ni and Cr content does not depend on the day variable d, so the information on which day a cylinder c came from is discarded anyway. The dependence on the variable c is slow and smooth, so it matters little if a cell has a specific c or a nearby value of c; in particular it matters little if 10 cells are selected from the same cylinder or not. Since there are 400 cells in a cylinder and we only choose 10, there is little chance of a repetition if we allow replacement.

4 Monte Carlo Estimate of the Expected Value of the Fine

To reproduce the sampling and fine method of the Safety Authority, we use the following method:

1. Take 100 samples uniformly without replacement from the integers $1, 2, \ldots, 10000$. Interpret the resulting numbers as $c + 1000(d - 1)$ to determine d and c values.

2. For each (c, d), take 10 samples uniformly without replacement from the integers $1, 2, \ldots, 400$. Using a precomputed array, map each number to a valid cell (x, y, z).

3. Use (3) and (4) to determine the Ni and Cr values for each of the 1000 values of (d, c, x, y, z).

4. For each violation of (1) obtained, add a fine of €1,000,000. If more than 50 violations of (2) were obtained, add a fine of €1,000,000.

We replicated this process $n = 5000$ times and averaged the fines to obtain an expected fine of €429200. We then used the boot library in R to bootstrap our Monte Carlo results to determine a 95% confidence interval of [410421, 447722].

This basic method is quite inefficient because it does many floating point operations per cell, whereas all we really care about is whether the cell violates the specifications (1) and (2). We can use the number 2 to mark a cell that violates (1) and thus also (2), 1 to mark a cell that violates (2) only, and 0 to mark a cell with no violations. In Section 3 we already counted how many cells violated the specifications. Since the sampling in each cylinder is uniform, it does not matter where these cells are positioned in (x, y, z), or where in the list of 400 possible cells. We can thus simplify the algorithm, without changing the results, to:

1. (unchanged)

2. For each c, if $200 < c$ do nothing; otherwise take 10 samples uniformly without replacement from the array of length 400:

   if $1 \le c \le 76$ use $[2, 2, 1, 1, 0, \ldots, 0]$ or

   if $76 < c \le 96$ use $[2, 1, 1, 1, 0, \ldots, 0]$ or

   if $96 < c \le 200$ use $[1, 1, 1, 1, 0, \ldots 0]$.

   Count and accumulate the number of 2's obtained and the number of nonzeros obtained.

3. For each 2 obtained, add a fine of €1,000,000. If more than 50 nonzeros were obtained, add a fine of €1,000,000.

We replicated this process $n = 500000$ times and averaged the fines to obtain an expected fine of €429732. In the left of Figure 8 we show the (estimated) probability distribution function of the fine. We then used the boot library in R to bootstrap our Monte Carlo results to determine a 95% confidence interval of $[427879, 431588]$. In the right of Figure 8 we show the bootstrap density function and its comparison with a normal distribution. We remark that in all of our simulations, we never saw a fine due to more than 50 samples violating (2).

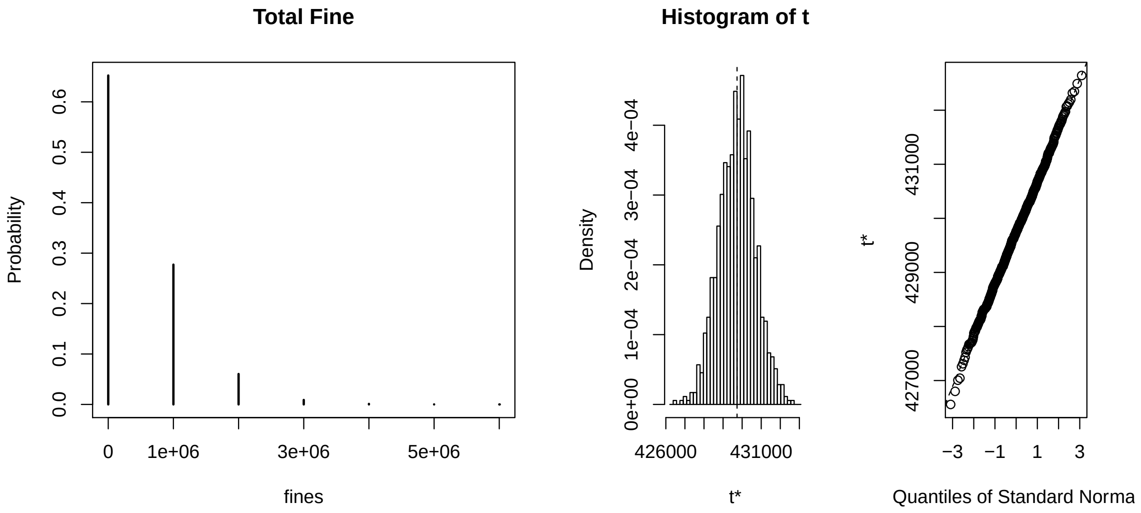

Figure 8: Left: Probability of Monte Carlo fines. Right: Bootstrap of Monte Carlo fines with t^* representing the fine: (unnormalized) density and comparison with a normal distribution.

5 Conclusion

By reverse-engineering the data provided, we determined how Ni and Cr depended on the sample location. Making simplifying assumptions on the sampling method we could then compute the expected value of the fine to be €430,000. Without making simplifying assumptions, we simulated the sampling method to obtain a Monte Carlo estimate of the expected value of the fine as €429732 with bootstrap 95% confidence interval of $[427879, 431588]$.

Using the method of determining the frequencies of cells violating the specification intervals, we noticed that only the first 200 cylinders each day ever have any defects. Therefore, we suggest that to lower the expected fine, the company should re-evaluate the method for starting the factory each day.

References

[1] Bernard Beauzamy. French Federation of Mathematical Games and Societe de Calcul Mathematique SA, 2013. Mathematical Competitive Game 2013-2014. URL: `http://scmsa.eu/archives/SCM_FFJM_Competitive_Game_2013_2014.pdf`.

[2] Martin J. Mohlenkamp. Math 4530/5530 Statistical Computing. Ohio University Department of Mathematics, 2014. URL: `http://www.ohio.edu/people/mohlenka/20142/4530-5530/`.

[3] R Core Team. *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria, 2013. URL: `http://www.R-project.org`.

[4] W. A. Stein et al. *Sage Mathematics Software*. The Sage Development Team, 2014. URL: `http://www.sagemath.org`.