Riddler March 5, 2021: Impressive baseball stats

Mark Girard

March 9, 2021

This week's riddler concerns the following interesting baseball stat: On February 23, 2021, professional baseball player Mike Trout had achieved a career batting average of .299 after having played a total of 299 games. Neat! But how remarkable is that, really?

Question 1. *Assuming that a baseball player has exactly 4 at bats in each game, what is the largest number of games for which it is* ***not*** *possible for the player to have a batting average (rounded to three digits) equal to the number of games divided by 1000?*

If a player gets a hit at every at-bat appearance in 1000 games, that player would have a batting average of 1.000. Hence this is clearly possible if the player plays 1000 games. (It is not possible to have a batting average greater than 1, so it only makes sense to consider cases where the number of games is at most 1000.) On the other hand, it is certainly impossible to have a batting average of .001 after only one game, as the only possibilities for batting averages are 0, .250, .500, .750, and 1.000 (representing, respectively, 0, 1, 2, 3, or 4 hits in the one game). Thus the answer we are looking for is somewhere between 1 and 999. Let's do some more work to find the answer.

Let n be the total number of games played and let k be the total number of hits that our player gets in those n games. (As we are assuming that the player has exactly four at bats per game, the number of hits is at most $4n$.) The player's batting average (rounded to three digits) is given by

$$\text{round}\left(\frac{k}{4n} \times 1000\right) \times \frac{1}{1000} = \frac{\text{round}\left(\frac{250k}{n}\right)}{1000}$$

where "round" is the function that rounds its input to the nearest integer (and rounds up at half-integers). Namely, for a given $x \in \mathbb{R}$, one has that

$$\text{round}(x) = \left\lfloor x + \frac{1}{2} \right\rfloor.$$

Assuming the player has k hits in n games, their batting average (rounded to three digits) is equal to the number of games divided by one thousand if and only if

$$n = \left\lfloor \frac{250k}{n} + \frac{1}{2} \right\rfloor.$$

The problem now is to determine the largest number of games for which it is *not* possible to have a batting average equal to the number of games divided by one thousand. We may therefore rephrase the problem as the following question.

Question 2. *What is the largest integer $n \in \{1, \ldots, 1000\}$ such that*

$$n = \left\lfloor \frac{500k + n}{2n} \right\rfloor \qquad (1)$$

is not satisfied for every choice of integer $k \in \{0, 1, \ldots, 4n\}$?

The equation in (1) holds if and only if there is an integer $r \in \{0, 1, \ldots, 2n-1\}$ satisfying

$$500k + n = 2n^2 + r.$$

Thus, for a given integer n, it is possible to have batting average equal to exactly $n/1000$ if and only if there exists an integer $r \in \{0, 1, \ldots, 2n-1\}$ satisfying

$$2n^2 - n + r \equiv 0 \mod 500.$$

Equivalently, this is possible if and only if $((n - 2n^2) \bmod 500) \leq 2n - 1$. As it must be the case that $(a \bmod 500) \leq 499$ for every integer a, this is clearly possible whenever $n \geq 250$. It remains to find the largest integer $n \leq 249$ satisfying

$$((n - 2n^2) \bmod 500) \geq 2n.$$

Here's some python code I used to find the answer:

```
n = 249
while (n - 2*(n**2)) % 500 < 2*n:
    n -= 1
print(n)
```

It turns out that the answer is 239.

Further investigation

Another thing we can do is to find all $n \in \{1, \ldots, 1000\}$ for which it is not possible to have a batting average equal to the number of games. It is straightforward to find these using the characterization above. The numbers for which it is not possible are displayed in the grid in Figure 1.

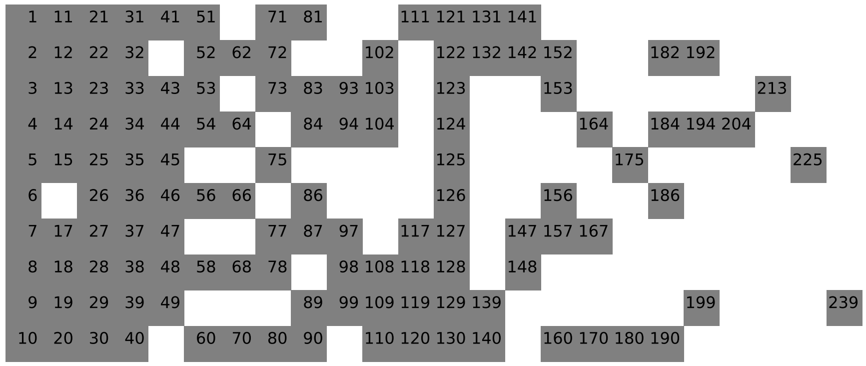

Figure 1: A grid depicting all numbers n (shaded gray) for which it is *not* possible to have a rounded batting average equal to $n/100$.

The code used to produce this grid is presented here:

```
import matplotlib.pyplot as plt
import matplotlib.patches as patches

rows = 10
cols = 25

plt.figure()
plt.xlim(0,cols)
plt.ylim(0,rows)
plt.axis('off')
ax = plt.gca()
ax.set_aspect('equal')
plt.rcParams.update({'font.size': 6})

for n in range(250):
    if (n-2*n**2)%500 > 2*n:
        a = (n - 1) //10
        b = rows - 1 - (n - 1) % 10
        patch = patches.Rectangle([a,b], 1, 1, facecolor='gray', zorder = 1)
        ax.add_patch(patch)
        plt.text(a+.9,b+.5,str(n),ha='right')

plt.savefig('grid.pdf')
```