6-1-2013

Dropping the Lowest Score: A Mathematical Analysis of a Common Grading Practice

Rosie Brown
Western Oregon University, rosio_west@hotmail.com

Dropping the Lowest Score: A Mathematical Analysis of a Common Grading Practice

By

Rosie Brown

An Honors Thesis Submitted in Partial Fulfillment of the Requirements for Graduation from the Western Oregon University Honors Program

Dr. Breeann Flesch,

Thesis Advisor

Dr. Gavin Keulks,

Honors Program Director

Western Oregon University

June 2013

Contents

1. Abstract

A common grading practice among educators has been to drop low test, quiz, or homework scores to help students in their classes. Although there exist simple cases in which it is easy to determine which scores should be kept and which scores should be dropped, there are many more complex cases that require an algorithm to solve. I will be discussing different ideas for how to easily determine the best set of scores to keep, why some methods do not work, and will go into detail about an efficient algorithm.

2. Introduction

Dropping low scores has become a rather common occurrence for many students in both high school and college. There are various ways teachers and professors can go about dropping low scores. Some teachers will drop one or more low test scores, while others will drop only low homework scores. There are various ways for teachers to go about dropping low scores, depending on how many they want to drop and the weight of each score. The main purpose of dropping low scores is to benefit the student's overall grade in the class, so a teacher should know the best method for determining which score(s) should be dropped.

3. Grade Dropping

We are going to let the number of quizzes a teacher has given be denoted k, which is greater than 0. We'll also denote the number of quiz scores to be dropped as r. On quiz j where $j = 1, 2, 3, ..., k$ a student scores m_j points out of a possible n_j points. We'll assume that the earned scores and possible scores are positive integers. Let N be an upper bound for all the n_j. We shall refer to the set of dropped scores as the *deletion set*, which has size r. Similarly, we shall refer to the set of not dropped scores as the *retained set*, which has size $k - r$. We want to find the deletion set that will result in the highest possible final grade for the student which we shall call the *optimal deletion set*.

Let's consider the example of Kyle's quiz scores:

Table 1: Kyle's Quiz Scores

Quiz	1	2	3
Score	9	12	14
Possible	10	20	16
Percentage	90	60	87.5

If the teacher was simply basing Kyle's final grade on his raw score, he would simply add all of the m_js, or $\sum_{j=1}^{k} m_j$. Then the teacher would simply drop the r smallest m_j values. If the teacher wanted to drop 1of Kyle's scores, Quiz 1 would be dropped.

This would leave Kyle with a total score of $12 + 14 = 26$. However, it is strange that we dropped Quiz 1 since it has the highest percentage of all the quizzes.

If the teacher decided to grade Kyle based on his ratio of total points earned out of total points possible, then the problem becomes more interesting. We need to find a subset $S \subset K = \{1, 2, 3, ..., k\}$ of $k - r$ retained scores so that the ratio of the raw scores with the possible scores, or $\dfrac{\sum_{j \in S} m_j}{\sum_{j \in S} n_j}$, is maximized. In Kyle's example, the optimal score to drop would be Quiz 2, leaving him with a percentage score of $(9 + 14)/(10 + 16) = 88.5\%$. Note that if all the n_j are equal, meaning all the quizzes are out of the same score, then the problem is easily reduced to finding the r lowest m_j values as we did when we were only considering the raw scores.

4. Paradoxical Behavior

Table 2: Maddy's Quiz Scores

Quiz	1	2	3	4	5
Score	21	81	28	18	7
Possible	75	100	50	60	28
Percentage	28	81	56	30	25

In the example of Maddy's quiz scores it would seem logical to drop the Quiz 5 score. It has the lowest raw score as well as the lowest percentage. If we were to drop Quiz 5, Maddy's new score is $\frac{21+81+28+18}{75+100+50+60} = 51.9\%$. But if we drop Quiz 1, her score is $\frac{81+28+18+7}{100+50+60+28} = 56.3\%$, which is in turn the optimal deletion set if we are dropping 1 score. If we drop Quiz 2 her score is 34.7%, Quiz 3 is 48%, and Quiz 4 is 54.2%. The reason this is true is because Quiz 5 has less impact on the overall score because of the low possible point value.

We should always keep the largest percentage grade. Otherwise, the average of the other kept scores will be less than that highest percentage. However, as shown by Maddy's example, we won't always drop the lowest percentage grade.[**1**]

Another possible way to find the optimal deletion set could be to pick one score at a time to drop. In other words, find the optimal score to drop, then find the next optimal score to drop from the remaining set and so on. For this, we'll consider Carl's quiz scores, and determine the optimal score to drop by using guess and check since there are so few quizzes.

Table 3: Carl's Quiz Scores

Quiz	1	2	3	4
Score	100	42	14	3
Possible	100	91	55	38
Percentage	100	46	25	8

[1]$Kane, Kane$

If we drop one grade, the optimal one would be Quiz 4, which leaves an average of $\frac{100+42+14}{100+91+55} = 63.4\%$ (as compared to dropping Quiz 1 which gives us 32.0%, Quiz 2 which gives us 60.6%, or Quiz 3 which gives us 63.3%). If we drop two grades, the optimal two would be Quiz 2 and Quiz 3, which gives us an average of $\frac{100+3}{100+38} = 74.6\%$ (as compared to to dropping 3 and 4 which gives us 74.3%, 2 and 4 which gives us 73.5%, or 1 and 4 which gives us 38.4%). Notice that this shows that the optimal deletion set for dropping two scores does not include the best single score to drop.[**1**]

Let's now consider the example of Dale's quiz scores to further explore the similarities and differences between different sizes of optimal deletion sets.

Table 4: Dale's Quiz Scores

Quiz	0	1	2	3	4	5
Score	$20+c$	$21-b_1$	$22-b_2$	$23-b_3$	$24-b_4$	$25-b_5$
Possible	40	42	44	46	48	50
Percentage	$50+\frac{c}{40}$	$50-\frac{b_1}{42}$	$50-\frac{b_2}{44}$	$50-\frac{b_3}{46}$	$50-\frac{b_4}{48}$	$50-\frac{b_5}{50}$

Quiz	6	7	8	9	10
Score	$26-b_6$	$27-b_7$	$28-b_8$	$29-b_9$	$30-b_{10}$
Possible	52	54	56	58	60
Percentage	$50-\frac{b_6}{52}$	$50-\frac{b_7}{54}$	$50-\frac{b_8}{56}$	$50-\frac{b_9}{58}$	$50-\frac{b_{10}}{60}$

$[1]Kane, Kane$

We know Quiz 0 is the highest scoring quiz since it's the only one above 50% . Thus, it will automatically be one of our retained scores because the average of any retained set not including Quiz 0 would be less that $50+\frac{c}{40}$.

With c and b_j being integers and the set A being the set of retained quiz scores excluding Quiz 0, the average score is

$$\frac{20+c+\frac{1}{2}\sum_{j\in A} n_j - \sum_{j\in A} b_j}{40+\sum_{j\in A} n_j}[\mathbf{1}] \tag{4.1}$$

We get $20+c$ in the numerator and 40 in the denominator from Quiz 0 and the two sums in the numerator from each $m_j = \frac{1}{2}n_j - b_j$.

Since $2 * [20 + \frac{1}{2} \sum_{j \in A} n_j]$ is $40 + \sum_{j \in A} n_j$ we can reduce equation (4.1) to

$$0.5 + \frac{c - \sum_{j \in A} b_j}{40 + \sum_{j \in A} n_j} [\mathbf{1}] \tag{4.2}$$

If we let $c = 4$ and all of the $b_j = 1$ and we drop 5 quiz scores, our new average becomes

$$0.5 + \frac{4 - \sum_{j \in A} b_j}{40 + \sum_{j \in A} n_j} = 0.5 + \frac{4 - 5 * 1}{40 + \sum_{j \in A} n_j} = 0.5 - \frac{1}{40 + \sum_{j \in A} n_j} [\mathbf{1}] \tag{4.3}$$

The first fraction equals the second because we let all of the $b_j = 1$, and since we are dropping 5 scores we are therefore keeping 5 scores, so $\sum_{j \in A} b_j = 5 * 1$. To maximize this score, we want the fraction on the right to be as small as possible because it is negative, meaning we want to maximize the denominator. To do this we want the largest possible n_j values. Thus, our optimal deletion set would be $[1, 2, 3, 4, 5]$.

However, if we change our c to equal 6, our average score becomes

$$0.5 + \frac{6 - \sum_{j \in A} b_j}{40 + \sum_{j \in A} n_j} = 0.5 + \frac{6 - 5 * 1}{40 + \sum_{j \in A} n_j} = 0.5 + \frac{1}{40 + \sum_{j \in A} n_j} [\mathbf{1}] \tag{4.4}$$

In this case, to maximize the score, we want the fraction on the right to be as big as possible because it is positive, meaning we want to minimize the denominator. To do this we want the smallest possible n_j values. Thus, our new optimal deletion set is $[6, 7, 8, 9, 10]$. This shows that by simply changing c, we can get a completely different deletion set with none of the same scores as the other deletion set.

What if we change the number of scores we drop, while keeping c and b_j constant.

Let $c = 22$ and all the $b_j = 3$. If we drop two quiz scores, our average becomes

$$0.5 + \frac{22 - \sum_{j \in A} b_j}{40 + \sum_{j \in A} n_j} = 0.5 + \frac{22 - 8 * 3}{40 + \sum_{j \in A} n_j} = 0.5 - \frac{2}{40 + \sum_{j \in A} n_j} \tag{4.5}$$

To maximize this score, we want the fraction on the right to be as small as possible, meaning we want to maximize the denominator. To do this we want to keep the largest possible n_j values. Thus, our optimal deletion set would be $[1, 2]$.

However, if we change the number of scores dropped from two to three, our new average becomes

$$0.5 + \frac{22 - \sum_{j \in A} b_j}{40 + \sum_{j \in A} n_j} = 0.5 + \frac{22 - 7 * 3}{40 + \sum_{j \in A} n_j} = 0.5 + \frac{1}{40 + \sum_{j \in A} n_j} \tag{4.6}$$

In this case, to maximize the score, we want the fraction on the right to be as big as possible, meaning we want to minimize the denominator. To do this we want to keep the smallest possible n_j values. Thus, our new optimal deletion set is $[8, 9, 10]$. This shows that by changing the number of quiz scores dropped by only 1, we get a completely different deletion set with none of the same scores as the other deletion set.

Let's see if we can use Dale's quiz scores to show that the optimal deletion set when we drop four scores can overlap the optimal deletion set when we drop five scores, as Kane and Kane did.

Let t be the number of scores the two deletion sets have in common. We know t can only equal $1, 2, 3$, or 4 because our smallest deletion set is size 4. Let $b_j = 3$ for all j

from 1 to t and $b_j = 2$ for all $j > t$ and let $c = 11$. If we drop four quiz scores, and s is the number of retained quiz scores with their $b_j = 3$, the average score becomes

$$0.5 + \frac{11 - \sum_{j \in A} b_j}{40 + \sum_{j \in A} n_j} = 0.5 + \frac{11 - [3 * s + 2 * (6 - s)]}{40 + \sum_{j \in A} n_j} = 0.5 - \frac{1 + s}{40 + \sum_{j \in A} n_j} \tag{4.7}$$

To maximize this score, we want the fraction on the right to be as small as possible. Thus, s needs to be as small as possible, or 0, and we want the largest possible n_j values. Thus, our optimal deletion set is $[1, 2, 3, 4]$.

If we drop five quiz scores instead of four, the average score becomes

$$0.5 + \frac{11 - \sum_{j \in A} b_j}{40 + \sum_{j \in A} n_j} = 0.5 + \frac{11 - [3 * s + 2 * (5 - s)]}{40 + \sum_{j \in A} n_j} = 0.5 + \frac{1 - s}{40 + \sum_{j \in A} n_j} \tag{4.8}$$

To maximize this score, we want our fraction on the right to be as big as possible. Thus, s needs to be 0 or the numerator would become 0 or negative. Hence we must drop all the scores with $b_j = 3$ and keep the smallest possible n_j values. Thus, our optimal deletion set is the set containing the quizzes with $b_j = 3$ and as many of the low scoring quizzes as needed. Therefore the overlap between the two deletion sets will be the set of t grades with $b_j = 3$.[**1**]

5. Algorithms for Finding the Optimal Deletion Set

Now we must return to our original question. How can we find the optimal deletion set of r grades to drop from k quiz scores. We could use brute force to simply calculate the average grade for each possible set of retained scores. Finding the average of each possible set is easy enough, but when k and r get large, it can become tediously long. The number of possible sets of scores is $\binom{k}{r}$.[**1**] Even a computer would take too long to determine the optimal set using the brute force method.

Unfortunately, we also can't use approaches that try to split our problem into smaller problems. We've shown from Carl's example that we cannot simply drop one grade at a time and it wouldn't be helpful to split the set of k quiz scores into different subsets and find the optimal scores to drop from those subsets.

6. The Optimal Drop Function

Our overall goal is to find a set of retained grades $S \subset K = \{1, 2, 3, ..., k\}$ with size $k - r$ such that our ratio

$$\frac{\sum\limits_{j \in S} m_j}{\sum\limits_{j \in S} n_j} = q \tag{6.1}$$

is maximized. For every j we define $f_j(q) = m_j - qn_j$. Then our equation 6.1 is equivalent to

$$\sum_{j \in S} f_j(q) = 0[\mathbf{1}] \tag{6.2}$$

We can get this equivalence because of the following:

$$f_j(q) = m_j - qn_j \tag{6.3}$$

$$\Rightarrow \sum_{j\in S} f_j(q) = \sum_{j\in S} m_j - \sum_{j\in S} q * n_j \tag{6.4}$$

$$\Rightarrow \sum_{j\in S} f_j(q) = \sum_{j\in S} m_j - q * \sum_{j\in S} n_j \tag{6.5}$$

$$\Rightarrow \sum_{j\in S} f_j(q) = \sum_{j\in S} m_j - \frac{\sum_{j\in S} m_j}{\sum_{j\in S} n_j} * \sum_{j\in S} n_j \tag{6.6}$$

$$\Rightarrow \sum_{j\in S} f_j(q) = \sum_{j\in S} m_j - \sum_{j\in S} m_j \tag{6.7}$$

$$\Rightarrow \sum_{j\in S} f_j(q) = 0 \tag{6.8}$$

From this equivalency, we know the left-hand side of equation 6.1 is greater than q if and only if the left-hand side of equation 6.2 is greater than 0. Notice that each $f_j(q)$ is a linear, decreasing function of q for any given set S. Thus, $\sum_{j\in S} f_j(q)$ is also a linear, decreasing function of q. For a particular set of retained grades, S, $\sum_{j\in S} f_j(q) = 0$ is satisfied by the value of q which represents the average of the quizzes in S.We will find the optimal retained set, S_{best}, when we find the set associated with the optimal average, q_{best}, which is as large as possible. We will define the *optimal drop function* to be

$$\max\{\sum_{j\in S} f_j(q) : S \subset K, |S| = k - r\} = F(q)[\mathbf{1}] \tag{6.9}$$

Since F is the maximum of some number of linear, decreasing functions, it is a piecewise, linear, decreasing, concave up function. We also know $F_{best} = 0$ since

$\sum_{j \in S} f_j(q_{best}) = 0$. If S were not S_{best}, but some other subset of K with $|S| = k - r$, then $\sum_{j \in S} f_j(q_{best}) \leq 0$.

Let's consider Carl's quiz scores from Table 3 when we dropped two of the four quiz scores. There are six possible sets of retained scores and six sums associated with those sets. We can graph them in Figure 1.

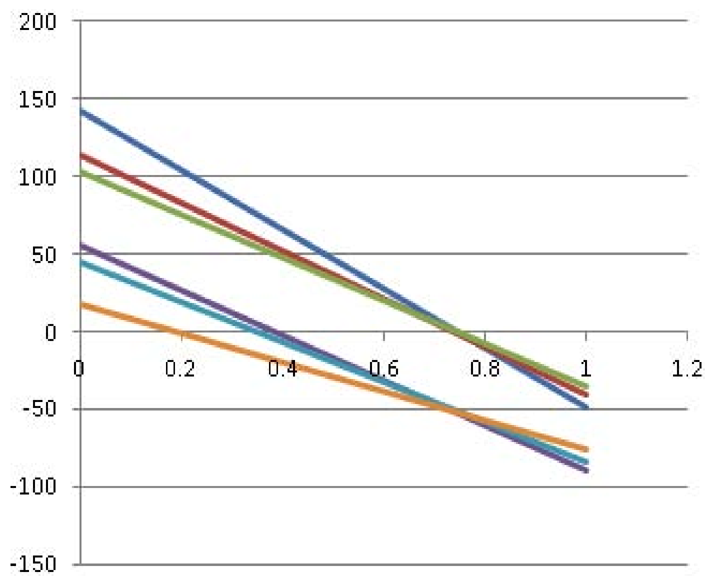

Figure 1: [1] Kane, Kane

The maximum sum, or function F, would be the following graph.

Our new goal has become finding a set of r grades to drop so that our subset S, where

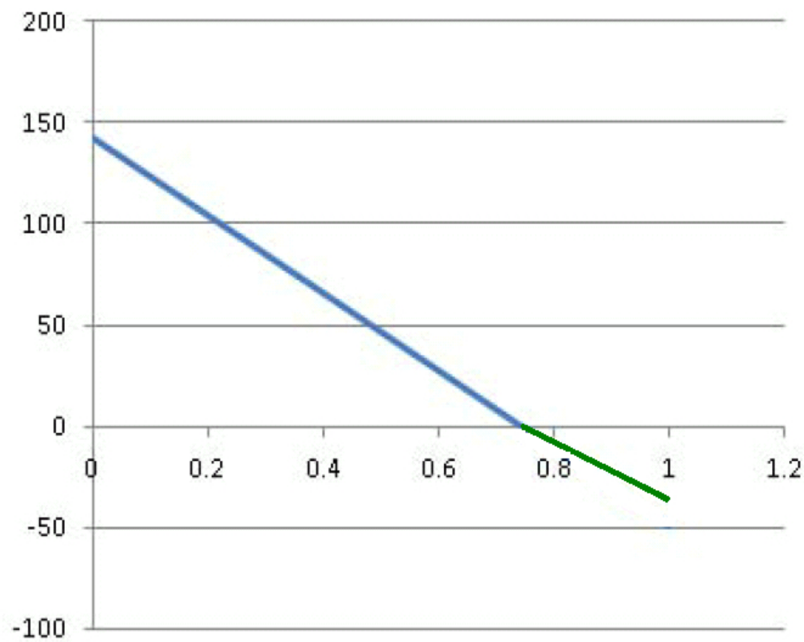

Figure 2: [1] Kane, Kane

$|S| = k - r$, and our rational number q cause $F(q) = \sum_{j \in S} f_j(q_{best}) = 0$. This makes the problem easier because we can easily evaluate $F(q)$ for some q. With a given list $m_1, m_2, ..., m_k$ of scores with their corresponding $n_1, n_2, ..., n_k$ possible scores, along with the r number of scores to be dropped and a real number q, we need only evaluate each $f_j(q) = m_j - qn_j$ for every $j = 1, 2, ..., k$. We then find the $k - r$ largest values of the f_j values, which gives us our set S_{best} for the correct j values. $F(q)$ can then be calculated from $\sum_{j \in S} f_j(q_{best})$. $F(q)$ can be calculated using any number of algorithms for identifying the largest values from a set of numbers.[**1**]

Now, we only need to find the value for q so that $F(q) = 0$. Since we know $\sum_{j \in S} f_j(q)$

is linear, F can change slope at some q where the associated set S changes at that q. For each q consider the collection of $f_j(q)$ values for $j = 1, 2, ..., k$. We can place these values in decreasing order. The order of f_j values will change as q changes. Set S depends on this order, so if it changes, S changes with it. Because each f_j is continuous, the order of $f_j(q)$ and some $f_i(q)$ where $1 \leq i \leq k$ with $i \neq j$ can only change for values of q where $f_j(q) = f_i(q)$, otherwise the switch would cause f_j to not always be continuous. Because f_j is linear, this can only occur once for every pair of j and i. So S cannot change more than $\binom{k}{2}$ values of q since that is the number of pairs of j and i.[**1**]

When $f_j(q) = f_i(q)$, we know $m_j - qn_j = m_i - qn_i$. Thus

$$\frac{m_i - m_j}{n_i - n_j} = q. \tag{6.10}$$

Therefore, if the F graph changes slope at some rational number q, the denominator must be bounded by N, the upper bound of all the n_j. This is because we know $n_j \leq N$, thus $n_j - n_i \leq N - n_i \leq N$. Since

$$\frac{\sum_{j \in S_{best}} m_j}{\sum_{j \in S_{best}} n_j} = q_{best} \tag{6.11}$$

then the denominator for q_{best} is no larger than $(k - r)N$ because $n_j \leq N$ which implies $\sum_{j \in S_{best}} n_j \leq \sum_{j \in S_{best}} N = (k - r)N$ since $|S| = k - r$.

We can use this information to help us find S_{best} and q_{best}. We could find all the values of q where $f_j(q) = f_i(q)$ for some values of i and j. We can then evaluate $F(q)$ at these points and construct the graph for F since F is linear, remembering that we only need two points draw a line. From there we can easily determine where $F(q) = 0$. But we can still find more efficient methods to figure out where $F(q) = 0$.

7. The Bisection Algorithm

We can use a *bisection method* to approximate the q value where $F(q) = 0$. We know q_{best} lies in the interval between the minimum and maximum values of m_j/n_j. From here we will set

$$q_{high} = max\{\frac{m_j}{n_j}\}, q_{low} = min\{\frac{m_j}{n_j}\}, q_{middle} = \frac{q_{high} + q_{low}}{2} \tag{7.1}$$

We then calculate $F(q_{middle})$ and the associated S set. If $F(q_{middle}) < 0$, we know q_{best} is between q_{middle} and q_{high}, so we reset q_{low} to q_{middle}. If $F(q_{middle}) > 0$, know q_{best} is between q_{low} and q_{middle}, so we reset q_{high} to q_{middle}. Afterwards we reset q_{middle} using the new q_{best} or q_{low} value.[**2**] We do this repeatedly until

$$q_{high} - q_{low} < \frac{1}{2(k-r)N^2} \tag{7.2}$$

At this point we know S is S_{best} because the distance between q_{high} and q_{low} is as small as possible. We can then calculate q_{best} using S_{best}.[**1**]

But are we sure that this final S is S_{best}. We'll consider the function F. We know F is piecewise linear, decreasing, and concave up. If F is linear around q_{best}, then the distance between q_{best} and the next q where F changes slope is the distance between a rational number with denominator at most N, which is how described our q_{best}, and a rational number with denominator at most $(k-r)N$, how we described the other qs. This leaves us with a number that is at least $1/[(k-r)N^2]$ by finding the common denominator. Because of this, our approximation of q_{best} must be closer to the actual value of q_{best} than to the closest q where F changes slope. So we know our S is S_{best}. If F were to change slope at q_{best}, then our q_{best} would be associated with two sets of grades that would be both equally good to drop.[**1**]

8. A More Efficient Algorithm

If we consider the geometry of our F graph we can improve our bisection algorithm. We know that F is a piecewise function. If we consider a value $q_1 < q_{best}$ then $F(q_1)$ is calculated and we are given the associated set S_1. We'll consider the piece of the F graph that passes through the point $(q_1, F(q_1))$. We'll let q_2 be the point where this piece crosses the x-axis. q_2 is the average of the grades in our set S_1. We know that since F is concave up, q_2 is between q_1 and q_{best}. We can continue this process until we reach q_{best} after finitely many steps. At this point, $F(q) = 0$. If we picked a $q_1 > q_{best}$ then our q_2 would be less than q_{best}.[**1**]

We don't know for sure if this method is any faster than the bisection method, but in most cases it tends to converge rather quickly. $Kane$ and $Kane$ used case of dropping 300 scores from $1,000$ and only needed up to five iterations of this process to determine the optimal deletion set.

9. Recap and Other Opportunities

We have been able to now find a quick way to determine the best set of grades to drop from a set of scores. We could expand on this idea by discussing how our algorithm would change if we decided to drop high scores as well as low scores. We could also discuss whether dropping low scores is beneficial to students, or if it causes them to not try as hard. Regardless of the overall benefit, it is important for teachers to make sure they are dropping the best set of scores so they can give their students the best grade possible.

REFERENCES

1. D. Kane, J. Kane, Dropping Lowest Grades, *Mathematics Magazine* **79**(3)(2006)181-189.
2. R.L. Burder, J.D. Faires, 2.1 The Bisection Algorithm, *Numerical Analysis* (3rd Ed.).