How to Divide Students into Groups so as to Optimize Learning: Towards a Solution to a Pedagogy-Related Optimization Problem

Olga Kosheleva\textsuperscript{1} and Vladik Kreinovich\textsuperscript{2}

Departments of \textsuperscript{1}Teacher Education and \textsuperscript{2}Computer Science
University of Texas at El Paso, El Paso, TX 79968, USA
olgak@utep.edu, vladik@utep.edu
1. Formulation of the Problem

- Students benefit from feedback.
- In large classes, instructor feedback is limited.
- It is desirable to supplement it with feedback from other students.
- For that, we divide students into small groups.
- The efficiency of the result depends on how we divide students into groups.
- If we simply allow students to group themselves together, often, weak students team together.
- Weak students are equally lost, so having them solve a problem together does not help.
- It is desirable to find the optimal way to divide students into groups. This is the problem that we study.
2. Need for an Approximate Description

- A realistic description of student interaction requires a multi-D learning profile of each student:
  - how much the students knows of each part of the material,
  - what is the student’s learning style, etc.

- Such a description is difficult to formulate and even more difficult to optimize.

- Because of this difficulty, in this paper, we consider a simplified description of student interaction.

- Already for this simplified description, the corresponding optimization problem is non-trivial.

- However, we succeed in solving it under reasonable assumptions.
3. How to Describe the Current State of Learning

- We assume that a student’s degree of knowledge can be described by a single number.
- Let $d_i$ be the degree of knowledge of the $i$-th student $S_i$.
- We consider subdivisions into groups $G_k$ of equal size.
- If two students with degrees $d_i < d_j$ work together, then the knowledge of the $i$-th student increases.
- The more $S_j$ knows that $S_i$ doesn’t, the more $S_i$ learns.
- In the linear approximation, the increase in $S_i$’s knowledge is thus proportional to $d_j - d_i$:
  \[ d'_i = d_i + \alpha \cdot (d_j - d_i). \]
- In a group, each student learns from all the students with higher degree of knowledge:
  \[ d'_i = d_i + \alpha \cdot \sum_{j \in G_k, d_j > d_i} (d_j - d_i). \]
4. Discussion: Group Subdivision Should Be Dynamic

- Students’ knowledge changes with time.
- As a result, optimal groupings change.
- So, we should continuously monitor the students’ knowledge and correspondingly re-arrange groups.
- Ideally, we should also take into account that there is a cost of group-changing:
  - before the student start gaining from mutual feedback,
  - they spend some effort adjusting to their new groups.
5. Possible Objective Functions

- First, we will consider the average grade $a \overset{\text{def}}{=} \frac{1}{n} \cdot \sum_{i=1}^{n} d_i$.

- Another reasonable criterion is minimizing the number of failed students.

- In this case, most attention is paid to students at the largest risk of failing, i.e., with the smallest $d_i$.

- From this viewpoint, we should maximize the worst grade $w \overset{\text{def}}{=} \min_{i=1,\ldots,n} d_i$.

- Many high schools brag about the number of their graduates who get into Ivy League colleges.

- From this viewpoint, most attention is paid to the best students, so we should maximize the best grade $b \overset{\text{def}}{=} \max_{i=1,\ldots,n} d_i$. 
6. Optimal Division into Pairs: Our Theorems

- To maximize the average grade $a$:
  - we sort the students by their knowledge, so that $d_1 \leq d_2 \leq \ldots \leq d_n$,
  - in each pair, we match one student from the lower half with one student from the upper half.

- To maximize the worst grade $w$:
  - we sort the students by their knowledge;
  - we pair the worst-performing student (corr. to $d_1$) with the best-performing student (corr. to $d_n$);
  - if there are other students with $d_i = d_1$, we match them with $d_{n-1}$, $d_{n-2}$, etc.;
  - other students can be paired arbitrarily.

- In this model, subdivision does not change the best grade $b$ (this is true for groups of all sizes $g$.)
7. Optimal Division into Groups of Given Size $g$

- To maximize the average grade $a$, we:
  - sort the students by their knowledge, and, based on this sorting, divide the students into $g$ sets:
    \[
    L_0 = \{d_1, d_2, \ldots, d_{n/g}\}, \ldots, L_{g-1} = \{d_{(g-1)(n/g)+1}, \ldots, d_n\};
    \]
  - in each group, we pick one student from each of $g$ sets $L_0, L_1, \ldots, L_{g-1}$.

- If there is only one worst-performing student, then, to maximize the worst grade $w$, we:
  - sort the students by their knowledge $d_1 \leq d_2 \leq \ldots$;
  - combine the worst-performing student (corr. to $d_1$) with best ones (corr. to $d_n, \ldots, d_{n-(g-2)}$);
  - group other students arbitrarily.

- If we have $s$ equally low-performing students $d_1 = d_2 = \ldots = d_s$, we match each with high performers.
8. Combined Optimality Criteria

- If we have several optimal group subdivisions, we can use this non-uniqueness to optimize another criterion.

- **Example:**
  - first, we optimize the average grade;
  - among all optimal subdivisions, we select the ones with the largest worst grade;
  - if there are still several subdivisions, we select the ones with the largest second worst grade, etc.
  - etc.

- Optimal subdivision into pairs:
  - sort the students by their knowledge, $d_1 \leq d_2 \leq \ldots$
  - match $d_1$ with $d_n$, $d_2$ with $d_{n-1}$, \ldots, $d_k$ with $d_{n+1-k}$, \ldots
9. Combined Optimality Criteria (cont-d)

- **Optimality criterion** (reminder):
  - first, we optimize the average grade;
  - among all optimal subdivisions, we select the ones with the largest worst grade;
  - if there are still several subdivisions, we select the ones with the largest second worst grade, etc.
  - etc.

- Optimal subdivision into groups of size $g$:
  - sort the students by their knowledge, and divide into $g$ sets $L_0, \ldots, L_{g-1}$;
  - match the smallest value $d_1 \in L_0$ with the largest values from each set $L_1, \ldots, L_{g-1}$,
  - match the second smallest value $d_2 \in L_0$ with the second largest values from $L_1, \ldots, L_{g-1}$, etc.
10. A More Nuanced Model

- In the above analysis, we assumed that only the weaker student benefits from the groupwork.
- In reality, stronger students benefit too:
  - when they explain the material to the weaker students,
  - they reinforce their knowledge, and
  - they may see the gaps in their knowledge that they did not see earlier.
- The larger the diff. $d_j - d_i$, the more the stronger student needs to explain and thus, the more s/he benefits.
- It is therefore reasonable to assume that the resulting increase in knowledge is also proportional to $d_j - d_i$:

$$d'_i = d_i + \alpha \cdot \sum_{j \in G_k, d_j > d_i} (d_j - d_i) + \beta \cdot \sum_{j \in G_k, d_i > d_j} (d_i - d_j).$$
11. Optimal Division into Groups: Case of a More Nuanced Model

- If we maximize the average grade or the worst grade, then the optimal subdivisions are exactly the same.
- Similarly, if we use the combined criterion, we get the exact same optimal subdivision.
- For pairs, the subdivision that optimizes the best grade is the same as for the worst grade.
- For $g > 2$, to optimize the best grade, we:
  - sort the students by their knowledge, $d_1 \leq d_2 \leq \ldots$;
  - group the best-performing student (corr. to $d_n$) with $g - 1$ worst ones (corr. to $d_1, d_2, \ldots, d_{g-1}$);
  - group other students arbitrarily.
12. Case of Uncertainty

- In practice, we rarely know the exact values of $d_i$.
- We only know approximately values $\tilde{d}_i$.
- We often also know the accuracy $\Delta$ of these estimates, i.e., we know that $d_i \in [\tilde{d}_i - \Delta, \tilde{d}_i + \Delta]$.
- In this case, we do not know the exact gain.
- So it is reasonable to select a “maximin” subdivision, i.e., a subdivision for which:
  - the guaranteed (= worst-case) gain
  - is the largest.
- One can prove that:
  - the subdivisions obtained by applying the above algorithms to the approximate value $\tilde{d}_i$
  - are optimal in this minimax sense as well.
13. Acknowledgment

- This work was supported in part:
  - by the National Science Foundation grants HRD-0734825 and DUE-0926721,
  - by Grant 1 T36 GM078000-01 from the National Institutes of Health, and
  - by a grant from the Office of Naval Research.

- The authors are thankful to the anonymous referees for valuable suggestions.
14. Proof of the Result About Average Grade

• Maximizing the average grade is equivalent to maximizing the sum $n \cdot a = \sum_{i=1}^{n} g_i'$ of the new grades.

• This is, in turn, equivalent to maximizing the overall gain $\sum_{i=1}^{n} g_i' - \sum_{i=1}^{n} g_i = \sum_{i=1}^{n} (g_i' - g_i)$.

• Let us take the optimal subdivision, and show that it has the form described in our algorithm.

• Indeed, in each pair, with degrees $d_i \leq d_j$, we have a weaker student $i$ and a stronger student $j$.

• Let us prove that in the optimal subdivision, each stronger student is stronger than each weaker student.

• In other words, if we have two pairs $d_i \leq d_j$ and $d_i' \leq d_j'$, then $d_i \leq d_j'$.

• We will prove this by contradiction.
15. Proof (by Contradiction) that $d_i \leq d_{j'}$

- Let us assume that $d_i > d_{j'}$.
- Let us then swap the $i$-th and the $j'$-th students, i.e., replace the pairs $(i, j), (i', j')$ with $(i, j')$ and $(i', j)$.
- The corresponding two terms in the overall gain are changed from $\alpha \cdot (d_j + d_{j'} - d_i - d_{i'})$ to $\alpha \cdot (d_j - d_{j'} + d_i - d_{i'})$.
- The difference between the two expressions is equal to $2\alpha \cdot (d_i - d_{j'})$.
- Since $d_i > d_{j'}$, the overall gain increases.
- This contradicts the fact that we selected the subdivision with the largest gain.
- This contradiction shows that our assumption $d_i > d_{j'}$ is wrong, and thus, $d_i \leq d_{j'}$. 
16. Proof (cont-d)

• Since every weaker-of-pair student is weaker than every stronger-of-pair student:
  – all weaker-of-pair students form the bottom of the ordering of the degrees $d_i$, while
  – all the stronger-of-pair students form the top of this ordering.

• This is exactly what we have in our algorithm.

• To complete the proof, we need to prove that every such subdivision leads to the optimal average grade.

• One can check that for each such subdivision, the overall gain is equal to $\sum_{i \in L_1} d_i - \sum_{j \in L_0} d_j$, where:
  – $L_1$ is the set of all the indices $i$ from the upper half;
  – $L_0$ is the set of all the indices from the lower half.
17. Proof: Final Part

- For each subdivision from the algorithm, the overall gain is equal to \(\sum_{i \in L_1} d_i - \sum_{j \in L_0} d_j\), where:
  - \(L_1\) is the set of all the indices \(i\) from the upper half;
  - \(L_0\) is the set of all the indices from the lower half.

- Thus, the overall gain for all such subdivisions is the same.

- So, this gain is equal to the gain of the optimal subdivision.

- Hence, all such subdivisions are indeed optimal.

- The result is proven.