Supplementary materials to TITLE

Martin Papenberg

Design

Motivating Example

- Explain the application and why the application / batch assignment is needed
- Explain "real" data set and synthetic data set
    - which variables are included (why is it relevant to balance them?)
    - (Why we do not share the original data)

The synthetic data set that we generated to resemble our actual application consisted of 370 samples from 191 unique patients. Samples belonging to the same patient were required to be assigned to the same batch. Most patients ($n = 100$) were included with 1 sample. The remaining 270 patients provided more than one sample: 47 patients provided 2 samples, 13 patients provided 3 samples, 24 patients provided 4 samples, 3 patients provided 5 samples, 3 patients provided 6 samples, and 1 patient provided 8 samples. We required the samples to be assigned to 16 equal-sized batches. As a perfect split was not possible using this constellation, we created 2 batches containing 24 samples and 14 batches containing 23 samples. We strived for balance among the batches with regard to two numeric variables (age, BMI) and several categorical variables (such as ethniciy, desease stage, and cycle phase). Table 1 illustrates the results of three batch assignments using three of the variables (age, BMI, ethnicity) for illustrative purposes. The anticlustering assignments were implemented using the `anticlust` R package, and the code and data to reproduce the assignment can be retrieved via XXX (**TODO**). The first assignment is based on "standard" anticlustering, which ignores the must-link constraints and assumes that all samples can be assigned independently, with the objective of maximum similarity among batches. This unrestricted anticlustering led to the most pronounced balance among the 16 batches: Average age ranged between 30 and 30.17 years, average BMI ranged between 29.38 and 29.43, and the percentage of samples belonging to hispanic patients ranged between 50 and 54%. The second assignment implemented the must-link restrictions, ensuring that samples belonging to the same patient were assigned to the same batch, while still optimizing balance via anticlustering. The constrained assignment reduced the overall similarity among batches slightly, but the variables were arguably still rather well balanced: Among the 16 batches, average age ranged between 29.91 and 30.35 years, average BMI ranged between 29.3 and 29.57, and the percentage of samples belonging to hispanic patients ranged between 50 and 57%. The third assignment used completely random allocation of samples to batches. Random assignment is often considered as the intuitive and simple method for allocation when no other method is available. However, it does not necessarily lead to similarity among batches (Yan et al. 2012; Papenberg and Klau 2021). In this application, random assignment by far led to the worst balance: Average age ranged between 28.04 and 31.96 years, average BMI ranged between 28.04 and 31.43, and the percentage of samples belonging to hispanic patients ranged between 30 and 65%.

In the next section, we will explain the methodology of anticlustering that led to the batch assignments illustrated in Table 1. We will reiterate the general anticlustering methodology first, before outlining how we included must-link constraints with anticlustering. The implementation of must-link constraints with anticlustering is a novel contribution of the current paper.

Anticlustering

Anticlustering is an optimization method that is characterized by (a) an objective function that quantifies the balance among batches, and (b) an algorithm that conducts the batch assignment in such a way that balance

Table 1: Balance among 16 batches after implementing three different assignment procedures

Batch	Unconstrained Anticlustering			Constrained Anticlustering			Random Assignment		
	Mean Age	Mean BMI	% Hispanic	Mean Age	Mean BMI	% Hispanic	Mean Age	Mean BMI	% Hispanic
1	30.00	29.42	54	30.17	29.38	50	30.62	30.17	42
2	30.08	29.38	50	30.12	29.38	50	28.04	29.29	62
3	30.17	29.43	52	30.13	29.39	52	29.43	29.04	52
4	30.13	29.39	52	30.09	29.39	52	30.17	29.00	57
5	30.13	29.43	52	30.00	29.43	52	31.78	29.78	52
6	30.13	29.39	52	30.35	29.52	52	30.00	29.61	39
7	30.13	29.39	52	30.09	29.43	52	28.78	30.04	61
8	30.13	29.43	52	30.17	29.30	52	29.43	29.61	57
9	30.17	29.43	52	30.00	29.52	52	29.74	28.13	65
10	30.13	29.43	52	30.04	29.43	52	31.65	28.26	35
11	30.13	29.43	52	30.30	29.30	52	31.52	30.48	57
12	30.17	29.43	52	30.22	29.57	57	29.83	28.39	30
13	30.13	29.39	52	30.35	29.43	52	28.87	28.04	57
14	30.13	29.43	52	30.09	29.43	52	28.52	30.13	65
15	30.17	29.43	52	29.91	29.39	52	31.83	29.26	52
16	30.17	29.43	52	30.09	29.39	52	31.96	31.43	52

among batches is maximized. Anticlustering owes its name to the fact that the objective functions it uses are the reversal of criteria used in cluster analysis. For example, Späth (1986) already recognized that by maximizing instead of minimizing the k-means criterion (the "variance"), he was able to create groups that are similar to each other, and presented it as an improvement over the more intuitive random assignment. Brusco, Cradit, and Steinley (2020) recognized that other objective functions known from cluster analysis can also be implemented in the context of anticlustering. Anticlustering has uses in many research fields including psychology (Brusco, Cradit, and Steinley 2020; Schaper, Kuhlmann, and Bayen 2023), education (Krauss, Lee, and Newman 2013; Baker and Powell 2002), artificial intelligence (Steghöfer et al. 2013), machine learning (Mauri, Apolloni, and Damiani 2023; Rahu, Kull, and Kruve 2024), network systems (Mohebi, Zanella, and Zorzi 2022), and operations research (Gallego et al. 2013; Gliesch and Ritt 2021). However, to the best of our knowledge, anticlustering has not previously been applied for batch assignment in high throughput sequencing.

In our application, we optimized the *diversity* objective to maximize similarity among batches. While the diversity is technically a measure of within-batch heterogeneity, its maximization simultaneously leads to minimal difference between the distribution of the input variables among batches (Feo and Khellaf 1990; cf. Papenberg 2024). Papenberg and Klau (2021) referred to the maximization of the diversity as "anticluster editing" because the minimization of the diversity is also well-known from the area of cluster analysis—under the term "cluster editing" (Shamir, Sharan, and Tsur 2004; Böcker, Briesemeister, and Klau 2011). The diversity is computed on the basis of a measure of pairwise dissimilarity among samples. In particular, it is defined as the overall sum of all dissimilarities among samples that are assigned to the same batch (Brusco, Cradit, and Steinley 2020). Hence, the diversity is not directly computed on the basis of the features of our samples, but instead it relies on a deduced distance measure. In the context of anticlustering, the Euclidean distance is the most common measure that translates features to dissimilarities (Gallego et al. 2013; Papenberg and Klau 2021). However, using other distance measures such as the squared Euclidean distance is also possible (Brusco, Cradit, and Steinley 2020). The Euclidean distance is defined as

$$d(x, y) = \sqrt{\sum_{i=1}^{M}(x_i - y_i)^2}$$

where M is the number of numeric features describing two samples $x = (x_1, \ldots, x_M)$ and $y = (y_1, \ldots, y_M)$. When samples are described by two features, the Euclidean distance corresponds to the geometric, "straightline"

distance between two points in a two-dimensional space; more similar data points are closer to each other. For categorical variables, we use binary coding before including them in the computation of the Euclidean distance (see Table 2 for an example). Figure 1 illustrates the computation of the Euclidean distance and the computation diversity for the numeric features BMI and age that were used in our motivating application.

An anticlustering algorithm assigns samples to batches in such a way that the objective function—here, the diversity—is maximized. Anticlustering usually employs heuristic optimization algorithms (Yang et al. 2022). While , heuristics generally provide satisfying results in the context of anticlustering (Papenberg and Klau 2021), they do not guarantee to find the globally best assignment among all possibilities. In principle, enumerating all possible assignments is a valid strategy to obtain an optimal assignment. However, this approach quickly becomes impossible with increasing N due to an exponential growth of the way in which assignments can be conducted (Papenberg and Klau 2021). Moreover, because anticlustering problems (with the exception of some special cases) are also NP-hard (Feo and Khellaf 1990), there is probably no algorithm that identifies the globally best assignment without considering all possibilities (at least in the worst case). In practice, heuristics are therefore indispensable. However, for smaller data sets, optimal approaches can be feasible. In the appendix, we discuss an optimal algorithm for anticlustering that we extended towards the inclusion of must-link restrictions. Its applicability is limited to about 40-50 samples. In the following, we present an anticlustering heuristic that is generally applicable to large data sets, and outline how we included must-link constraints with this algorithm.

Table 2: Illustrates the recoding of the categorical variable Race using four binary variables.

Race	Black	Native American	Pacific Islander	White
Black	1	0	0	0
Pacific Islander	0	0	1	0
Native American	0	1	0	0
White	0	0	0	1
Asian	0	0	0	0

We use an exchange algorithm to maximize the diversity (Späth 1986; Papenberg and Klau 2021; Weitz and Lakshminarayanan 1998.). It consists of two steps: an initialization step and an optimization step. As initialization, it randomly assigns samples to K equal-sized batches. In principle, unequal-sized batches would also be possible, but equal-sized batches were required in the current application. After initialization, the algorithm selects the first sample and checks how the diversity would change if the sample were swapped with each sample that is currently assigned to a different batch. After simulating each exchange—that is $N - \frac{N}{K}$ exchanges—it realizes the one exchange that increases the diversity the most. It does not conduct an exchange if no improvement in diversity is possible. This procedure is repeated for each sample, leading to the evaluation of $N \cdot \frac{N}{K}$ exchanges in total; it terminates after the last sample was processed. The procedure might also restart at the first element and reiterate through all samples until no exchange leads to an improvement any more, i.e., until a local maximum is found. In `anticlust`, we also implemented this local maximum search, which corresponds to the algorithm LCW by Weitz and Lakshminarayanan (1998). For better results, it is also possible to restart the search algorithm multiple times using different (random) initializations (Späth 1986). For the anticlustering assignment illustrated in Table 1, we used 10 repetitions of the local maximum search LCW, with the squared Euclidean distance as measure of pairwise dissimilarity. To ensure comparable weight of the three features, we also applied a standardization of the input variables before the distance was computed, which is recommended if the variables differ strongly in their ranges (Papenberg 2024).

Including Must-Link Constraints with Anticlustering

In our application, samples belonging to the same patient were required to be assigned to the same batch. We refer to a set of samples that must be assigned to same batch as a must-link *clique*. In our application, not all samples were part of a clique. For the synthetic data set that resembled our actual application, we simulated
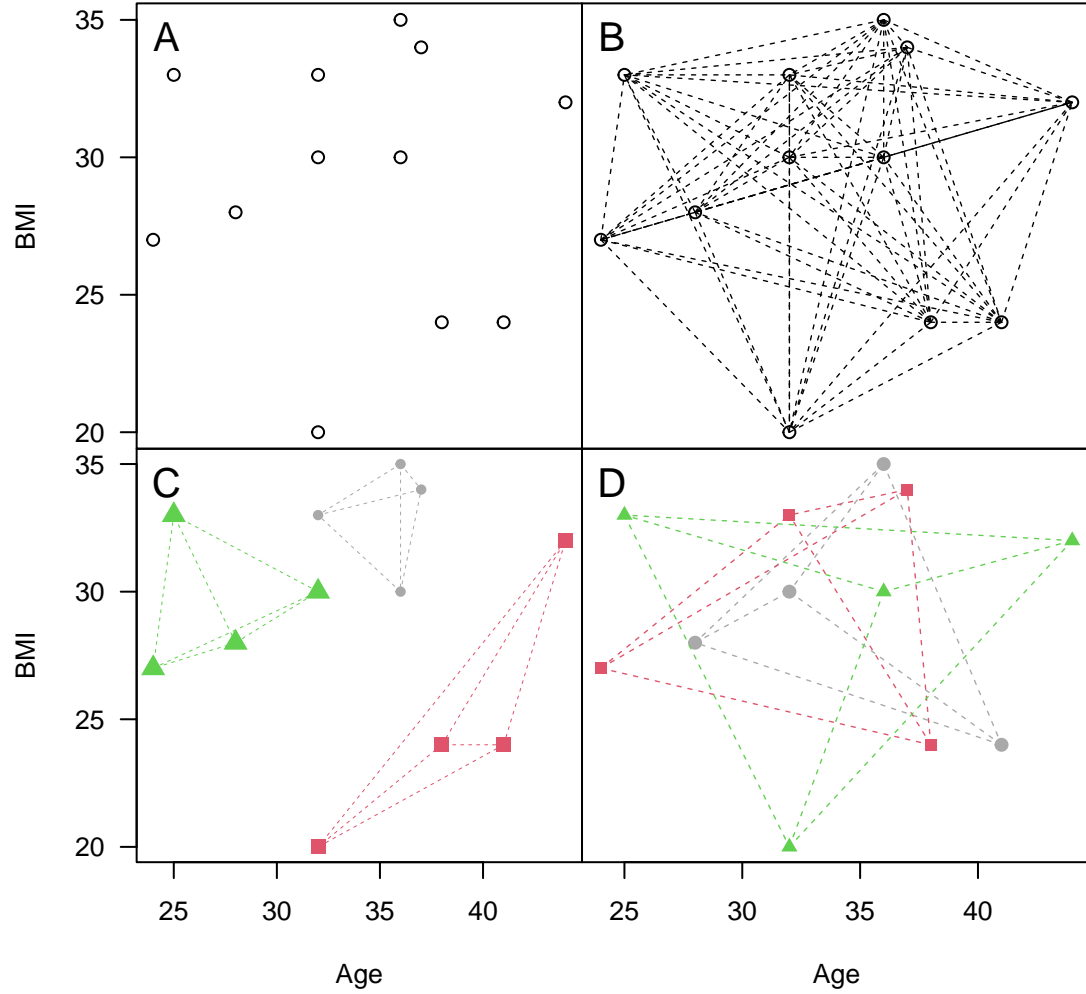
Figure 1: Illustrates the conversion from numeric features to Euclidean distance, and (anti)clustering assignments based on minimum and maximum diversity using the Euclidean distance. Panel A illustrates the BMI and age of twelve women in our synthetic data in a scatter plot. Panel B represents the Euclidean distances between features as a straight line in the two-dimensional space. The Euclidean distance is proportional to the length of the connecting lines in panel B. Panel C illustrates a clustering assignment of the 12 data points to $K = 3$ equal-sized groups via *minimum* diversity. Panel D illustrates an anticlustering assignment of the 12 data points to $K = 3$ equal-sized groups via *maximum* diversity. The diversity is computed as the sum of within-(anti)cluster distances, which are highlighted in Panel C and Panel D through connecting lines. Maximizing the diversity simultaneously leads to similar distribution of the input features among batches.

370 samples from 191 unique patients. To include the must-link constraints with anticlustering, we basically use a downscaled data set where each unique patient—but not each single sample—constitutes a unit in the anticlustering process. Hence, in our application, the effective sample size was 191 instead of 370. Some adjustments of the exchange method are required to ensure (a) we still obtain a valid partitioning regarding the size constraints (equal-sized batches) and (b) the diversity is computed correctly during optimization. We therefore had to adjust both the initialization phase as well as the optimization phase of the exchange algorithm.

During initialization, we first assign all samples to a batch that are part of a clique (270 samples). Each clique must be assigned completely to one of the K batches and samples within a clique must not be split apart. At the same time, the maximum capacity of each batch must not be exceeded. Using this conceptualization, the initialization step corresponds to a bin packing problem, which is one the classical NP complete problem in computer science (Garey and Johnson 1979). That is, we assign a weight to each clique, corresponding to the number of samples it contains. When filling batches, the sum of the weights of the cliques in each batch must not exceed its capacity. Many optimal and heuristic algorithms have been developed to address such a bin packing problem. As the default method, we use a randomized fit heuristic to fill batches. For each must-link clique, we iterate through the K batches in random order and assign it to the first batch where it fits. The process is expected to evenly distribute the must-link cliques among batches, and the random component is particularly useful if we use multiple restarts for the exchange algorithm. After assigning the must-link cliques to batches, the remaining samples can be assigned randomly to fill the remaining space. Note that our randomized fit algorithm is a heuristic that may not find an assignment of must-link groups to batches even if one is theoretically available. If the heuristic indicates that the batches cannot hold the must-link groups, we therefore use an optimal algorithm based on integer linear programming as fallback option, which allows us to verify if the constraints really cannot be fulfilled (Martello and Toth 1990). We include the details of this model in the supplementary materials.

The optimization phase of the exchange algorithm also uses a downscaled data set where each patient rather than each sample corresponds corresponds to a unit of analysis. To obtain a reduced distance matrix that preserves all information of the original distance matrix, we sum up all pairwise distances between samples in different cliques (Böcker, Briesemeister, and Klau 2011). In the context of maximizing the diversity, this transformation sufficiently preserves the relevant information in the original distance matrix. Using the initial assignment and the reduced distance matrix, we apply the same exchange algorithm as we described for the unrestricted anticlustering application. However, during the exchange process, we only exchange cliques of the same size (e.g., patients providing the same number of samples) to ensure that the cardinality constraints are respected throughout (i.e., equal-sized groups).

Evaluation

We evaluated the usefulness of anticlustering by comparing it to two alternative approaches for batch assignment: (1) the OSAT method by Yan et al. (2012), which already has been used frequently for the purpose of batch assignment, and (2) propensity score batch assignment (PSBA) by Carry et al. (2023), which is a more recent method that has not yet been applied frequently. The code and data for all evaluations is openly retrievable from XXX (**TODO**). As first evaluation, we seize up on the example applications that were provided by Yan et al. (2012) and Carry et al. (2023) to illustrate the usage of OSAT and PSBA, respectively. In each case, we illustrate how anticlustering performs on the example data sets provided by the authors of the alternative methods. It should be noted that neither package implements the must-link restrictions that we provide, so this evaluation is limited to the case of unrestricted assignment. As a second evaluation, we conducted a large scale simulation study that compared anticlustering with OSAT and PSBA using 10,000 synthetic data sets. As a secondary contribution of the simulation study, we investigated whether the quality of batch assignment is reduced when must-link constraints are included with anticlustering, as compared to an unconstrained anticlustering assignment.

Implementation of methods

To obtain the most fair comparison to the alternative approaches, anticlustering was implemented using the default settings of the `anticlust` package (package version, Papenberg and Klau 2021): We optimized the diversity objective using the (non-local maximum) exchange method without additional repetitions, using the Euclidean distance as measure of dissimilarity. We used the OSAT implementation that is freely available as an R package from Bioconductor [version; citation]. We used the default OSAT algorithm optimal shuffle. **TODO say something about the algorithm**. We increased the number of random initializations to 5000, which is recommended in the vignette that accompanies the OSAT package. Carry et al. (2023) provided an R implementation of PSBA that is available from an accompanying online repository.[1] In the original publication, the authors presented a version of PSBA that investigates all possible batch assignments and selects the one that minimizes discrepancy in propensity scores among batches. However, investigating all possible assignment is only feasible for small data sets and therefore not generally applicable. In the online repository, the authors therefore included an implementation that randomly generates a user defined number of batch assignments, and selects the best one among them. For comparability with OSAT, we set the number of random assignments to 5000. When generating the assignments, PSBA allows to treat one variable a stratification variable; if requested by the user, PSBA only generates assignments that have a perfect split according to the stratification variable. In the simulation, we did not declare a variable as stratification variable, because there is no obvious prioritization for any one variable in synthetic data sets. Moreover, the PSBA implementation provided by the authors only allows stratification variables that contain two levels, which was too limiting for the settings of our simulation.

Example applications

Alongside the OSAT package, the authors included a vignette that illustrates the basic usage of the OSAT method on an example data set. The data set includes features of 576 persons who provided gene expression samples. In their application, the samples had to be assigned to 6 equal-sized batches, while they attempted to balance age, race and sample type (case vs. control). As the OSAT method does not deal with numeric variables, age was treated as a categorical variable with the levels (0,30], (30,40], (40,50], (50,60], and (60,100]. For their evaluation, the authors conducted the OSAT assignment and then computed a χ^2-test that quantifies imbalance among batches for each of the 3 variables. A larger p-value associated with the χ^2-test indicates that the distribution of the variables is more similar among batches. The authors reported p-values of .99058 for sample type, .99963 for race, and 1.00000 for age group. Anticlustering led to p-values of .99999 for sample type, .99998 for race, and 1.00000 for age group. Hence, in this example provided by the OSAT authors, anticlustering led to improved balance among the six batches compared to OSAT.

The developers of PSBA also provided an example application in a vignette. The application uses the data set GSE122288 from the GEO data base[2]. The data set includes 61 cases who provided blood samples to measure DNA methylation levels. In the hypothetical application, these samples had to be assigned to three batches, two batches containing 20 samples and one batch containing 21 samples. They used PSBA to balance the covariates neonate sex, birth weight, and gestational age, and employed the R package `table1` (citation) to illustrate the balancing. Table 3 reproduces the results and also includes the balance that was obtained via the application of anticlustering. As shown, both approaches led to a similar level of balance in this example.

Simulation Study

For the simulation, we generated 2000 data sets. Data sets were processed via (a) OSAT, (b) PSBA, (c) unconstrained anticlustering and (d) anticlustering subject to must-link constraints. Because the OSAT method is only applicable to categorical variables, we used categorical variables in our simulation. For anticlustering and for PSBA, the categorical variables were binary coded before the methods were applied (see Table 2). For each data set, we randomly determined the number of categorical variables (2-5), the number of classes per variable (2-5; the distribution of classes was uniform), the total sample size N (between 50 and 500) and the number of batches K (2, 4, or 10 equal-sized batches). PSBA was only applied for

---

[1] https://github.com/carryp/PS-Batch-Effect
[2] https://www.ncbi.nlm.nih.gov/geo/query/acc.cgi?acc=GSE122288

Table 3: Comparison of the Propensity score approach and anticlustering.

	B1	B2	B3
	(N=20)	(N=20)	(N=21)
Propensity Score			
neonate.gender			
female	9 (45.0%)	9 (45.0%)	9 (42.9%)
male	11 (55.0%)	11 (55.0%)	12 (57.1%)
birth.weight			
Mean (SD)	2980 (376)	2980 (384)	2980 (356)
Median [Min, Max]	2970 [2230, 3610]	2950 [2390, 3620]	3000 [2360, 3670]
gestational.age			
Mean (SD)	38.7 (1.22)	38.8 (1.29)	38.8 (1.26)
Median [Min, Max]	39.0 [37.0, 41.0]	39.0 [37.0, 41.0]	39.0 [37.0, 41.0]
Anticlustering			
neonate.gender			
female	9 (45.0%)	9 (45.0%)	9 (42.9%)
male	11 (55.0%)	11 (55.0%)	12 (57.1%)
birth.weight			
Mean (SD)	2980 (357)	2980 (367)	2980 (390)
Median [Min, Max]	2980 [2230, 3480]	2960 [2360, 3610]	2990 [2370, 3670]
gestational.age			
Mean (SD)	38.8 (1.21)	38.8 (1.29)	38.7 (1.27)
Median [Min, Max]	39.0 [37.0, 41.0]	39.0 [37.0, 41.0]	39.0 [37.0, 41.0]

$K = 2$ and $K = 4$ ($n = 1375$ data sets) because the authors' implementation only allows the assignment to a maximum of four batches. Must-link constraints were generating a random integer between 1 and N (with equal probability), and using the resulting numbers as must-link grouping variables. This rule resulted in a distribution of constraints that resembled our motivating application: 58% of all elements had no must-link partner; 29% had 1 must-link partner; 10% had 2 must-link partners, and 3% had 3 or more must-link partners.

For each simulation run, we computed χ^2-tests to assess the imbalance among batches for each of the 2-5 variables, for each of the three competing methods. We stored the p-value associated with each test. A higher p-value indicates that there is less imbalance among batches, i.e., that the batches are more similar. The simulation revealed that in 83% of all variable comparisons, balance among batches was better when using anticlustering as compared to OSAT. Balance was equal in 16% of all comparisons, and only in 1% OSAT outperformed `anticlust`. In 51% of all variable comparisons, balance was better when using anticlustering as compared to PSBA. Balance was equal in 16% of all comparisons, and in 34% PSBA outperformed anticlustering. Figure 2 illustrates the average p-values in dependence of the number of variables (M) and the number of batches (K). The online supplement also includes additional Figures illustrating average p-values in dependence of the other factors that varied in the simulation (the sample size N and the number of classes per categorical variables; **TODO**). Figure 2 shows that increasing the number of variables posed severe challenges for OSAT, but hardly affected anticlustering. PSBA also showed decreased performance with an increasing number of variables, but less so than OSAT.

The anticlustering assignment that was subjected to must-link constraints on average achieved 99.8% of the objective value of the unconstrained assignment. Hence, must-link constraints are not only desirable from a user's point of view, but they also do not decrease batch balance considerably; in 57% of all cases, balance was not at all reduced by the constraints. Remarkably, the constrained anticlustering assignment led to better balance than the OSAT and PSBA assignments that did not employ any constraints (see Figure 2).
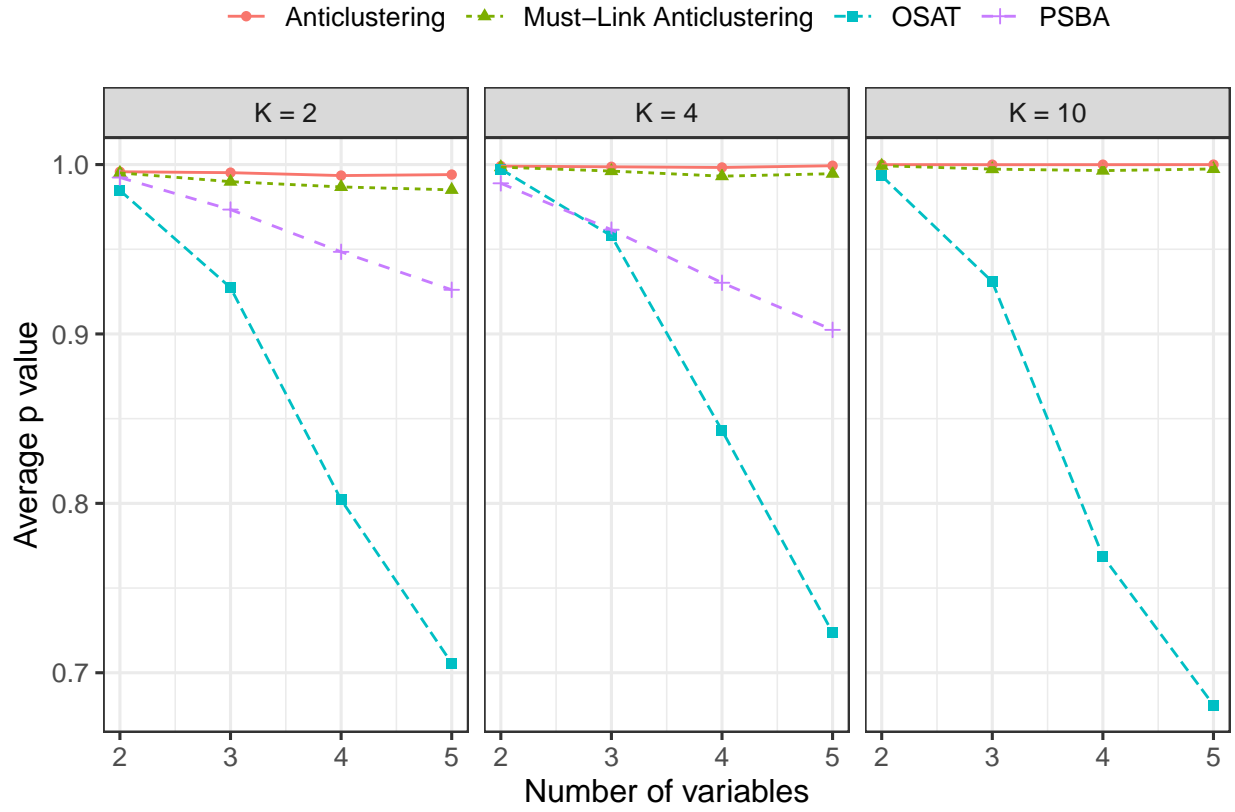
Figure 2: Average p values in dependence of the number of batches and the number of variables. Higher p values indicate better balance. Anticlustering maintained a comparable level of balance in all conditions. OSAT's performance decreased with increasing number of variables most strongly.

References

Baker, Kenneth R, and Stephen G Powell. 2002. "Methods for Assigning Students to Groups: A Study of Alternative Objective Functions." *Journal of the Operational Research Society* 53 (4): 397–404. https://doi.org/10.1057/palgrave.jors.2601307.

Böcker, Sebastian, Sebastian Briesemeister, and Gunnar W Klau. 2011. "Exact Algorithms for Cluster Editing: Evaluation and Experiments." *Algorithmica* 60: 316–34. https://doi.org/10.1007/s00453-009-9339-7.

Brusco, Michael J, J Dennis Cradit, and Douglas Steinley. 2020. "Combining Diversity and Dispersion Criteria for Anticlustering: A Bicriterion Approach." *British Journal of Mathematical and Statistical Psychology* 73 (3): 375–96. https://doi.org/10.1111/bmsp.12186.

Carry, Patrick M, Tim Vigers, Lauren A Vanderlinden, Carson Keeter, Fran Dong, Teresa Buckner, Elizabeth Litkowski, Ivana Yang, Jill M Norris, and Katerina Kechris. 2023. "Propensity Scores as a Novel Method to Guide Sample Allocation and Minimize Batch Effects During the Design of High Throughput Experiments." *BMC Bioinformatics* 24 (1): 86.

Feo, Thomas A, and Mallek Khellaf. 1990. "A Class of Bounded Approximation Algorithms for Graph Partitioning." *Networks* 20 (2): 181–95.

Gallego, Micael, Manuel Laguna, Rafael Marti, and Abraham Duarte. 2013. "Tabu Search with Strategic Oscillation for the Maximally Diverse Grouping Problem." *Journal of the Operational Research Society* 64 (5): 724–34. https://doi.org/10.1057/jors.2012.66.

Garey, Michael R, and David S Johnson. 1979. *Computers and Intractability: A Guide to the Theory of NP-completeness.* New York: Freeman.

Gliesch, Alex, and Marcus Ritt. 2021. "A Hybrid Heuristic for the Maximum Dispersion Problem." *European Journal of Operational Research* 288 (3): 721–35. https://doi.org/10.1016/j.ejor.2020.06.011.

Krauss, Binyamin, Jon Lee, and Daniel Newman. 2013. "Optimizing the Assignment of Students to Classes in an Elementary School." *INFORMS Transactions on Education* 14 (1): 39–44.

Martello, S, and P Toth. 1990. *Knapsack Problems: Algorithms and Computer Implementations.* Wiley.

Mauri, Lara, Bruno Apolloni, and Ernesto Damiani. 2023. "Robust ML Model Ensembles via Risk-Driven Anti-Clustering of Training Data." *Information Sciences* 633: 122–40.

Mohebi, Salman, Andrea Zanella, and Michele Zorzi. 2022. "Pilot Reuse in Cell-Free Massive MIMO Systems: A Diverse Clustering Approach." *arXiv Preprint arXiv:2212.08872.*

Papenberg, Martin. 2024. "K-Plus Anticlustering: An Improved k-Means Criterion for Maximizing Between-Group Similarity." *British Journal of Mathematical and Statistical Psychology* 77 (1): 80–102.

Papenberg, Martin, and Gunnar W Klau. 2021. "Using Anticlustering to Partition Data Sets into Equivalent Parts." *Psychological Methods* 26 (2): 161–74. https://doi.org/10.1037/met0000301.

Rahu, Ida, Meelis Kull, and Anneli Kruve. 2024. "Predicting the Activity of Unidentified Chemicals in Complementary Bioassays from the HRMS Data to Pinpoint Potential Endocrine Disruptors." *Journal of Chemical Information and Modeling.*

Schaper, Marie Luisa, Beatrice G Kuhlmann, and Ute J Bayen. 2023. "Metacognitive Differentiation of Item Memory and Source Memory in Schema-Based Source Monitoring." *Journal of Experimental Psychology: Learning, Memory, and Cognition* 49 (5): 743–65. https://doi.org/10.1037/xlm0001207.

Shamir, Ron, Roded Sharan, and D Tsur. 2004. "Cluster Graph Modification Problems." *Discrete Applied Mathematics* 144: 173–82. https://doi.org/10.1016/j.dam.2004.01.007.

Späth, H. 1986. "Anticlustering: Maximizing the Variance Criterion." *Control and Cybernetics* 15 (2): 213–18.

Steghöfer, Jan-Philipp, Pascal Behrmann, Gerrit Anders, Florian Siefert, and Wolfgang Reif. 2013. "HiS-PADA: Self-Organising Hierarchies for Large-Scale Multi-Agent Systems." In *Proceedings of the IARIA International Conference on Autonomic and Autonomous Systems (ICAS)*, edited by Michael Bauer, Radu Calinescu, Michael Grottke, and Bruno Dillenseger, 71–76. Lisbon, Portugal: International Academy, Research, and Industry Association.

Weitz, RR, and S Lakshminarayanan. 1998. "An Empirical Comparison of Heuristic Methods for Creating Maximally Diverse Groups." *Journal of the Operational Research Society* 49 (6): 635–46. https://doi.org/10.1057/palgrave.jors.2600510.

Yan, Li, Changxing Ma, Dan Wang, Qiang Hu, Maochun Qin, Jeffrey M Conroy, Lara E Sucheston, et al. 2012. "OSAT: A Tool for Sample-to-Batch Allocations in Genomics Experiments." *BMC Genomics* 13: 1–7.

Yang, Xiao, Zonghui Cai, Ting Jin, Zheng Tang, and Shangce Gao. 2022. "A Three-Phase Search Approach

with Dynamic Population Size for Solving the Maximally Diverse Grouping Problem." *European Journal of Operational Research* 302 (3): 925–53.