Getting More Out of Data Competition Results with Pareto Fronts

Christine M. Anderson-Cook, Los Alamos National Laboratory, candcook@lanl.gov
Lu Lu, University of South Florida
Sarah E. Burke, The Perduco Group

Abstract

Data competitions have attracted considerable attention among the world's community of data and analytics scientists, as well as discipline-specific subject matter experts. Their broad involvement provides a model of crowdsourcing for business and government to solve tough high-impact problems in a cost-effective way. Typically, winners are determined through a leaderboard formula that needs to be static throughout the competition, with fixed rewards and penalties for patterns of correct and incorrect responses for different aspects of the solution. However, for different uses of the solution, these aspects might be more or less important. By using the existing capability for constructing flexible high-dimensional Pareto fronts in JMP and a developed Add-In, it is possible to explore and identify the strengths and weaknesses of various solutions. Pareto fronts allow the user to identify all of the objectively superior solutions across all possible weightings of the different elements of the solution, and discard non-competitive solutions. The approach to using multiple Pareto fronts to highlight different "best" solutions is demonstrated through a recently completed data competition focused on detecting, identifying and locating radioactive sources in an urban environment (https://www.topcoder.com/lp/detect-radiation).

Introduction

Data competitions, sometimes called machine learning competitions, have attracted considerable attention among the world's community of data and analytics scientists and discipline-specific subject matter experts. This broad involvement provides a model of crowdsourcing for business and government to solve tough high-impact problems in a cost-effective way. Competition hosts often use a commercial platform, such as Kaggle (www.kaggle.com), to hold the competition, rank competitors, and provide a prize (from thousands to millions of dollars) to reward winners. By bringing in new approaches to solving problems, there is potential to accelerate cutting-edge research through the use of data science approaches and the involvement of a more technically diverse set of experts.

The kinds of data competitions we consider here are conducted in a supervised learning framework (Hastie and Tibshirani, 2009). That is, competitors are provided two sets of data: a training set, for which the answers are provided, and a test set, for which the competitors will provide their predicted answers for scoring. Competitors develop or "train" their algorithms using the training set, then refine them based on the feedback given by their reported scores on the test set. Typically, the platform that runs the competition will provide a real-time public leaderboard that reflects each competitor's best score and ranking based on their predicted answers for the test set. Competitors can make multiple submissions over the course of the competition, and each is scored and incorporated into the leaderboard if it reflects an improvement over the previous best submission for that competitor.

Within the test set, there is a further division of the data that is not disclosed to the participants. A fraction of test data forms the public test set, which is used to score and rank the competitors on the public leaderboard while the competition is running. The remaining test data form the private test set. The final score for each competitor is based on the private test set and is not shared with the competitors until the competition closes. The private leaderboard, based on the scores on the private test set, specifies the final ranking and winners of the competition.

The host has the flexibility to specify what data comprises the training, public test, and private test sets, often subject to a practical limit on the total amount of available data. The host also chooses a static evaluation formula or scoring metric to define the score of each submission and its ranking on the public and private leaderboards. It is our understanding that the vast majority of current data competitions rely exclusively on the leaderboard to evaluate and rank the submitted solutions. To provide timely, succinct feedback on competitors' performance during the competition, the scoring metric is usually a simple scalar summary that quantifies the accuracy and effectiveness of a solution for solving the competition task(s). This metric, when properly defined, encompasses the key aspects of the problem under investigation with the competition, and seeks to identify its top solutions.

However, there are several potential limitations to this approach: First, by necessity, the scoring metric is created before the competition opens. Hence anything that the host learns by observing competitor contributions cannot be incorporated into revisions of the metric. Second, the leaderboard summary is a global number that amalgamates responses across a large number of instances, each of which could represent different regions of the problem space. Finally, since many data competitions involve multiple tasks, the scoring metric for the leaderboard must combine evaluation of all of these tasks and may be too simplistic to allow deeper understanding of the relative performance of the different solutions and address multiple questions of interest to the host.

We consider an aspect of the post-competition analysis that seeks to identify specific solutions for a subset of the overall objectives of the competition. At the conclusion of the competition, the host may have access to details of all of the submission results which are partitioned into contributions from each of the sub-tasks that were required of the competitors. For example, it is possible to construct a table that has one row for each submission from the competition, and one column with a score/summary for the individual sub-tasks. Our goal is to illustrate how to use JMP software to identify promising candidate solutions for a particular subset of objectives. Pareto fronts are an established tool that allows for the elimination of inferior solutions. We complement this with a graphical summary, called the trade-off plot, that allows the competition host to evaluated and compare solutions on the Pareto front.

The Urban Radiation Search Competition

This competition used simulated measurements mimicking those collected by a radiation detector being driven along typical urban streets. The simulations were performed at Oak Ridge National Laboratory, where they could flexibly simulate data for a wide variety of scenarios. The inputs for these scenarios were chosen to mimic the diversity of urban environments seen in practice.
A key feature of urban radiological search is being able to separate the background signal (generated from benign emitters of radiation, like buildings and pavement, in the urban environment) from a localized source. We divided the input factors into several categories: characteristics of the background, characteristics of the sources, and characteristics of the detector's movement. For the background factors, several versions of urban streets were used with different configurations and compositions for the buildings and features.

For the source factors, we considered five different radioactive source types, plus an additional source defined as a combination of two of the sources. These sources include weapons grade materials and isotopes common in medical or industrial settings:

1. HEU: Highly enriched uranium
2. WGPu: Weapons grade plutonium
3. 131I: Iodine, a medical isotope
4. 60Co: Cobalt, an industrial isotope
5. 99mTc: Technetium, a medical isotope
6. A combination of HEU and 99mTc

The other source factors included its location on the street, its strength, and whether it was shielded in a dampening container. With close engagement from the subject matter experts, we combined the location, strength, and shielding factors into a measure of the signal-to-noise ratio (SNR). For the detector factors we considered its speed in meters per second as it traveled along the street, the traffic lane of travel, and the starting / ending points within a street.

The data were generated using a stochastic simulation code developed at Oak Ridge National Laboratory. Each "run" or instance of data in the training and test sets was specified by selecting values for more than 100 parameters. For more details on the competition, see Anderson-Cook et al. (2019a and 2019b).

Two versions of the competition were run with two different sets of data. The first competition was open to government employees from February to May of 2018, and the second competition was hosted on Topcoder and was open from March to May of 2019. To illustrate the methods, we consider the resulting data from the second competition. In this competition, there were 69 teams that competed, with a total of 1479 valid submissions. For this example, we focus on 13 sub-tasks that the competitors were asked to perform: For each of the 6 sources, the fraction runs where the source was correctly detected (1-6) and the fraction of runs where the source was correctly identified (7-12). Note that in order for a source to be correctly identified, it must necessarily have been correctly detected. Finally, there were runs where there was no source present. For these runs, the fraction of correctly classified "no source" runs (13) was the key summary. We have also included an alternative summary for the "no source" runs: the false positive rate, which is simply 1- the fraction of correctly classified no source runs. A desirable outcome for this criterion is to minimize it (This will be helpful to illustrate some features of the Add-in later). A 14th summary was also considered, the leaderboard score, since it was viewed as a global summary of the goodness of the algorithm across all of the subtasks.

It should be noted that the detection/identification summaries focus on the type of error where a source is missed. This would be problematic if an adversary placed some radioactive material and it was not found. The consequences of such an error would be potential harm to people in these surroundings. The no source runs focus on a different type of error, namely where there is nothing to be found, but an alarm triggers from the algorithm. In this case, the consequences are that the first responders rush to the scene to react to a problem that does not exist. Since first responders are unlikely to keep using a system that gives too many false alarms (or false positives), this is a key priority in the choice of which solution to use for any scenario.

Pareto Front Basics

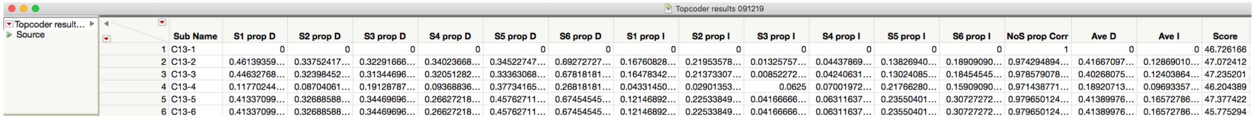
A fundamental problem with ranking choices based on multiple criteria is that there is no unique ordering of the choices. Depending on how much we value different criteria, the best choice changes. Consider a simple example with cost and quality. If we emphasize cost exclusively, we would select the cheapest item, regardless of quality. If we emphasize quality exclusively, we would select the highest quality item, regardless of cost. If we strove to achieve some balance between alternatives, then we might choose a "reasonably priced item with reasonable quality". The choice of balancing the cost and quality criteria depends on our subjective valuing of these objectives.

A Pareto front (Lu et al, 2011) consists of all of the non-dominated solutions from our set of choices. A solution is defined to be dominated, if there exists another solution that is at least as good for all of the criteria, and strictly better for at least one criterion. For example, dominated solutions in the cost-quality example would be (a) one for which there is another choice that is simultaneously cheaper and better quality, (b) one with an alternative that is the same quality, but cheaper, or (c) one with an alternative that is the same price, but better quality. Choices that are not on the Pareto front should not considered rational. So a Pareto front provides an objective set of rational choices from which we can choose a "best solution" based on our subjective prioritization of the quantities of interest. It can be constructed without any scaling of the different criteria, and allows us to see which combinations should be under consideration as a subset among all of the alternatives. The collection of solutions on the Pareto front can also provide valuable information about the range of criterion values that are sensible to consider among the alternatives. In the Define-Measure-Reduce-Combine-Select (DMRCS) process described in Anderson-Cook and Lu (2015), the Pareto front is considered a key tool in the Reduce stage.

Illustrating the Use of Pareto Fronts in JMP

Consider the data from the second data competition. Table 1 shows a snapshot of the first few rows of the summary table that we use to evaluate different solutions. Each row represents a submitted solution by one of the competitors, and the columns summarize different sub-tasks previously defined. The first column, Sub Name, indicates a label for the submission, with the competitor's number listed first, and then which submission of theirs it is after the "-". In addition to the fraction of correct detection (S1 prop D, S2 prop D, ….) and identification (S1 prop I, S2 prop I, ….) for each source, the no source correct classification was also listed. "Ave D" and "Ave I" correspond to the average correct detection and identification across the 6 sources. The Score column represents the score for that run posted on the leaderboard.

Table 1: Sample of Data Table from Data Competition



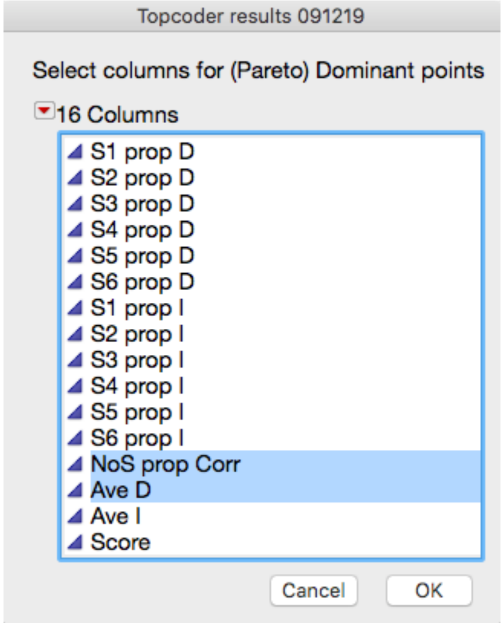
	Sub Name	S1 prop D	S2 prop D	S3 prop D	S4 prop D	S5 prop D	S6 prop D	S1 prop I	S2 prop I	S3 prop I	S4 prop I	S5 prop I	S6 prop I	NoS prop Corr	Ave D	Ave I	Score
1	C13-1	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	46.726166
2	C13-2	0.46139359…	0.33752417…	0.32291666…	0.34023668…	0.34522747…	0.69272727…	0.16760828…	0.21953578…	0.01325757…	0.04437869…	0.13826940…	0.18909090…	0.974294894…	0.41667097…	0.12869010…	47.072412
3	C13-3	0.44632768…	0.32398452…	0.31344696…	0.32051282…	0.33363068…	0.67818181…	0.16478342…	0.21373307…	0.00852272…	0.04240631…	0.13024085…	0.18454545…	0.978579078…	0.40268075…	0.12403864…	47.235201
4	C13-4	0.11770244…	0.08704061…	0.19128787…	0.09368836…	0.37734165…	0.26818181…	0.04331450…	0.02901353…	0.0625	0.07001972…	0.21766280…	0.15909090…	0.971438771…	0.18920713…	0.09693357…	46.204389
5	C13-5	0.41337099…	0.32688588…	0.34469696…	0.26627218…	0.45762711…	0.67454545…	0.12146892…	0.22533849…	0.04166666…	0.06311637…	0.23550401…	0.30727272…	0.979650124…	0.41389976…	0.16572786…	47.377422
6	C13-6	0.41337099…	0.32688588…	0.34469696…	0.26627218…	0.45762711…	0.67454545…	0.12146892…	0.22533849…	0.04166666…	0.06311637…	0.23550401…	0.30727272…	0.979650124…	0.41389976…	0.16572786…	45.775294

In the next illustrations, we show how starting with a target for the ideal solution for a particular scenario guides the selection of relevant criteria and this allows for the creation of a tailored Pareto front and graphical summaries highlighting promising solutions.

Case 1: Goal of good detection across all of the sources

For this scenario, the chosen criteria are "Ave D" and "NoS prop Corr", since they represent simple summaries to provide a good chance of detecting any of the 6 sources as well as protecting against too many false positives when no source is present.

Using the data table with all of the competitor submissions, select Rows, then Row Selection ▶ then Select Dominant... . When the dialog box opens with the list of the columns, select the two criteria that we wish to consider:

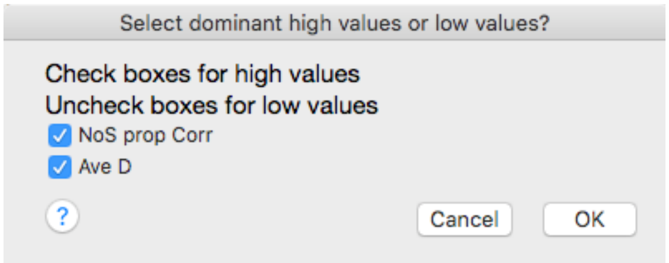


Topcoder results 091219

Select columns for (Pareto) Dominant points

16 Columns

- S1 prop D
- S2 prop D
- S3 prop D
- S4 prop D
- S5 prop D
- S6 prop D
- S1 prop I
- S2 prop I
- S3 prop I
- S4 prop I
- S5 prop I
- S6 prop I
- NoS prop Corr
- Ave D
- Ave I
- Score



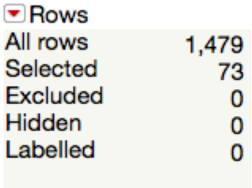
Select dominant high values or low values?

Check boxes for high values
Uncheck boxes for low values

☑ NoS prop Corr
☑ Ave D

Next, specify whether we wish to minimize or maximize each of the quantities. Since we want to have both a high proportion of correctly detected sources for the runs that contain sources, and a high proportion of no source runs that are correctly classified, both boxes should remain checked.

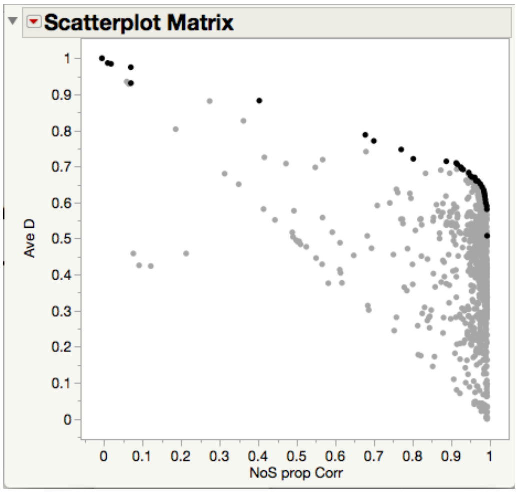
When we click “OK”, the rows on the Pareto front are highlighted, with the non-dominant solutions not highlighted. In the bottom right corner, we can see that of the 1479 submissions, only 73 are on the Pareto front. This represents a helpful reduction of the number of items that we need to consider.

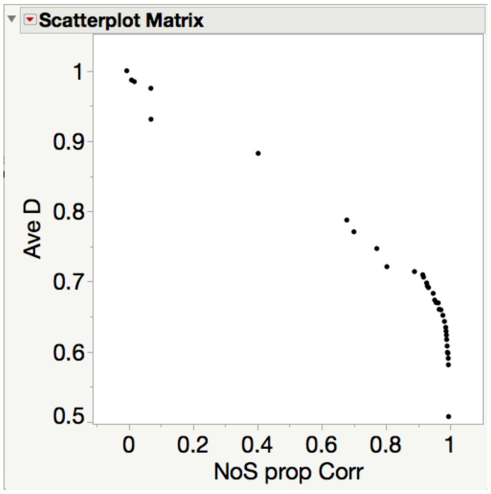


Rows

All rows	1,479
Selected	73
Excluded	0
Hidden	0
Labelled	0

With the rows still highlighted, it can be helpful to plot the data to see which solutions have been identified and how they compare to the overall set of all solutions. As anticipated, those solutions which are on the leading edge of the solutions closest to the top right corner (which represents high detection rates and high correctly classification of the no source runs).

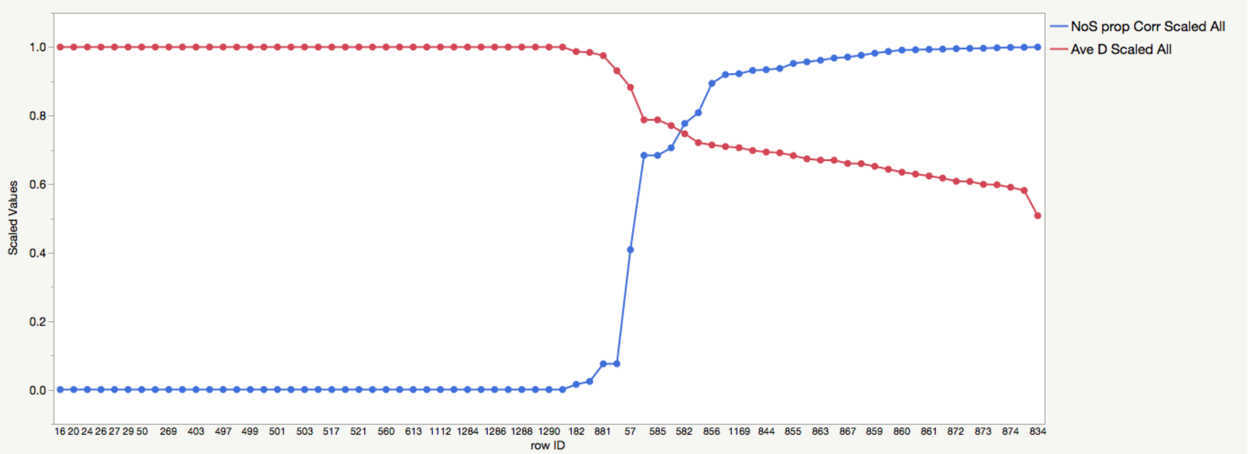




To work with only the Pareto front solutions, we can create a new data table with only those rows. Under “Tables”, select “Subset” and choose to use the selected rows. From this new table, we can create a scatterplot of just the Pareto front solutions, which shows the trade-off between good detection and low false positive rates.

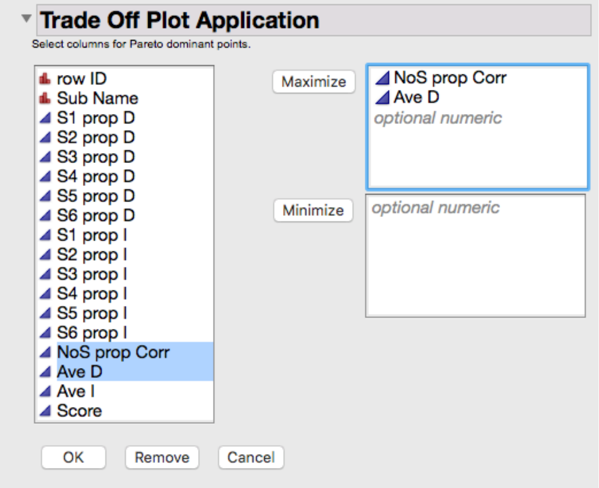
Note that the top left corner of the Pareto plot corresponds to (0,1), which means that we have 100% correct detection, but 0% correct classification of no source runs (a solution where all the runs are treated as having a source). It is quite easy to create solutions that have 100% correct classification of the no source runs (submit a solution where everything is classified as no source). This would correspond to the point (1,0), but we see that this is not on the Pareto front. One of the solutions is able to detect ~50% of the sources, while still maintaining 100% correct classification of the no source runs. Hence this solution dominates the point (1,0) and prevents it from being on the Pareto front.

Visualizing the trade-off between two criteria can be done well with the Pareto front itself, and hence there is no need for additional graphical tools. However, we now introduce the trade-off plot, as it will prove useful when we consider more than 2 criteria for our selection of a best algorithm. The figure below shows the trade-off plot for the same scenario. In this plot, we use a desirability scale to show the ranges of the criteria, with the best available value being scaled to 1 (which corresponds to the 100% correct for our two criteria, but in general these values will not be the same). The worst value, which is 0 can be chosen based on either the entire data set (shown here) or based on just the Pareto front values. The solutions are sorted from left to right from worst to best for one of the criteria (here “NoS prop Corr”), with the other criterion (or multiple criteria, when more than 2 are being considered) shown with a different colored line. By sliding from left to right, we can see solutions that prioritize one criterion more or less strongly.



For this example, we see that there are many tied solutions of the left, where competitors had 100% correct detection. Since the implementers of the solution do not wish to have high false positive rates, the solutions that they would find most desirable would lie closer to the right side of the plot, where the proportion of correct no source classification was greater than 90%. By examining these solutions more closely, it is possible to determine which solution is best to balance the proportion of correct detections with the proportion of correct no source classification.

To generate this plot, requires an Add-in developed by Sarah Burke. After installing the Add-in, then select it with the data table open. Specify which criteria you wish to maximize and minimize.



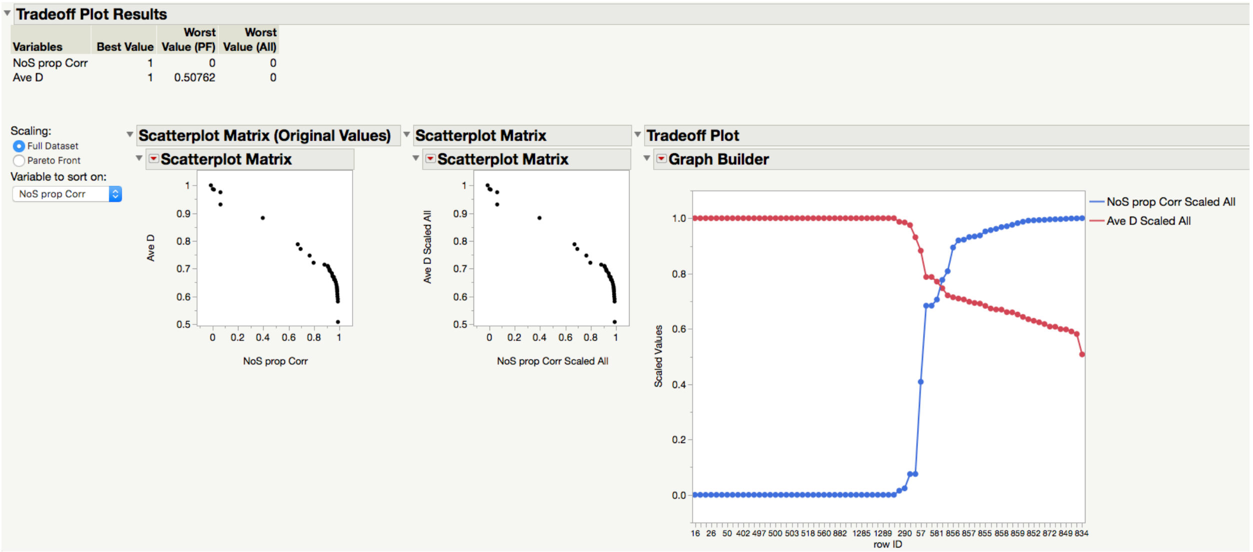
Maximize: NoS prop Corr, Ave D

After clicking "OK", the Pareto front and trade-off plot are created, with a separate data table containing the solutions found on the Pareto front. In the top left corner the ranges of the values are summarized, with the universal best value for each criterion listed, as well as the worst values for the entire data set and on the Pareto front.

At the bottom of the summary, the left side shows a pairwise scatterplot matrix of all of the criteria on their original scaling. The middle plot is a pairwise scatterplot matrix of the criteria scaled on the desirability scale (1 being best, and 0 being worst). On the right hand side, the trade-off plot shows the ordered solutions with a separate line for each criterion.

There are options to scale the trade-off plot to be based on either of the two worst values. In addition, the plot can be sorted based on any of the criteria, with the selected column being sorted from worst to best going from left to right.

The plots are dynamic to allow points to be selected in any plot to be highlighted in the created table as well as the other plot. This allows access to details of the chosen solution for all of the criteria and for results on the original scale.

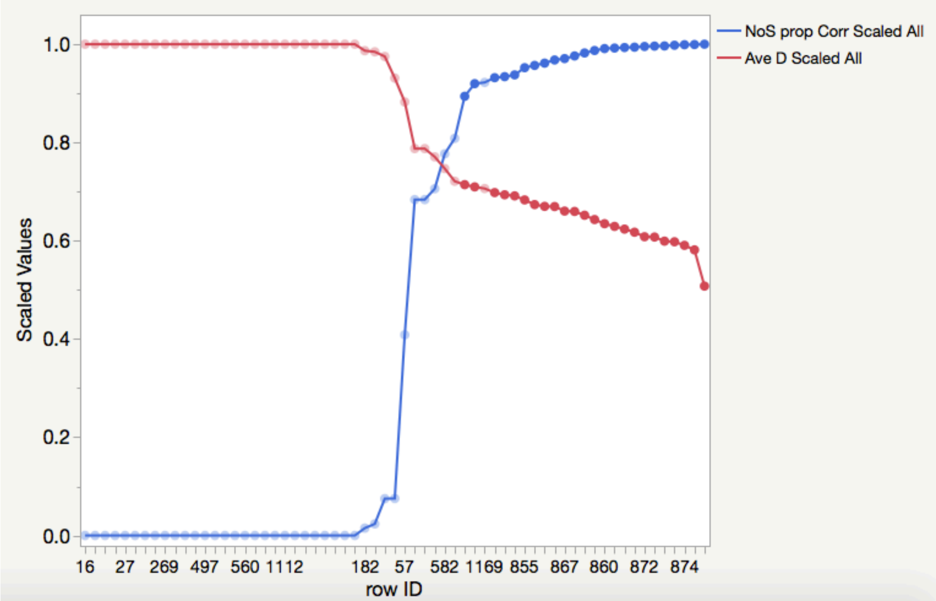


Tradeoff Plot Results

Variables	Best Value	Worst Value (PF)	Worst Value (All)
NoS prop Corr	1	0	0
Ave D	1	0.50762	0

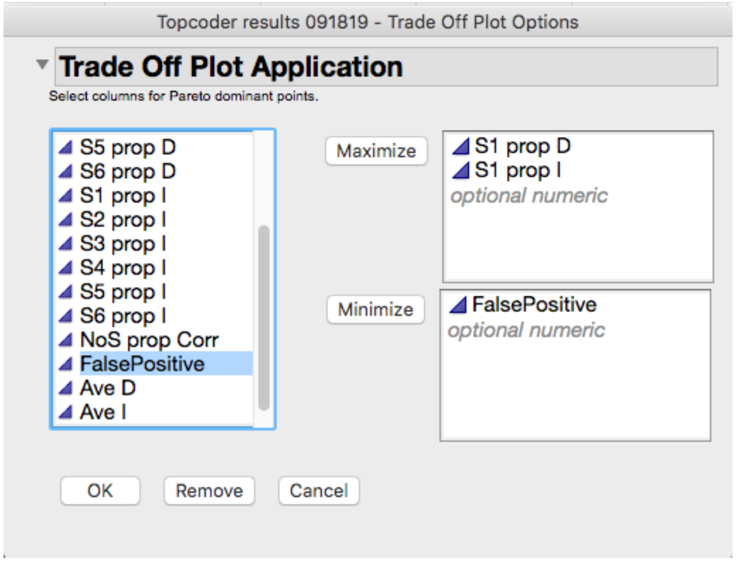
Scaling:
● Full Dataset
○ Pareto Front
Variable to sort on: NoS prop Corr

Also included in the Trade-off Add-in is a constructed data table with the solutions on the Pareto front. when we examine the results, we see that 24 of 73 solutions were submitted by the winning competitor (Comp1). Below is the plot with the submissions from Comp1 highlighted. They represent almost all of the solutions on the right hand side of the plot, where the "NoS prop Corr" values are in the desired range with values greater than 90% correct.



Case 2: Goal of good performance for Source 1

We next consider a scenario where we consider 3 criteria. If there was particular interest in good performance for detection and identification of Source 1, HEU (Highly enriched Uranium), then we might select the following criteria as our focus. Note that either “NoS prop Corr” or “False Positive” are always included in our Pareto fronts, since no solution will be considered desirable unless it moderates this type of error. Here we select “False Positive” as the criterion, which will be helpful to illustrate the minimization capability of the Trade-off Add-in.



Topcoder results 091819 - Trade Off Plot Options

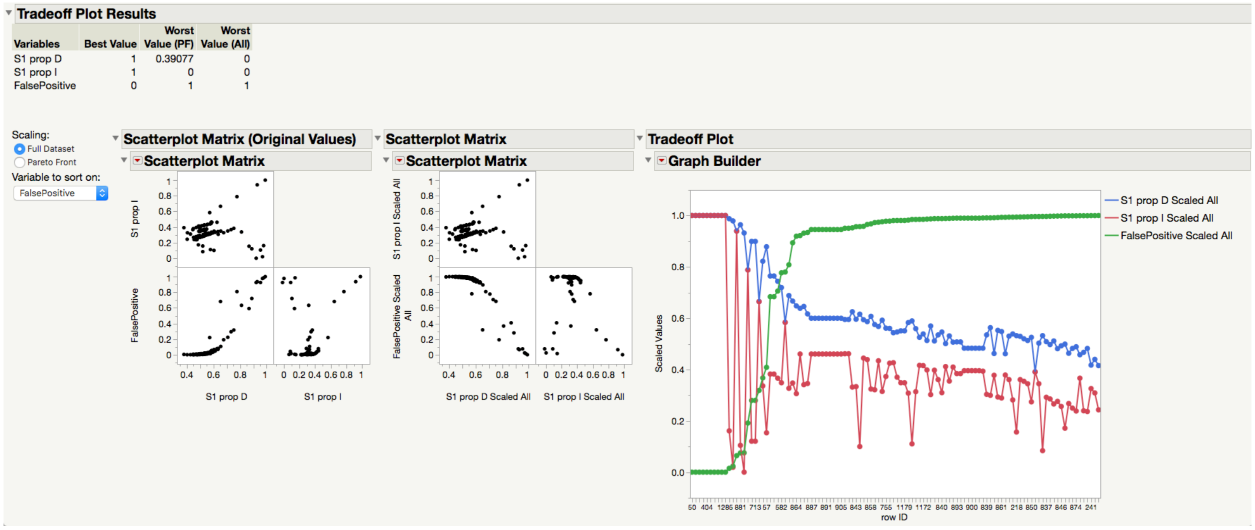
Trade Off Plot Application

Select columns for Pareto dominant points.

Maximize: S1 prop D, S1 prop I

Minimize: FalsePositive

The Pareto front for this scenario has 110 solutions, and a version of the summaries are shown below, after selecting to scale based on the entire dataset, and sort based on “False Positive”. Note that when we minimize a criterion (here “False Positive”), the best value is smaller than the worst value. Different from the previous example, the left and middle plots are now different. On the left, the original values are shown, while in the middle, the scaled response are shown. Note how the plots involving “False Positive” are now flipped vertically. The motivation for the trade-off plot should be clearer now that the Pareto front plots on the left are no longer able to show the 3-D surface of the plot.



Tradeoff Plot Results

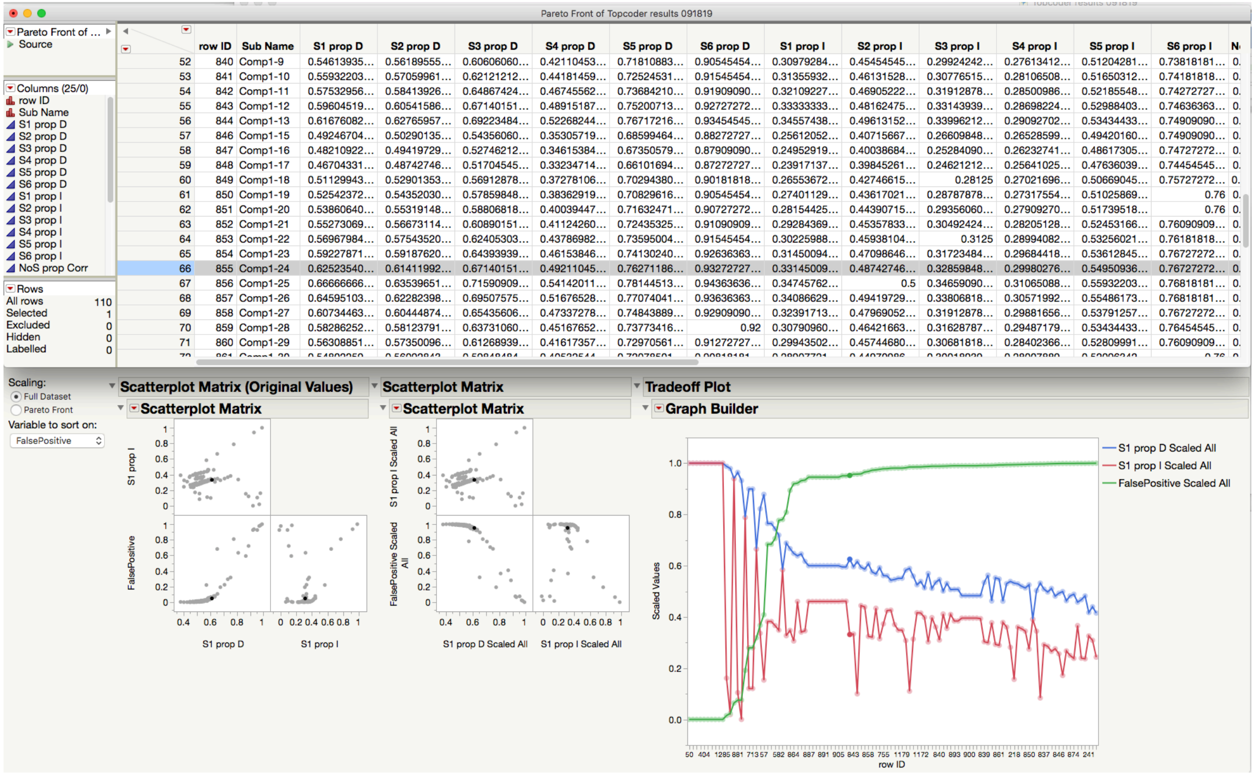
Variables	Best Value	Worst Value (PF)	Worst Value (All)
S1 prop D	1	0.39077	0
S1 prop I	1	0	0
FalsePositive	0	1	1

Scaling:
● Full Dataset
○ Pareto Front

Variable to sort on: FalsePositive

In this case there are a large number of solutions with the proportion of correct no source classification was greater than 90%. As noted earlier, the proportion of correct identification must necessarily be less than or equal to the proportion of correct detection, and this plot makes it straightforward to see the difference between these two sub-tasks.

The figure below shows one selected solution from the Trade-off plot, with the highlighted solution shown highlighted on the scatterplots and in the Pareto front table. This makes it easy to extract values for the original criteria graphically and numerically.



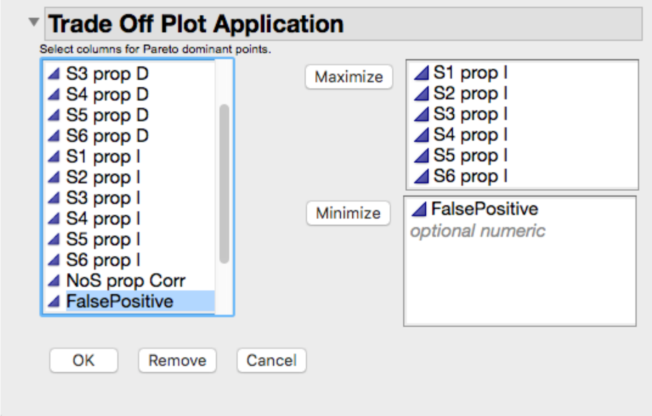
Rows	
All rows	110
Selected	1
Excluded	0
Hidden	0
Labelled	0

Scaling:
● Full Dataset
○ Pareto Front
Variable to sort on: FalsePositive

	row ID	Sub Name	S1 prop D	S2 prop D	S3 prop D	S4 prop D	S5 prop D	S6 prop D	S1 prop I	S2 prop I	S3 prop I	S4 prop I	S5 prop I	S6 prop I	N
52	840	Comp1-9	0.54613935...	0.56189555...	0.60606060...	0.42110453...	0.71810883...	0.90545454...	0.30979284...	0.45454545...	0.29924242...	0.27613412...	0.51204281...	0.73818181...	0.
53	841	Comp1-10	0.55932203...	0.57059961...	0.62121212...	0.44181459...	0.72524531...	0.91545454...	0.31355932...	0.46131528...	0.30776515...	0.28106508...	0.51650312...	0.74181818...	0.
54	842	Comp1-11	0.57532956...	0.58413926...	0.64867424...	0.46745562...	0.73684210...	0.91909090...	0.32109227...	0.46905222...	0.31912878...	0.28500986...	0.52185548...	0.74272727...	0.
55	843	Comp1-12	0.59604519...	0.60541586...	0.67140151...	0.48915187...	0.75200713...	0.92727272...	0.33333333...	0.48162475...	0.33143939...	0.28698224...	0.52988403...	0.74636363...	0.
56	844	Comp1-13	0.61676082...	0.62765957...	0.69223484...	0.52268244...	0.76717216...	0.93454545...	0.34557438...	0.49613152...	0.33996212...	0.29092702...	0.53434433...	0.74909090...	0.
57	846	Comp1-15	0.49246704...	0.50290135...	0.54356060...	0.35305719...	0.68599464...	0.88272727...	0.25612052...	0.40715667...	0.26609848...	0.26528599...	0.49420160...	0.74909090...	0.
58	847	Comp1-16	0.48210922...	0.49419729...	0.52746212...	0.34615384...	0.67350579...	0.87909090...	0.24952919...	0.40038684...	0.25284090...	0.26232741...	0.48617305...	0.74727272...	0.
59	848	Comp1-17	0.46704331...	0.48742746...	0.51704545...	0.33234714...	0.66101694...	0.87272727...	0.23917137...	0.39845261...	0.24621212...	0.25641025...	0.47636039...	0.74454545...	0.
60	849	Comp1-18	0.51129943...	0.52901353...	0.56912878...	0.37278106...	0.70294380...	0.90181818...	0.26553672...	0.42746615...	0.28125	0.27021696...	0.50669045...	0.75727272...	0.
61	850	Comp1-19	0.52542372...	0.54352030...	0.57859848...	0.38362919...	0.70829616...	0.90545454...	0.27401129...	0.43617021...	0.28787878...	0.27317554...	0.51025869...	0.76	0.
62	851	Comp1-20	0.53860640...	0.55319148...	0.58806818...	0.40039447...	0.71632471...	0.90727272...	0.28154425...	0.44390715...	0.29356060...	0.27909270...	0.51739518...	0.76	0.
63	852	Comp1-21	0.55273069...	0.56673114...	0.60890151...	0.41124260...	0.72435325...	0.91090909...	0.29284369...	0.45357833...	0.30492424...	0.28205128...	0.52453166...	0.76090909...	0.
64	853	Comp1-22	0.56967984...	0.57543520...	0.62405303...	0.43786982...	0.73595004...	0.91545454...	0.30225988...	0.45938104...	0.3125	0.28994082...	0.53256021...	0.76181818...	0.
65	854	Comp1-23	0.59227871...	0.59187620...	0.64393939...	0.46153846...	0.74130240...	0.92636363...	0.31450094...	0.47098646...	0.31723484...	0.29684418...	0.53612845...	0.76727272...	0.
66	855	Comp1-24	0.62523540...	0.61411992...	0.67140151...	0.49211045...	0.76271186...	0.93272727...	0.33145009...	0.48742746...	0.32859848...	0.29980276...	0.54950936...	0.76727272...	0.
67	856	Comp1-25	0.66666666...	0.63539651...	0.71590909...	0.54142011...	0.78144513...	0.94363636...	0.34745762...	0.5	0.34659090...	0.31065088...	0.55932203...	0.76818181...	0.
68	857	Comp1-26	0.64595103...	0.62282398...	0.69507575...	0.51676528...	0.77074041...	0.93636363...	0.34086629...	0.49419729...	0.33806818...	0.30571992...	0.55486173...	0.76818181...	0.
69	858	Comp1-27	0.60734463...	0.60444874...	0.65435606...	0.47337278...	0.74843889...	0.92909090...	0.32391713...	0.47969052...	0.31912878...	0.29881656...	0.53791257...	0.76727272...	0.
70	859	Comp1-28	0.58286252...	0.58123791...	0.63731060...	0.45167652...	0.73773416...	0.92	0.30790960...	0.46421663...	0.31628787...	0.29487179...	0.53434433...	0.76454545...	0.
71	860	Comp1-29	0.56308851...	0.57350096...	0.61268939...	0.41617357...	0.72970561...	0.91272727...	0.29943502...	0.45744680...	0.30681818...	0.28402366...	0.52809991...	0.76090909...	0.

Case 3: Goal of good identification for all sources

We next consider a scenario where we consider 7 criteria. In this case, the goal is to identify solutions that perform well for the identification of all 6 sources, as well as maintaining an acceptable false positive rate. In this case, the selected criteria are shown below:



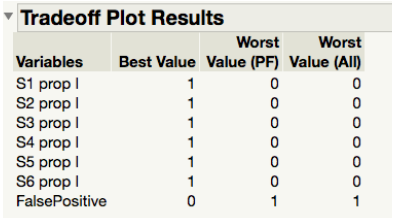
Trade Off Plot Application

Select columns for Pareto dominant points.

S3 prop D
S4 prop D
S5 prop D
S6 prop D
S1 prop I
S2 prop I
S3 prop I
S4 prop I
S5 prop I
S6 prop I
NoS prop Corr
FalsePositive

Maximize: S1 prop I, S2 prop I, S3 prop I, S4 prop I, S5 prop I, S6 prop I

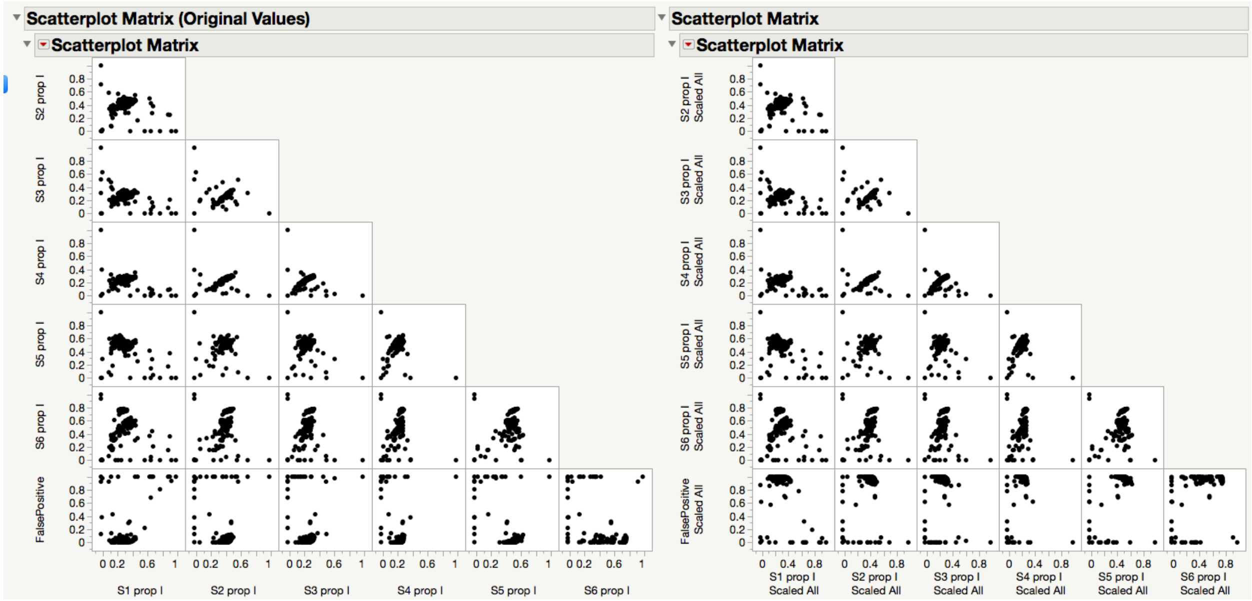
Minimize: FalsePositive, *optional numeric*

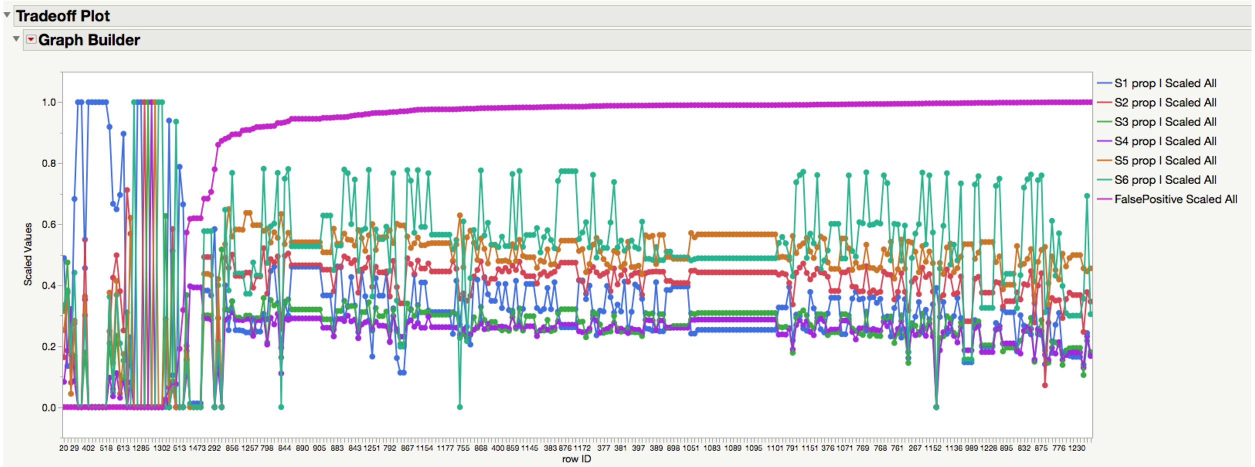


Tradeoff Plot Results

Variables	Best Value	Worst Value (PF)	Worst Value (All)
S1 prop I	1	0	0
S2 prop I	1	0	0
S3 prop I	1	0	0
S4 prop I	1	0	0
S5 prop I	1	0	0
S6 prop I	1	0	0
FalsePositive	0	1	1

Screenshots of different parts of the Trade-off Add-In results are shown below:





The pairwise scatterplots of the criteria (both the original values and the desirability scaled versions) illustrate the benefits of using the trade-off plot as it is challenging to get a sense of the relationships between criteria. From the trade-off plot, we can extract a number of features:

- There are a large number of solutions with the proportion of correct no source classification was greater than 90%.
- Different solutions vary in how well they are able to identify different sources, but there is a general ranking of the difficulty of the sources with Source 3 being generally easiest (typically highest of the source lines on the right of the plot)
- The range of correct identification for any of the sources fluctuates considerably across the range of similar proportion of correct no source classification alternatives.

Given that there are many different scenarios that might be of interest for the use of the urban radiation detection algorithm, it makes sense that there are many subsets of the criteria that could be selected for the construction of the Pareto fronts and trade-off plots. In the competition scenario, here are some additional combinations of criteria that were considered for specific scenarios:

- "NoS prop Corr" combined with "Ave Identification" (this scenario is similar to Case 3, but does not allow exploration of the individual identification rates).
- "NoS prop Corr" combined with "S1 prop D", "S2 prop D", … "S6 prop D" (this scenario allows more detailed understanding of the detection performance of each of the sources, and expands the level of understanding beyond what was considered in Case 1)
- "NoS prop Corr" combined with "Ave Identification" and "Ave Detection" (this scenario allows simultaneous consideration of both detection and identification, but groups the individual performance across the sources into a single summary).
- "NoS prop Corr" combined with "S1 prop D" and "S2 prop D" (this scenario emphasizes good performance for the weapon related radioactive sources, and does not consider the more benign medical and industrial isotopes).

Adding additional criteria must necessarily make the set of solutions on the Pareto front larger, so for each considered scenario, it is important to identify the key objectives and to streamline the number of criteria included. This will provide a smaller number of choices that need to be considered in the decision-making phase where selecting the best option is done. If we consider all of the 14 criteria listed, then the Pareto front consists of 308 solutions. So 1169 solutions (1479 – 308) can be removed from consideration as best for any subset of criteria. This simplification is helpful to make choosing a best option for a given scenario more streamlined.

Conclusions

Data competitions are becoming increasingly common as a cost-effective approach for generating diverse solutions to difficult problems. However, relying exclusively on the constructed leaderboard to determine which solution to use might be limiting and miss potentially valuable solutions that would target particular sub-tasks well. The Pareto front and trade-off plot provide an easy way of examining the differences between the solutions and being able to select a tailored solution to match the specific goals of different scenarios.

References

1. Anderson-Cook, C.M., Lu, L. (2015) "Much-Needed Structure: A New 5-Step Decision-Making Process Helps You Evaluate, Balance Competing Objectives" **Quality Progress** 48(10) 42-50.
2. Anderson-Cook, C.M., Lu, L., Myers, K., Quinlan, K., Pawley, N. (2019a) "Improved Learning from Data Competitions through Strategic Generation of Informative Data Sets" **Quality Engineering** 31(4) 564-580.
3. Anderson-Cook, C.M., Myers, K., Lu, L., Fugate, M.L., Quinlan, K., Pawley, N. (2019b) "Data Competitions: Getting More from a Strategic Design and Post-Competition Analysis" **Statistical Analysis and Data Mining** 12 271-289**.**
4. Hastie, T., Tibshirani, R. ***The Elements of Statistical Learning: Data mining, Inference, and Prediction*****.** New York: Springer, 2009, pp. 485–586.
5. Lu, L., Anderson-Cook, C.M., Robinson, T.J. (2011) "Optimization of Designed Experiments Based on Multiple Criteria Utilizing a Pareto Frontier" **Technometrics** 53 353-365.