Evolutionary Optimization of Flavors

Kalyan Veeramachaneni
Massachusetts Institute of Technology
Cambridge, MA
kalyan@csail.mit.edu

Katya Vladislavleva
University of Antwerp
Belgium
katya@vanillamodeling.com

Matt Burland
Givaudan Flavors Corp.
Cincinnati, OH
matt.burland@givaudan.com

Jason Parcon
Givaudan Flavors Corp.
Cincinnati, OH
jason.parcon@givaudan.com

Una-May O'Reilly
Massachusetts Institute of Technology
Cambridge, MA
unamay@csail.mit.edu

ABSTRACT

We have acquired panelist data that provides hedonic (liking) ratings for a set of 40 flavors each composed of the same 7 ingredients at different concentration levels. Our goal is to use this data to predict other flavors, composed of the same ingredients in new combinations, which the panelist will like. We describe how we first employ Pareto-Genetic Programming (GP) to generate a surrogate for the human panelist from the 40 observations. This surrogate, in fact an ensemble of GP symbolic regression models, can predict liking scores for flavors outside the observations and provide a confidence in the prediction. We then employ a multi-objective particle swarm optimization (MOPSO) to design a *well* and *consistently* liked flavor suite for a panelist. The MOPSO identifies flavors that are well liked (high liking score), and consistently-liked (of maximum confidence). Further, we generate flavors that are well and consistently liked by a cluster of panelists, by giving the MOPSO slightly different objectives.

Categories and Subject Descriptors

I.1.2 [**Computing Methodologies**]: Artificial Intelligence—*Problem Solving, Control Methods, and Search*

General Terms

Algorithms, Design, Experimentation

Keywords

sensory evaluation, genetic programming, ensemble learning, particle swarm optimization

GECCO'10, July 7–11, 2010, Portland, Oregon, USA.

1. INTRODUCTION

Givaudan Flavors Corporation is a world renowned fragrance and flavor company that employs a large team of flavorists to design flavors for clients in the food industry. Its flavor design process can be long and might involve many experts from diverse specialities such as sensory science and food chemistry. They have extensive expertise in understanding how particular ingredients relate to hedonic response, how multiple ingredients interact, and the complexity of flavor composition. To figure out the ingredient driving hedonic response (positively or negatively) in a mixture of ingredients, Givaudan uses designed experiments to define a small set of mixtures that are (see Figure 1) evaluated by a panel of consumers. Each panelist is asked how much they like the flavor, ranging from "like extremely" to "dislike extremely" with 9 distinctions. The responses and their corresponding numeric scores in the dataset are shown in Table 1.

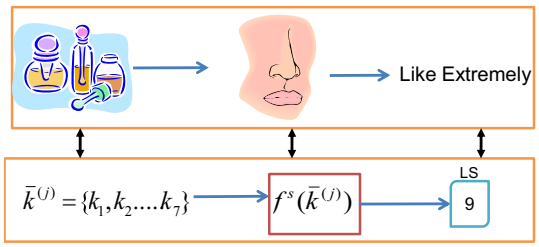


Figure 1: Hedonic evaluation system for flavors.

In this contribution, our goal is to use the data of this consumer evaluation study and attempt to optimally determine a set of flavors any panelist (or group of panelists with common flavor preferences) would highly like. Technically there are two broad challenges. First, there are only 40 flavor samples while a flavor has 7 dimensions (each a different ingredient) with significant range in concentration levels of each ingredient (some range between [0 200], while others range between [0-20]). In this respect the data is sparse. Secondly, the sampled flavors are chosen via design of experiment methodology so they are regularly spaced (al-

though sparsely) at extreme and center points of the ingredient ranges of principally important ingredients. However, we are advised by Givaudan that very interesting non-linear variable relationships could exist in between the samples due to non-linear ingredient interactions and ingredient-concentration-hedonic response relations. This information indicates that, as we proceed, we should always recognize the inherent uncertainty when interpolating from the observations.

Table 1: Category anchoring of the 9 point hedonic scale

Panelist Hedonic Rating	Liking Score
Like Extremely	9
Like Very Much	8
Like Moderately	7
Like Slightly	6
Neither Like Nor Dislike	5
Dislike Slightly	4
Dislike Moderately	3
Dislike Very Much	2
Dislike Extremely	1

We proceed as follows: Section 2 presents related work. Section 3 overviews the two-step process to find consistently well liked flavors. Section 4 covers the first step by describing model ensemble derivation using ParetoGP and our methods to derive a prediction and the model's confidence in it for un-observed flavors. Section 5 presents the multi objective particle swarm optimization algorithm (MOPSO) that is used to optimize the flavors by searching through the key volumes. Section 6 presents and analyzes experimental results. Section 7 summarizes and mentions future work.

2. RELATED WORK

The field of sensory science is broad and encompasses psychophysics, food chemistry and consumer psychology. See [11] for a modest overview of sensory evaluation practices. Within sensory science, the study of hedonic response is actively pursued because of its commercial and societal implications. Hedonic study has protocol issues: how to identify a reliable panelist, how to assemble a panel, what questions to pose and what responses should be admitted [2, 7]. Conventional data analysis uses multivariate techniques [3]. Some research in hedonics has focused on inter panelist differences, e.g. [9]. One recent study involving machine learning and sensory evaluation studied the reliability of wine flavor categorization by experts based upon aroma [4].

3. FLAVOR OPTIMIZATION OVERVIEW

We proceed in two steps, using notation defined in Table 2 and according to Figure 2: Our first step, (see Section 4) is to model the panelist in terms of a liking score function of 7 ingredient concentration levels. This function must be capable of expressing a non-linear relationship between liking score and ingredient levels. In consideration of this and the inherent uncertainty between observation points, we use ParetoGP[10, 5, 12]. ParetoGP is a robust genetic programming symbolic regression technique that generates an ensemble of multiple models aggregated from multiple runs. Symbolic regression is a non-parametric modeling technique without restrictions on variable correlation. ParetoGP's model ensemble provides collectively diverse and plausible explanations of a response. Each model is sufficiently accurate with respect to its observations but differs from the others with respect to its responses on unobserved points. As in conventional GP ParetoGP's models do not presume any model structure *a priori* and are selected to be robust to overfitting. Their structure and generality arises from evolutionary interaction between accuracy and complexity objectives. We provide technical details of ParetoGP in Section 4.

Our intent is to use this ensemble as a surrogate for the panelist in Step 2. When we want to "know" how much the panelist would like a (unobserved) flavor, we feed the flavor's ingredient levels into each model of the ensemble and receive a prediction. The issue then is how to fuse the predictions from the multiple models. In fact, "fusing" (or combining) the multiple predictions from an ensemble is an open area of research. In this contribution we choose a median-average method for its robustness (see [12]), also defined in section 4.1. We additionally exploit the ensemble to derive another piece of valuable information: model confidence, expressed locally, for a specific prediction. We experiment with two confidence measures: inter-decile range and entropy. These are explained in detail in Sections 4.1 and 4.2.

In our second step, detailed in Section 5, we use a multi-objective particle swarm optimization (MOPSO) algorithm to find a Pareto front of flavors that are both highly and consistently liked according to the model ensemble of the panelist. The objectives of the MOPSO are to maximize liking (as a score) while minimizing the uncertainty of the prediction. When a group of panelists is considered for flavor optimization, modeling will generate a set of model ensembles. The fused predictions from these ensembles must themselves be combined. In this case, we combine by slight modification of the objectives. The first objective expresses the goal of maximizing the mean of panelists' liking scores weighted by prediction confidence. The second objective is based on a measure we call consistency which is intended to assure minimum variance among the panelists' liking scores. More details are provided in Section 5.2.

4. ENSEMBLE MODELING

We wish to first derive multiple liking score models based on ingredient concentration levels, then specify a means of combining their predictions and providing a confidence measure. For this we use ParetoGP. ParetoGP iterates over 3 steps for a user specified time duration after initialization.

Initialization consists of generating symbolic regression models in the form of executable expressions (represented as parse trees) using a set of basic functions or operators that have one or two arguments. We use $\{+, -, /, *, log, exp, power\}$ as functions here in addition to the explanatory variables for our problem, $\{k_1, k_2, k_3, \ldots, k_7\}$ which are ingredient concentration levels that in the modeling context we call "keys". **[Step 1]** Model evaluation: The models are evaluated under two criteria, (a) accuracy and (b) complexity. A correlation metric between the model prediction $\widehat{LS}$ and the actual values of the LS for the known set of flavors, F_o, is defined as

$$R(\widehat{LS}, LS) = \frac{cov(\widehat{LS}, LS)}{std(\widehat{LS})std(LS)} \quad (1)$$

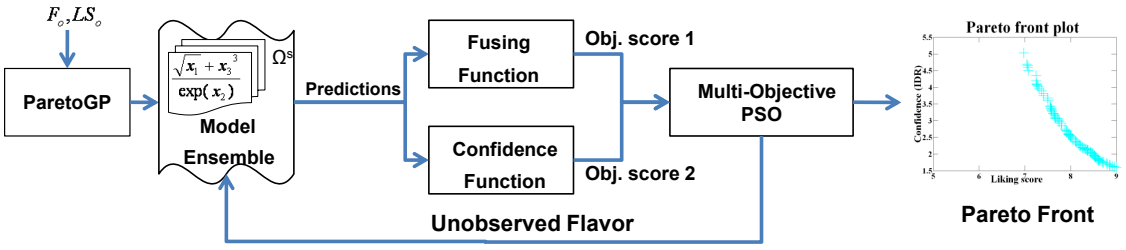


Figure 2: Evolutionary optimization of flavors for a panelist.

Variable	Notation	Details
Observed Flavor space	F_o	The flavors from the dataset
Ingredients or Keys	k_i	$i \in \{1...7\}$
Flavor	$\overline{k}$	A mixture of 7 ingredients, $\overline{k} = \{k_1, ...k_7\}$
Panelist	s_n	$n \in \{1..69\}$
Liking score function	$f^s(\overline{k}^{(j)}) = LS$	Relationship between a $\overline{k}^{(j)}$ and LS
Model	m	Model m for Panelist s
Prediction	$y^{s,m,j}$	Model m's prediction for a $\overline{k}^{(j)}$
Model Ensemble	Ω^s	All models in the ensemble
Prediction Set	$\overline{Y}^s$	$\overline{Y}^s = \forall m \in \Omega^s \ \{y^{s,m,j}\}$

Table 2: Problem Notation

Using (1) for scaled LS and $\widehat{LS}$ we define error as

$$E = 1 - R^2 \tag{2}$$

The second evaluation metric for a model is its complexity. Complexity of a model is evaluated as the number of nodes in all the subtrees of a given tree.

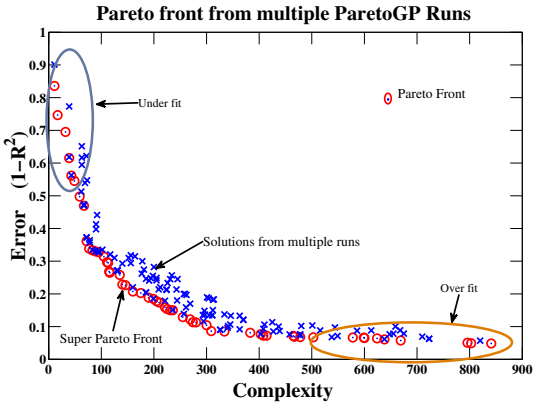


Figure 3: A "Super" Pareto front aggregating multiple ParetoGP runs

[Step 2] Model selection and archiving: When no *a priori* information about the problem is known, ParetoGP fosters a trade-off between model complexity and model accuracy to prevent premature and inaccurate decisions about the model. The accuracy and complexity metrics are used as two minimization objectives. ParetoGP evolves a set of solutions approximating the true Pareto front, which is the set of optimal trade-off (a.k.a. non-dominated) points in the two objective space. It employs the population and an archive to robustly identify the front.

[Step 3] Model update: Parents are drawn from the archive and population with tournament selection that uses rank of the front as its comparative measure. Ties are broken randomly. The current population is updated by conventional subtree mutation and crossover of the selected parents [6] .

ParetoGP aggregates and sorts each run's Pareto front into a "super" Pareto front and other non-dominated solutions. These solutions for 1 panelist are shown in Figure 3 which also indicates the regions where the models can be considered to *overfit* and *underfit* the data. From this set, an engineer down selects by choosing models within an accuracy and complexity range. These are further weeded to eliminate models with correlated prediction error, ρ, on a test set of inputs which consists of observations or arbitrary samples. In the following subsections, we present methods to derive prediction and confidence from these predictions.

4.1 Combining Multiple Predictions

Considering the predictions generated by the ensemble, $\overline{Y}^s$, our next goal is to fuse them into a prediction for the liking score. We use the following method:

Median-Average The median prediction and its two neighbors in the prediction space are identified and averaged.

4.2 Confidence Measures

We present two different methods to provide a measure of confidence in our final prediction derived above.

Inter-Decile Range: This is the inverse of the difference between the 80th and 20th percentile of predictions in $\overline{Y}^s$. Given $\overline{Y}^s$ consisting of the predictions $\{y^1,..., y^m\}$, the sorted vector (in ascending order) of $\overline{Y}^s$ is given by $\{v^1, ...v^m\}$. The index for the p^{th} percentile in this new vector is given by

$$n = \frac{p}{100}(m-1) + 1, \tag{3}$$

where p is the percentile value in percentage, m is the total number of predictions. The value at this index is approximated using

$$v_p = \begin{cases} v_1, & \text{for } l = 1 \\ v_N, & \text{for } l = N \\ v_k + d(v_{l+1} - v_l), & \text{for } 1 < l < N, \end{cases} \quad (4)$$

where d is the decimal component of n and l is the integer component of n. The confidence is given by

$$C_I(\overline{k}^{(j)}) = \frac{1}{v_{80} - v_{20}} \quad (5)$$

Entropy In this method the discrete probability mass function is formed for the liking score predictions, in $\overline{Y}^s$, that are approximated to nearest integer value, and the entropy [1] is evaluated using

$$H = -\sum_{i=1}^{9} p(a_i) log(p(a_i)) \quad (6)$$

where $p(a_i)$ is the probability mass generated from the predictions for the liking score values $a = \{1, 2, ...9\}$. Higher

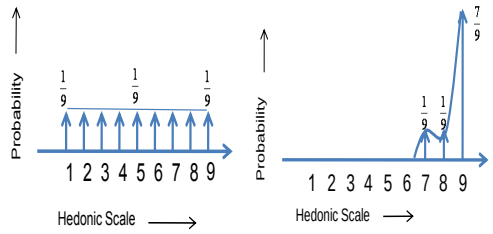


Figure 4: Discrete probability mass of predictions for cases of (a) maximum entropy - low confidence and (b) low entropy - high confidence

entropy implies higher random behavior. Figure 4 illustrates the properties of the entropy based confidence placement. It shows two different scenarios of the predictions. In the first scenario, the predictions in the $\overline{Y}^s$ are equally divided among the different values on the rating scale, giving each a probability of 1/9. In the second scenario, only three values are chosen by the predictions, i.e. 7 (1/9), 8 (1/9) and 9(7/9). The entropy is higher for the first scenario indicating random nature of the predictions. In fact the entropy is maximally bounded by this number. Higher entropy implies lower confidence. In the second scenario, due to the spiky characteristic of the discrete probability mass function, the entropy is lower and implies high confidence because many predictions converge to the same rating. Thus we define confidence as

$$C_E(\overline{k}^{(j)}) = 1 - H \quad (7)$$

5. MOPSO FLAVOR OPTIMIZATION

Using our models derived with ParetoGP, we now use a multi-objective particle swarm optimization detailed in Algorithm 1 and [8] to identify flavors that maximize the liking score function for (a) a single panelist, (b) a cluster of panelists. The algorithm uses the same velocity update and position update functions as in the classical particle swarm. The algorithm differs in the way the *pbest* is updated. In presence of multiple objectives the algorithm evolves solutions towards the Pareto front. In our problem, a candidate solution is the concentration levels of the keys (a.k.a. ingredients) that compose a flavor. Specifically the swarm is defined as $\{k_{i1}, k_{i2}....k_{i7}\}$, where subscript i represents the particle number and each particle has 7 keys.

Algorithm 1 Multi objective particle swarm optimizer for flavor optimization

(k_{i_d}): $d \Leftarrow$ key index, $i \Leftarrow$ particle index, $k \Leftarrow$ Key;
$\overline{k_i} \Leftarrow$ particle, $\overline{P_i} \Leftarrow$ *pbest*, $N \Leftarrow$ number of particles
1. Initialize the particles, $\overline{k}$ randomly in the search space with in the range for each key, $[l_i, u_i]$
2. Initialize $\overline{P_i}$ to be the same, $\forall i$
3. Initialize parameters of PSO, $\omega = 0.8$, $\psi_1 = 1$, $\psi_2 = 1$
4. Evaluate objective function (o_1) and objective function (o_2) for the given $\overline{\overline{k}}$
5. Randomly initialize the *'gbest'*, g
for $t = 1$ to maxiter **do**
  for $i = 1$ to N **do**
    for $r = 1$ to d **do**
      $V_{ir}^{(t+1)} = \omega V_{ir}^{(t)} + \psi_1(P_{ir}^{(t)} - k_{ir}^{(t)})U[0,1] + \psi_1(P_{gr}^{(t)} - k_{ir}^{(t)})U[0,1]$
      $k_{ir}^{(t+1)} = k_{ir}^{(t)} + V_{ir}^{(t+1)}$
    end for
  end for
  for $i = 1$ to number of N **do**
    Evaluate o_1 and o_2 for $\overline{k_i}$
  end for
  Update the $\overline{P}$ with the non-dominated solutions
  Identify the *'gbest'*, $g = 5 * U[0,1]$
  Store the *pbest* vector for iteration t
  $t \leftarrow t + 1$
end for
6. Output $\overline{P}^{(t)}$

5.1 Objectives for single panelist optimization

To find optimal flavors for a single panelist, our objectives are (1) maximize fused liking score prediction (2) maximize the confidence. In Section 6, we examine optimizing with the two objectives using the median average fusing method and each of the two confidence measures: inter-decile range: Case A, and entropy: Case B.

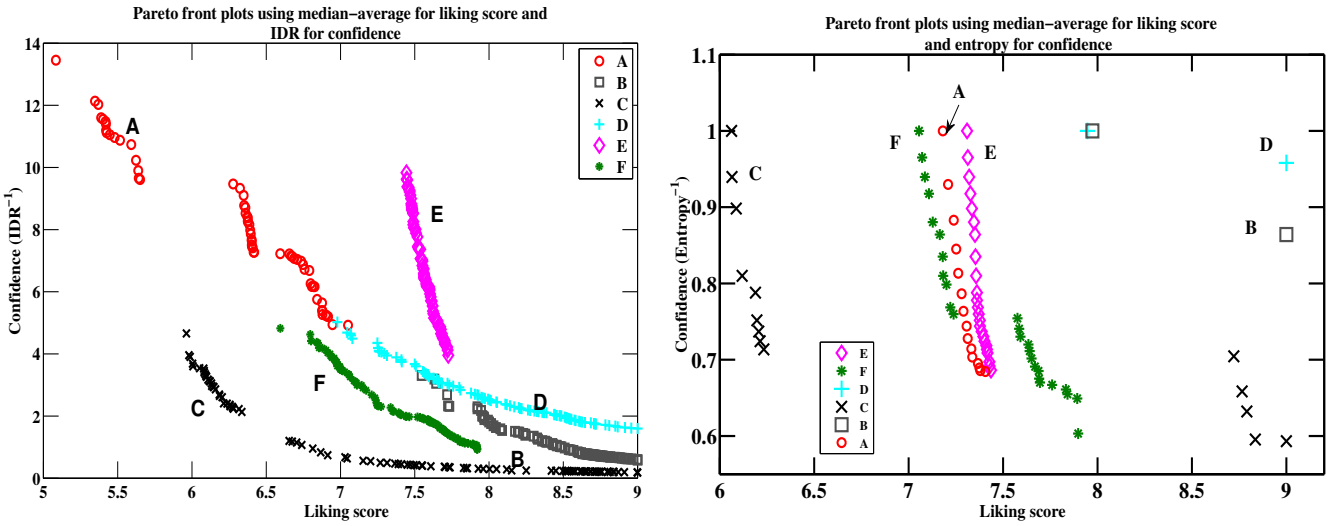
5.2 Objective design for a cluster of panelists

To identify flavors that are well-liked across the panelists, the first objective is to maximize the mean of the predicted liking scores of different panelists where each liking score prediction is weighted by its confidence:

$$LS^c = \frac{\sum_{s=1}^{N} C_s LS_s}{\sum_{s=1}^{N} C_s} \quad (8)$$

C_s is the confidence of the s^{th} panelist's prediction derived from $\overline{Y}^s$, and LS_s is the liking score derived from $\overline{Y}^s$. The confidence and liking score can be estimated using any of the two methods described in Section 4.1 and 4.2.

We define a second measure called consistency that helps in two scenarios to, (a) counter the case in which the mean is driven by a few panelists that like a particular $\overline{k}^{(j)}$, and (b)



(a) (b)

Figure 5: Flavor optimization achieved by Multi Objective particle swarm optimization (a) Pareto front plots for 6 panelists obtained for Case A; (b) Pareto front plots for 6 panelists for Case B

push the flavors to the design space where they are mostly liked. The measure is derived using the LS_s derived from the $\overline{Y}^s$. The measure is given by

$$V = \sqrt{\frac{1}{N}\sum_{s=1}^{N}(LS_s^c - \overline{LS^c})^2}, \quad (9)$$

where $\overline{LS^c}$ is the mean of the liking score prediction of multiple panelists. Equation 9 simply evaluates the standard deviation of the liking scores generated by the multiple panelists. The second objective is to minimize the variability among the liking scores defined by (9).

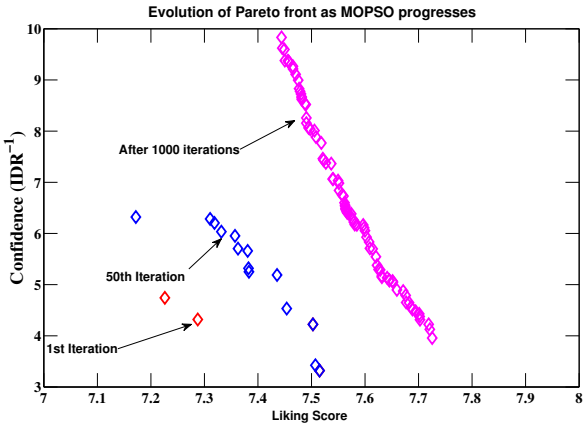


Figure 6: Progress of MOPSO as a function of iterations for the single panelist problem.

6. EXPERIMENTS

Experimental Parameters ParetoGP is run 6 times with a population size of 500 where each run executes for 600 seconds. Subtree crossover probability is 0.9 and mutation probability = 0.1. We select models from the super Pareto front and its dominated models that have accuracy better than 0.5 and complexity less than 400. We set the correlation threshold to $\rho = 0.92$. The MOPSO has a swarm size of 100 and iterates 1000 times. The cognitive and social learning rates are: $\psi_1 = 1.0$ and $\psi_2 = 1.0$. Inertia is controlled by $\omega = 0.8$.

6.1 Results and Discussion for Single Panelist

We experimented with 6 different panelists (each of whom rated the same 40 flavors) and named the panelists A, B, C, D, E, F. Table 3 gives the number of models ParetoGP identifies after run aggregation, how many of them are selected into the ensemble and how many optimized flavors lie on the Pareto front after completion of the MOPSO. The ensemble selection methods reduced the number of models by approximately 60%. The MOPSO identifies approximately 100 flavors for panelists B, C, D and F on respective Pareto fronts.

Table 3: One panelist: models and optimized flavors

Panelist	GP Models	Ensemble	Case A	Case B
A	109	35	62	15
B	100	41	100	2
C	92	41	100	14
D	145	67	100	2
E	234	85	86	24
F	223	83	100	26

Case A: Prediction by Median-average, Confidence by Inter-Decile Range Figure 5(a) shows the Pareto curves for each of the 6 panelists. Each Pareto front in this set has unique properties. For example, E has very low variance in liking score but high variance in confidence. B on the other hand has comparatively low variance in the confidence, but

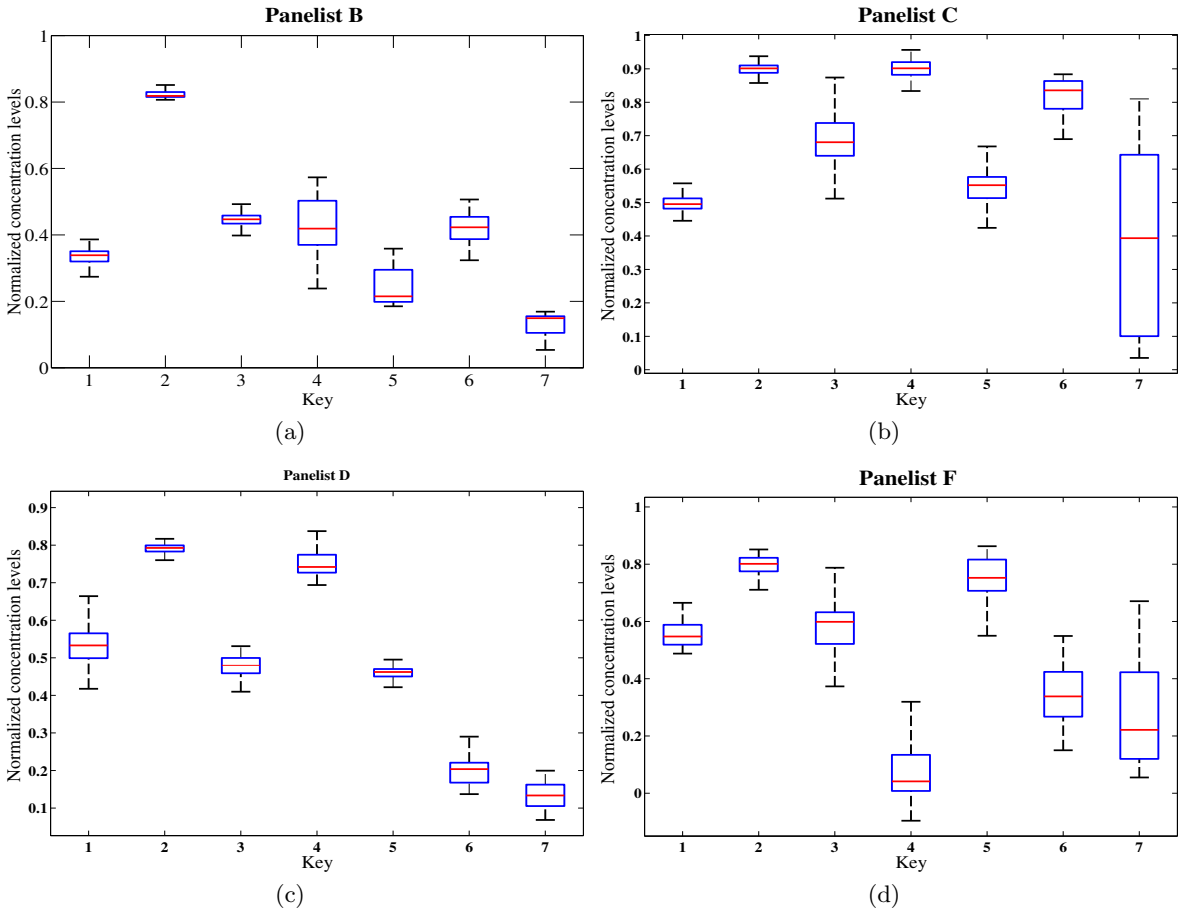


Figure 7: Variability of different key values on the pareto front solutions for panelists, B, C, D, F (a) Panelist B (b) Panelist C (c) Panelist D (d) Panelist F

spreads across the liking score range. F has a Pareto front that is spread proportionately in both confidence and liking score. D varies significantly in liking score, but also achieves a liking score of 9 with a high confidence ≈ 2.0. C performs similar to D but could only achieve the liking score of 9 with very low confidence. Flavors on A's Pareto front have the highest confidence of all, but could only achieve a maximum liking score of ≈ 7.

To analyze the range of flavors on the Pareto front, we create box plots for the 7 key concentration levels of 4 panelists (B, C, D, F) in Figure 7. In these plots, on each box, the line close to the center of the box is the median, the edges of the box are the 25th and 75th percentiles, the whiskers extend to the most extreme data points not considered outliers.

These box plots show the range of each key, within which the liking of a panelist lies. The plot also shows that different panelists differ in the key ranges where there liking lies.

Since we were attempting to maximize the liking score, for some keys, we observe that the range of their variance was very narrow, implying that panelists liking score is sensitive to this range of concentration levels. For example, for key 2 the range is between $[0.8 - 0.85]$ for all the panelists. Also for each panelist the variation for this key is very narrow indicating the sensitivity of liking score for this key.

We further observe, panelists C, F have very high variance for key 7, indicating that these panelists are less sensitive to this key's level. In other words, varying this key does not lower their liking score significantly.

Figure 6 shows the progress of MOPSO as a function of iterations for the panelist E. The higher the curve the better the solution is. One can see, that MOPSO starts with 2 solutions on the front and slowly fills up solutions with better confidence measure as iterations progress. Our goal is to approach towards the right top corner, i.e., solutions with high liking score and high confidence.

Case B: Prediction by Median-average, Confidence by Entropy Plots (a) and (b) of Figure 5 contrast results from the optimization when it employs the two different confidence measures. In Figure 5(b) the *y-axis* is, by definition, bound at 1.0. A liking score with confidence of 1 (entropy of 0) implies that the model predictions for this flavor (when rounded to nearest integer) all agreed. We can observe that this liking score is generally between 7 and 7.5 for panelists E, A, F, at 8 for B and D, and at 6 for C. Notice that the range of solutions that appear in Case A with liking score less than 6 but confidence of $[6 \ldots 12]$ do not appear in this case . This is because a lower interdecile range is equivalent to lower entropy (≈ 0) and such points are dominated by the points that have higher liking score but equal entropy, i.e. 0. Thus this method bounds the pareto front plots to higher

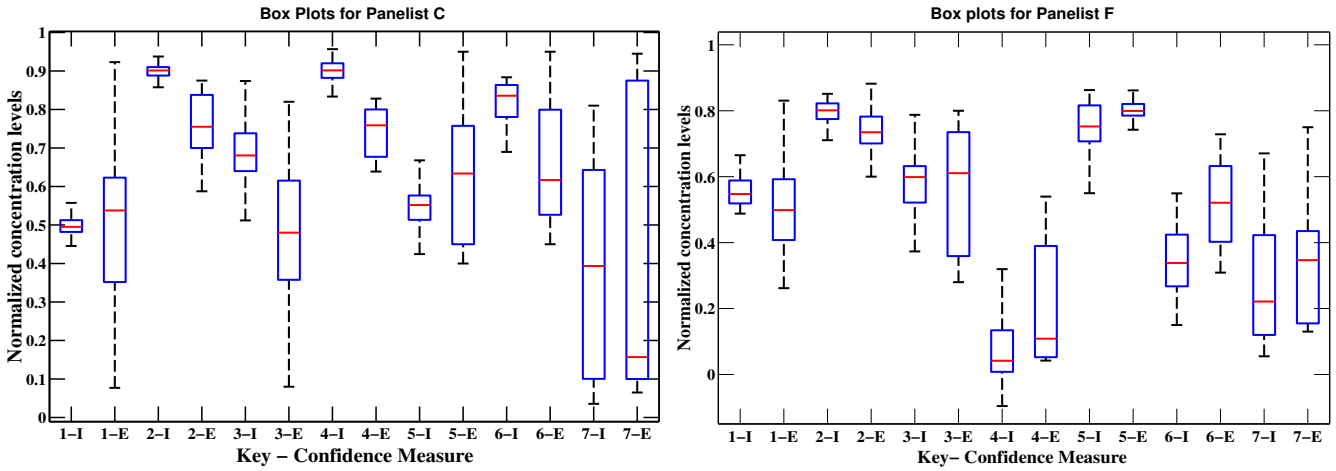


Figure 8: Comparison of variability of normalized key values on the pareto front solutions for panelists, C and F for the Cases A and B. I- Inter-Decile Range, E- Entropy. (a) Panelist C (b) Panelist F

range of liking scores. We examined the key variance of C and F. We noticed higher variance in the keys compared to the previous case. This is shown in Figure 8.

Discussion One is tempted to seek a conclusion from these experiments as to whether expressing confidence with inter-decile range or entropy is preferable. We believe, in general, this must be determined on an application basis when the results are passed to a human engineer. In our context of flavor optimization, an expert flavorist wishes to inspect many different flavors and gain insight by comparing them while considering ingredient hedonic knowledge and integrating knowledge of other flavor-oriented factors such as cost and chemical robustness during food preparation, etc. Thus, aggregating the results of Cases A and B and identifying all significantly distinguishable flavors is desirable. For this identification we provide Figure 8. For all flavors on each case's Pareto front, it provides box plots of normalized concentration level for each key. We observe that using confidence as measured by entropy provides more diverse exploration for all keys. Figure 11 visualizes the absolute distance (using normalized key differences) between the 5 flavors with highest liking score identified by the Pareto front of each case. Each entry shows this distance value and, it shows a bar with progressively increasing grey scale for each entry. We observe that the entropy flavors are very similar to each other while the IDR flavors differ slightly. We observe that the flavors identified in Case A are different (by this metric) than those of Case B.

6.2 Panelist Cluster Results and Discussion

We experimented with 1 cluster of panelists selected for similar flavor preferences. In this experiment each ensemble computes confidence with the entropy measure. Figure 9 shows the iterative progress of the MOPSO. In the first few iterations there are only a few solutions on the Pareto front. The front then moves up and rightward and the algorithm identifies flavors with higher liking scores,$(6 \ldots 6.5)$, that are lower in variance before finally identifying flavors with confidence-weighted liking scores ranging from $[5.5 \ldots 7]$ which have low variance (higher inverse of variance). Figure 10 shows the box plots for the flavors on the Pareto front. The plots demonstrate the variance in key values among these flavors is very low for all the keys, although the liking score varies from $[5.8 \ldots 7.0]$.

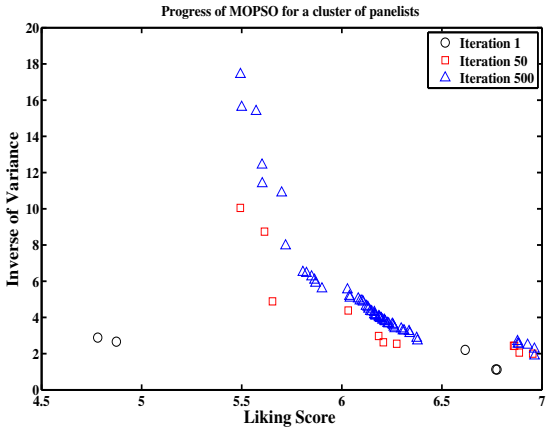


Figure 9: MOPSO progress for cluster optimization

Discussion While a liking score of 6 and above is only in the "slightly-like" and "moderately-like" hedonic range, it is understandable that more highly liked flavors cannot be identified because multiple panelists remain "individual" even when they are clustered by similar flavor preference. This is still consistent with the expectation that similar flavors will have a range of liking scores because of the panelists' preferential similarities. It appears that limiting the concentration to a certain range for all the keys is necessary to arrive at a reasonable consensus. Overall the results reflect the challenges in working with hedonic-based panel aggregates.

7. SUMMARY AND FUTURE WORK

In this paper, we presented an approach to model a liking score function for consumers employed for sensory evalua-

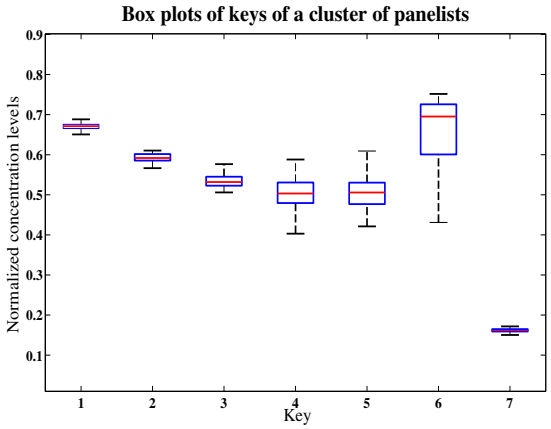


Figure 10: Variability of key values for the flavors on the Pareto front for the cluster of panelists.

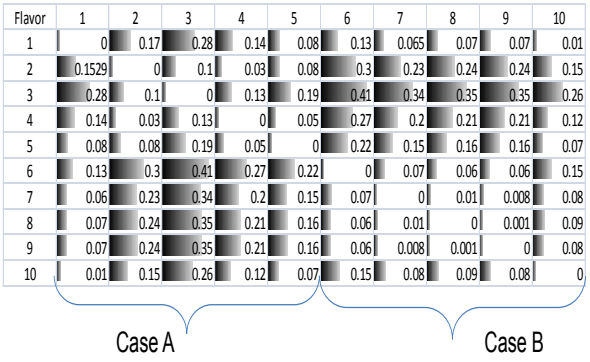


Figure 11: A map of the normalized absolute distances between 10 flavors. 5 from Case A and and 5 from Case B. Each entry is the distance between flavor i and flavor j. The length of the gray bar is indicative of the distance. The diagonal is zero distance

tion. A GP based symbolic regression methodology was used to generate all plausible models that explain the given data. From these plausible models, a subset was selected that best represent the liking score function for the human expert. We thus formed a bank of liking score functions, that are diverse and that explain as many as possible behavior that the human expert can exhibit when a previously unseen flavor is presented. We then integrated these banks of liking score functions per panelist into two multi objective algorithms that identify the flavors that maximize (a) a panelist's liking score, or (b) a group of panelists' liking scores. We used a particle swarm optimization algorithm for this. We observed that for an individual panelist, the method identified flavors with key concentration levels in narrow ranges relative to input ranges. These flavors also differed from panelist to panelist even though they all have predicted liking scores between $[5.5 \ldots 8]$. For multiple panelists grouped together, we get flavors with very minute changes in the key ranges.

We plan to work further on distinguishing differences among optimized flavors. If flavor difference is considered from the perspective of perception rather than concentration level, difference is very complex. Often a concentration level is perceptually invisible until a threshold is reached. Then perception is non-linear and it finally "flattens" at a perceptual saturation threshold. This work will be integrated with our related studies in aggregating panelists by preference of flavors and by propensity to express specific ranges of liking scores.

8. REFERENCES

[1] A. Antos and I. Kontoyiannis. Estimating the entropy of discrete distributions. In *Information Theory, 2001. Proceedings. 2001 IEEE International Symposium on*, pages 45–, 2001.

[2] Committee on Sensory Evaluation of Materials and Products. *Guidelines for the Selection and Training of Sensory Panel Members, ASTM STP 758*. American Society for Testing and Materials, 1981.

[3] G. B. Dijksterhuis. *Multivariate Data Analysis in Sensory and Consumer Science*. Wiley-Blackwell, USA, 1997.

[4] J. G. Ferrier and D. E. Block. Neural-network-assisted optimization of wine blending based on sensory analysis. In *American Journal of Enology and Viticulture*, volume 52, pages 386–395. American society for Enology and Viticulture, 2001.

[5] M. Kotanchek, G. Smits, and E. Vladislavleva. Trustable symoblic regression models. In R. L. Riolo, T. Soule, and B. Worzel, editors, *Genetic Programming Theory and Practice V*, Genetic and Evolutionary Computation, chapter 12, pages 203–222. Springer, Ann Arbor, 17-19May 2007.

[6] J. R. Koza. *Genetic Programming: On the Programming of Computers by Means of Natural Selection*. MIT Press, Cambridge, MA, USA, 1992.

[7] J. Kuznicki, R. Johnson, and A. Rutkiewic, editors. *Selected Sensory Methods: Problems and Approaches to Measuring Hedonics, ASTM STP 773*. American Society for Testing and Materials, 1982.

[8] X. Li. A non-dominated sorting particle swarm optimizer for multiobjective optimization. In *Lecture Notes in Computer Science*, volume 2723/2003, page 198. Springer, 2003.

[9] H. R. Moskowitz and R. Bernstein. Variability in hedonics: Indications of world-wide sensory and cognitive preference segmentation. *Journal of Sensory Studies*, 15(3):263–284, 2000.

[10] G. Smits and M. Kotanchek. Pareto-front exploitation in symbolic regression. In U.-M. O'Reilly, T. Yu, R. L. Riolo, and B. Worzel, editors, *Genetic Programming Theory and Practice II*, chapter 17. Springer, Ann Arbor, 13-15 May 2004.

[11] H. Stone and J. L. Sidel. *Sensory evaluation practices*. Academic Press, New Jersey, USA, 2004.

[12] E. Vladislavleva. *Model-based Problem Solving through Symbolic Regression via Pareto Genetic Programming*. PhD thesis, Tilburg University, Tilburg, the Netherlands, Aug. 2008.