Learning a Lot from Only a Little: Genetic Programming for Panel Segmentation on Sparse Sensory Evaluation Data

Katya Vladislavleva[1], Kalyan Veeramachaneni[2], Una-May O'Reilly[2],
Matt Burland[3], and Jason Parcon[3]

[1] University of Antwerp, Belgium, katya@vanillamodeling.com
[2] Massachusetts Institute of Technology, USA, kalyan,unamay@csail.mit.edu
[3] Givaudan Flavors Corp., USA, matt.burland,jason.parcon@givaudan.com

Abstract. We describe a data mining framework that derives panelist information from sparse flavour survey data. One component of the framework executes genetic programming ensemble based symbolic regression. Its evolved models for each panelist provide a second component with all plausible and uncorrelated explanations of how a panelist rates flavours. The second component bootstraps the data using an ensemble selected from the evolved models, forms a probability density function for each panelist and clusters the panelists into segments that are *easy to please*, *neutral*, and *hard to please.*

Key words: symbolic regression, panel segmentation, survey data, ensemble modeling, hedonic, sensory evaluation

1 Introduction

Givaudan Flavours, a leading fragrance and flavour corporation, is currently trying to integrate evolutionary computation techniques into its design of flavours. In one step of its design process, Givaudan conducts a hedonic survey which presents aromas of flavours to a small panel of targeted consumers and queries how much each flavour is liked. Each panelist is asked to sniff roughly 50 flavours.

To best exploit the restricted sample size, Givaudan flavourists first reduce the ingredients they experimentally vary in the flavours to the most important ones. Then they use experimental design to define a set that statistically provides them with the most information about responses to the entire design space.

The specificity of sensory evaluation data is such, that "the panelist to panelist differences are simply too great to ignore as just an inconvenience of the scientific quest," [1], because "taste and smell, the chemical senses, are prime examples of inter-panelist differences, especially in terms of the hedonic tone (liking/disliking)," [1]. Givaudan employs reliable statistical techniques that regress a single model from the survey data. This model describes how much the panel, as an aggregate, likes any flavour in the space. But since the differences in the liking preferences of the panelists are significant, Givaudan is also using several proprietary methods to deal with the variation in the panel and is interested in alternative techniques.

A goal of our interaction with Givaudan is to generate innovative information about the different panelists and their liking-based responses by developing techniques that will eventually help Givaudan design even better flavours. Here we describe how Genetic Programming (GP) can be used to model sensory evaluation data without suppressing the variation that comes from humans having different flavour preferences. We also describe how GP enables a knowledge mining framework, see Figure 1, that meaningfully segments (i.e. clusters) the panel. With an exemplar Givaudan dataset, we identify the panelists who are "easy to please", i.e. that frequently respond with high liking to flavours, "hard to please" and "neutral". This is, in general, challenging because the survey data is sparse. In this particular dataset there are only 40 flavours in the seven-dimensional sample set and 69 panelist responses per flavour.

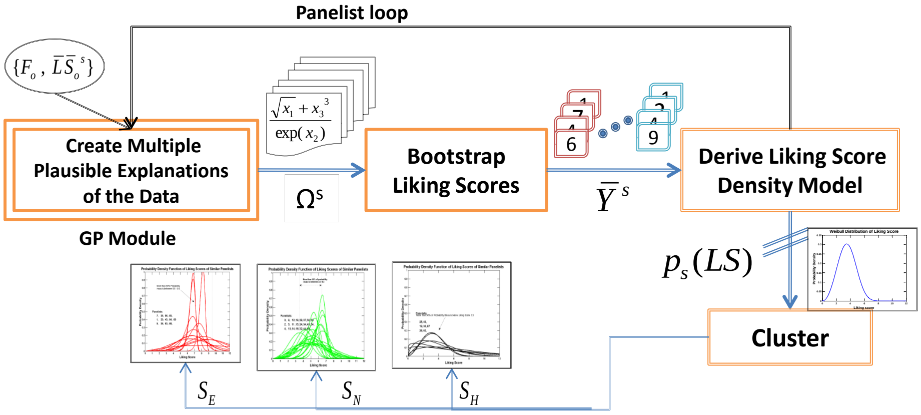
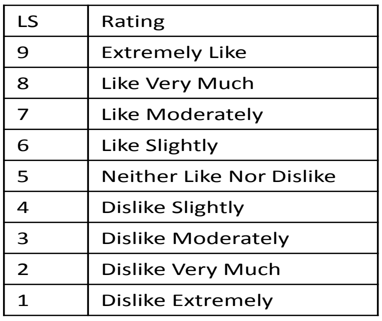

Panelist loop
$\{F_o, \overline{LS}_o^s\}$
Create Multiple Plausible Explanations of the Data
GP Module
$\frac{\sqrt{x_1} + x_3^3}{\exp(x_2)}$
Ω^s
Bootstrap Liking Scores
1 7 4 6
1 2 4 9
$\bar{Y}^s$
Derive Liking Score Density Model
$p_s(LS)$
Cluster
S_E
S_N
S_H

Fig. 1. Knowledge mining framework for sparse sensory data with a focus on panel segmentation. Read clockwise. The top portion is repeated for each panelist

We proceed as follows: Section 2 introduces our flavour-liking data set. Section 3 discusses why GP model ensembles are well suited for this problem domain and briefly cites related work. Section 4 outlines the 5 steps of our method. Section 5 describes Steps 1 and 2, the ensemble derivation starting from ParetoGP. Section 6 presents Steps 3-5 – how the probability density functions and clusters that ultimately answer the questions are derived from this ensemble, and our experimental results. Section 7 concludes and mentions future work.

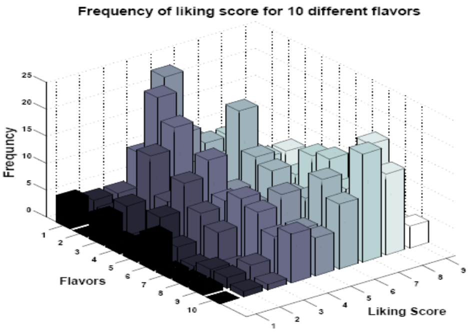
2 The Givaudan Flavour Liking Data Set

In this data set, flavour space consists of seven ingredients called *keys*, k_i. A flavour in the flavour space is a mixture by volume of these seven ingredients and the jth flavour is denoted by $\overline{k}^{(b)}$. 69 panelists sniff 40 different flavours and select a rating that is translated to its *liking score*, *LS* per Figure 2(a). Figure 2(b) illustrates the variation in the liking preferences of 69 panelists for the first ten flavours (for each flavour a histogram of 69 likling scores is depicted using 9 bins). Table 1 gives the notation for different variables used in this paper. Givaudan may pre-process these scores to adjust them for how different panelists use the range of the response scale. We scale all key data to the same range in this study.



LS	Rating
9	Extremely Like
8	Like Very Much
7	Like Moderately
6	Like Slightly
5	Neither Like Nor Dislike
4	Dislike Slightly
3	Dislike Moderately
2	Dislike Very Much
1	Dislike Extremely

(a)



(b)

Fig. 2. (a)Category anchoring of the 9 point hedonic scale (b) Liking score frequency for 10 different flavours over all 69 panelists shows preference variance.

3 Related Work

Because our data is sparse, it is not justifiable to presume that there is solely one model that explains a panelist's preferences. Thus presuming any structure for a model (which parametric regression requires) is tenuous. Model over-fitting must also be avoided. This makes the non-parametric, symbolic regression modeling capability of GP desirable. GP symbolic regression is also population-based and can be run over the data multiple times with different random seeds to generate multiple, diverse models. Complexity control and interval arithmetics can be used to mitigate data over-fitting. Symbolic regression works without *a priori* assumptions of model structure (except primitive selection).

However, with a few exceptions, GP symbolic regression has been focused on obtaining the most accurate single model from multiple runs. In Section 5, we will describe ParetoGP [2] as one means of explicitly refocusing GP symbolic regression so it generates a robust set of models. The idea of using ensembles for improved generalization of the response prediction is by far not new in regression. It has been extensively used in neural networks (e.g., [3–7]), and even more

Table 1. Problem Specific Variable Description

Variable	Notation	Details
flavour Space	F	The design space of ingredient mixtures
Keys	k_i	$i \in \{1...7\}$
flavour	$\overline{k}$	A mixture of 7 keys, $\overline{k} = \{k_1, ...k_7\}$
A specific flavour	$\overline{k}^{(b)}$	A specific flavour denoted by superscript b
Panelist	s_n	$n \in \{1..69\}$
Set of Panelists	S	$S = \{s_1, s_2, ...s_{69}\}$
Observed flavours	F_o	$F_o = \{\overline{k}^{(1)}....\overline{k}^{(40)}\}$
Bootstrapped flavours	F_B	$F_B = \{\overline{k}^{(1)}......\overline{k}^{(10,000)}\}$
Likability Function	$f^s(\overline{k}^{(j)}) = LS$	Relationship between a $\overline{k}^{(b)}$ and LS
lsd	$p(LS \mid s)$	Liking score density function for a panelist s
Cumulative density	$P_x(LS \geq x \mid s)$	Probability of Liking score $\geq x$
Panelist Cluster	S_c	A subset of S, $c \in \{E, N, H\}$
Model	m	Model m for Panelist s
Prediction	$y^{s,b,m}$	Model m's prediction for a $\overline{k}^{(b)}$
Model Ensemble	Ω^s	All models in the ensemble
Prediction Set	$\overline{Y}^{s,b}$	$\overline{Y}^{s,b}= \forall m \in \Omega^s \ \{y^{s,b,m}\}$
Set of Liking Scores s	$\overline{Y}^{s}$	$\overline{Y}^{s}= \forall b \in F_B \ \{\overline{Y}^{s,b}\}$

extensively in boosting and machine learning in general (albeit, mostly for classification). See [8–14] for examples. [7] presented the idea of using disagreement of ensemble models for quantifying the ambiguity of ensemble prediction for neural networks, but the approach has not been adapted to symbolic regression.

4 Panel Data Mining Steps

Our GP ensemble-based "knowledge mining" method has five steps:

1. Generate a diverse model set for each panelist from the sparse samples.
2. Thoughtfully select an *ensemble* of models meeting accuracy and complexity limits to admit generalization and avoid overfitting and a correlation threshold to avoid redundancy.
3. Use *all* models of the ensemble to generate multiple predictions for many unseen inputs.
4. With minor trimming of the extremes and attention to the discrete nature of liking scores, fit the predictions to a Weibull distribution.
5. Cluster based on the Weibull distribution's probability mass.

It is significant to note that these steps respect the importance of avoiding premature elimination of any plausible information because the data is sparse. The ensemble provides all valid values of the random variable when it is presented with new inputs. This extracts maximum possible information about the random variable, which supports more robust density estimation.

We proceed in Section 5 to detail how we assemble a symbolic regression ensemble, i.e. Steps 1 and 2. In Section 6, we detail Steps 3 through 5.

5 A Symbolic Regression Ensemble

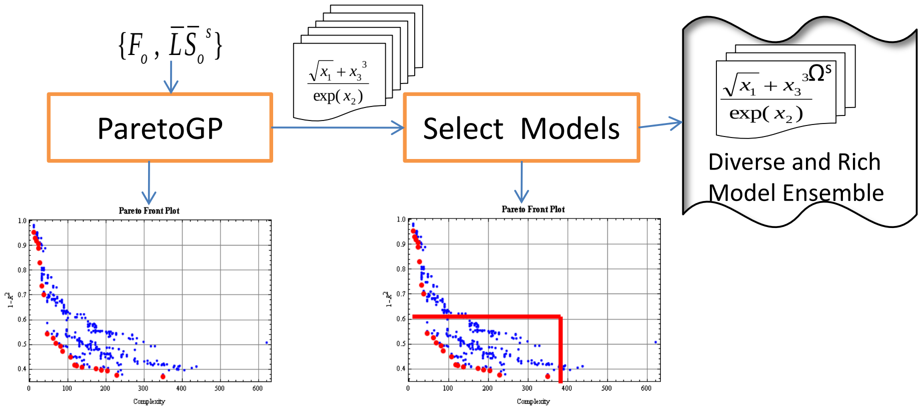

$\{F_0, \overline{LS}_0^{\,s}\}$
ParetoGP
$\frac{\sqrt{x_1+x_3^3}}{\exp(x_2)}$
Select Models
$\frac{\sqrt{x_1+x_3^3}}{\exp(x_2)}\Omega^S$
Diverse and Rich Model Ensemble
Pareto Front Plot
$1-R^2$
Complexity
1.0
0.9
0.8
0.7
0.6
0.5
0.4
0
100
200
300
400
500
600
Pareto Front Plot
$1-R^2$
Complexity

Fig. 3. Ensemble based symbolic regression

Traditionally symbolic regression has been designed for generating a single model. Researchers have focused on evolving *the* model that best approximates the data and identifies hidden relationships between variables. They have developed multiple competent approaches to over-fitting. There are a number of demonstrably effective procedures for selecting the final model from the GP system. Machine learning techniques such as cross validation and bagging have been integrated. Multiple ways of controlling expression complexity are effective. See [15] for a thorough justification of the above assessment.

Modelers who must provide all and any explanations for the data are not well served by this emphasis upon a single model. Any algorithm variation of symbolic regression, even one that proceeds with attention to avoiding over-fitting, is as fragile as a parametric model with respect to the accuracy of its predictions and the confidence it places in those predictions *if it outputs one model*. The risks are maximal when the best-of-the-run model is selected from the GP system as the solution. Our opinion is supported by the evidence in [16] which shows that symbolic regression performed with complexity control, interval arithmetic, and linear scaling still produces over-fitted best-error-of-the-run models that frequently have extrapolation pathologies.

Symbolic regression can handle dependent and correlated variables and automatically perform feature selection. It is capable of producing hundreds of candidate models that explain sparse data via diverse mathematical structure and parameters. But the combined information of these multiple models has been conventionally ignored. In our framework, we exploit rather than ignore them. During a typical run, GP symbolic regression explores numerous models. We capture the combined explanatory content of fitness-selected models, and pool *as many* explanations as we can from whatever *little data* we have.

An explicit implementation of this strategy, such as ParetoGP, must embed operators and evaluation methods into the GP algorithm to specifically aggregate a rich model set after combining multiple runs. The set will support deriving an ensemble of high-quality but diverse models. Within an ensemble, each model must approximate *all* training data samples well – *high quality.* As an ensemble, the models must collectively diverge in their predictions on unobserved data samples –*diverse.* If a GP symbolic regression system can yield a sufficient quantity of "strong learners" as its solution set, all of them can and should be used to determine both a prediction, and the ensemble disagreement (lack of confidence) at any arbitrary point of the original variable space. In contrast to boosting methods that are intended to improve the prediction accuracy through a combination of weak learners into an ensemble, this ensemble derivation process has the intent of improving prediction robustness and estimating reliability of predictions.

5.1 Model set Generation

All experiments of this paper use the ParetoGP algorithm which has been specifically designed to meet the goals of ensemble modeling. Any other GP system designed for the same goals would suffice. ParetoGP consists of the tree-based GP with multi-objective model selection optimizing the trade-off between a model's training error and expressional complexity; an elite-preservation strategy (also known as archiving), interval arithmetic, linear scaling and Pareto tournaments for selecting crossover pairs. In each iteration of the algorithm, it tries to closely approximate the (true) Pareto curve trade-offs between accuracy and complexity. It supports a practical rule-of-thumb: "use as many independent GP runs as the computational budget allows", by providing an interface where only the budget has to be stated to control the length of a run. It also has explicit diversity preservation mechanisms and efficiently supports a sufficiently large population size. The training error used in experiments is $1 - R^2$, where R is a correlation coefficient of the scaled model prediction and the scaled observed response. The expressional complexity of models is defined as the total sum of nodes in all subtrees of the tree-based model genome. The following primitives are used for gp trees of maximal arity of four: $\{+, -, *, /, inverse, square, exp, ln\}$. Variables $x_1 - x_7$ corresponding to seven keys and constants from the range $[-5, 5]$ are used as terminals. ParetoGP is executed for 6 independent runs per panelist data before the models from runs are aggregated and combined. The population size equals 500, the archive size is 100. Crossover rate is 0.9, and sub-tree mutation rate is 0.1. ParetoGP collects all models on the Pareto front of each run and for information purposes identifies a "super" Pareto front from among them. All models move forward to ensemble selection. We now have to make a decision about which models will be used to form an ensemble.

5.2 Ensemble Model Selection

In [17], the authors describe an approach to selecting the models which form an ensemble: collect models that differ according to complexity, prediction accuracy and specific predictions. Complexity can be measured by examining some quantity associated with the GP expression tree or by considering how non-linear the expression is. Accuracy is the conventional error measure between actual and predicted observations. Specific predictions are considered to assess correlations and eliminate correlated models. Generally, each ensemble combines:

- A "box" of non-dominated and dominated models in the dual objective space of model prediction error and model complexity.
- A set of models with uncorrelated prediction errors on a designated test set of inputs. Here a model is selected based on a metric which expresses how its error vector correlates with other models' error vector. The correlation must not exceed a value of ρ. The input samples used to compute prediction errors can belong to the training set, test set (if available), or be arbitrarily sampled from the observed region.

The actual ρ and box thresholds for the ensemble selection depend on the problem domain's goals. For this knowledge mining framework, where the next step is to model a probability density function of a liking score, all plausible explanations of the data are desired to acknowledge the variation we expect to see in human preferences. The box thresholds are $accuracy = 0.5$ and $expressional\ complexity < 400$. This generates models with sufficient generality (since we allowed accuracy as low as 0.5) and restricts any models with unreasonably high complexity with no obvious improvement in accuracy. We chose a value of $\rho = 0.92$ to weed out correlated models. A set of models selected after applying the criteria above is called the *ensemble*, Ω^s.

6 Modeling a Panelist's Propensity to Like

With methods that support refocusing GP based symbolic regression to derive a rich and diverse set of models and the methods [17] that select an ensemble, our GP system becomes a competent cornerstone in our knowledge mining framework. The framework can next use the ensemble, Ω^s designed for a panelist s to answer the question: "How likely is a panelist to answer with a liking score/rating higher than X?". The answer to this question allows us to categorize panelists as: (1) Easy to Please, (2) Hard to Please, (3) Neutral. We accomplish this by modeling the probability density function given by $p(LS|s)$ for a panelist s. To describe our methodology, we rely upon the notations in Table 1.

Density estimation poses a critical challenge in machine learning, especially with sparse data. Even if we assume that we have finite support for the density function and it is discrete, i.e. $LS = \{1, 2, ...8, 9\}$, we need sample sizes of the order of *"supra-polynomial"* in the cardinality of support [18]. In addition, if the decision variables are inter-dependent, as they are here, estimating a conditional distribution increases the computational complexity. Most of the research in density estimation has focused on identifying non-parametric methods to estimate distribution of the data. Research on estimation of density from very small sample sizes is limited [18, 19].

Figure 4 presents the steps taken to form this liking score probability density model. We first generate 10,000 untested flavours We use the model ensemble Ω^s, which gives us a set of predictions $\overline{Y}^{s,b}$.For each untested flavour we get a set of predictions (*not just one*), which plausibly represents all possible liking scores the panelist would give. We use these to construct the *lsd*, liking score density function, for an individual panelist.

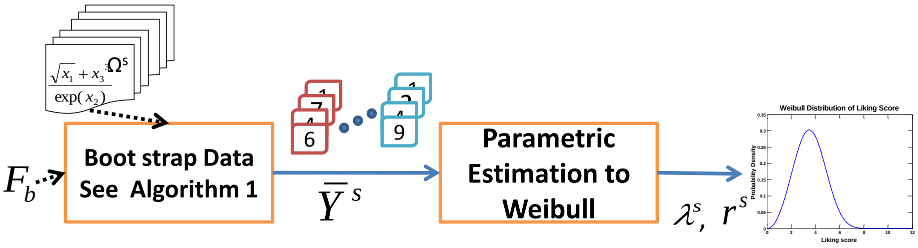


Fig. 4. Bootstrapping the Data and Deriving the Liking Score Probability Density Model

6.1 Deriving Predictions by Bootstrapping the Data

To generate the bootstrapped data of liking scores for the $F_B = \{\overline{k}^{(1)}......\overline{k}^{(10,000)}\}$ we follow the steps described in Algorithm 1.

Algorithm 1 Bootstrapping the LS data for Panelist s

```
Generate 10,000 flavours randomly, i.e., $F_b = \{\overline{k}^1....\overline{k}^{10,000}\}$ (we use a fixed uniform lattice in the experiments, same for all panelists)
for ($\overline{k}^b \in F_b\ \forall b$) do
    (i)Collect all the predictions from Model Ensemble, $\Omega^s$: $\overline{Y}^{s,b}$
    (ii) Sort the vector $\overline{Y}^{s,b}$
    (iii) Remove the bottom and top 10% of $\overline{Y}^{s,b}$ and call this vector $\overline{R}^{s,j}$
    (iv) Append $\overline{R}^{s,j}$ to $\overline{Y}^s$
end for
Fit the $\overline{Y}^s$ to a Weibull distribution. See Section 6.2
```

6.2 Parametric Estimation of the Liking Score Density Function

We use a parametric Weibull distribution to estimate $p(LS|s)$. The two parameters for the Weibull distribution, λ and r are called scale and shape respectively. A Weibull distribution is an adaptive distribution that can be made equivalent to an Exponential, Gaussian or Rayleigh distributions as its shape and scale parameters are varied. For our problem this is a helpful capability as a panelist's liking score follows any one of the three distributions. The derived Weibull distribution is:

$$p(LS; \lambda, r|s) = \begin{cases} \frac{r}{\lambda}(\frac{LS}{\lambda})^{r-1}e^{-(\frac{LS}{\lambda})^r} & \text{if } LS \geq 0 \\ 0 & \text{if } LS < 0. \end{cases} \tag{1}$$

In addition to steps taken in Section 6.1, we map the bootstrapped data to a range of the support of Weibull and the hedonic rating scale i.e., $[1, 9]$. There are some predictions in the $\overline{Y}^s$ which are below 1 or are above 9. We remove 80% of these predictions as outliers. We assign a liking score of 1 for the remaining 20% of predictions that are less than '1' in the prediction set. We similarly assign the liking score of '9' for the ones that are above 9. We use these 20% in $\overline{Y}^s$ to capture the scores corresponding to the *"extremely dislike"* and *"extremely like"* condition. Each plot line of Figures 6 (b), (c) and (d) is a *lsd*.

6.3 Clustering Panelists by Propensity to Like

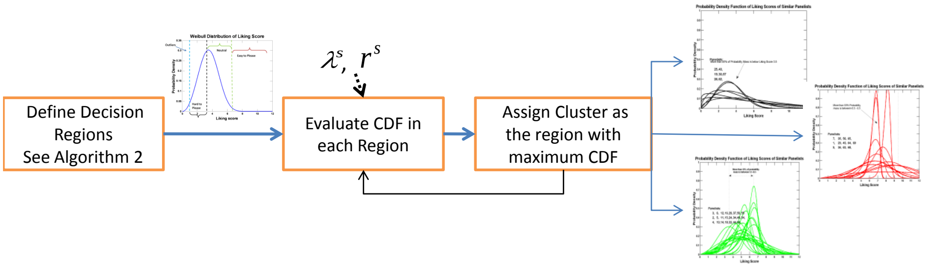

Define Decision Regions See Algorithm 2
λ^s, r^s
Evaluate CDF in each Region
Assign Cluster as the region with maximum CDF
Weibull Distribution of Liking Score
Liking score
Probability Density Function of Liking Scores of Similar Panelists
Liking Score

Fig. 5. Clustering the Panelists

Having estimated the data generated from the models for 10,000 flavours in $F_B = \{\overline{k}^{(1)} ......\overline{k}^{(10,000)}\}$ using the methods described in Section 6.2, we can classify the panelists into three different categories (see Figure 5). We divide the liking score range [1..9] into three regions as shown in Figure 6. The panelists are then classified by identifying the region in which the majority (more than 50%) of their probability mass lies (see Algorithm 2). This is accomplished by evaluating the cumulative distribution in each of these regions using:

$$P_{(l_1,l_2]}(LS; \lambda, r|s) = e^{-(\frac{l_1}{\lambda})^r} - e^{-(\frac{l_2}{\lambda})^r}. \tag{2}$$

Algorithm 2 Clustering the Panelists

```
for $\forall s \in \{S\}$ do
    1. Calculate $P_{l_1,l_2}$ using estimated $(\lambda^s, r^s)$ for $(l_1, l_2] \rightarrow (1, 3.5]$, $(3.5, 6.5]$ and $(6.5, 9.5]$
    2. Assign the panelist $s$, to the cluster corresponding to the region where he/she has maximum cumulative density
    $s \leftarrow s + 1$
end for
```

6.4 Results on All Panelists

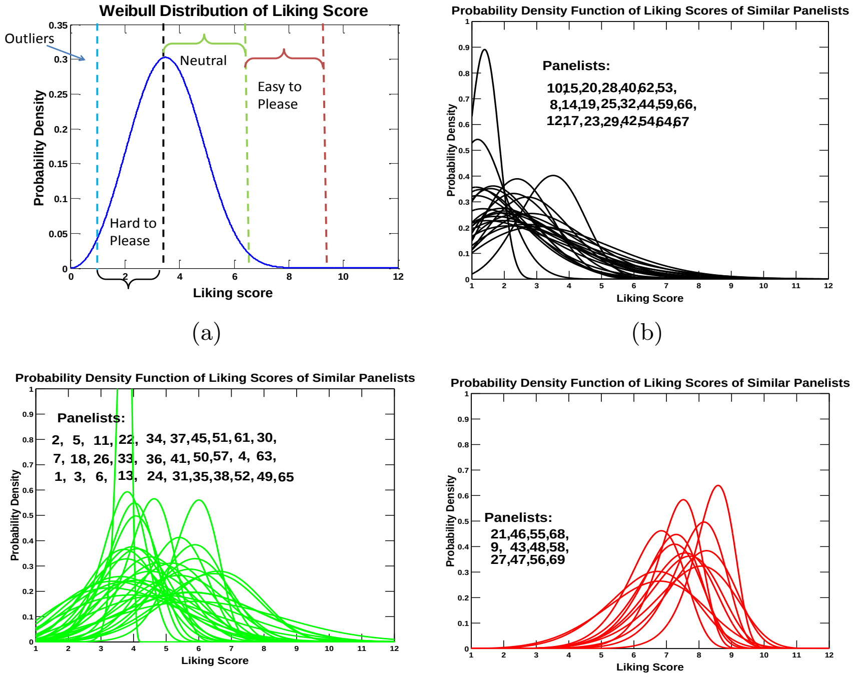
We applied our methodology to the dataset of 66 panelists who can be individually modeled with adequate accuracy. The first cluster is the "hard to please" panelists. We have 23 panelists in this cluster which is approximately 34.8% of the panel. These panelists have most of their liking scores concentrated between 1-3.5 range. We call these "hard-to-please" since low liking scores might imply that they are very choosy in their liking.

The second cluster is the cluster of "neutral panelists". These panelists rarely choose the liking scores which are *extremely like* or *extremely dislike*. For most of the sampled flavours they choose somewhere in between and hence the name *neutral*. There are 31 panelists in this cluster which is 47% of the total panel.

The final cluster of panelists is the "easy to please" panelists. This cluster of panelists reports a high liking for most of the flavours presented to them or may report moderate dislike of some. They rarely report "extremely dislike". There are 12 panelists in this cluster which is close to 18% of the total panel.

7 Conclusions and Future Work

This contribution described an ensemble-based symbolic regression approach for knowledge mining from a very small sample of survey measurements. It is only a first small step towards GP-driven flavour optimization and also demonstrates the effectiveness of GP for sparse data modeling. Our goal was to model behavior of panelists who rate flavours. Our methodology postpones decision making regarding a *model*, a *prediction*, and a *decision boundary* until the very end. In Step 1 ParetoGP generates a rich set of models consisting of the multiple plausible explanations for the data from multiple run aggregation of its best models. In Step 2 these are filtered into an efficient and capable ensemble and no valid explanation is eliminated. In Step 3 *all* the models are consulted, and with minor trimming, their predictions are fit to a probability density function. Finally, in Step 4, when macro-level behaviour has emerged and more is known about the panelists, decision boundaries can be rationally imposed on this probability space to allow their segmentation. Our approach allowed us to robustly identify segments in the panel based on the liking preferences. We conjecture from our results that there are similar potential benefits across any sparse, repeated mea-



(a)

(b)

(c)

(d)

Fig. 6. Liking Score Density Models: (a)Decision regions for evaluating cumulative distribution, (b) Hard to please panelists (c) Neutral Panelists (d) Easy to Please Panelists

sure dataset. We will focus our efforts in the future on the theory and practice of efficient techniques for ensemble derivation in the context of GP.

Acknowledgements

We acknowledge funding support from Givaudan Flavors Corp. and thank Guido Smits and Mark Kotancheck.

References

1. Moskowitz, H.R., Bernstein, R.: Variability in hedonics: Indications of world-wide sensory and cognitive preference segmentation. Journal of Sensory Studies **15**(3) (2000) 263–284
2. Smits, G., Kotanchek, M.: Pareto-front exploitation in symbolic regression. In O'Reilly, U.M., Yu, T., Riolo, R.L., Worzel, B., eds.: Genetic Programming Theory and Practice II. Springer, Ann Arbor (2004)
3. Liu, Y., Yao, X., Higuchi, T.: Evolutionary ensembles with negative correlation learning. IEEE Transactions on Evolutionary Computation **4**(4) (2000) 380

4. Liu, Y., Yao, X.: Learning and evolution by minimization of mutual information. In: PPSN VII: Proceedings of the 7th International Conference on Parallel Problem Solving from Nature, London, UK, Springer-Verlag (2002) 495–504
5. Hansen, L.K., Salamon, P.: Neural network ensembles. IEEE Trans. Pattern Anal. Mach. Intell. **12**(10) (1990) 993–1001
6. Wolpert, D.H.: Stacked generalization. Neural Networks **5**(2) (1992) 241–259
7. Krogh, A., Vedelsby, J.: Neural network ensembles, cross validation, and active learning. In Tesauro, G., Touretzky, D., Leen, T., eds.: Advances in Neural Information Processing Systems. Volume 7., Cambridge, MA, USA, The MIT Press (1995) 231–238
8. Paris, G., Robilliard, D., Fonlupt, C.: Applying boosting techniques to genetic programming. In Collet, P., Fonlupt, C., Hao, J.K., Lutton, E., Schoenauer, M., eds.: Artificial Evolution 5th International Conference, Evolution Artificielle, EA 2001. Volume 2310 of LNCS., Creusot, France, Springer Verlag (2001) 267–278
9. Iba, H.: Bagging, boosting, and bloating in genetic programming. In Banzhaf, W., Daida, J., Eiben, A.E., Garzon, M.H., Honavar, V., Jakiela, M., Smith, R.E., eds.: Proceedings of the Genetic and Evolutionary Computation Conference. Volume 2., Orlando, Florida, USA, Morgan Kaufmann (1999) 1053–1060
10. Schapire, R.E.: The strength of weak learnability. Machine Learning **5**(2) (1990) 197–227
11. Freund, Y., Seung, H.S., Shamir, E., Tishby, N.: Information, prediction, and query by committee. In: Advances in Neural Information Processing Systems 5, [NIPS Conference], San Francisco, CA, USA, Morgan Kaufmann Publishers Inc. (1993) 483–490
12. Sun, P., Yao, X.: Boosting kernel models for regression. In: ICDM '06: Proceedings of the Sixth International Conference on Data Mining, Washington, DC, USA, IEEE Computer Society (2006) 583–591
13. Freund, Y.: Boosting a weak learning algorithm by majority. Inf. Comput. **121**(2) (1995) 256–285
14. Folino, G., Pizzuti, C., Spezzano, G.: GP ensembles for large-scale data classification. IEEE Trans. Evolutionary Computation **10**(5) (2006) 604–616
15. Vladislavleva, E.: Model-based Problem Solving through Symbolic Regression via Pareto Genetic Programming. PhD thesis, Tilburg University, Tilburg, the Netherlands (2008)
16. Vladislavleva, E.J., Smits, G.F., den Hertog, D.: Order of nonlinearity as a complexity measure for models generated by symbolic regression via pareto genetic programming. IEEE Transactions on Evolutionary Computation **13**(2) (2009) 333–349
17. Kotanchek, M., Smits, G., Vladislavleva, E.: Trustable symoblic regression models. In Riolo, R.L., Soule, T., Worzel, B., eds.: Genetic Programming Theory and Practice V. Genetic and Evolutionary Computation. Springer, Ann Arbor (2007) 203–222
18. Taylor, J.S., Dolia, A.: A framework for probability density estimation. In Lawrence, N., ed.: Proceedings of the Eleventh International Conference on Artificial Intelligence and Statistics, Journal of Machine Learning Research (2007) 468–475
19. Mukherjee, S., Vapnik, V.: Multivariate density estimation: a support vector machine approach. In: In NIPS 12, Morgan Kaufmann Publishers (1999)