Will Smith

Dr. Shyam Gouri-Suresh

ECO 316 – Fall 2020

December 7, 2020

**Food Hall Fitting – Computational Economics Final Project**

1. **Introduction**

In my childhood and youth, I observed several storefronts in my hometown demonstrate significant turnover of tenants over time, especially restaurants. Since each of these locations seemed to be attractive pieces of real estate with a positive neighborhood atmosphere, my family and I wondered why those properties so frequently failed to establish successful restaurants for us to enjoy. Some of the attempts certainly performed better than others – perhaps those restaurants’ cuisines better fit the gastronomic demands of our community. I develop this project based on this curiosity of how property developers or restaurateurs determine the best-fitting restaurant types/cuisines for their market over time. How might observing or sharing information with other restaurateurs help them get it right?

A recent local development that has exhibited great success and adoption is a place called the Pizitz Food Hall. Located on the ground floor of a historic apartment building in downtown Birmingham, Alabama, this food hall is home to a collection of food stalls which serve customers with a common seating and bar area. I believe a key factor behind the success of this location was the site’s flexibility in changing the tenants of their food stalls – the managers had the opportunity to react to the patterns observed in the customer base and invite new restaurants or keep successful ones. Perhaps each stall had its own proprietor who would make those decisions, potentially after observing the successes or failures of the others, as well. I take inspiration from the Pizitz as I formulate a research question: what kind of management structure will allow the stalls in a food hall to learn to provide the greatest satisfaction to the customer base?

Considering this question, I identify two major factors that I believe will be consequential in the outcomes of the model: the competition allowed between different stalls in the food hall and the extent to which information sharing takes place. Naturally, an appropriate model will consider a number of different contexts and parameters to reflect different versions according to these factors. Ahead of running the model, I expect that a higher degree of competition will result in greater utility for the customer base, and a higher degree of information sharing will result in more profit maximization by the stalls. These two forces likely will conflict if they are combined into a single model.

Additionally, it is possible that the ‘optimal’ management structure depends on the composition of the customer population in terms of their tastes and preferences for different cuisines. An appropriate model will account for a diverse population in the customer base, and different distributions of people with varying tastes and preferences. I address these considerations of management structure and population when constructing a model to approach my research question.

1. **Model Description**

Creating a model to approach this topic, I consider a hypothetical food hall which has three internal stalls, three cuisine possibilities for those stalls, and a population of 1200 customers with specified cuisine preferences. To properly address my research question, I use three different management structures (Versions) under which the model will run:

1. Each stall operates independently
2. Each stall operates independently, but with knowledge of the other stalls’ service volumes
3. The food hall itself manages all of the stalls

For each of these management versions, I use a reinforced learning model based on the entities making cuisine decisions and the extent of information sharing. I utilize the structure of the learning model as developed by my professor Dr. Shyam Gouri-Suresh based on the paper “Predicting How People Play Games: Reinforcement Learning in Experimental Games with Unique, Mixed Strategy Equilibria” by Roth and Erev (1998).[[1]](#footnote-2)

|  |  |  |  |
| --- | --- | --- | --- |
| **Table 1: Customer Personas, with Cuisine Preferences** | | | |
|  | **Cuisine 1** | **Cuisine 2** | **Cuisine 3** |
| **Persona 1** | .15 | .5 | .85 |
| **Persona 2** | .15 | .85 | .5 |
| **Persona 3** | .5 | .15 | .85 |
| **Persona 4** | .5 | .85 | .15 |
| **Persona 5** | .85 | .15 | .5 |
| **Persona 6** | .85 | .5 | .15 |

Since learning occurs as a result of services volumes at each stall, the distribution of preferences in the customer population is an important factor. Managing this dependency, I consider six different customer “personas,” which have preferences for the three cuisines as specified in Table 1. In this model, I separate results based on distributions of these personas in four 1200-customer “cohorts” as detailed below:

* *Cohort 1*: Even distribution of Personas 1-6
* *Cohort 2*: Homogenous population of Persona 1
* *Cohort 3*: Even mix of Personas 1 and 6
* *Cohort 4*: Uneven, 1:3 ratio of Personas 1 and 6 (a minority and a majority)

Given the three management versions and four cohorts present, there are 12 different contexts under which I run repeated trials of the model to produce results. I focus my analysis on comparing results between management versions for each of the four customer cohorts.

The model runs over a duration of 5000 days. In the daily operations of the food hall, each of the 1200 customers makes a decision to patronize one of the three stalls or to stay home. While the customers do not go through a learning process, their daily decisions are determined according to their preferences for the cuisines available on a given day, as discussed in the next section. Then, the appropriate reinforced learning ‘buckets’ for the given Version are updated to according to the service volumes at each stall or the service volume of the food hall as a whole. Importantly, the stalls’ cuisines are updated on the first day then again on every 30th day (always before the daily operations) to represent monthly adjustments.

1. **Model Implementation**

I implement the Food Hall learning model in MATLAB through a master script called *main*, which calls four externally defined functions: *initPreferences*, *getSatisfaction*, *updateCuisines*, and *processCustomers*. In this section, I explain the iterative structure of the master script and describe these four critical functions in the order in which they are called, explaining how they execute the initial, daily, and monthly operations of the model as described in the Model Description above.

At the top of the *main* script, I define a number of model parameters, including the number of customers, the number of day iterations per trial, and the number trials per combination of Version and Cohort. I then define a matrix called *outcomes* to represent all possible combinations of cuisines which could result from the cuisine choices of the stalls or the chosen strategy of the singular food hall management.[[2]](#footnote-3) The following array shows number representations of these combinations:

combinations = [111,112,113,122,123,133,222,223,233,333]

In lines 8-12 of *main*, I convert this number array into a usable matrix called *outcomes*, where each of the ten rows contains the cuisines present in a given outcome, with multiples allowed as shown above. These outcomes are shown in Table 2. The script then proceeds into the iterative structure of four nested for-loops: a loop over the four cohorts on the outside, containing a loop over the three versions, containing a loop over 20 trials, finally containing a loop over 5000 days for each trial. At the beginning of the outermost loop, I call *initPreferences* to create a 1200x3 *CustomerPreferences* matrix to hold all 1200 customers’ cuisine preferences.

|  |  |  |  |
| --- | --- | --- | --- |
| **Table 2: Possible Outcomes of Cuisine/Strategy Choices** | | | |
| **Outcome 1** | Cuisine 1 | Cuisine 1 | Cuisine 1 |
| **Outcome 2** | Cuisine 1 | Cuisine 1 | Cuisine 2 |
| **Outcome 3** | Cuisine 1 | Cuisine 1 | Cuisine 3 |
| **Outcome 4** | Cuisine 1 | Cuisine 2 | Cuisine 2 |
| **Outcome 5** | Cuisine 1 | Cuisine 2 | Cuisine 3 |
| **Outcome 6** | Cuisine 1 | Cuisine 3 | Cuisine 3 |
| **Outcome 7** | Cuisine 2 | Cuisine 2 | Cuisine 2 |
| **Outcome 8** | Cuisine 2 | Cuisine 2 | Cuisine 3 |
| **Outcome 9** | Cuisine 2 | Cuisine 3 | Cuisine 3 |
| **Outcome 10** | Cuisine 3 | Cuisine 3 | Cuisine 3 |

*initPreferences* accepts the number of customers and the cohort number (1 – 4) as arguments and returns the above *CustomerPreferences* matrix. Lines 7-9 of this function create the six customer Personas as described in the last section by taking the six permutations of [ .85 .5 .15 ] to distribute high, medium, and low preferences for the three cuisines. In the remainder of the function, I populate the returnable matrix with the preferences of these Personas according to the given cohort distribution.

I then call the function *getSatisfaction*, which takes the defined *outcomes* and *CustomerPreferences* matrices as arguments and returns a 1x10 *CustomerSatisfaction* array, which holds the given cohort’s total satisfaction from each of the 10 possible outcomes. For each of these outcomes, this function uses nested for-loops of all the customers and the three cuisines, increasing the outcome’s total satisfaction by a customer’s preference for a cuisine (.85, .5, or .15) only if that cuisine is available in the outcome. With this structure, the function naturally places a high value on diversity of options available – even for cohort 4, in which the 3:1 majority prefers Cuisine 1 at the .85 level, outcome 5 gives the greatest customer satisfaction because it offers all three cuisines. The *CustomerSatisfaction* arrays for all four customer cohorts are visualized in Figure 1.

Inside the version loop, I initialize MATLAB’s random number generator with the version number as a seed before running 20 trials of the 5000-day learning model for the given cohort and version. I initialize two matrices to hold the learning “buckets” for the different versions of the model: *StallCuisinePropensities* is a 3x3 matrix initialized at 10,000 to represent the propensities of each of the three stalls to choose each of the three cuisines, and *StrategyPropensities* is a 1x10 matrix initialized at 80,000 to represent the propensity of the food hall to choose each of the 10 strategies/outcomes.

The core functionality of the model is implemented within the for-loop over the 5000 days. Inside this loop, the cuisines are updated with a call to the *updateCuisines* function on the first day, then every 30th day. This updating always occurs at the beginning of the day, before daily operations. These daily operations are handled by a call to the *processCustomers* function, which processes all customer decisions to visit a stall or stay home and updates the reinforced learning buckets according to the version. Finally, I implement a small degree of forgetting in these learning buckets.

Chart, bar chart

Description automatically generated

The *updateCuisines* function takes in several matrices and variables representing the current state as arguments and returns a 1x3 matrix *Cuisines* representing the updated cuisines available and a variable *bucket* indicating the outcome number (1 – 10) associated with this combination of cuisines. I incorporate branching in this function to handle the decisions differently based on the given management version. Under both Version 1 and Version 2, each stall’s decision is determined by where a random number *x* falls on a cumulative probability spectrum generated from that stall’s learning buckets for the three cuisines. Under Version 3, the strategy decision of the food hall is determined by this same process, but instead using the learning buckets for the 10 strategies/outcomes.

The *processCustomers* function also takes in several matrices and variables and returns the matrices *StallCuisinePropensities* and *StrategyPropensities* with updated learning buckets, as well as an array *Volumes* containing the service volumes of each of the three stalls on that day. First, the function processes each customer’s decision using an algorithm similar to that mentioned in the previous paragraph, though the customers do not experience learning. It defines a probability matrix for each customer on the given day, based on their preferences for the cuisines available in the three stalls. A fourth element, representing the probability of staying home, is calculated as the maximum possible sum of three preferences (.85 \* 3) less the sum of the preferences for the available cuisines. This matrix is then converted into a cumulative probability spectrum from 0 to 1, for use with a random number *x*. The customers’ decisions according to this process are aggregated in the *Volumes* matrix, indicating the service volume at each of the three stalls. In both Versions 1 and 2, each stall’s volume is added to their learning bucket for their current cuisine in the *StallCuisinePropensities* matrix. In Version 2 only, the other stalls’ volumes are multiplied by a factor of 0.5 and added to their respective cuisine buckets – this represents the sharing of information between stalls. In Version 3, the sum of all three volumes is added to the bucket in *StrategyPropensities* for the current strategy/outcome.[[3]](#footnote-4)

After the 5000-day model loop closes, I record data for the total satisfaction of customers over the whole model period, using four matrices called *SatisfactionDataCohort1, SatisfactionDataCohort2, SatisfactionDataCohort3,* and *SatisfactionDataCohort4*. Each of these matrices is 20x3 and holds the satisfaction data for each trial of Versions 1, 2, and 3 for their respective customer cohort for use in quantitative analysis. The calculation of these data is explained in the following Results section.

1. **Results**

Approaching my research question using the model described in the preceding sections across the 12 contexts (Cohort – Version combinations), I identify both visual trends in the learning process and calculable variations in measured. In this process, I analyze both qualitative features observable in live visualizations and quantitative, statistical results.

In order to observe qualitative trends for the various Cohort – Version combinations in the model, I define a function called *visualize*, which uses much of the same code as the *main* script but runs only one trial for a cohort and version specified in the function arguments. Rather than recording statistical data, this function displays a live subplot visualization to show how the decision-making entities learn over the model and how their decisions compare with the outcomes that deliver the greatest satisfaction to the given customer cohort. I have included two examples of these subplots as Figures 2 and 3. As I monitor a visualization, I largely focus on the top-left plot, showing the learning process of the individual stalls or of the food hall as a whole. Reading the plots in Figure 2, it follows that, at the end of the 5000 days, Stall 3 would be likely to choose Cuisine 3 in its next update. Similarly, in Figure 3, the food hall would most likely choose strategy 1 or strategy 2 in its next update.

The likelihood of an entity making a particular decision increases by the payoff from that decision, so it is easy and very common to see momentum effects in the development of these ‘propensities.’ As these decisions are made, the outcome frequencies plot updates, showing which outcomes become the most common over the course of the model trial. One limitation of the implementation of this model is that it does not account for the costs of changing a stall’s cuisine – the inclusion of which would likely cause a stall to maintain its initial cuisine, unless there was truly insufficient demand for it. Without this element, the Stall Cuisine Propensities as seen in Figure 2 tend to be much more split between the three cuisines than would be expected in a real-world scenario.

**Figure 2**

Chart, bar chart

Description automatically generated

**Figure 3**

**Chart, bar chart, histogram

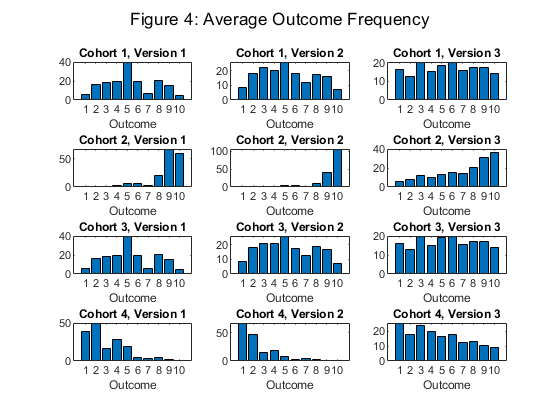
Description automatically generated**

|  |  |  |  |
| --- | --- | --- | --- |
| **Table 4: p-values, Two-Sample t-tests for difference of means** | | | |
|  | **Version 1** | **Version 2** | **Version 3** |
| **Cohort 1** | - | 0.000647 | .00000000894899 |
| **Cohort 2** | - | 0.000205 | 0.000306 |
| **Cohort 3** | - | 0.000317 | .00000000911784 |
| **Cohort 4** | - | 0.000899 | 0.001047 |
| *Using Version 1 means as controls.* | | | |

Quantitatively, I record two types of data for each of the 12 Cohort-Version contexts: the average total satisfaction experienced by the cohort for the given management version and the average frequencies of each of the 10 outcomes over all 20 trials. Analyzing these data, I identify the management structure that, on average, delivers the greatest satisfaction to the customer base.

|  |  |  |  |
| --- | --- | --- | --- |
| **Table 3: Average Total Satisfaction Scores** | | | |
|  | **Version 1** | **Version 2** | **Version 3** |
| **Cohort 1** | 2.14080  ­­­­­­­(.13647) | 1.99350  (.11216) | 1.83960  (.11238) |
| **Cohort 2** | 2.27981  (.23772) | 2.02275  (.13170) | 2.01794  (.16901) |
| **Cohort 3** | 2.14410  (.13687) | 1.98900  (.10773) | 1.84320  (.12176) |
| **Cohort 4** | 2.09476  (.21447) | 1.89091  (.12406) | 1.88035  (.16149) |
| Data over 20 trials. Given in 105. *Standard deviations in parentheses.* | | | |

The total satisfaction ‘score’ for a given trial is calculated as a “SUMPRODUCT” of two arrays: the 1x10 *CustomerSatisfaction* array for a given cohort and the 1x10 *OutcomeFrequencies* array, which contains the frequencies as discussed above.[[4]](#footnote-5) These scores are then averaged over 20 trials to produce the data presented in Table 3. Examining this data, we see a general trend that, for all cohorts, the mean values are highest in Version 1 and declining for Version 2 and Version 3 in sequence. To determine the statistical significance of these differences, I conducted two-sample t-tests with assumed unequal variances for Versions 2 and 3, using Version 1 as a control. The p-values for these tests are given in Table 4. Examining these p-values, we observe that, with the exception of Version 3 for Cohort 4, all of the differences are statistically significant at the α = .001 level or lower. Despite this statistical significance, the arbitrary nature of this ‘satisfaction score’ measurement prevents any ‘real’ assessment of the economic significance of these differences. Keeping in mind the statistically significant trend of mean satisfaction scores between the three versions, let us examine the data for outcome frequency.

Given the nature of this data, it is best analyzed visually – Figure 4 displays the average frequencies of the 10 outcomes for each cohort-version pair over 20 trials. In order to gain insight from these charts, we must compare them to the corresponding Total Satisfaction charts for each cohort in Figure 1. While the focus of my analysis is comparing the results across Versions 1-3 for each cohort, this comparison also produces some interesting insights across cohorts. Notice that the optimal outcome (outcome 5, which has all three cuisines) is fairly dominant for Cohorts 1 and 3, with the degree of dominance decreasing across the versions. Note also how the bar graphs for Cohorts 1 and 3, Versions 1 and 2 closely resemble the bar graphs for those cohorts’ total satisfaction in Figure 1, while the graphs for Version 3 appear more evenly distributed between outcomes, reflecting relatively ineffective learning. This supports the conclusion that competition between stalls, as in Versions 1 and 2, functions better than single management for fitting the preferences of a population with diverse and evenly distributed preferences. Further, the learning process without shared information in Version 1 proved more frequently produced the optimal outcome, indicating that the shared knowledge may have distracted stalls from identifying and filling an existing market segment.

In contrast, the graphs for Cohorts 2 and 4 appear more skewed, favoring outcomes which favor the cuisine strongly preferred by the majority population (or the homogenous population, in the case of Cohort 2). For example, outcome 1 is one of the top two bars in all of the graphs for Cohort 4 – that outcome has three stalls with Cuisine 1, which is the top-preference cuisine for the majority group (Persona 6). This outcome leaves the minority group (Persona 1) with zero stalls which have their first or second-preference cuisine. However, outcome 2, also a frequent outcome, offers two stalls of the majority’s top-preference cuisine, plus one stall with the second-preference cuisine of *both* the majority and the minority. We observe that, in Version 1, this outcome is the most common. Similarly, for Cohort 2 (the homogenous population), Version 1 most commonly produces outcome 9, which also gives the population a combination of two first-preference stalls and one second-preference stall. While Version 3 does produce the theoretical best-satisfaction outcome more often, Versions 1 and 2 prove more effective for meeting market need because they less frequently low-satisfaction outcomes, such as outcomes 7-10 for Cohort 4. Thus, for majority-dominated populations, Versions 1 and 2 still better fit the population’s preferences, though for reasons different than for evenly distributed populations.

1. **Conclusions**

The process of fitting the stalls in a food hall to the tastes and preferences of a variety of customer bases proved quite complicated and difficult to implement in the complexity that it deserves. Nonetheless, my model provides some valuable insight into the curiosity of which management structures deliver the greatest satisfaction to the customer base. In my statistical analysis of the total customer satisfaction scores for each Cohort – Version context, I identified the trend that, for each of the Cohorts 1 – 4, the average score was the highest under individual stall management in Version 1, followed by individual management with information sharing in Version 2, followed by the whole-hall management in Version 3. In all but one of the models for Versions 2 and 3, a hypothesis test at the α = .001 level proved that these averages were different from that of Version 1 for the same cohort.

While these differences cannot be assumed to be economically significant, my analysis of the frequencies of the 10 outcomes for each of the abovementioned contexts indicates that Versions 1 and 2 do produce more favorable outcomes for the given population. For the more diverse and evenly distributed Cohorts 1 and 3, these versions with individual management produced the optimal outcome 5 with a much greater frequency than Version 3. For the more homogenous or unevenly distributed Cohorts 2 and 4, Versions 1 and 2 more reliably produced outcomes which provided sufficient variety and reasonable satisfaction for all groups within the population, even the minority group in Cohort 4. From the analyses described above, I conclude that competition between stalls is the driving force which provides the optimal customer satisfaction for the population.

The model I developed to address my research question could be improved upon with further consideration for the daily operations in a food hall, and the business challenges faced by individual stalls or the space as a whole. For example, a crowding effect could be incorporated into customers’ decision algorithm, such that they might be less inclined to visit a stall at which a many of customers are already waiting in line. As mentioned previously, a cost of changing cuisines could be added to the decision-making process for stalls or the food hall manager, which would deter from changing cuisines unnecessarily. Ultimately, I believe that these changes would support the claim that greater variety and competition between stalls produce the optimal outcome for customers.

Bibliography

Erev, Roth. “Predicting How People Play Games: Reinforcement Learning in Experimental Games with Unique, Mixed Strategy Equilibria.” *The American economic review* 88, no. 4 (September 1, 1998): 848–881.

1. Ido Erev and Alvin Roth, “Predicting How People Play Games: Reinforcement Learning in Experimental Games with Unique, Mixed Strategy Equilibria,” *The American economic review* 88, no. 4 (September 1, 1998): 848–881. [↑](#footnote-ref-2)
2. I thank Dr. Shyam Gouri-Suresh for his guidance in structuring my learning models with these outcome “buckets.” [↑](#footnote-ref-3)
3. Note that the total volume of customers will vary based on how many customers decide to stay at home rather than visit one of the three stalls. [↑](#footnote-ref-4)
4. Note that, while there are 5000 days in each trial, changes in ‘outcomes’ only occur on the first day and every 30th day. Thus, there are only 167 outcomes changes in the duration of each trial. [↑](#footnote-ref-5)