CHAPTER

13

Case Studies

These case studies are included to describe real projects that use principles described in this book to create predictive models. Throughout any project, decisions are made at key points of the analysis that influence the quality of the final solution; this is the *art* of predictive modeling. Rarely does an analyst have time to consider all of the potential approaches to a solution, and therefore decisions must be made during Data Preparation, Modeling, and Model Evaluation.

The case studies should be used as motivational rather than as recipes for predictive modeling. The analyses in these case studies don't present perfect solutions, but they were successful. The survey analysis case study had the luxury of trying two approaches, each with pros and cons. The help desk case study succeeded because of the perseverance of the analysts after the first modeling approach failed. In both cases, the final solution used the science of predictive analytics plus creative twists that were unconventional, but productive.

Survey Analysis Case Study: Overview

This case study describes a survey analysis project for the YMCA, a cause-driven charity whose core mission is to improve the lives of its members and build communities. The YMCA achieves these objectives primarily through facilities and programs that promote the physical well-being of its members.

The YMCA expends considerable time and effort to understand how to help members achieve their fitness goals and build community. These analyses are grounded in sound social science and must be diagnostic and predictive so they are understandable by decision makers. Moreover, decisions are made at the branch level—the individual YMCA facility.

One source of data for achieving the analytics objectives is the annual member survey. The YMCA, as a national organization with more than 2,500 branches across the country, developed a member survey for use by its branches. Tens of thousands of these surveys are completed each year.

SEER Analytics (`http://www.seeranalytics.com`) was, and continues to be, the leader in analyzing YMCA surveys and creating actionable insights based on the analyses. I was a subcontractor to Seer Analytics for the work described in the case study.

Business Understanding: Defining the Problem

For the project, 32,811 surveys with responses were made available for the year 2001. Modeling data from the survey contained 48 multiple choice questions coded with values between 1 and 5, where 1 was the most positive response and 5 the most negative. Questions were primarily attitudinal, related to the member's experience with the Y, though four questions were demographic variables. There were two free-form text questions and two related fields that categorized the text into buckets, but these were not used in the analysis.

Throughout this case study, questions will be identified by a Q followed by the question number, so Q1 for question 1. Sometimes an explanation of the key idea of the question will be provided in parentheses, such as Q1 (satisfaction), indicating that the key idea in question 1 is member satisfaction.

Defining the Target Variable

Three questions in the survey were identified as key questions related to attitudes and behaviors in members that are indicative of how well the Y is meeting the needs and expectations of the members. These three questions were Q1, Q32, and Q48, with the full text of the questions shown in Table 13-1.

Table 13-1: Three Questions for Target Variables

QUESTION	FULL QUESTION	SUMMARY OF IDEA REPRESENTED IN QUESTION	PERCENT WITH RESPONSE = 1
Q1	Overall, how would you rate the [Branch Name] YMCA?	Satisfaction	31%
Q32	All things considered, do you think you will belong to this Club a year from now?	Intend to renew	46%
Q48	Would you recommend the Club to your friends?	Recommend to a friend	54%

The three target variables had high association with each other. Of the 31 percent of Q1 = 1 responders, 86 percent of them also had Q48 = 1, a lift of 1.6 over the 54 percent baseline for Q48 = 1. Conversely, of the 54 percent with Q48 = 1, 49 percent have Q1 = 1, a lift of 1.6 as well.

Because of this phenomenon and the desire to create a single model to characterize members with a highly positive opinion of their Y branch, we created a new derived target variable: a simple linear combination of the three questions, called the *Index of Excellence* (IOE). In order to have larger values of IOE considered better than smaller values and have its maximum value equal to 10, we reversed the scale by subtracting the sum from the maximum possible value (15), or

$$\text{IOE} = 10 \times [(15 - \text{Q1} - \text{Q32} - \text{Q48}) \div 12]$$

The three target variables are all skewed toward the lower end of their respective distributions, and therefore IOE, with its distribution reversed, was skewed toward the upper end of its range; the vast majority of values exceed 7, as can be seen in Figure 13-1.

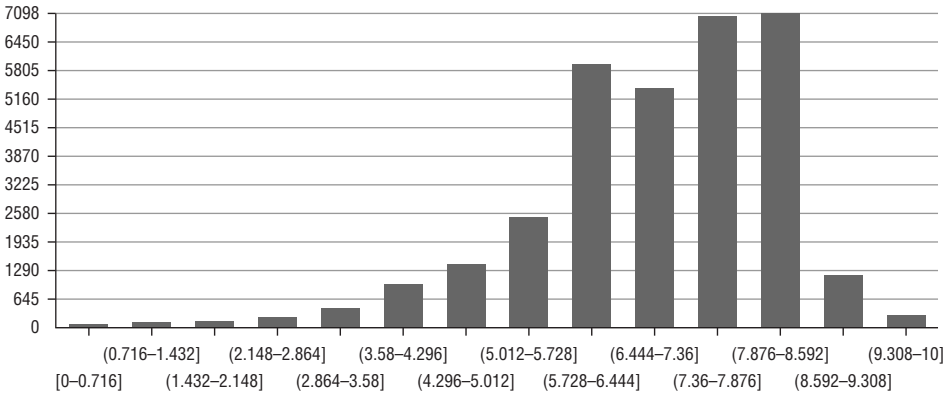

Figure 13-1: Index of Excellence distribution

Data Understanding

The candidate input questions are shown in Table 13-2, along with their category labels as determined by the analysts.

Table 13-2: Questions and Categories for Model Inputs and Outputs

CATEGORY	QUESTIONS
Staff	Q2, Q3, Q4, Q5, Q6, Q7, Q8, Q9, Q10, Q11, Q12
Building	Q13, Q14, Q15, Q16, Q17, Q42
Equipment	Q18, Q19, Q20, Q37, Q38, Q39, Q40, Q41
Programs	Q21, Q23, Q43
Value	Q22
Goals	Q44
Relationships	Q24, Q25, Q26, Q27
Other	Q28, Q29, Q30, Q31, Q45, Q46, Q47
Targets	Q1, Q32, Q48
Text	Q33, Q34, Q35, Q36
Membership	Q49, Q50, Q51
Demographics	Q52, Q53, Q54, Q55

First, two of the quirks and potential problems in the data are as follows:

- In the full data set, 232 surveys had no responses (0.7 percent of the total survey population); all the response values were zero. However, these

232 surveys had text comments, indicating the members still wanted to communicate a viewpoint. These surveys were included in the analysis (an oversight), though had little effect on the patterns found.

- One question that arose was whether there was a significant number of members who merely checked the same box for all the questions: all 1s, all 2s, all 3s, and so on. The good news was that the worst instance of this pattern was in 274 surveys that had all responses equal to 1 (only 0.8 percent of the population), more than the number of surveys with all 2s (58), all 3s (108). Therefore, the responses to the questions were believed to be an accurate reflection of the true member response.

Because neither of these problems cause significant numerical issues, no corrective action was taken.

Data Preparation

Very little data preparation was needed for the data set. The responses were coded with values 0 through 5, so there were no outliers. There were few NULL values. Any NULL values were recoded as 0.

The most significant decision about the data was determining how to use the questions in the algorithms. The data was ordinal and not truly continuous. However, for regression models, it was far more convenient to use the data as continuous because in this case, there is only one column in the data per question. If the data were assumed to be categorical, the questions would be exploded to dummy variables, leading to up to five or six columns per question if the zeros were included as dummy columns.

Missing Data Imputation

The next most significant issue was the problem with responses coded with 0—the code for no response. If the data were treated as categorical, these could be left as the value 0 and treated as just another level. However, if the variables are treated as continuous or ordinal, leaving these as 0 communicates to the algorithms that 0 is the best answer of all (smaller than the "top" response value of 0). These values therefore were similar in function to missing values and needed to be corrected in some way.

On the surface, mean imputation seemed to be a misleading way to recode these values; coding non-responses as 3 didn't necessarily convey the relationship between the non-responses and the target variables. Instead, the approach was to identify the relationship between the 0s and the target variable first, and then recode the 0s to values that did the least harm.

Figure 13-2 shows two plots. The plots show the relationship between the mean value of the target Q1 = 1 and Q2. Q2 = 5 was omitted because for this question and most others, the counts for the worst response code, 5, was very small and did not follow the trend of the other responses. Note that the relationship between the percentage of responses for Q1 = 1 decreases monotonically with Q2. The percentage of Q1 = 1 responses for non-responsive Q2 was 16.24 percent, shown at the far left of both plots.

The plot at the right shows a linear curve fit of the data points and the plot at the left an exponential curve fit of the percentage of Q1 = 1 vs. Q2, and the resulting formulas for the curve fits are shown on the plots. The equations were solved for "x" (Q2), which yielded Q2 = 2.2 for the exponential equation and Q2 = 2.8 for the linear fit. Replacing the 0s with either of these values would bias the models less than using the mean (1.8) or the median (2). The final formula for imputation places the 0s at a value in between these, 2.5 in the case of question Q2.

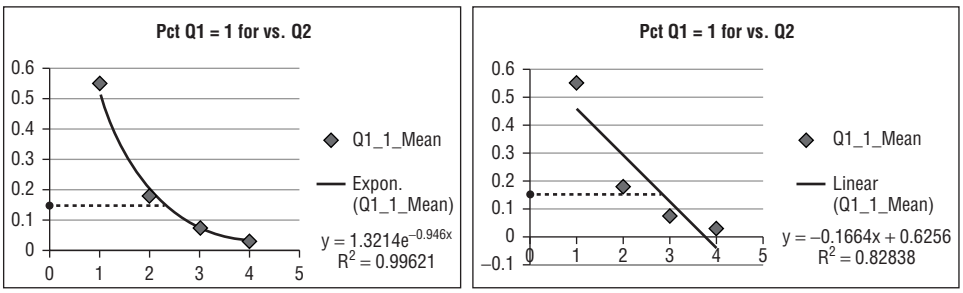

Figure 13-2: Missing value imputation

Was this cheating, to use the target variable in the imputation? The answer technically was "yes"; using the target variable in decisions for how to recode and transform variables is inherently dangerous, possibly leading to biased decisions that will increase accuracy on training data but lessen accuracy on new data. On the other hand, from a practical standpoint, the answer is clearly "no." Because the values used in imputation were nearly always 2.5, they were stable and not prone to overfitting data based on the imputation value.

This process of imputing missing values using a linear fit of a question to the target variable was repeated for every question. Very often, the value 2.5 was a good value to use like Q2, though this wasn't always the case.

Feature Creation and Selection through Factor Analysis

The questions in a survey were intended to capture attitudes of the responders. Often, multiple questions were asked to uncover an attitude or opinion in different ways, which led to a high correlation between questions. One approach often used by social scientists to uncover the underlying ideas represented by the

questions is Factor Analysis, a technique closely related to Principal Component Analysis, discussed in Chapter 6. Even though the algorithm has some difference with PCA, the use of Factor Analysis in this case study mirrors the use of PCA described in Chapter 6.

After applying Factor Analysis using default settings, we determined that the top ten factors were interesting and comprised enough of the variance to be a good cutoff. As is the usual practice, each of the factors was named according to the questions that loaded highest on the factors. The factor loadings for five of the factors are shown in Table 13-3. The highest loading questions on the factors appear in bold. The highest loading questions generally tend to be questions neighboring each other because of the way the survey was laid out, though it isn't always the case. In addition, a few questions loaded high on multiple factors, indicating a clean set of ideas identified by Factor Analysis.

Table 13-3: Factor Loadings for Five of Six Top Factors

FACTOR DESCRIPTION	STAFF CARES	FACILITIES CLEAN/SAFE	EQUIPMENT	REGISTRATION	FRIENDLY STAFF
Factor Number	**1**	**2**	**3**	**4**	**6**
Q2	0.295	0.238	0.115	**0.458**	0.380
Q3	0.217	0.143	0.093	**0.708**	0.077
Q4	0.298	0.174	0.106	**0.601**	0.266
Q5	0.442	0.198	0.087	0.173	**0.613**
Q6	0.417	0.254	0.142	0.318	**0.584**
Q7	0.406	0.277	0.167	0.252	**0.461**
Q8	**0.774**	0.058	0.041	0.093	0.113
Q9	**0.733**	0.175	0.108	0.145	0.260
Q10	**0.786**	0.139	0.079	0.110	0.218
Q11	**0.765**	0.120	0.101	0.132	0.015
Q12	**0.776**	0.090	0.049	0.087	0.014
Q13	0.145	**0.728**	0.174	0.112	0.110
Q14	0.191	**0.683**	0.163	0.151	0.124
Q15	0.102	**0.598**	0.141	0.090	0.070
Q16	0.100	0.370	0.133	0.082	0.035
Q17	0.128	**0.567**	0.229	0.102	0.080
Q18	0.148	0.449	**0.562**	0.116	0.114
Q19	0.129	0.315	**0.811**	0.101	0.103
Q20	0.171	0.250	**0.702**	0.086	0.078

Table 13-4 rearranges the information in Table 13-3, showing a summary of the factors, the top loading questions, and the percent variance explained. It is perhaps more clear from this table that questions did not overlap between factors.

Table 13-4: Summary of Factors, Questions, and Variance Explained

FACTOR	FACTOR DESCRIPTION	TOP LOADING QUESTIONS	CUMULATIVE PERCENT VARIANCE EXPLAINED
Factor 1	Staff cares	Q8, Q9, Q10, Q11, Q12	12.2
Factor 2	Facilities clean/safe	Q13, Q14, Q16	20.1
Factor 3	Equipment	Q18, Q19, Q20	25.3
Factor 4	Registration	Q3, Q4	29.7
Factor 5	Condition of locker rooms, gym, swimming pool	Q39, Q40, Q41	34.1
Factor 6	Friendly/competent staff	Q5, Q6, Q7	38.2
Factor 7	Financial assistance	Q28, Q29	42.2
Factor 8	Parking	Q16, Q42	46.0
Factor 9	Good place for families	Q26, Q30	49.1
Factor 10	Can relate to members / feel welcome	Q23, Q24, Q25	52.1

To help understand why the factors were labeled as they were, Tables 13-5, 13-6, 13-7, and 13-8 show the question numbers and corresponding descriptions of the questions that loaded highest for the top four factors.

Table 13-5: Top Loading Questions for Factor 1, Staff Cares

Q8	Know your name
Q9	Care about your well-being
Q10	Take the initiative to talk to members
Q11	Check on your progress & discuss it with you
Q12	Would notice if you stop coming

Table 13-6: Top Loading Questions for Factor 2, Facilities Clean/Safe

Q13	Overall cleanliness
Q14	Security and safety
Q16	Adequate parking

Table 13-7: Top Loading Questions for Factor 3, Equipment

Q18	Maintenance of equipment
Q19	Has the right equipment
Q20	Has enough equipment

Table 13-8: Top Loading Questions for Factor 4, Registration

Q3	Ease of program or class registration
Q4	Staff can answer questions about schedules, classes, etc.

Modeling

Sampling for building the predictive models was standard: 50 percent of the data was used for training, 50 percent for out-of-sample testing. The first models used a traditional approach of stepwise linear regression to predict IOE with a few key questions and the factor analysis projections as inputs. Some descriptions of Factor Analysis and Principal Component Analysis (like Wikipedia) describe these techniques as performing data reduction or dimensionality reduction. A more accurate description is that these techniques perform candidate input variable reduction. All 48 of the original questions are still needed to compute the factors or principal components, so all of the data is still needed even if the Factor Analysis or Principal Component Analysis reduces the questions down to ten factors like the ones shown in Table 13-4.

For this project, two questions were identified as key questions on their own: Q22 (Value for the money) and Q44 (How has the YMCA helped meet your fitness goals) based on the recommendations of domain experts and the strong predictive relationship of these questions in preliminary modeling. These were included in addition to the ten factors so that there were 12 candidate inputs. A stepwise linear regression variable selection approach was taken in building the model summarized in Table 13-9.

Table 13-9: Regression Model with Factors as Inputs

VARIABLE	VALUE	STD. ERROR	T VALUE	PR(>|T|)
(Intercept)	10.2566	0.0172	597.0967	0.0000
Q44	–0.5185	0.0074	–70.4438	0.0000
Q22	–0.4893	0.0068	–71.5139	0.0000

Continues

Table 13-9 (*continued*)

VARIABLE	VALUE	STD. ERROR	T VALUE	PR(>|T|)
Factor2	–0.2761	0.0055	–50.5849	0.0000
Factor1	–0.2397	0.0051	–47.0156	0.0000
Factor6	–0.2242	0.0056	–39.9239	0.0000
Factor9	–0.2158	0.0054	–40.0539	0.0000
Factor10	–0.1917	0.0057	–33.4452	0.0000
Factor3	–0.1512	0.0051	–29.472	0.0000
Factor4	–0.1068	0.0055	–19.2649	0.0000
Factor5	–0.0798	0.0054	–14.846	0.0000

All of the factors included in the model were significant predictors. Factors 7 and 8 were removed as a result of the stepwise procedure, as their reduction in accuracy did not justify their inclusion per the *Akaike information criterion* (AIC). However, the two key questions, Q44 and Q22, had a much larger influence on the model as evidenced by their coefficients and t values.

An alternative approach was tried as well. Rather than using the factors as inputs to the models, the question that loaded the highest on each factor was selected as the representative of the factor. This approach has an advantage over using the factors as inputs: It is more transparent.

The factors, while representing an idea, still require all of the inputs so the factors can be computed; each factor is a linear combination of all the survey questions. However, if you use just one representative question for each factor, the question that loaded the highest on the factor can be used instead of the entire factor. In this approach, rather than needing all the questions to run the model, only 12 questions (at most, if no variable selection took place) are candidate inputs. After further assessment of the data, a third key variable, Q25 (Feel welcome at the YMCA) was added, making 13 candidate inputs to the model. That model, with seven inputs found by a stepwise regression procedure, is summarized in Table 13-10. The four factors represented in the model are 1, 2, 3, and 6.

Table 13-10: Regression Model with Representative Questions as Inputs

INPUT	QUESTION	QUESTION DESCRIPTION	FACTOR
1	Q25	Feel Welcome	NA
2	Q44	Y Helps Meet Fitness Goals	NA
3	Q22	Value for the Dollar	NA
4	Q13	Facilities clean	Factor 2
5	Q18	Equipment Maintained	Factor 3

6	Q9	Staff Cares about Well-Being	Factor 1
7	Q6	Competent Staff	Factor 6

After a comparison of the two approaches, using individual questions as inputs rather than the factors, generated higher R-squared, and therefore was the model selected for use.

Model Interpretation

The regression model was not built so that the three target questions or the combined form of the three questions, IOE, could be predicted. They were built to identify member attitudes related to these target questions. The models identified the key questions (inputs) related to IOE nationally. However, the models needed to be able to inform individual branches how well they were achieving their own IOE, and in what areas they could improve their Y so that IOE could be increased.

To explain the models, Seer Analysis focused on the questions included as inputs to the models. If a YMCA branch was doing well with those questions, they were necessarily doing well on the IOE scale. The branches then could incorporate changes in their staff, programs, facilities, or equipment to improve the responses to the inputs of the model, thus increasing their IOE. The desire was to show these relationships in a concise and informative way through data visualization.

The key visual shown for each branch was like the one shown in Figure 13-3. The seven key questions found by the regression model were shown along the x axis, with the importance as found by the regression equation increasing as one goes to the right; the most important question nationally was at the far right. All of the questions were important though, not just the ones at the right. The same visualization could be created for each branch.

On the y axis, relative measures of the impact of these questions on IOE were shown. Several key ideas were incorporated in the visualization. First, the order of the balls on the x axis reflected the level of relative importance of the question in the model. Feel welcome was the most important question and Staff cares was the least important of the top seven questions.

Second, Figure 13-3 compares the effectiveness of a YMCA branch to its peers. The average value of the peer group the branch belonged to appeared as a horizontal line across the middle of the plot. Peer groups were branches whose members are from similar socio-economic status, level of education, and ethnic background. A ball whose center was above the line was performing better for that question than its peers, whereas if the ball center fell below the line, the branch was performing worse than its peers.

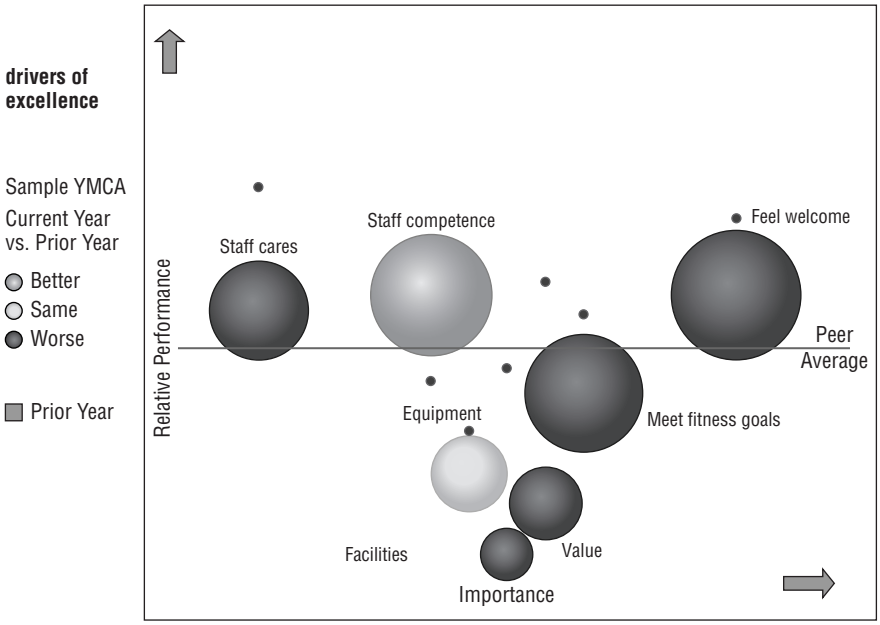

Figure 13-3: Visualization of Drivers of Excellence

Third, Figure 13-3 compares the effectiveness of the branch compared to the values of each factor for the prior year. If the ball was medium gray, the branch was improving compared to the year prior; if it was dark gray, the branch was doing worse; and if the ball was light gray, it was approximately the same as the year prior. To provide a measure of how much better or worse the branch was performing, a small round dot was placed on the plot to indicate the prior year value for each question.

Fourth, the size of the balls indicated the relative importance of the question to IOE for that particular branch; larger balls indicated higher importance, smaller balls lower importance. The order of the balls from left to right was kept the same for every branch, indicating the national trend relating the questions to IOE. The individual branch could therefore identify if it was behaving in ways similar to or different from the national trends.

For the branch shown in Figure 13-3, Staff competence improved compared to the prior year, but most of the other factors were worse than the prior year. Feel welcome had the most significant influence on IOE (just like the average influence nationally). The Facilities question is the smallest ball, and therefore, for this branch, has the least influence on IOE, although it was the fourth most influence nationally.

The second example, Figure 13-4, shows a very successful YMCA branch. All seven of the key factors show that this branch performs better than its peer group, and five of the seven questions show an improvement for this branch compared to its prior year measures. Six of the seven questions were worse than their peer group the year before (the dots are below the peer average line), indicating this branch took steps in the past year to improve the attributes. Why is this so important? If responses to the questions were improved, IOE would improve; this is what the predictive model demonstrated empirically from the data.

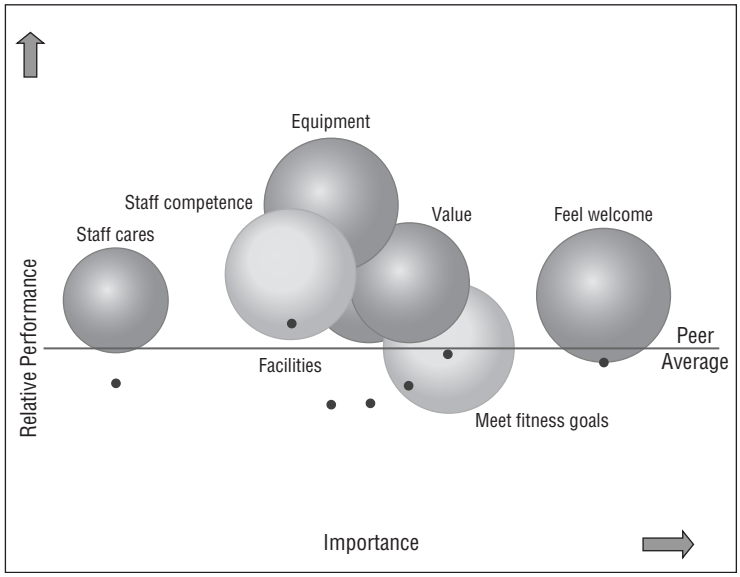

Figure 13-4: Drivers of Excellence, example 2

In the example shown in Figure 13-5, the branch performed at or below its peer group average for every factor, and its performance was flat or mildly improving from the prior year, indicating that the branch is making some improvements to close its gap with other branches. The biggest areas of deficit were related to the facilities and equipment, good candidates for improvement in the next year.

In the example shown in Figure 13-6, the branch was performing worse in six of the seven factors compared to its prior year surveys, and was below its peer group average in the current year. Clearly this branch was heading in the wrong direction and needed extensive improvements.

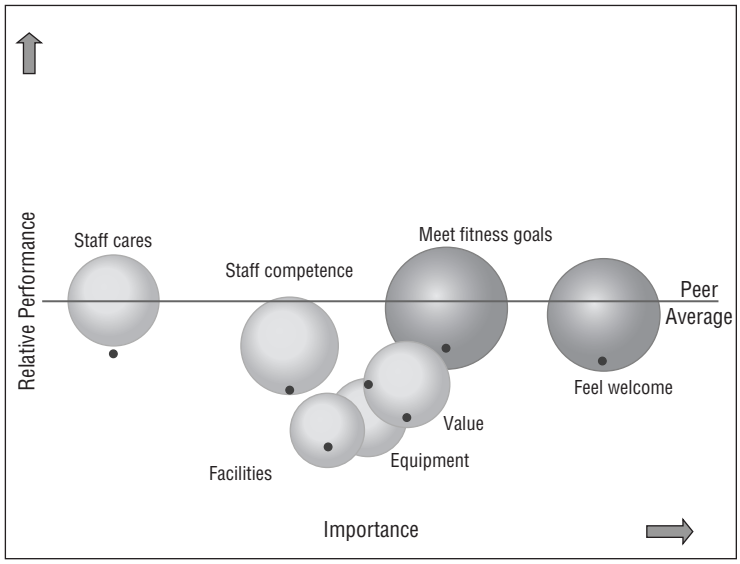

Figure 13-5: Drivers of Excellence, example 3

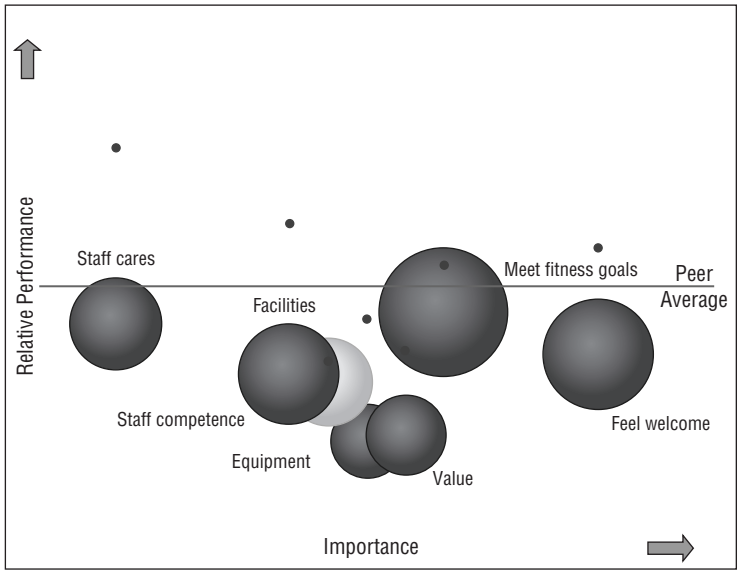

Figure 13-6: Drivers of Excellence, example 4

Finally, in Figure 13-7, the branch performance is mixed. On some dimensions, such as Staff competence, the branch did well: It was higher than its peers and doing better than it was in the prior year. The Staff cares question was also much

better than its peers. On the other hand, the Facilities and Value of the branch were well below the values of its peers and the branch was doing worse than the prior year. The factors Facilities and Value clearly needed improvement.

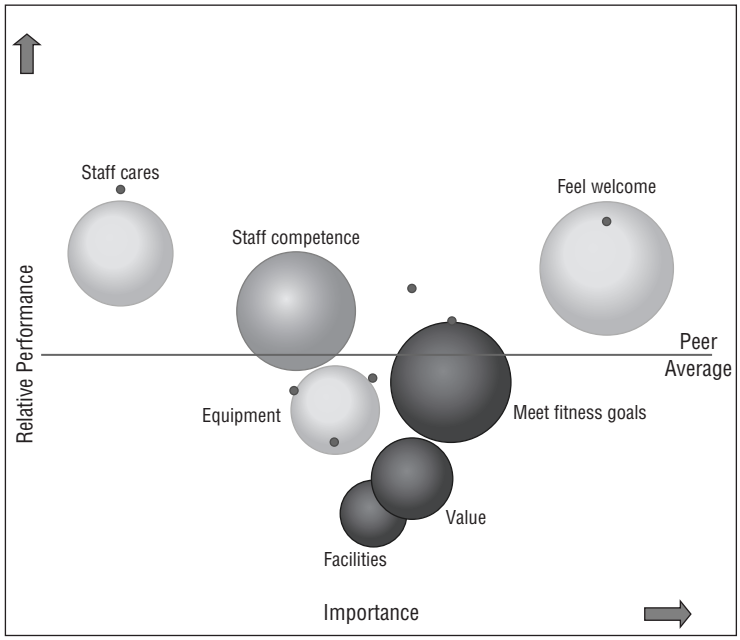

Figure 13-7: Drivers of Excellence, example 5

The idea behind the visualization, therefore, was to provide a report and an explanation of the report to every branch; each branch would only see its own report and could focus on factors where the branch was performing worse than its peers or worse than its performance in the prior year.

Deployment: "What-If" Analysis

One additional deployment strategy considered was incorporating a "what-if" analysis based on model predictions. The premise was that if the branch could change a significant percentage of members that checked the "2" box on a survey question to a "1," how would that change IOE? Figure 13-8 shows the results of some of the analysis. The entire simulation took place in a spreadsheet, where a random proportion of a single question changed in value from 2 to 1—10 percent, 20 percent, and 50 percent—and IOE was recalculated based on the new proportions of 1s and 2s. The percent change in IOE was recorded in the figure.

Recall that the regression models showed that "Feel Welcome" was the most important factor in predicting IOE. However, this factor did not have the highest

sensitivity to change from 2 to 1; that honor belonged to Value for the money. In other words, nationally, on average for all branches, the best way to improve IOE was to do something at the branch that caused a significant percentage of members to change their survey score for Value for the money. In second place was "Staff Cares." Therefore, getting 10 percent or more of the members to believe the staff cares about their well being would result in a 4.2 percent increase in IOE, thus increasing Satisfaction, Intend to renew, and Recommend to a friend.

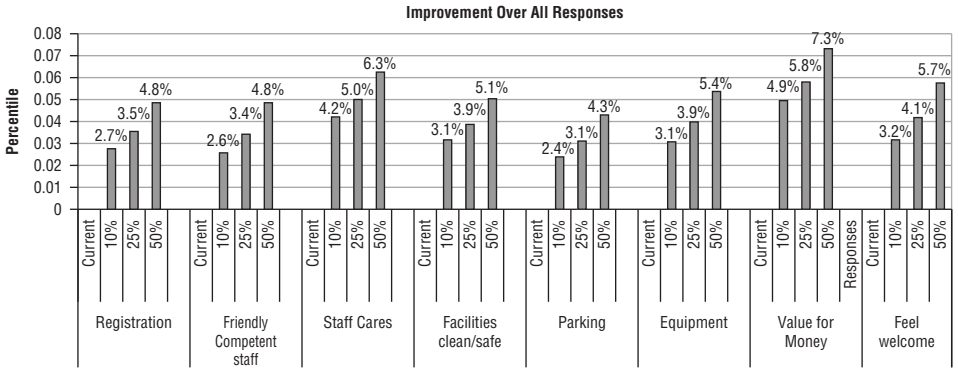

Figure 13-8: What-if scenarios for key questions

This simulation was never delivered to the YMCA but provides a way to use the models that is not directly included in the predictive modeling business objective. In fact, the predictive models didn't directly influence the simulation at all; the role of the regression models was to identify which questions to focus on in the simulation, and therefore was intended to complement the visualizations shown in Figures 13-3 to 13-7.

Revisit Models

Unfortunately, decision-makers found the visualization too complex; it wasn't clear to them exactly what the information meant for their individual branch. There are other predictive modeling approaches that are more accessible, however, including decision trees. Therefore, decision trees were created to uncover key questions in the survey related to the three target variables that would hopefully provide a more transparent interpretation of the survey.

Business Understanding

The business objective remained the same: Identify survey questions related to the three target questions so that individual YMCA branches can evaluate how they can improve member satisfaction, likelihood to renew, and recommendations to friends. However, this time, a key part of the business objectives is making the insights transparent.

Decision trees are often used to gain insights into data because rules are easier to interpret than mathematical equations, especially if the rules are simple. The original three target variables were modeled directly instead of modeling IOE, in keeping with the desire to make the models as easy to understand as possible.

Data Preparation

Because the decision tree algorithm used in the analysis could handle missing values, no data preparation was done to recode NULLs or 0s. However, for each question, dummy variables were created to indicate if the responder checked the "1" box or not; these were the only inputs used in the models.

Modeling and Model Interpretation

We built decision trees using a CART-styled algorithm and determined complexity using cross-validation, a standard CART algorithm practice. The algorithm identified surrogate splits in the tree which we used to help understand the variables that were the most important to predict to target variables. Models were built for each of the three target variables: Satisfaction, Intend to renew, and Recommend to a friend.

Satisfaction Model

The tree for target variable Q1 = 1 (Satisfaction) is shown in Figure 13-9. If the question response was 1, the path of the tree goes down the left side of the split. The number at the bottom of a branch labels the terminal node and indicates it is one of the most significant terminal nodes. For example, terminal node 1 follows the rule Q25 = 1 and Q13 = 1, whereas terminal node 2 follows the rule Q25 = 1 and Q13 ≠1 and Q22 = 1. Table 13-11 shows a list of variables included in the tree.

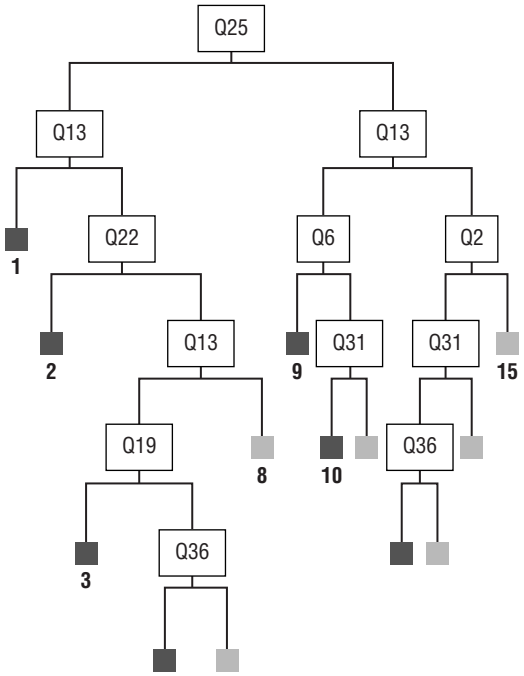

Figure 13-9: Member satisfaction tree

Table 13-11: Key Variables Included in the Satisfaction Tree

QUESTION	DESCRIPTION
Q2	Efficiency of front desk procedures
Q6	Staff competence
Q13	Overall cleanliness
Q19	Has the right equipment
Q22	Value for the money
Q25	You feel welcome at the Club.
Q31	Compared to other organizations in your community or companies you deal with, please rate your loyalty to the Club.

Characterizing the rules in English rather than in mathematical symbols is necessary to facilitate interpretation. Decision tree rules can be read as sequences of "and" conditions. The rules in the tree indicate the best way to predict the target variable according to the algorithm, though they aren't necessarily the only way to achieve comparable predictive accuracy; sometimes other combinations of splits in the tree can yield nearly identical accuracy.

Table 13-12 shows a summary report based on the tree in Figure 13-9. The key terminal nodes were defined as those terminal nodes with much higher than average satisfaction proportions and those with much lower than average satisfaction proportions. The rules defined by the branch of the tree were put into English to make it easier for decision-makers to understand what the rules were communicating about the member attitudes represented by the branch.

Table 13-12: Rule Descriptions for the Satisfaction Model

TERMINAL NODE	RULE
1	If strongly agree that facilities are clean and strongly agree that member feels welcome, then highly satisfied.
9	If strongly agree that facilities are clean, and strongly agree that staff is competent, even if don't strongly agree feel welcome, then highly satisfied.
2	If strongly agree that feel welcome and strongly agree Y is value for money, even if don't strongly agree facilities are clean, then highly satisfied.
3	If strongly agree that Y has the right equipment and strongly agree that feel welcome, and somewhat agree that facilities are clean, even though don't strongly feel Y is good value for the money, then highly satisfied.
10	If strongly agree that loyal to Y and strongly agree that facilities are clean, even though don't strongly agree that feel welcome nor strongly agree that staff is competent, then highly satisfied.
8	If don't strongly agree that facilities are clean and don't strongly agree that the Y is good value for the money, even though strongly agree that feel welcome, member isn't highly satisfied.
15	If don't strongly agree that staff is efficient and don't strongly agree that feel welcome, and don't strongly agree that the facilities are clean, then member isn't highly satisfied.

Finally, Table 13-13 shows several key terminal nodes from the satisfaction model, including two key statistics. The first key statistic is the proportion of the population with high satisfaction. The highest value is found in terminal node 1 (72.8 percent). The second key statistic is also important, however: the proportion of all highly satisfied members identified by the rule. The top terminal node by percent satisfaction is terminal node 1, but also important is that the rule finds nearly half of all highly satisfied members (49.1 percent). In fact, the top three rules, terminal nodes 1, 2, and 9, comprise over 70 percent of all highly satisfied members. The key questions included in these rules are summarized in Table 13-14, with a "yes" if the question has the value 1 in the

branch, "no" if the question has a value greater than 1 in the branch, and "NA" if the question is not in the branch.

Note as well that terminal node 15, the terminal node with the lowest satisfaction, is primarily the converse of the best rule: If the member doesn't agree that they feel welcome, the facilities are clean and the staff is efficient, only 6 percent of the members were highly satisfied with the branch; it makes one wonder why these 6 percent were still highly satisfied!

Table 13-13: Key Terminal Nodes in the Satisfaction Model

TERMINAL NODE	NUMBER OF SURVEYS IN NODE	PERCENT OF ALL SURVEYS FALLING INTO NODE	NUMBER OF HIGHLY SATISFIED IN TERMINAL NODE	PERCENT OF HIGHLY SATISFIED IN TERMINAL NODE	PERCENT OF ALL HIGHLY SATISFIED IN TERMINAL NODE
1	10,014	20.80	7,289	72.8%	49.1
9	1,739	3.60	904	52.0%	6.1
2	4,578	9.50	2,317	50.6%	15.5
3	1,014	2.11	471	46.5%	3.2
10	998	2.08	431	43.2%	2.9
8	1,364	2.80	141	10.3%	1.0
15	19,323	40.20	1,231	6.4%	8.3

The highest satisfaction comes from clean facilities and a feeling of being welcome at the Y. However, the branch can overcome a lack of cleanliness by demonstrating value in being a member of the Y (terminal node 2), and the branch can overcome a lack of feeling welcome with high staff competence. These "even if" insights from the decision tree provide nuance to the picture of why members are satisfied with their Y branch. These interaction effects cannot be learned or inferred from the regression models without introducing the interaction effects directly.

Table 13-14: Key Questions in Top Terminal Nodes

TERMINAL NODE	FEEL WELCOME	OVERALL CLEANLINESS	STAFF COMPETENCE	VALUE FOR MONEY
1	yes	yes	NA	NA
2	yes	no	NA	yes
9	no	yes	yes	NA

Recommend to a Friend Model

The Recommend to a friend model appears in Figure 13-10, which shows that the key questions are Feel welcome (Q25) and Loyal to the Y (Q31), with terminal node summary statistics shown in Table 13-15. The loyalty question was interesting because the other two models did not use this question at all, nor was it a competitor or surrogate split in the other models. The top rule—terminal node 1—found a population with 88.6 percent highly likely to recommend to a friend, comprising nearly half of all members who strongly recommend the Y. Once again, the top three terminal nodes represented more than 70 percent of the target variable population. These three terminal nodes (1, 2, and 4) included the questions related to Loyalty to the Y, Feel welcome, and Value for the money (Q31, Q25, and Q22). The descriptions of top rules for Recommend to a friend are shown in Table 13-16.

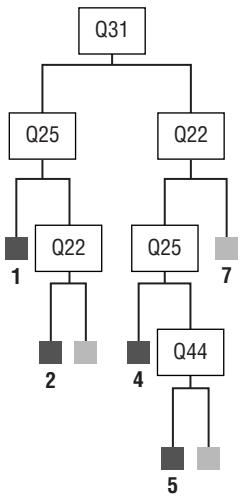

Figure 13-10: Recommend to a Friend decision tree

Table 13-15: Terminal Node Populations for the Recommend to a Friend Model

TERMINAL NODE	NUMBER OF SURVEYS IN NODE	PERCENT OF ALL SURVEYS FALLING INTO NODE	NUMBER OF HIGHLY RECOMMEND TO FRIEND IN TERMINAL NODE	PERCENT OF HIGHLY RECOMMEND TO FRIEND IN TERMINAL NODE	PERCENT OF ALL HIGHLY RECOMMEND TO FRIEND IN TERMINAL NODE
1	13678	28.46	12122	88.60	47.0
2	6637	13.80	4744	71.50	18.4

Continues

Table 13-15 (*continued*)

TERMINAL NODE	NUMBER OF SURVEYS IN NODE	PERCENT OF ALL SURVEYS FALLING INTO NODE	NUMBER OF HIGHLY RECOMMEND TO FRIEND IN TERMINAL NODE	PERCENT OF HIGHLY RECOMMEND TO FRIEND IN TERMINAL NODE	PERCENT OF ALL HIGHLY RECOMMEND TO FRIEND IN TERMINAL NODE
4	2628	5.50	1932	73.50	6.1
7	21865	45.50	5461	25.00	21.2
5	814	1.70	509	62.50	2.0

Table 13-16: Rule Descriptions for the Recommend to a Friend Model

TERMINAL NODE	RULE
1	If strongly agree that loyal to Y and strongly agree that feel welcome, then strongly agree that will recommend to a friend.
2	If strongly agree that loyal to Y and agree that Y is a good value for the money, even though don't strongly agree feel welcome, strongly agree will recommend to a friend.
4	If strongly agree that Y is a good value for the money and strongly agree that feel welcome, even though not strongly loyal to Y, strongly agree will recommend to a friend.
7	If don't strongly agree that loyal to Y and don't strongly agree that Y is value for the money, then will not highly recommend to a friend.
5	If strongly agree that Y is good value for the money, and strongly agree that Y helps meet fitness goals, even though not strongly loyal to the Y and don't strongly feel welcome, will highly recommend to a friend.

The interesting contrast of the Recommend to the Friend model compared to the satisfaction model is the inclusion of loyalty here; when a member feels a sense of loyalty to their branch, they are more likely to recommend it to a friend. If a member is loyal, if the member feels welcome or believes the Y is a good value for the dollar, they are likely to recommend it. But even without the loyalty, meeting fitness goals and good value for the dollar is enough for the member to recommend the Y to a friend.

Intend to Renew Model

The Intend to Renew model appears in Figure 13-11, which shows that the key questions were Feel welcome (Q25), Y helps to meet fitness goals (Q44), and Value for the money (Q22), with terminal node summary statistics shown in Table 13-17. Rules using the fitness goals question were interesting because the other

two models did not use this question in their top three terminal nodes, though it was in the fourth best terminal in the Recommend to a friend model, and in that same model was a surrogate for the Loyal to the Y split. As a reminder, surrogate splits identify a variable that splits the most like the winning split for a node in the tree.

The top rule—terminal node 1—found a population with 73.9 percent highly likely to intend to renew, comprising nearly half of all members who intended to renew. Once again, the top three terminal nodes represented more than 70 percent of the target variable population. The description of top rules for Intend to renew are shown in Table 13-18.

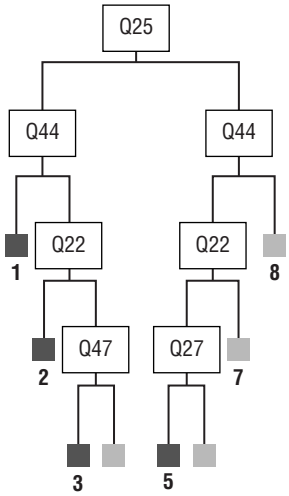

Figure 13-11: Intend to Renew decision tree

Table 13-17: Terminal Node Populations for the Intend to Renew Model

TERMINAL NODE	NUMBER OF SURVEYS IN NODE	PERCENT OF ALL SURVEYS FALLING INTO NODE	NUMBER OF HIGH INTEND TO RENEW IN TERMINAL NODE	PERCENT OF HIGH INTEND TO RENEW IN TERMINAL NODE	PERCENT OF ALL HIGH INTEND TO RENEW IN TERMINAL NODE
1	13397	27.90	9903	73.90	48.40
2	3051	6.30	1823	59.80	8.90
5	5704	11.90	3201	56.10	15.60
8	18547	38.60	3130	16.90	15.30
7	2178	4.50	578	26.50	2.80

Table 13-18: Rule Descriptions for the Intend to Renew Model

TERMINAL NODE	RULE
1	If strongly agree that feel welcome and strongly agree that Y helps meet fitness goals, then strongly agree that intend to renew.
2	If strongly agree Y is good value for the money and strongly agree that feel welcome, even if don't strongly agree that Y helps meet fitness goals, then strongly agree that intend to renew.
5	If strongly agree that feel sense of belonging, and agree that Y is value for the money, and strongly agree that Y helps meet fitness goals, even if don't feel welcome, then strongly agree intend to renew.
8	If don't strongly agree that feel welcome and don't strongly agree that Y helps meet fitness goals, then don't strongly agree that intend to renew.
7	If don't strongly agree that Y is good value for money and don't strongly agree that feel welcome, even if strongly agree Y helps meet fitness goals, don't strongly agree that intend to renew.

Fitness goals figured strongly in the Intend to renew models when coupled with feeling welcome (73.9 percent). Conversely, negating both of these reduced the Intent to renew to 16.9 percent, a ratio of more than four to one. Value for the money can help overcome fitness goals not being met, but only partially. Terminal nodes 2 and 5 had Intent to renew percentages of 59.8 and 56.1, respectively, well below the top terminal node.

Summary of Models

The Satisfaction model was more complex than the other two models, which implied that the reasons for satisfaction are more complex. Each of the models captured different characteristics of the members. The differences also highlight the importance of defining the target variable well. For this project, all three of the target variables provided insights into the attitudes of members.

Feel welcome was a top question for all three models and provided an important insight into the mindset of the members; the connection to the Y was more important than the conditions of the facility, parking, programs, or staff competence. Value for the money was also a key question in all three models, showing that member costs were significant contributors to the attitudes of members toward their branch.

However, each model also had one or two questions that differed from the other models, summarized in Table 13-19. These differences could be useful in helping decision-makers tune the changes to the target they are most interested in. For example, if the branch wishes to improve member satisfaction, in addition to making members feel welcome and revisiting the value of the Y,

they can make sure staff is trained well (Staff competence) and the facilities are kept clean.

Table 13-19: Key Questions That Differ between Target Variables.

MODEL	KEY QUESTION(S)
Satisfaction	Facility cleanliness, Staff competence
Recommend to a friend	Loyalty to the Y
Intend to renew	Y helps to meet fitness goals

Deployment

The models were not deployed nationwide; individual branches decided to what degree the models were used to influence what changes were made within their branches. After years of using models, a 32 percent improvement in satisfaction (Q1 = 1) was measured, which clearly indicates improvement at the branch level in meeting the expectations of members. Additionally, the Recommend to a friend (Q48 = 1) improved by 6 percent, easily a statistically significant improvement based on the number of surveys, though operationally not much higher than the original value.

Summary and Conclusions

Two predictive modeling approaches were described in this chapter, the first a mostly traditional approach to modeling, including the use of factor analysis and stepwise regression. The model visualization reports provided a rich, branch-level summary of the status of key questions on the survey that influence satisfaction. It ultimately was not used because of its complexity.

The second took a machine learning approach, building decision trees to predict the target variable. Simple rules were built that provided a more transparent view of which questions influence Satisfaction, Intend to renew, and Recommend to a friend, including how much interactions in key questions relate to these three target questions.

Neither modeling approach is right or wrong; they provide different ways to understand the surveys in complementary ways. A more sophisticated approach could even incorporate both to find main effects and interactions related to IOE and its components. Given more time and resources, more avenues of analysis could have been attempted, but in most projects, time and resources are limited, requiring the analysts to make decisions that may not be optimum, but hopefully are reasonable and will help the company improve decisions.

This case study also demonstrates the iterative nature of predictive modeling solutions. The data was plentiful and relatively clean, and the target variables were well defined. Nevertheless, translating the insights from the models was not straightforward and required considerable thought before communicating these insights to the decision makers. Success in most predictive modeling projects hinge on how well the information gleaned from the models—both predictions and interpretations—can ultimately be leveraged.

Since the completion of work described in this case study, Seer Analytics has progressed well beyond these models and has developed a framework that addresses the target variable from a different perspective; they readdressed the objectives of modeling to better match the business needs. Instead of predicting influencers for factors that drive satisfaction, they now define six key components of the member experience and create a pyramid of factors that are hierarchical. These six factors are, in order from bottom to top, Facility, Value, Service (of staff), Engagement (with staff and members), Health (meeting goals), and Involvement. This approach has resonated much better with YMCA branches and is in use today.

Help Desk Case Study

In the second case study, a combination of text mining and predictive modeling approaches were used to improve the efficiency of the help desk of a large U.S. corporation responsible for hardware services and repairs of devices it manufactured, sold, and supported.

The problem the company needed to address is this: Could the company use the description of problems transcribed from help desk calls to predict if a part would be needed to resolve the problem. After receiving a help desk phone call and after the call was processed, a support engineer was assigned to the ticket to try to resolve it. The engineer first tried to resolve the problem over the phone, and if that wasn't successful, the engineer went to the customer site to resolve the ticket.

In particular, the problem was efficiency. Knowing whether a part was needed for a repair or not before going to the customer site was helpful. Even more important was predicting which parts were most likely to be needed in the repair. These models were built for the modeling project but will not be described in this case study. A second set of models were built to predict the actual part that was needed to complete the repair, though that part of the modeling is not described here.

Concerns over revealing too much to competitors prevents me from revealing the client. Additionally, specifics about the project are not provided, such as actual performance numbers, precise definitions of derived variables, and

the identities of the most predictive variables in the models. The project was so successful that the details about the models and how much the models improved efficiency became a strategic corporate asset. Nevertheless, even without the details, the principles used to solve the problem can be applied broadly.

The accuracy of the model was paramount for successful deployment. The decision makers determined the minimum accuracy of the "parts needed" model for the model to be successful, a number that cannot be revealed here. The entire data set did not need to be classified at this rate. If even 20 percent of the tickets that needed a part to complete the repair could be identified correctly, the model could be successfully deployed. The remaining tickets would then be processed as they had always been processed.

Data Understanding: Defining the Data

The unit of analysis was a support ticket, so each row contained a unique ticket ID and columns contained descriptions of the ticket. More than 1,000,000 tickets were available for modeling. For each support ticket, the following variables were available as candidate inputs for models:

- Date and time of the call
- Business/individual name that generated the ticket
- Country of origin
- Device error codes for the hardware
- The reason for the call that generated the ticket: regularly scheduled maintenance call or a hardware problem
- Warranty information
- The problem description (text transcribed from the call)
- The outcome of the ticket: ticket closed, solution found, and so on
- The part(s) needed to close the ticket

The target variable had two parts: Did the ticket require a part to close, and which part or parts were needed to close the ticket. Only the PartsUsed target variable models are described in this case study.

Data Preparation

Extensive preparation was done on the data to examine all codes, correcting incorrect codes when they were identified as incorrect. The date of the call was transformed into a single column: day of week. The business name was not used in building models, but helped in interpreting them.

Problems with the Target Variable

The target variable definition was clear: Create a 1/0 dummy variable indicating if the ticket required a part to complete the repair. The target variable was labeled the PartsUsed flag. There were, however, ambiguities even with this definition. First, if a part was not necessary for the repair, although it helped the repair, it was still coded as a part being used.

In other cases, a part was used to fix a problem that was not the one specified on the ticket. If this situation was discovered, the PartsUsed flag was set to 0 because a part the engineer specified to close the ticket was not actually needed to fix the problem on the ticket, and the engineer should have opened a new ticket for the second problem. These ambiguities were difficult to discover, however, and it is presumed that these problems persisted in the data.

A third problem occurred when a fix was made using a part and the ticket was closed, only to reopen later when the problem reappeared. If the problem was subsequently solved using a different part, the PartsUsed flag would still be coded as a 1 (though a different part was used to fix the problem). However, if a part was not needed to fix the problem, the PartsUsed flag would have to be changed to a 0. If the ticket was coded properly, this would work itself out.

Feature Creation for Text

The company realized early on in the project that extracting features from the transcribed text was critical to the success of the models. Initially, text was extracted using SQL text matching rather than a text mining software package, and it was done quite well. After finding some success using SQL to extract text, they initiated a search for text mining software and technology to give it the ability to leverage the transcription of help desk calls.

Problems with the text were those problems common in text mining including misspellings, synonyms, and abbreviations. Fortunately, the client had an excellent database programmer who spent considerable time fixing these problems.

Table 13-20 shows a sample of the kinds of misspellings found in the data. The database programmer not only recoded these to a common spelling, but even without any knowledge of stemming, he essentially stemmed these words, capturing the concepts into one keyword. The keyword STICK from the table therefore included all of the variants listed in the table.

Table 13-20: Data Preparation for Help Desk Text

ID	WORD	KEYWORD	COUNT
1220	STCK	STICK	19
1221	STIC	STICK	16

1222	STICK	STICK	61
1223	STICKIN	STICK	15
1224	STICKING	STICK	1,141
1225	STICKS	STICK	179
1226	STICKY	STICK	176

As a second example, the keyword FAILURE included words FAILURE, FAIL, FAILURES, FAILED, and FA found in the text. These words were combinations of inflected words and abbreviations.

Domain experts and database programmers worked together to build a list of keywords deemed to be potentially useful in building the predictive models. After combining synonyms, stemming, and converting abbreviations to the appropriate keyword, the team identified more than 600 keywords and phrases.

Modeling

Several algorithms could have been built to achieve the business objectives, including neural networks, support vector machines, and decision trees, but two considerations pushed the team to select decision trees. First, the data was largely categorical and the number of candidate inputs was very large, two areas decision trees handle well. Second, the company needed to build models that were easy to interpret so support engineers could understand why a part was predicted to be needed for the repair. For these reasons, decision trees were used for modeling.

Hundreds of decision trees were built from the modeling data using different partitions of training and testing data records, different variable subsets of keywords, and different tree settings for priors and complexity penalties. The tree in Figure 13-12 is typical of the trees that were built. The column name for the PartsUsed target variable in the tree is Parts. Terminal nodes had the percentage of records needing a part (Class equal to Parts) ranging from a low of 1.5 percent in Terminal Node 2 to 50.5 percent in Terminal Node 8. The nodes in Figure 13-12 are color coded by percentage of records needing a part to facilitate seeing where the percentages are highest.

However, 50.5 percent fell far short of the minimum value required in the business objectives, even though the 50.5 percent Parts rate represented a lift of more than two over the baseline rate of 23.2 percent. This tree and the hundreds of others were clearly insufficient to achieve the goals of the model.

What was the problem? The trees were able to handle the large number of keywords but struggled to find combinations that produced high percentages of tickets with parts use. Decision trees sometimes struggle with sparse data

because of the greedy search strategy: Each keyword dummy variable was a sparse variable, populated in a small minority of tickets, and individual keywords didn't necessarily provide enough information on their own to create good splits.

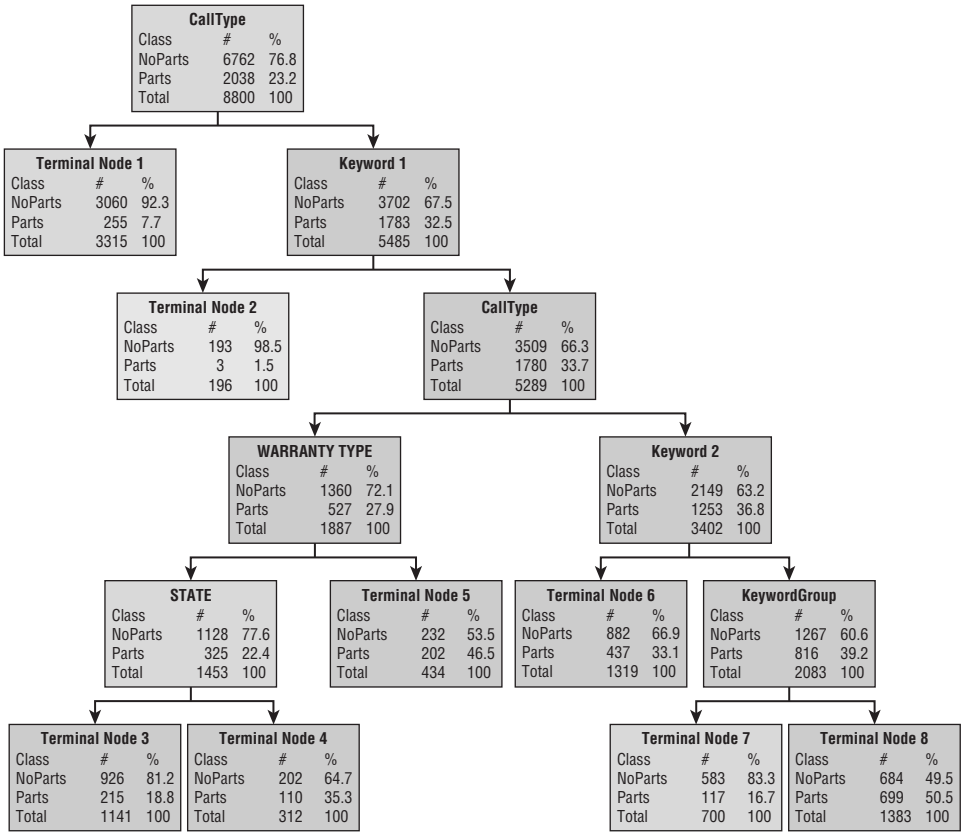

Figure 13-12: Typical parts prediction decision tree

Model ensembles, such as Random Forests, helped the accuracy some, but still not enough to achieve the business objective. Random Forests, in particular, represent a good strategy for this kind of problem because of the random variable subsets that are part of the algorithm, forcing the trees to find many ways to achieve high accuracy. But the modest improvement in accuracy also came at the expense of transparency of the rules, an important factor to communicate to the support engineer before going to the company site; even if the Random Forests solution was accurate enough, it is unlikely the company would have adopted it as the model to deploy.

Some trees were very interesting because they did not determine if parts were needed from the keywords, but rather from the business making the call,

the level of expertise of their internal support within the company making the support call, or specific devices needing repair. Trees, to some degree, were therefore measuring which companies had better internal support (if they had good support, parts were more likely to be needed because they fixed the easy problems themselves).

The best branches in the trees were actually ones that predicted a part *not* being needed. For example, in one tree, when both keywords "machine" and "down" existed in the ticket, parts were rarely needed for the repair. Rules like this one were valuable, but not the kind of rule that the company needed for successful deployment.

Revisit Business Understanding

The research team went back to the drawing board to try to find a different way to solve the problem. They asked good questions, such as "What additional information is known about problems at the time tickets are submitted?" and "How would an informed person think about the data to solve the problem?" The answers to these questions were in the historical context of the tickets themselves.

When a ticket came in, a support engineer who was trying to predict what the problem might be and whether a part was needed for a repair would refine their conclusion by eliminating options that were not likely, just as decision trees do. But the engineer would also group together problems naturally, using an "or" condition, an operation that trees do not do in a single split. But even these conditions did something subtle in the mind of the support engineer.

The keywords and other features represented a historical idea: the experience of the engineer. The experiences of the engineer also included temporal information; one combination of keywords may have been a problem for particular kinds of devices at one time, but two years prior to that time the problem didn't exist.

The analysts and decision makers then determined the following. Rather than using codes and keywords directly, they would use historic information about parts being needed when those codes and keywords appeared in the ticket as inputs. Let's say "paper jam" was a keyword in the data. Rather than using this as a dummy variable, use the percentage of times this pair needed a part in the past year.

Figure 13-13 is a visual representation of the thought process. Ticket 1 was resolved at some timestamp prior to "Now," where "Now" represents the date the data for modeling was created. The date of resolution for Ticket 1, called "Ticket 1 Timestamp" in the figure, means Ticket 1 has a target variable. The input variables for Ticket 1, PartsUsed percentages for keywords and codes in Ticket 1, were generated for the time period range represented by the curly bracket with the label "Time Period to Compute PartsUsed for Keywords in Ticket 1."

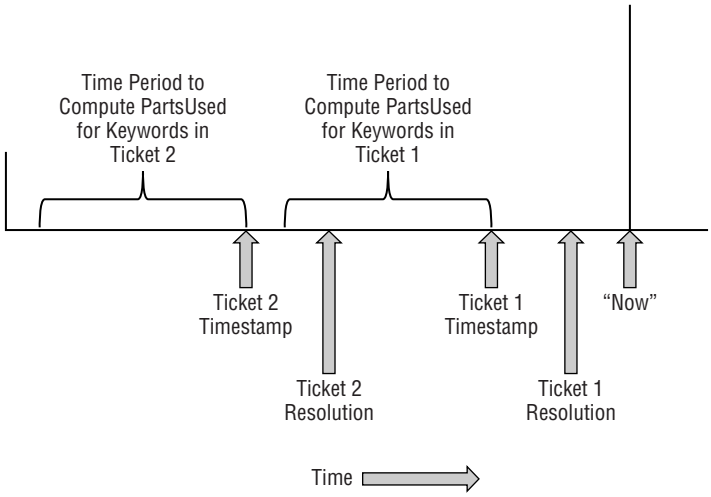

Figure 13-13: Temporal framework for new features

The same process was done for each ticket, including Ticket 2 shown in the figure, and all other tickets (more than one million).

What's the difference between this and just using the flag? What is the difference between this data and the data used in the first modeling pass? After all, doesn't the tree compute the average PartsUsed rate for each keyword in the data and show us those in the nodes and terminal nodes of the tree?

First, the new representation contains richer information: instead of a 1/0 dummy, you have a percentage. Second, there is additional temporal information in the new features missing from the dummy variables. When models were built from the dummy variables, all occurrences of the keywords and error codes were included in the tree regardless of when the ticket occurred. In the new representation, tickets occurring after the date for Ticket 1 ("Ticket 1 Resolution") are not included in the computation of the PartsUsed percentage. Therefore, the new variables take trends into account in ways the first data set couldn't; the first models included all tickets in the modeling data regardless of when the tickets occurred.

New features included not only historic PartsUsed, but also counts of how many tickets contained the keyword or code.

Modeling and Model Interpretation

Decision trees were used once again for building the predictive models because they could scale well. In addition, many decision trees could be built easily by modifying parameters, such as priors or misclassification costs and the set of inputs that could be included in the models.

Nevertheless, the trees still did not achieve the classification accuracy required by the business objectives if measured by Percent Correct Classification (PCC) or by using a confusion matrix. What was new was that there were finally at least some terminal nodes in most of the trees that did meet the business objective PartsUsed rate, even if only for a small percentage of the overall records.

When examining these terminal nodes, the analysts discovered that the rules describing the path to the terminal nodes differed from one another, as one would expect from decision trees. Moreover, these terminal nodes were not identifying the same populations: There were more than a few ways to predict a high percent of PartsUsed for different groups of tickets. Interestingly, some of the rules contained six or more conditions (the trees were six or more levels deep), but even in these situations, the rules made sense to domain experts.

At this point, an astute analyst may see that this kind of data is ideal for ensembles. Random Forests (RF), in particular, is a good match for generating trees from different input variables and achieving higher accuracy than individual trees can achieve. However, even though the RF models had higher accuracy than individual trees, the RF models had two problems. First, the full trees did not achieve overall accuracy that was high enough to deploy. Second, the individual branches of RF trees were (purposefully) overfit and unsuitable to use as a way to interpret why a part was needed.

Therefore, the rules that showed the most potential had to be plucked from thousands of trees. The most interesting rules were those that related to high percentages of PartsUsed, but the rules matching very low PartsUsed percentages were also interesting as cases that could be solved without parts being needed and therefore were good candidates for phone support.

An alternative to picking the best rules from decision trees is to use association rules. Advantages of association rules are that they find (exhaustively) all combinations of rules rather than just the best paths found by decision trees. A disadvantage is that the inputs must all be categorical, so the PartsUsed percentages would have to be binned to create the categorical representation for every keyword and code.

Deployment

More than 10,000 decision trees were built, creating more than 20,000 terminal nodes with a high percentage of PartsUsed to complete the repair. Each terminal represents a rule—a series of "and" conditions found by the decision tree. These rules were collected and sorted by the predicted percent PartsUsed for the terminal node. The sorted list of rules then was treated as a sequence of rules to fire, from highest PartsUsed likelihood to lowest, though even the lowest contained a high percentage of PartsUsed.

The algorithm for applying rules in this way can be thought of in this way:

1. Sort the rules by percent PartsUsed on training data in descending order. Each rule contains not only the PartsUsed percentage, but also the variables and splits (the rules) used to generate the PartsUsed percentage.
2. Choose one or more tickets to apply the rules to.
3. Run the first ticket through the rules, stopping when a rule matches (or "fires") with the highest PartsUsed percentage.
4. Repeat for all tickets in the set.

Table 13-21 contains a sample list of rules. As a reminder, these numbers are not real and are used for illustration purposes only; they should not be used to infer the PartsUsed percentages found and used by the company. The PartsUsed percentage found during training is shown in the table, but the more important number is the PartsUsed Percentage in Sequence, a number that can be significantly different than the Training PartsUsed Percentage.

Table 13-21: Sample List of Rules to Fire

RULE ID	RULE SEQUENCE NUMBER	NUMBER TICKETS MATCHING RULE	TRAINING PARTSUSED PERCENTAGE	PARTSUSED PERCENTAGE IN SEQUENCE	CUMULATIVE PARTSUSED PERCENTAGE
278	1	11809	93.6	93.6	93.56
2255	2	7215	93.5	91.7	92.85
1693	3	1927	93.5	93.0	92.85
2258	4	16	93.5	70.5	92.84
1337	5	20	93.5	89.3	92.83
993	6	2727	93.5	84.1	91.82
977	7	2	93.5	94.0	91.82
2134	8	2516	93.5	92.8	91.91
1783	9	4670	93.5	93.1	92.09
984	10	6	93.5	94.0	92.09

For example, consider Rule ID 2258. This rule matched only 16 tickets because the thousands of tickets it matched during training already matched one of the prior rules in the sequence: 278, 2255, or 1693. Moreover, the best tickets from rule 2258 were gone: The remaining 16 tickets had only a 70.5 percent PartsUsed rate. This rule should therefore be pruned from the list. The same applies to rules 1337, 977, and 984.

For the company in this case study, these rules were particularly attractive because they could be converted to SQL so easily, making deployment simple.

Summary and Conclusions

Several aspects of this case study were critical to the success of the project. First, deriving new, innovative features to use for modeling made the difference between failure and success. Sometimes one hears that more data will overcome algorithms and domain expertise. In this case, this was clearly false. It wouldn't matter how many tickets were included in the models. If the new features weren't created, the performance would still be below the requirements. Subject matter experts were essential to defining the new features.

Second, using decision trees allows the modeling to proceed quickly because trees handle wide data so easily and efficiently. Moreover, adjusting learning parameters for the trees produced thousands of trees and interesting terminal nodes. Third, the decision trees were not used blindly and as black boxes. When they were considered feature creators themselves, finding interesting business rules, the fact that no tree solved the entire problem was irrelevant. Finally, using the decision trees to find business rules meant the list of rules was intuitive and easy to communicate to the support engineers.

The models went into production and were very successful for the company, and so much so that the modeling was expanded to try to include more tickets in the models and to produce models that matched more tickets with high PartsUsed scores. Cost savings for the company as a result of the models were significant.