Practical Challenges of Portfolio Optimization

A Fair Isaac White Paper

Mac Belniak

May 2003

1 800 999 2955 *from the US*   1 415 472 2211 *from anywhere*   info@fairisaac.com *email*

www.fairisaac.com

Table of Contents

Introduction

The word "optimization" is often used informally to describe any technology that improves business results. In more formal terms, optimization is a mathematical methodology used to make decisions that achieve an overall objective by allocating finite resources subject to the constraints imposed by the environment. It is an analytic methodology that can be used to unlock value for people, business, and organizations.

Fair Isaac leverages optimization to help customers develop superior decisioning strategies. Traditionally, methods such as champion-challenger have been applied to evolve decision strategies. A decision maker works to develop a candidate strategy (or "challenger") that has the potential to surpass the status quo (or "champion"). Analytics and intuition are used to assign actions that may yield better results while still adhering to constraints. The challenger is then tested against the current decision strategy to determine which strategy is best. The decision strategy is improved iteratively by developing and testing new challengers. Much iteration is required to find the best possible strategy. This has made the process of improving a decision strategy time-consuming. An approach based on optimization provides a path to finding the best possible challenger strategy much faster.

While the theory of optimization is well studied by academics in fields ranging from operations research to computer science, there are many practical challenges when applying optimization to develop superior decisioning strategies. For the first time, Fair Isaac is offering directly to its customers the optimization software environment and methodology that it has developed to address the unique challenges. This solution goes beyond optimization software components and algorithms to provide a workspace for developing and deploying the best challenger strategy. Strategy Science and the Business Science tools supporting it empower statisticians, data miners, analysts, and business users to apply optimization. Before Strategy Science, only those teams with software engineering capability and optimization expertise could leverage global optimization by weaving together solvers and software components using languages like Java or C++.

This whitepaper begins with an overview of optimization. It then discusses the common issues that Fair Isaac faces when applying optimization. By exploring these issues, we develop an understanding of what Fair Isaac means by optimization and what is required of optimization software. Next it explains the Strategy Science methodology Fair Isaac uses to approach optimization problems and the benefits of using this approach. Finally, it discusses how the analytic software, Decision Optimizer, is designed to conquer the practical challenges. The whitepaper also has a sidebar that shows how Strategy Science leverages prior investment in testing, analytics, and predictive models to significantly impact performance.

Although optimization broadly addresses applications in a range of industries, an example from credit card account management is used to clarify important issues throughout the whitepaper. The example assumes that there is a bank with a portfolio of credit card customers. The bank has an overarching goal of maximizing lifetime profit from the portfolio of card customers. For each individual customer, the bank can periodically make adjustments to things like credit limits and interest rates. The adjustments drive the behavior of the customers in terms of, utilization, delinquency, loss, attrition, and ultimately profitability. Deciding on the appropriate pattern of credit limits and interest rates for each customer is challenging. The key challenge is not merely to maximize profitability. It is to do so while ensuring hard limits across the entire portfolio on things like delinquency, loss exposure, attrition, loss ratio, and receivables growth while managing uncertainty about the performance of individual customers and the portfolio as a whole.

Optimization Basics

There are many standard, and non-standard, terms used when discussing optimization problems. As a baseline, the critical terms needed to understand the Fair Isaac approach to optimization are defined.

Variables in an optimization model are either non-controllable or controllable. Non-controllable variables contain the information that is known before the optimization. Thus, in customer account management, non-controllable variables describe demographic data such as income and education as well as payment and purchase histories. Controllable variables are those that can be adjusted by the optimization to achieve the desired goal. For example, controllable variables in retention management might represent the decisions for each account to offer a balance transfer and the introductory interest rate on the transfer. Controllable variables are often called **decision variables** because a particular setting of the controllable variables represents a decision.

Constraints restrict the allowable values for the decision variables. In commercial marketing applications, we might mandate the length of intervals between mailings and sales visits, the order of the solicitations made, or the price offered to the customer. In managing accounts, we may enforce the card issuer's policies via constraints. Some constraints directly restrict the decision variables. Perhaps all APR's are mandated to be above 8%. In other cases, constraints restrict the decision variable indirectly, e.g. there may be a constraint limiting the total expected losses due to defaults in the next year.

Almost all optimization problems have a single **objective function**. This is the measure or metric that we seek to minimize or maximize. For instance, in many business applications, we want to maximize net profit or, if revenues are not impacted by the decisions being modeled, to minimize total cost. In account management, we seek to maximize the value of the portfolio over time.

Some problems have multiple objectives. For instance, in the customer management problem, we may choose to minimize the risk of losses while maximizing revenues. In this case, the **multiple objectives** oppose each other. In practice, problems with multiple objectives can be reformulated to have a single objective. The customer retention objective could be reformulated as maximize revenues subject to the constraint that losses not exceed $X million. In fact, this type of reformulation sets the stage for a **sensitivity analysis** that can yield valuable insights about the trade-off between profit and the loss threshold value.

The **optimization problem** is to find values for the decision variables that minimize or maximize the objective function while satisfying the constraints. The broadest term for the subject of constrained optimization is mathematical programming. Despite its name, mathematical programming has little to do with computer programming. Programming is a British word for scheduling that was chosen since scheduling problems had been solved as math programs. Some of the specialties within the field of math programming include linear programming, nonlinear programming, integer and mixed-integer programming. Each class of problem has its own characteristics and solution techniques. We refer to all of these as **constrained optimization** problems.

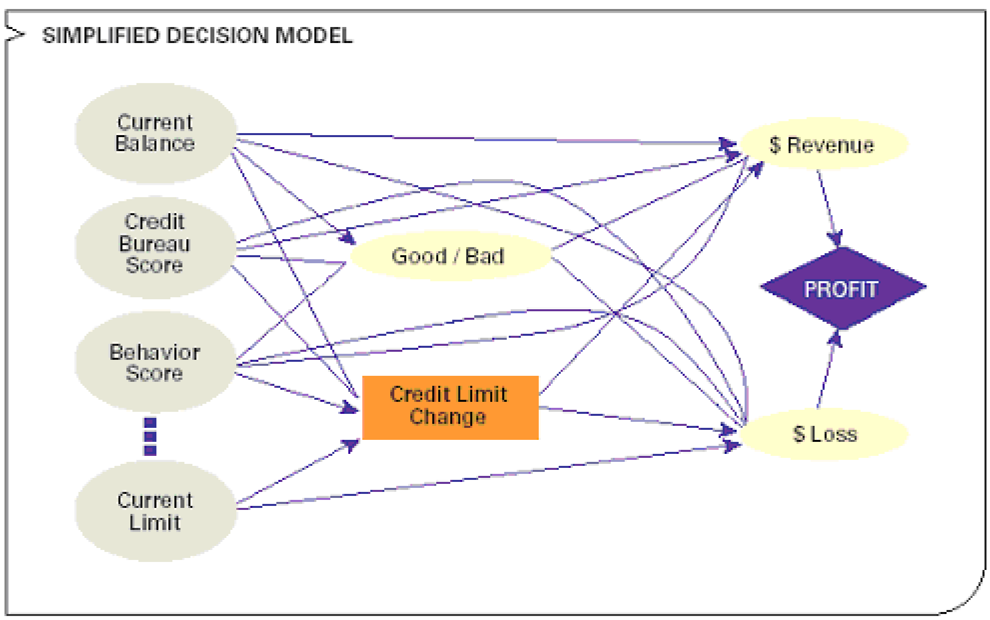
Typically, optimization problems assume that the values of the non-controllable variables are known with certainty. For many problems, this is not the case. Within the literature of Management Science and Operations Research, we find the concept of **Robust Optimization**. It is defined as constrained optimization that seeks good solutions across a range of scenarios. For account management, scenarios could be defined to represent uncertainty about upcoming Federal Reserve rate changes. Considering only one scenario can lead to a solution that performs quite badly if there are significant uncertainties that affect the objective. This is because the future may differ significantly from the scenario considered by the optimization.

Optimization in Practice

Although the basic principals of constrained optimization are understood, applying optimization raises a number of practical issues. By exploring the issues, we can develop an understanding of what capabilities should be present in decision optimization software.

Formulating an Optimization Model

Optimization juggles a multitude of entities such as goals, policies, constraints, historical data, facts, scores, predictions, forecasts, and uncertainties. All of this information must be marshaled into an optimization that achieves a goal and adheres to constraints while considering uncertainties. Formulating a well-structured optimization problem from this wealth of business knowledge is quite challenging and often requires skills and expertise in operations research and computer programming. It is useful to introduce a framework for this task from the field of decision analysis called a Decision Model**.** Decision Models are elsewhere called Influence Diagrams. They incorporate all of the information that is important for formulating and executing optimization. A **Decision Model** maps the relationship between information such as historical data and facts to predictions, possible actions to their effects and likely outcomes, outcomes to business goals, and finally relates everything to constraints. In this sense, a Decision Model leverages previous investment in data collection, predictive modeling, and scores as it incorporates this disparate information into a unified model of the decision situation facing the business. Decision Models abstract the complexity of formulating optimization problems to make advanced optimization accessible to statisticians, analysts, and businesspersons. Decision models can also be represented graphically as Influence Diagrams (Clemen 1990).



In this credit line management example, the decision model identifies how each customer (with a certain Current Balance, Credit Bureau Score, etc.) is likely to respond (in terms of Revenue and Loss) to the organization's action (Credit Limit Change), and what the result will be (Profit). Based on this Decision Model, we can analytically determine which action to take for each customer in order to optimize profit, given business constraints on the portfolio such as acceptable losses or revenue targets.

Multiple Decisions and Treatments

Most decision situations involve two or more decisions. In the retention example there are many decisions, e.g. offer a balance transfer, offer a line increase, introductory rate, introductory period, go-to rate, etc. Each decision has a handful of alternatives for each customer, e.g. increase the line $200, $1000, $2000, or $5000. In most practical cases, it is not limiting to assume that decision alternatives are discrete and can be explicitly enumerated, i.e. no continuous decision variables as in a linear program. When there are multiple decisions, it is practical to introduce the concept of treatments. Treatments specify combinations of these decisions, e.g. offer balance transfer at 0% APR for six months and then go to a rate of 12.9%.

Sequential Decisions

In the above example, all of the decisions occur at a single point in time. Many decision optimizations can benefit from considering a sequence of decisions. In the retention example, there is likely to be an opportunity to extend another transfer offer at a later date.

Sequential decision models are important because when choosing an alternative today, the decision maker has to know what additional alternatives she will have available to choose from in the future, as well as what type of information will be observable *then* which is still uncertain *now*. The key to deriving benefit from this is that more is known about the customer before making the second decision in the sequence, e.g. the customer has performed well after accepting the first balance transfer offer.

If the optimization only considers the present decision and past information, the choice could be less than optimal over a longer period. For example optimizing the balance transfer offer without thinking about subsequent decisions may lead to overly conservative decisions, because the opportunity to observe performance and limit losses with subsequent decisions is ignored. The best choice for the current decision may be quite different, and the long-run objectives significantly improved, if the optimization considers today all, or at least most, of the future decision making opportunities. Such a consideration can only be addressed explicitly and quantitatively in a sequential decision model. Sequential decision models can be very valuable, but can be significantly more complicated to construct with the data limitations that exist in many practical applications.

Constraints

Another practical challenge is that organizations often have policies that limit the alternatives that can be considered. These constraints come in two varieties, record-level and cross-record. Record-level constraints are those that place restrictions one record at a time. Sometimes these are also called local constraints. In account management, there is one record per customer. There are a number of policies that apply at the account level, e.g. never send an offer to a customer two months in a row. Record-level constraints may also be used to specify treatments (combinations of decisions), e.g. never offer a treatment with a go-to rate that is lower than the introductory rate.

A common practice is to segment records before optimization. This is often done so that predictive models can be tuned for individual segments or so that segments can be treated differently. For example, assume that each record has a variable that assigns the customer to a segment. Then there could be a record-level constraint such as, if the customer is in the high-revenue low-risk segment never increase the APR. The key to identifying record-level constraints is that they can be applied one record at a time without considering the decisions made for other records. In general, record-level constraints do not add significant complexity to the optimization process.

The second type of constraint is the cross-record constraint. Cross-record constraints are fundamentally different. It is not possible to look at only one record of information at a time and apply such a constraint. In retention management, there are a handful of cross-record constraints. Some apply across the entire portfolio, e.g. marketing budget of $X, losses for the portfolio must be less than $Y next year. When cross-record constraints apply to all the records they are sometimes called global or portfolio-level constraints. Other cross-record constraints might apply only across specific segments, e.g. losses for the high-risk low-reward segment must be less than $Z next year. The values of all the controllable and non-controllable variables must be known in order to check whether a cross-record constraint has been satisfied. Constraints that apply across the entire portfolio, across segments, or across more than a handful of records add significant complexity to the optimization problem. In practice there are usually fewer than 20 such constraints.

While business rules management technologies and many optimization packages are capable of enforcing and accounting for record-level constraints, they are not capable of finding the optimal decision strategy in the presence of cross-record constraints. However, once an optimal decision strategy has been developed, business rules management software can be used to maintain and execute the strategy, allowing it to be put easily into production.

Cash Flows and Forecasts

There is a subtlety in computing the values of objectives and constraints. The best objectives are often measured over longer terms to cover the full effects of the decisions. It is desirable not only to have a summary of that objective at the current point in time, e.g. net present profit, but also to have the ability to see how decisions affect that objective at future points in time. This requires a framework for working with forecasts and discounted cash flows. For example, in account management, the portfolio manager may want to model how balances evolve over time based on the decisions made.

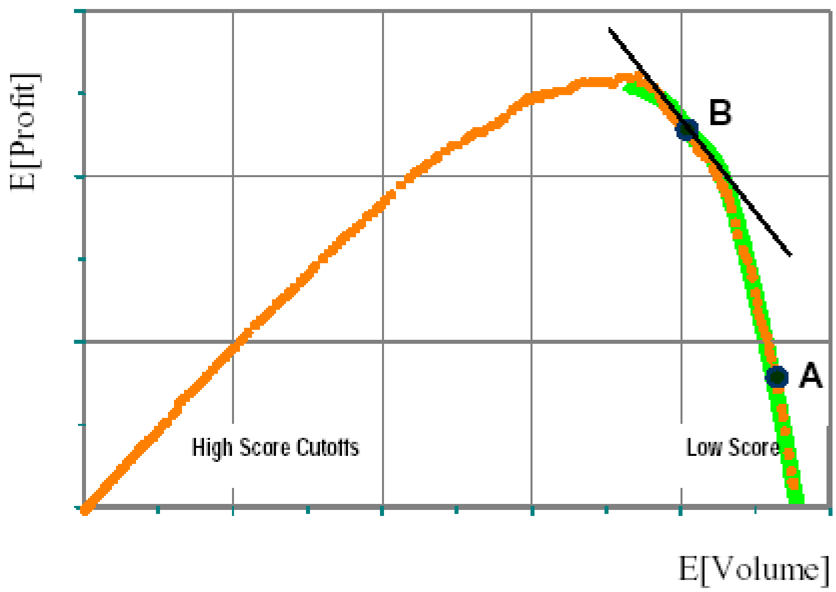
Multiple Objectives

A key issue arises when specifying the objectives for an optimization. The objectives are the business or organizational goals. The first challenge is that there are almost always multiple conflicting goals, for example, maximizing revenue and minimizing risk. Another example is maximizing profit while maintaining acceptance volume of new loans. In practice, problems with multiple objectives can be reformulated to have a single objective either by forming a weighted combination of the different objectives (where the weights represent their relative importance), or by representing some of the objectives as constraints whenever those objectives must attain known desired values. The account management objective could be reformulated as maximize revenues subject to the constraint that losses not exceed $X million.

Sensitivity Analysis

Framing an optimization problem in terms of a single objective and multiple constraints sets the stage for a sensitivity analysis that can yield valuable insights. By repeating the optimization with different loss constraint thresholds, the trade-off between profit and the threshold can be evaluated.

Consider the trade-off between expected-Volume and expected-Profit associated with an accept-reject policy in a credit portfolio, using a single risk score. It illustrates that the lower the score cutoff, above which applicants are accepted, the higher the volume. In the high range of score values, where decreasing the cutoff mainly accepts "Good" applicants, the profit also increases. In the low range, however, continued decrease of the score cutoff results in reduced profit because more and more "Bad" applicants are accepted. Only the green portion of the frontier is efficient in these two dimensions, in the sense that, for any given level of expected Volume, all decision makers would prefer a higher, rather than a lower, expected Profit.



The choice of the operating point on the efficient frontier can be selected judgmentally, by considering the subjective trade-off between profit and volume. A portfolio manager for which volume is relatively more important may choose a lower score cutoff, corresponding to point A, say, while another for which volume is relatively less important will choose a higher score cutoff, corresponding to point B. In fact, if the portfolio manager can assess the trade-off factor between Profit and Volume, the optimal point on the efficient frontier would be the one where the slope of the tangent line equals the trade-off value.

Stress Testing

Sensitivity analysis demonstrates how the objective value and optimal strategy changes across different constraint thresholds. Stress testing shows how a given strategy and its objective are affected by changes in key variables and uncertainties. Stress testing can be used to evaluate how robust the objective value of the optimal strategy is. For example, assume that there is an optimal decision strategy that is expected to make a profit of $Y million, but one would like to know the range or distribution of profits to expect when executing the strategy. Stress testing can compute the objective, profit, for a wide range of values for the uncertain driving factors, thus yielding a range of profits. It is desirable to have a strategy where the objective is robust in the face of a wide range of circumstances.

Uncertainty

Another practical aspect of many optimization problems is managing uncertainty. As with constraints, there are two basic types of uncertainty. First is uncertainty at the record-level. In account management there is uncertainty surrounding the performance of an individual customer. This uncertainty is independent of the performance of the other customers. Second is uncertainty that affects the behavior of the entire portfolio simultaneously in the same way. For example, an unexpected economic downturn may curb spending for everyone in the portfolio or segment at the same time. It is important to account for both types of uncertainty when optimizing decisions. This helps to ensure that the decision strategy will be robust.

The academic literature offers several approaches to such decision making under uncertainty within the context of constrained optimization. We favor Robust Optimization techniques (Mulvey, 1995). Stochastic robust optimization provides an effective means of representing uncertainty, if it's possible to generate appropriate sets of scenarios and scenario probabilities. This approach seeks solutions that maximize the expected value of the objective function and therefore provide the best answer across all scenarios (Clemen 1990). Ideally the scenarios are formed and their probabilities are assigned in harmony with the experience of the decision maker as well as the patterns that are evident in the available data.

There are also different formulations for modeling risk neutrality or risk aversion on the part of the organization. For example, in account management the portfolio manager may be risk averse, i.e. the pain of loosing $1,000 in profit is much greater than the reward of an additional $1,000 in profit. This affect can be explicitly accounted for in the objective by incorporating a utility function.

Another key challenge is developing predictive models that compute the affect of taking alternative actions. In account management one must model the effects of giving a credit line increase on key factors such as the likelihood of delinquency. Unfortunately, some of the actions may not have been taken before, thus there is no data available to build a model. Experimental design techniques can be employed to collect data to fill this void. It is also desirable to have a software environment that can use all of the data available, yet allow a human expert to extend the capabilities of the model to predict for previously untested actions. Without customized software and a robust methodology, developing these **Action-Effect Models** is cumbersome.

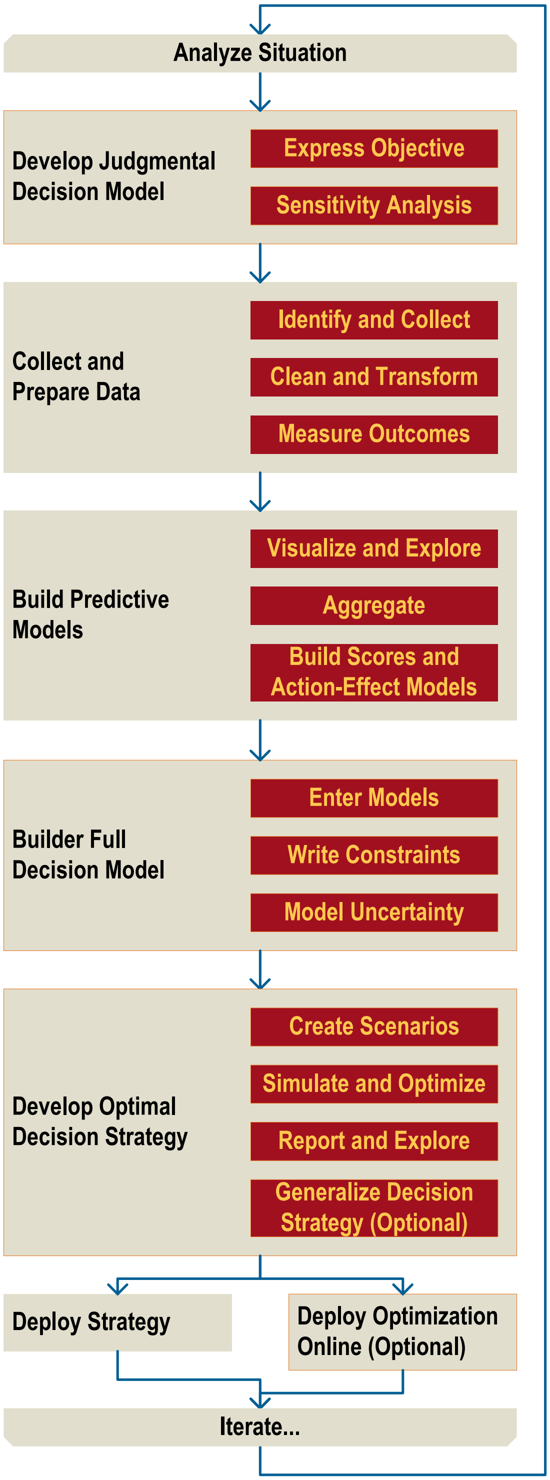
These are all details encountered under the banner of uncertainty. The value of embellishments always depends on the nature of the problem being solved. The critical point is that many problems are inherently rich with uncertainty and robust optimization techniques adequately address that uncertainty while simultaneously allowing for the inclusion of a long-term objective function and both record-level and cross-record constraints.

Deploying Optimization

The final practical aspect of applying optimization is deploying the optimization to the production environment. Often it is sufficient to deploy the results of an optimization based on the data used in developing the optimization model. One can either deploy a database that contains actions assigned to each record or develop a decision tree that generalizes the optimal strategy. In effect, these tables or trees are deployable action plans that Fair Isaac calls strategies. Exporting a database that contains actions assigned to each record ensures that all cross-record constraints are satisfied by the business strategy, but also requires that all records be considered in the optimization. Often results need to be generalized so that they can be applied to records that have not thus far been considered.

For example, in account management a new customer might be acquired after the optimization. If a credit score of 749 was not considered in the optimization and the new customer has a score of 749, the optimization does not prescribe an action for the new customer. In this case, the challenge is to develop a decision tree or a set of business rules that encodes the optimal actions while still having a high likelihood of adhering to the cross-record constraints. For example, suppose the optimization did consider customers with scores of 748 and 750. If these customers were assigned the same action, it would be reasonable to use the action for the new customer who has a score of 749. Moreover, suppose that all inactive customers with a score above 700 were offered a balance transfer by the optimization, then we could generalize the strategy into a rule or tree branch, e.g. if score is greater than 700 then offer a transfer. For practical applications it is extremely important to be able to deploy optimization results directly to custom applications developed using business rules products such as Blaze Advisor or to rule-based decisioning systems such as Triad. The caveat is that applying a tree to the entire customer portfolio including the new customers may violate a cross-record constraint. If the risk is judged too great, then optimization should be performed again on the new portfolio. Decision trees are also extremely useful as a visualization of the action plan. They provide unprecedented visibility into exactly how each decision is reached as well as an understanding of the key drivers of value.

Sometimes, it is not acceptable to deploy the decision strategy based on the data used in developing the model, e.g. if many new customers are added to the portfolio on a regular basis. If this is the case, one can deploy the optimization model into the production environment so that optimizations can be run on production data periodically to assign actions. Each run can then consider all of the records including any new additions. This is different from recalibration where components of the decision model are updated, e.g. recalibrating the coefficients of an equation that predicts bankruptcy and generates a bankruptcy score. Predictive models and scores used in the optimization model should be recalibrated periodically as is normally required.



Strategy Science Methodology

Fair Isaac has created a methodology for decision optimization that captures the best practices it has developed over the years. The Strategy Science methodology addresses the challenges of applying optimization to business problems, creates workable solutions within an enterprise, and serves as a guide for the evolution of Decision Optimizer. The methodology is iterative providing a means to adapt and accelerate learning that leads to better decision strategies. The following graphic shows elements of the methodology; elements outlined in **orange** are completed using Decision Optimizer.

Analyze Situation

Define the decision situation. State your goals as well as your non-negotiable requirements. Explore conflicts and trade-offs. List your policies and constraints. Generate creative decision alternatives. Identify uncertainties.

Develop Judgmental Decision Model

Build the framework for the decision model without filling in the details. This is called building a Judgmental Model because numbers are assessed by human experts instead of from data.

Express Objective

Express the business problem in the form of an objective function. This function should depend on all of the factors and possible uncertainties that have the potential to drive value.

Sensitivity Analysis

Perform a sensitivity analysis on the objective to understand which uncertain factors need to be modeled with the greatest detail and the value of collecting additional data and information.

Collect and Prepare Data

Identify and Collect

Identify existing data and data sources. Identify gaps in existing data and sources for required and desirable data. Plan a comprehensive design of experiments to collect the best data.

Clean and Transform

Clean and filter all data, correcting detectable errors. Make judgments about what data is reliable, useful, and correct.

Measure Outcomes

Measure the outcomes by their ability to contribute to a company's bottom line. Use metrics like Lifetime Customer Value (LCV) that measure the present value of future profits.

Build Predictive Models

Visualize and Explore

Visualize the data for better understanding. Provide statistical and graphical summaries of the data.

Aggregate

Aggregate the data enough that erratic variations are smoothed out, but not so much that useful distinctions are lost.

Build Scores and Action-effect Models

Model the important relationships with best-practices methods that use unique, custom-fitted models for each individual project whenever they provide a lift over reusable, “canned” models. Predictive models must be developed to yield scores while Action-Effect models are required for assessing the affects of potential actions on key value-drivers.

Build the Full Decision Model

Enter Models

Import all the Predictive models, scoring formulas, and action-effect models into the decision model.

Write Constraints

Enter all viable constraints, including cross-record constraints, such that the feasible solution space closely reflects the real-world policies and environment.

Model Uncertainty

Model the uncertainty by considering multiple scenarios of feasible outcomes that are derived and weighted in harmony with observed patterns in the data, but adjusted to be compatible with intuition regarding the future.

Develop Optimal Decision Strategy

Create Scenarios

Create multiple scenarios that will explore different combinations of constraint thresholds, fact values, e.g. mailing cost, and uncertain parameters.

Simulate and Optimize

Simulate the scenarios to find the affect of taking all possible actions for each row, e.g. account. Optimize across the rows to find the best decision strategy that adheres to the constraints.

Report and Explore

Generate reports to explore the scenarios that contain visualizations and numerical summaries. Use OLAP technology to query by, scenario, and segment. Also query by business cases that consider what the result would be with and without uncertainty and with and without constraints. Gain critical insight into trade-offs.

Generalize Decision Strategy (optional)

Develop a decision tree that generalizes, refines, and adequately encodes the optimal decision rules. Ensure that the tree will be likely to adhere to cross-record constraints when it assigns treatments/actions to accounts. Use the tree as a visualization to gain insight into the key drivers of value and the reasoning behind the recommended decisions.

Deploy Strategy

Deploy the optimal challenger strategy into the solution alongside the current champion. Compare the results. Assuming the challenger outperforms, make it the new champion.

Iterate

Document the success of the solution. Implement the methodology repeatedly as desired.

Deploy Optimization Online (optional)

Deploy the optimization model to a Decision Optimizer application that will perform optimization directly on production data on a regular basis.

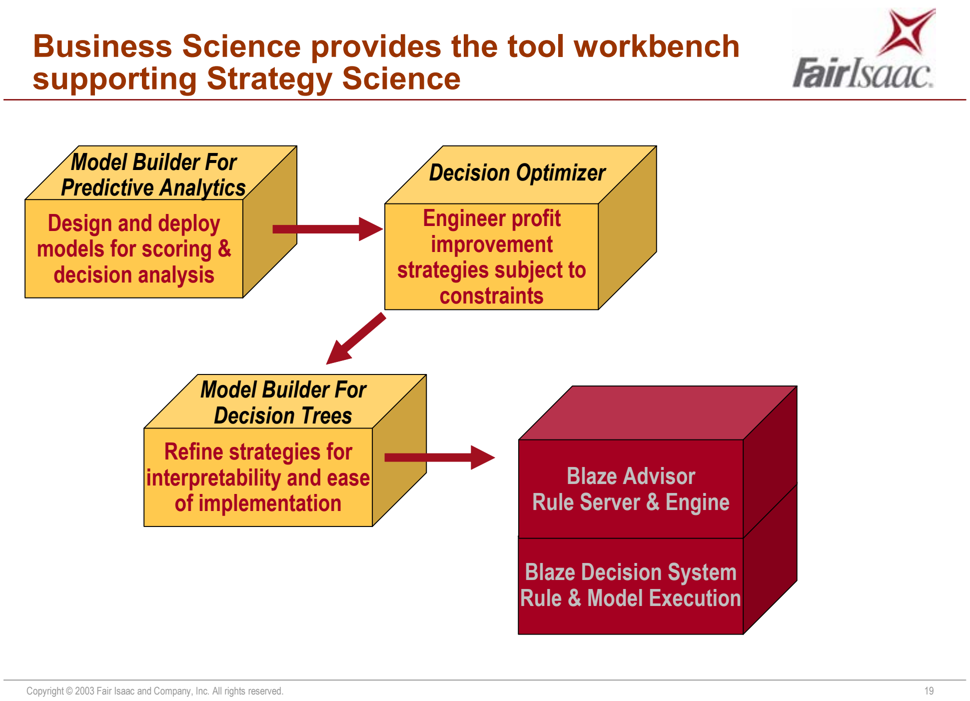
Decision Optimizer

Decision Optimizer is an environment for finding the best challenger decision strategy. As an environment, it goes beyond the capabilities of optimization software components and tools. Optimization software components do provide access to implementations of cutting-edge optimization algorithms, but they do not provide a workspace for implementing a decision optimization methodology or features required to address the challenges discussed above.

Decision Optimizer is engineered to support fully the best practices of Fair Isaac and the Strategy Science methodology. It includes features that address the issues that arise when applying optimization in practice. Specifically, Decision Optimizer supports, multiple decisions and definition of treatments, sequential decisions, record and cross-record constraints, cash flows and forecasting, sensitivity analysis for analyzing competing objectives, robust optimization for addressing uncertainty both at the record level as well as the portfolio level, CART decision tree technology for generalizing and deploying strategies, and a Java application for easily deploying optimization to the production environment.

Business Science

The Strategy Science methodology is supported by a suite of software tools that integrate support for the development and deployment of predictive models, decision optimization, strategy refinement, and deployment. Decision Optimizer works seamlessly with Model Builder for creating and deploying predictive and action-effect models, with Model Builder for Decision Trees for strategy refinement, and with Blaze Advisor for designing and integrating business rules.



Sidebar: Strategy Science for Credit Line Management Results

Client: Large and medium credit card issuers

Challenge: Design optimal credit line strategies. Specifically, find the most profitable segments for credit line increases—without exceeding risk parameters.

Decisions: Increase credit line, by how much.

Objectives: Maximize profitability, increase revenue, limit risk exposure, and control delinquency rates.

Uncertainties: Will the customer "go bad", if so for how much? What would be the effect of an economic downturn on the portfolio?

Solution: Fair Isaac developed a decision model that accounts for the complex relationships among the facets of the decision. The decision model leveraged data such as master file information (time on books, utilization levels, revolving frequency), scores (risk, revenue, attrition), and models to predict outcomes based on actions (e.g. revenue, loss). Decision Optimizer was applied to design a new decision strategy. This optimized challenger strategy was tested in the marketplace against the existing champion.

Result: Over 16 months incrementally increased receivables by over $90 per active account while maintaining losses at historical levels. This translated into a $9.5 million dollar profit increase for each million accounts held in portfolio.

References

Clemen, R., “Making Hard Decisions an Introduction to Decision Analysis,” Duxbury Press, Belmont, California, 1990.

Bai, D., Carpenter, T., and Mulvey, J., "Making a Case for Robust Optimization Models," *Management Science*, 43 (1997), 895-907.

Charnes, D. and W. W. Cooper, "Chance-Constrained Programming," *Management Science*, 5 (1959), 197-207.

Grover, M. B., "Land Grab," *Forbes*, August 23, 1999, 90-92.

Infanger, G., *Planning Under Uncertainty: Solving Large-Scale Stochastic Linear Programs*, Boyd and Fraser Publishing Company, Danvers, MA, 1994.

Mulvey, J. M., R. J. Vanderbei, and S. A. Zenios, "Robust Optimization of Large-Scale Systems," *Operations Research*, 43 (1995), 264-281.

Turban, E. and J. R. Meredith, *Fundamentals of Management Science, Fourth Edition*, Business Publications, Inc., Plano, TX, 1988.

About Fair Isaac

Fair Isaac Corporation (NYSE:FIC) is the preeminent provider of creative analytics that unlock value for people, businesses and industries. The company's predictive modeling, decision analysis, intelligence management, decision management systems and consulting services power more than 25 billion mission-critical customer decisions a year. Founded in 1956, Fair Isaac helps thousands of companies in over 60 countries acquire customers more efficiently increase customer value, reduce fraud and credit losses, lower operating expenses and enter new markets more profitably. Most leading banks and credit card issuers rely on Fair Isaac solutions, as do insurers, retailers, telecommunications providers, healthcare organizations and government agencies. Through the *www.myfico.com* Web site, consumers use the company's FICO® scores, the standard measure of credit risk, to manage their financial health. As of August 2003, HNC Software Inc., a leading provider of high-end analytic and decision management software, is part of Fair Isaac. For more information, visit *www.fairisaac.com*.

Corporate Headquarters:
200 Smith Ranch Road
San Rafael, CA 94903-5551
1 800 999 2955 *from the US*
1 415 472 2211 *from anywhere*
info@fairisaac.com *email*

Offices Worldwide:
Brazil, Canada, France, Germany,
Japan, Singapore, Spain,
United Kingdom, United States

www.fairisaac.com

1397WP MM/YY PDF