Proceedings of the 2007 IEEE Symposium on Computational Intelligence in Multicriteria Decision Making (MCDM 2007)

Automated Risk Classification and Outlier Detection

Naresh Iyer, *IEEE Member*, Piero P. Bonissone, *IEEE Fellow*

Abstract— Risk assessment is a common task present in a variety of problem domains, ranging from the assignment of premium classes to insurance applications, to the evaluation of disease treatments in medical diagnostics, situation assessments in battlefield management, state evaluations in planning activities, etc. Risk assessment involves scoring alternatives based on their likelihood to produce better or worse than expected returns in their application domain. Often, it is sufficient to evaluate the risk associated with an alternative by using a predefined granularity derived from an ordered set of risk-classes. Therefore, the process of risk assessment becomes one of classification. Traditionally, risk classifications are made by human experts using their domain knowledge to perform such assignments. These assignments will drive further decisions related to the alternatives. We address the automation of the risk classification process by exploiting risk structures present in sets of historical cases classified by human experts. We use such structures to pre-compile risk signatures that are compact and can be used to classify new alternatives. Specifically, we use Dominance relationships, exploiting the partial ordering induced by the monotonic relationship between the individual features and the risk associated with a candidate alternative, to extract such signatures. Due to its underlying logical basis, this classifier produces highly accurate and defensible risk assignments. However, due to its strict applicability constraints, it covers only a small percentage of new cases. In response, we present a weaker version of the classifier, which incrementally improves its coverage without any substantial drop in accuracy. Although these approaches could be used as risk classifiers on their own, we found their primary strengths to be in validating the overall logical consistency of the risk assignments made by human experts and automated systems. We refer to potentially inconsistent risk assignments as outliers and present results obtained from implementing our technique in the problem of insurance underwriting.

***Index Terms* — Risk classification, Automated insurance underwriting, inconsistency detection, Pareto Dominance, rational risk assignment Pareto Optimality.**

I. Introduction

The problem of assigning risk to an alternative is common in many domains. It is also fairly typical for these risk assignments to come from a predefined set of risk classes, in which case the problem becomes one of risk classification. Typically human experts apply their domain knowledge to evaluate a given alternative before assigning an appropriate risk class to it based on its characteristics. In this paper, we tackle the problem of automating risk classification processes.

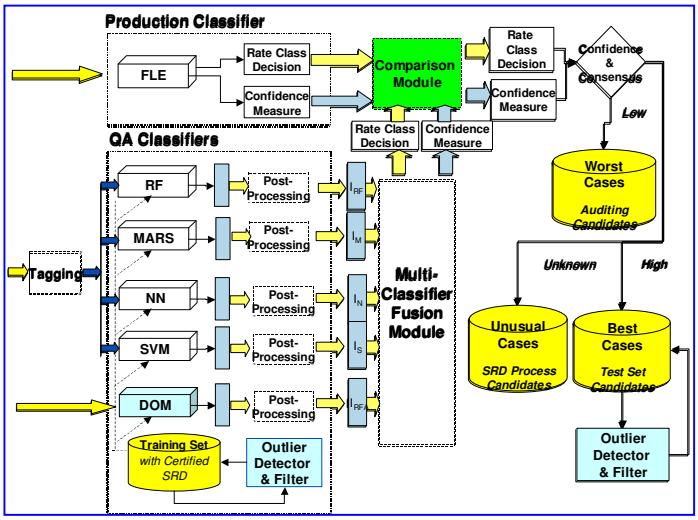
In any risk classification problem, it is fair to assume that the human expert reasons with the help of his domain knowledge to assign a risk class to a given alternative. The reasoning principles used by the expert for risk classification is expected to be embedded as risk signatures in a set of alternatives that have already been classified by the expert. We describe a technique based on the concept of *Pareto dominance* [1], [2], which extracts such risk signatures. More specifically, we use the set of alternatives classified by the expert to produce two subsets for each risk class in the problem: the *Pareto best subset* and the *Pareto worst subset* by using Dominance. Algorithmically, producing these two subsets is equivalent to producing what is more commonly known as the Pareto-optimal set. The two subsets are different from each other only in the sense that they optimize the multiple objectives along opposing directions of goodness. By doing so, these two subsets can be seen as representing the least risky (the Pareto-best) and the most risky (Pareto worst) candidates within a given risk class. If there are a sufficient number of candidates in these two subsets, then the candidates in these two subsets can be seen as samples from the two hypothetical risk surfaces in the feature space that bound the risk class from above and below respectively.

We would like the mention that for the problem being described in this paper, we tried out other traditional classification approaches with great success. The other approaches include the use of fuzzy rule-based and case-based classifiers. The fuzzy rule based classifier [3] used the concept of soft constraint satisfaction to determine the degree to which an applicant would belong to a given rate class. The case-based classifier [4] used a similarity metric in the feature space to retrieve applications or cases similar to an application at hand; the retrieved cases were now used to determine the risk class of the given application. Reference [5] describes and compares both these approaches; it also presents other details relevant to the maintenance of a classifier over its life cycle. The approach described in this paper presents a technique that possesses high classification accuracy but potentially low coverage. Hence it is more suitable for use as a complementary technique as far as classification is concerned.

In a hybrid approach described in [6] we show how the classifier based on Dominance can be used in Quality Assurance of a more traditional rule-based fuzzy classifier. The use of Dominance as an *outlier detector* for the detection of logically inconsistent risk assignments by human experts is also shown to be quite effective.

Naresh Iyer and Piero Bonissone work at the General Electric Global Research, One Research Circle, Niskayuna, NY 12309, USA. (Corresponding author: Naresh Iyer; phone: 518-387-5419; fax: 518-387-6845; e-mail: iyerna@crd.ge.com).

Figure 1 indicates the architecture of the hybrid approach from paper [6], which describes the use of a fusion module to combine the outputs of several classifiers to determine the correct rate class for an insurance application. The purpose of the fusion module is one of Quality Assurance (QA) for testing and monitoring a Production Decision Engine that makes the rate class assignment in real-time. The fusion module exploits the possibility that the various classifiers contribute to decisions that rely on different informational aspects of the data; as a result, these decisions are independent and can therefore be either in agreement or conflict with each other. The degree of agreement or conflict is used to quantify the confidence in the output of the fusion module. The fusion is expected to produce a classification that is more accurate as result of its dependence upon multiple, but independent, sources of decision-making. As shown, both the dominance-based classifier and the outlier detector are part of this quality assurance architecture.



LEGEND

Abbreviation	Description
FLE	Fuzzy Rule Based reasoning
RF	Random Forests
MARS	Multivariate Adaptive Regression Splines
NN	Neural Networks
SVM	Support Vector Machine
DOM	Dominance-based

Fig 1. Architecture for a hybrid approach with the outlier detector for detecting inconsistent risk assignments and quality assurance.

Since our approach is analogous to clustering-based classification, we describe some previous work done in clustering as well as *dominance* in section II, which also provides a more detailed description of the insurance underwriting problem that we have attempted to automate. Section III and its subsections motivate and define key principles of *dominance-based risk classification* that are central to our approach; we also present results of applying the classifier to the underwriting problem. In section IV we present a slightly modified version of our approach in an effort to improve the coverage of the classifier along with the associated results. We describe how our approach can be used to detect potentially inconsistent risk assignments in section V some instances of such potentially inconsistent assignments that were found using our technique are also presented in the same section. Finally, the conclusions are summarized in section VI.

II. Background

A. Previous Work

In some sense, our approach parallels approaches in clustering for classification problems. However, our technique applies only to a subclass of classification problems where there is a total ordering of the individual classes, and where the class of a candidate is a monotonic function of the feature-values of the candidate (as is true of {\em risk classification} problems). In traditional clustering approaches, distance and similarity measures are used to represent individual clusters in terms of centroids; in our approach, we represent a cluster in terms of superior and inferior Pareto boundaries. We then verify if a given point lies inside these boundaries before classifying it as belonging to the cluster. Since the definition of a cluster in our approach is based on the logical equivalence of the *riskier_than* relation and the dominance relation, the use of our approach to assign a risk class to a given candidate is logically defensible. Because of the same reason, the violation of this condition by a labeled candidate can be interpreted as the candidate being inconsistently labeled by the expert (we refer to such a candidate as an *outlier*). Our technique can be extended to detect such outliers, which we also describe in this paper.

There are several endeavors in literature related to the use of dominance in clustering although we could not locate any that uses dominance in the manner described in this paper. In [7] for example, the author presents equivalence theorems between dominance and circuit faults in order to establish a parsimonious number of faults that need to be tested for circuit correctness. This approach is slightly similar to our own in its philosophy since both approaches tend to use dominance relations to find a parsimonious set of instances needed to solve the problem; whereas [7] compresses the number of faults that need to be tested to ensure circuit correctness, we compress the number of labeled risk assignments that need to be stored in order to assign risk class to a new instance. Both [8] and [9] deal with the problem of finding the optimal dendrogram by using dominance relations between competing sub-clustering schemes. Reference [8] uses *homogeneity* and *separation* as criteria while [9] uses *cluster diameter* and *intercluster separation* as the criteria. Reference [10] presents

a technique called *CPEA* (Clustering Pareto Evolutionary Algorithm) where each population is clustered in the genotypic space and locally Pareto-optimal points of the individual clusters in the phenotypic space are maintained and resolved as the algorithm proceeds from one generation to another. The primary aim of the CPEA approach is to ensure a suitably diverse representation of solutions in the genotypic space although they are only locally Pareto-optimal. This concludes our survey of some of the previous efforts that seem related to what we describe in this paper.

For ease of understanding, we present our work in the context of a concrete problem on which we implement our automation technique - it is the problem of assigning appropriate premium classes to insurance applications. However, we want to state that the technique described applies generically to risk classification processes that are similar in structure to the problem of premium-class assignment for insurance applications.

B. Domain Description: Insurance Underwriting

Insurance underwriting is a complex task traditionally performed by trained experts or underwriters. An underwriter evaluates each insurance application in terms of its potential risk, such as mortality in the case of term life insurance. Typically, an application is compared against standards adopted by the insurance company, which are derived from actuarial principles related to mortality. Based on this comparison, the application is classified into one of the risk categories available for the type of insurance requested by the applicant. The accept/reject decision is also part of this risk classification, since risks above a certain tolerance level will typically be rejected. The estimated risk, in conjunction with other factors such as gender, age, and policy face value together determine the appropriate price (premium) for the insurance policy. In order to keep the expected return on a policy to a fair value, a higher risk typically corresponds to a higher premium, all other factors remaining the same.

An insurance application can be represented by a set of features taking values that can be continuous, discrete, or categorical in nature. These features represent the applicant's medical and demographic information that have been identified as pertinent to the estimation of the applicant's claim risk, based on actuarial studies. Since it is difficult to represent the risk associated with an application by a single number, a discrete number of risk classes or bins are identified to which a given application is assigned based on assessment of its associated risk. Therefore, the insurance underwriting problem is a discrete classifier that maps a given set of features to a discrete decision space.

There are several properties that make automating this problem a complex one:

1. The mapping to the decision space is highly nonlinear; in other words, incremental changes to one of the input components can change the most appropriate risk class for the application.
2. Most features are derived and based on interpretations made by the underwriter. Therefore, the underwriter's subjective judgment will almost always play a role in the overall risk assignment process, thereby making it crucially dependent on factors such as underwriter training and experience leading to potential variability in the decision.
3. The risk assignment is essentially a process of balancing tolerance for risk (in order to preserve price competitiveness) with aversion to risk (in order to prevent overexposure to risk). The making of these tradeoff judgments requires flexibility.
4. Legal and compliance regulations require that the models used to make the underwriting decisions be transparent and interpretable.

The approach presented in this paper tries to address most of the above properties. In response to the issue of subjectivity, our approach reduces the irrationality that can be potentially introduced due to subjectivity by two means:

1. With respect to the labeled set of applications, our approach imposes logical consistency on the labeled set by finding outliers that are potentially incorrect risk assignments. As a result, it uses only a logically consistent set of labeled applications to capture the risk profiles related to a risk class.
2. All new applications that are classified by our automated classifier are logically defensible in the context of the labeled set applications. This property addresses the issue of transparency and interpretability quite well.

A comprehensive description of the Automated Underwriting (UW) System addressing all the above requirements can be found in reference [3]. In the next section we will describe the concept of dominance-based risk classification, which was the logical constraint used to validate the results of such automated system.

III. DOMINANCE BASED RISK CLASSIFICATION

In insurance underwriting, as in most risk classification problems, the classification proceeds according to the feature-values taken by the insurance application along a predefined set of features. The direction in which the risk associated with the application changes with change in a feature value is also typically known. Based on this knowledge, we can state if an insurance application is *better than* another application *along a feature* using the feature values taken by two applications. For example, if the *Cholesterol Level* of some applicant C were to be higher than the *Cholesterol Level* of another applicant D, then since higher *Cholesterol Level* implies higher risk in an application, we say that D is *better than* C along the feature *Cholesterol Level.* If two applications take the same

value along some feature, then we say that the two applicants are *equally good* along that feature. We introduce the term *A dominates B* involving two applicants A and B, and denoted by *dominates*(*A,B*), as per the following definition of Dominance.

Dominance:
Given two applicants A and B we say that A dominates B if and only if A is at least as good as B along all the features and there is at least one feature along which A is better than B, i.e.:

$$Dominates(A,B) \Leftrightarrow \forall i\,(A_i \le B_i) \wedge \exists j\,(A_j < B_j)$$

where A_i denotes the value taken by applicant A on feature i, and so on. Note that without loss of generality we assume lower values along features to be better in the above definition. The relation *dominates*(*A,B*) is a trichotomous relation, meaning that given two applicants *A* and *B*, either *A* dominates *B*, or *B* dominates *A*, or neither dominates the other. In the case where neither applicant dominates the other, each applicant will be better than its counterpart along different features. In such a case we say that *A* and *B* are dominance-tied. For example if we consider three applicants *A*, *B*, and *C* with feature values as indicated in Table I.

TABLE I
EXAMPLE OF DOMINANCE

Applicant	BMI[1]	Cholesterol	Systolic Blood Pressure
A	25	255	115
B	26	248	120
C	24	248	112

By the definition, we see that C dominates both *A* and *B*, since *C* is at least as good (i.e. as low) as *A* and *B* along each feature and moreover there is at least one feature along which *C* is better than (i.e. strictly lower than) *A* and *B*. On the other hand, applicants *A* and *B* are dominance-tied since each is better (i.e. lower) than the other along some feature (*A* has the better *Cholesterol Level* value while *B* has the better *BMI* value).

As mentioned previously, the risk categories that can be assigned to the insurance applications can be totally ordered so that a higher risk category pertains to a higher premium or a riskier application. We represent the ordering between the risk categories by using "<". In other words, if r_A is a higher risk category compared to r_B, then we use the relation $r_B < r_A$ to assert that applications assigned to category r_B are less risky than those assigned to category r_A. Based on this, we next introduce the *No_Riskier_Than*(*A,B*) relation between two applicants *A* and *B*.

[1] BMI is the Body-Mass-Index, which is defined as the ratio of the weight (in kilograms) divided by the square of the height (in meters). It measures body fat as a function of an individual's height and weight; it is a good surrogate to estimate the individual's health.

No_Riskier_Than(A,B):
The relation No_Riskier_Than(*A,B*) *is said to hold if and only if the risk class assigned to A (say r_A is no better than that assigned to B (say r_B).* Or,

$$No\_Riskier\_Than(A,B) \Leftrightarrow r_A \le r_B\,.$$

Based on the knowledge that the risk associated with an applicant is a monotonic, non-decreasing function of the feature values, it can be seen that, for any pair of insurance applications, if the *dominates*-relation holds between the two applications in a certain direction (say *A dominates B*), then the relation *No_Riskier_Than* will also hold in the same direction - i.e., *No_Riskier_Than*(*A,B*) holds. In other words, the *dominates*-relation is a sufficiency condition for the *No_Riskier_Than*-relation. That is:

$$dominates(A,B) \rightarrow No\_Riskier\_Than(A,B).$$

Now, we define the trinary relation *Bounded_Within*(*B,*(*A,C*) as follows

Bounded_Within(B,(A,C)):
The relation Bounded_Within(B,(A,C)) is said to hold if and only if the relations, dominates(A,B) and dominates(B,C), hold together. Or,

$$Bounded\_Within(B,(A,C)) \Leftrightarrow dominates(A,B) \wedge dominates(B,C)$$

We read the above relation as *B is bounded within A and C*. We are now ready to express and prove the principle underlying the technique described in this paper.

Principle of Dominance based risk classification:
If A, B, and C are applicants such that B is bounded within A and C, and if the risk category assigned to A and C is the same, say r, then the risk category of applicant B must be the same as that of A and C. In other words,

$$[Bounded\_Within(B,(A,C)) \wedge (r_A=r_C=r)] \rightarrow (r_B=r).$$

Proof
Suppose we have,

$$[Bounded\_Within(B,(A,C)) \wedge (r_A=r_C=r)]$$

This implies that,

$$dominates(A,B) \wedge dominates(B,C) \wedge (r_A=r_C=r).$$

Or,

$$No\_Riskier\_Than(A,B) \wedge No\_Riskier\_Than(B,C) \wedge (r_A=r_C=r).$$

Based on our definitions, we can rewrite the above as,
$(r_A \le r_B) \wedge (r_B \le r_C) \wedge (r_A = r_C = r)$, which leads to, $(r_B = r)$.

This proves the *Principle of Dominance based risk classification*. ■

This principle is the basis of our approach. For any given application B with unassigned risk category, we check if there exist two applications A and C such that the left hand side of the principle is satisfied; if so, we declare the risk category of B to be the same as that of A and C.

The proof works because of the equivalence established between the *No_Riskier_Than* relation and the *dominates* relation. Since the *No_Riskier_Than* relation is not described in terms of a strict inequality, it is possible for the *dominates* relation to hold between pairs of applications even if they belong to the same risk category. This implies that we can further partition the applications *within a risk category* into the *best, non-dominated* subset and *worst, non-dominating* subset, which are defined as below:

Best, non-dominated subset:
The best, non-dominated subset, labeled O, for a given risk category, is one that contains all such applications that are not dominated by another application within that risk category.

Worst, non-dominating subset:
The worst, non-dominating subset, labeled P, for a risk category is one that contains all those applications that do not dominate even a single application in that risk category.

Because the two subsets are defined in terms of *Pareto Dominance*, we call the set *O* the *Pareto best* subset and the set *P*, the *Pareto worst* subset, for a given risk category. To visualize these two subsets geometrically, consider Figure 2, which shows a plot of features *f1* and *f2* for 1000 insurance applications. The insurance applications are plotted as points in the 2-dimensional feature space. Suppose for simplicity that these are the only two features used while assigning a risk category to an application, and that lower values along a feature correspond to lower risk. Then, the circles denote the *Pareto best* set while the squares denote the *Pareto worst* set. Also, based on the definition of the Pareto best and the Pareto worst sets, we know that each of remaining points is such that at least one circle dominates it, and it dominates at least one square. In other words we know that for each point *X* that is not in the *Pareto best* set *(O)* or in the *Pareto worst* set *(P)* in the figure, there is at least one square *S* and one circle *C* such

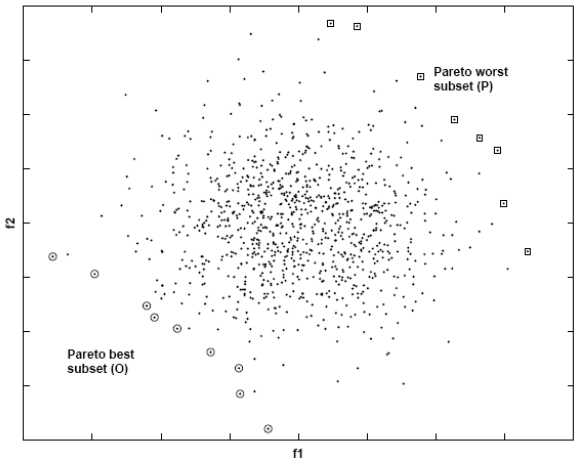


Figure 2. Indicating the Pareto-best (circles) and Pareto-worst (squares) subsets for a set of applicants.

that *Bounded_Within(X,(C,S)* holds. If we additionally knew that the risk categories assigned to *C* and *S* are the same, then we can infer the risk category for application *X* to be the same as well. In general, for the example shown in Figure 2, if we knew that every circle and square in the figure was assigned the same risk category, say *r,* then by applying *Principle of dominance-based risk classification*, all the points shown in the figure can
be assigned the risk category *r* as well. The production of the two subsets *O* and *P* is identical to the production of the Dominance subset in discrete alternative decision problems. Algorithms presented in [12], [13] can create these subsets in $O(n\log^{m-1}(n))$ time, where *n* is the number of candidates involved and *m* is the number of features along which the dominance comparisons are being done. For an insurance underwriting problem with *r* risk categories, there will be *2r* such subsets, or one pair for each risk category representing the risk surfaces that form the upper bound and the lower bound.

A. Dominance based risk classification: System Description

In this section, we describe the algorithms for application of *dominance based risk classification*. We use the term *Dominance(X,k)* to indicate the application of the above algorithm to the set *X(n,m),* where *k* is either *+1* or *-1* depending upon whether we want higher or lower feature values to be respectively considered as better during dominance comparisons. Basically, there are two modules involved in the risk classification process:

1. The Tuning module computes the Pareto best and Pareto worst subsets for each risk category from set A of labeled applications. A pseudocode appears below

```
TUNE(A,i)
{
 for each risk category r_i
 Compute O(r_i)=Dominance(A,-1).
 Compute P(r_i)=Dominance(A,+1).
}
```

2. The Classification module uses the results of tuning to classify the set *U* of new applications. The pseudocode for the Classification module is presented below

```
CLASSIFY(U,i)
{
 for each unlabeled application z ∈U
 for each risk category r_i {
  if(x ∈ O(r_i), y ∈P(r_i) : Bounded_Within(z,(x,y)))
  assign risk category r_i to z
  break
 }
}
```

During classification, given an insurance application, if the system is unable to find a risk category for which the *bounded within condition* mentioned in the pseudocode is satisfied, then the application at hand is marked as unresolved by the system.

Figure 3 shows results obtained by applying the automated dominance based risk classification to a set of applications for an example insurance underwriting problem with 3 risk categories (labeled as *R1, R2, R3*). In the experiment, the system initially used Tuning in order to compute the *Pareto best* and the *Pareto worst* subsets for each of the 3 risk categories. The system was then made to classify a set of applicants, which were not used during the Tuning. For these applications, risk assignments were also obtained from human underwriters. This allows us to compare the performance of the automated system with that of the experts using the comparison matrix shown in Figure 3.

		Dominance based Risk Classification				Σ
		R1	R2	R3	UW	
Underwriter Classification	R1	28	0	0	32	60
	R2	0	8	0	62	70
	R3	0	0	3	33	36
Σ		28	8	3	127	166

Figure 3. Comparison matrix from a pilot run conducted for the dominance based risk classifier.

As mentioned earlier, an application that is not *bounded within* any of the 3 risk categories is marked as unresolved by the system. These applications are shown in the column labeled *UW*. As can be seen, quite a large number of applicants are marked as unresolved by the system. However, for the applications that do get assigned a risk category by the system, we see that the system is accurate \$100% of the time. Hence, the principle of dominance based risk classification presented here has the potential to produce risk assignments with a high degree of confidence. In order for an application to be assigned a risk category, it is required to be dominated by some application in the *Pareto best* set for the risk category and additionally it should dominate at least one application in the *Pareto worst* set for the same risk category. This requirement can end up being too strict especially if the size of the Pareto subsets for a risk category is small. As a result, many of the applications will end up with unresolved risk assignments. It is possible to improve the coverage by allowing a minor relaxation to the classification rule for the extreme rate classes (i.e. the best and worst rate class). This modified version of the classifier is described next.

IV. Modifying the Classification Rule for the Extreme Risk Categories

This section shows how the principle of dominance based risk classification can be relaxed for the *best* and the *worst* risk categories. This relaxation is expected to improve the coverage of the automated classification system. The statements and proofs of the relaxed principles of risk classification for the best and the worst risk classes appear below.

Relaxed principle of dominance based risk classification for the best risk category:
If application X dominates application A such that the risk category assigned to A is the best risk category for the problem, say r_{best} , then the risk category of X is also r_{best} i.e.,

$$[dominates(X,A) \wedge (r_A = r_{best})] \rightarrow (r_X = r_{best}).$$

Proof:
Let us start by assuming that there is an application X such that it dominates application A, where it is known that A has been assigned the best risk category. In other words,

$$r_A = r_{best}.$$

Now since applicant A is assigned the best risk category, no other applicant can be assigned a better risk category that A. Or,

$$r_X \geq r_A.$$

However, since dominance is a sufficiency condition for an applicant to be no riskier than another applicant, and since we know that *X* dominates *A*, we must also have,

$$r_X \leq r_A.$$

Based on the previous 3 expressions we can therefore infer that,

$$r_X = r_{best},$$

which proves the relaxation condition for the best risk category. ■

Relaxed principle of dominance based risk classification for the worst risk category:
If applicant A dominates applicant X such that the risk category assigned to A is the worst risk category for the problem, say r_{worst}, then the risk category of X is also r_{worst}.

$$[dominates(A,X) \wedge (r_A = r_{worst})] \rightarrow (r_X = r_{worst})].$$

Proof
Let us start by assuming that there is an application X such that it is dominated by application A, where it is known A is assigned the worst risk category. i.e., $r_A = r_{worst}$. Now, since applicant A belongs to the worst risk category, every other applicant belongs to a risk category that is better than or equal to that of A. In other words,

$$r_X \leq r_A.$$

However since we also know that A dominates X and that dominance is a sufficiency condition for an applicant to be no riskier than another applicant, we must also have

$$r_X \geq r_A .$$

From the previous 3 expressions we can therefore infer that,

$$r_X = r_{worst} ,$$

which proves the condition for the worst risk category. ■

In other words, for the extreme risk classes an application is required to satisfy only one dominance condition instead of the two required for the intermediate risk classes. This is expected to reduce the number of applications that are left unresolved by the system for the extreme risk categories. Figure 4 presents the comparison matrix for same problem presented previously but with the relaxed principle of risk classification being used.

		Dominance based Risk Classification				Σ
		R1	R2	R3	UW	
Underwriter Classification	R1	38	0	0	22	*60*
	R2	0	8	0	62	*70*
	R3	0	0	11	25	*36*
Σ		*38*	*8*	*11*	*109*	*166*

Figure 4. Comparison matrix from a pilot run conducted for the modified dominance based risk classifier.

As seen from the matrix, the number of applications that are left unresolved reduces for both the extreme classes (by 10 applications for the best risk category and by 3 applications for the worst risk category). This improves the overall coverage of the classification system without loss to accuracy as seen from the matrix.

V. Detecting Inconsistent Risk Assignments

The technique described here can also be used to check the global consistency of the risk assignments made by human experts. In other words, ensuring that every application that is riskier than a counterpart is not assigned a worse risk category relative to the counterpart. More specifically, our system can identify all such pairs of applications for which at least one risk assignment must be changed in order to maintain global consistency. We refer to such pairs of applications as *outliers* defined as follows.

Outliers:

Applications X and Y are marked as outliers if and only if one of the following two conditions is satisfied:

1. *X dominates Y, and X is assigned a risk category that associates greater risk with X compared to Y, or*
2. *Y dominates X, and Y is assigned a risk category that associates greater risk with Y compared to X.*

In other words,

(X, Y are *outliers*) $\Leftrightarrow$ [(*dominates*(X,Y) $\wedge$ ($r_X > r_Y$)) $\cup$ (*dominates*(Y,X) $\wedge$ ($r_Y > r_X$)].

Claim: *Outliers are potentially inconsistent risk assignments.*

Proof: Suppose applications X and Y are outliers. By definition, this implies that,

[*dominates*(X,Y) $\wedge$ ($r_X > r_Y$) $\cup$ [*dominates*(Y,X) $\wedge$ ($r_Y > r_X$)].

Since, *dominates*(X,Y) $\rightarrow$ *No_Riskier_Than*(X,Y) $\rightarrow r_X \leq r_Y$,

we must have

$$[(r_X \leq r_Y) \wedge (r_X > r_Y)] \cup [(r_Y \leq r_X) \wedge (r_Y > r_X)],$$

which results in a contradiction. ■

Since dominance depends upon the feature values of the applications, if the feature values are accurate and if the same features were used by the human underwriter to assign risks to the applications, then the only way to resolve the above contradiction is to change either r_X or r_Y or both so that the reassignments correspond to the *dominance* relation between the two applications. In other words, all outliers are potentially inconsistent risk assignments. We applied the outlier detection technique to a set of 541 labeled applications. This resulted in the identification of more than a dozen inconsistencies in risk assignment by the experts. An example of the results produced by the outlier detector is shown in Figure 5 with a few relevant features.

Underwriter Classification	Application Feature Values												
	f1	f2	f3	f4	f5	f6	f7	f8	f9	f10	f11	f12	f13
R1	62	146	112	80	258	4.1	21	16	17	0	27	0	0
R2	77	229	132	84	278	4.6	25	22	17	0	27	0	0

Figure 5. Example of an inconsistency identified as an outlier.

Each row represents an insurance application for which the risk classification had already been determined, as shown in the first column. The risk class labeled *R2* is a better risk class compared to the risk class labeled *R1*. In other words, the premium associated with risk class *R2* is lower than that associated with class *R1*. Yet, based on the above feature values, we see that the application indicated in row 1 of the table *dominates* (and is therefore less riskier than) the application in the lower row. Upon sending these two applications to the underwriters for reconsideration, the risk classifications for the applications were reversed. This simple example illustrates the use of an outlier to obtain more consistent risk assignments from human underwriters.

VI. Conclusions

We presented an automated, dominance based risk classification technique using insurance underwriting as an example. The technique produces highly confident and accurate risk assignments to new insurance applications. We also presented an outlier detection technique based on dominance which can be used to detect potentially inconsistent

risk assignments by human experts. The use of the two Pareto surfaces to characterize risk-classes is admittedly weak (as characterized by the low coverage of the dominance-based classification approach). However, this weakness is, in principle, because the dominance condition becomes difficult to obtain for *large number of criteria/features*. What is interesting is that the approach is able to find outliers violating the dominance condition, despite the fact that it was based on a large number of criteria, and flag them for review by the underwriter. Information overload from large number of features is precisely the situation under which we expect human underwriters to become inconsistent in their labeling. As stated by Tversky [14], [15] "*in choosing among many complex alternatives… optimal policies for choosing among such alternatives require involved computations based on weights assigned to the various relevant factors, or on the compensation rates associated with the critical variables. Since man's intuitive capacities are quite limited, the above method is quite difficult to apply."* In such circumstances, humans have been known to use various decision heuristics to solve the decision-making problem so as to strike a reasonable balance between the complexity of the problem and the resources at hand [16]-[18]. As shown in [14], [15], the use of a decision heuristic will not always lead to the best choices from the viewpoint of the decision-maker. Conversely, if the number of criteria for the problem being considered happens to be low, we expect the approach to do well in terms of its coverage because the dominance condition is easier to obtain.

The identification of an outlier is identification of an indefensible and inconsistent risk labeling by the underwriter along with witness-labels to prove it. Hence the number of outliers produced by the underwriter is in some sense intrinsic to the underwriter and his global consistency. From points of view of *fairness*, *transparency* and *accountability*, these are necessary virtues an underwriting company would desire in its underwriting process. What the outliers show is consistent with findings in behavioral research, namely how when faced with complex information processing, humans have no mechanism of ensuring global consistency across all labeled applications. The outlier detector is meant to be an automated mechanism to provide a process for that global check.

For all classification problems, the borderline where two risk-classes meet are intrinsically the most difficult instances to classify (given true lack of separability) and most classifiers would require added complexity to have a good performance for such borderline cases. What is interesting about the dominance-based approach is that it would mostly declare such borderline cases as 'unknown risk class'. In terms of ROC (receiver operating curves), the dominance-based classifier can be thought of as the classifier at one of the corner points of the ROC curve with 100% true positive rate but a large false negative rate (false negative being one where it is unable to identify the class of an application), with the false negative rate deteriorating as the number of features increases. The goal of the technique is not to replace human underwriters as much as to augment their rationality and computational capacities. A very important dimension of decision support is the provision of computational support to the decision maker so that the employment of suboptimal decision heuristics is curtailed or at least limited [19]. We contend that the technique presented here serves this goal of valuable decision support for the complex decision-making task of risk classification.

REFERENCES

[1] V. Pareto, "Cours d'economie politique," University of Lausanne, Lausanne, Switzerland, Tech. Rep., 1896.

[2] K. R. MacCrimmon, "An overview of multiple objective decision making,"in *Multiple Criteria Decision Making*, J. Cochrane and M. Zeleny, Eds., 1973, pp. 18–44.

[3] P. Bonissone, R. Subbu, and K. Aggour, "Evolutionary optimization of fuzzy decision systems for automated insurance underwriting," in *Proceedings of the IEEE International Conference on Fuzzy Systems, Honolulu, Hawaai*, 2002, pp. 1003–1008.

[4] K. Aggour, P. Bonissone, and W. Cheetham, "Soft-cbr: A self-optimizing fuzzy tool for case-based reasoning," in *Proceedings of ICCBR*, 2003.

[5] P. Bonissone, "The life cycle of a fuzzy knowledge-based classifier," in *Proceedings of NAFIPS, Chicago, IL.* IEEE Computer Press, 2003.

[6] P. Bonissone, "Automating the quality assurance of an on-line knowledge-based classifier by using multiple off-line classifiers," in *Proceedings of the 10th International Conference on Information Processing and Management of Uncertainty in Knowledge-Based Systems, Perugia, Italy.*, vol. 1, July 2004, pp. 309–316.

[7] A. Lioy, "Advanced fault collapsing (logic circuits testing)," *Design and Test of Computers, IEEE*, vol. 9, no. 1, pp. 64–71, 1992.

[8] J. M. Malard, "Pareto annotation of the dendrogram of performance data," Pacific Northwest National Laboratory, WA, Tech. Rep., 2002.

[9] P. B. Kantor and K. B. Ng, "Comparison of systems using pairs-out-of order," Computer Systems Laboratory, NIST, Tech. Rep., 1998.

[10] A. K. Molyneaux, G. B. Leyland, and D. Favrat, "A new, clustering evolutionary multi-objective optimisation technique," in *Proceedings of the Third International Symposium on Adaptive Systems-Evolutionary Computation and Probabilistic Graphical Models, Mathematics and Physics*, 2001, pp. 41–47.

[11] A. Patterson, P. Bonissone, and M. Pavese, "Six Sigma Applied Throughout the Lifecycle of an Automated Decision System", *Quality and Reliability Engineering International*, 21:275-292, 2005

[12] H. T. Kung, F. Luccio, and F. P. Preparata, "On finding the maxima of a set of vectors," *Journal of the Association for Computing Machinery*, vol. 22, no. 4, pp. 469–476, 1975.

[13] H. C. Calpine and A. Golding, "Some properties of Pareto optimal choices in decision problems," *International Journal of Management Science*, vol. 4, no. 2, pp. 141–147, 1976.

[14] A. Tversky, "Intransitivity of preferences," *Psychological Review*, vol. 76, no. 1, pp. 31–48, 1969.

[15] A. Tversky, "Elimination by aspects: a theory of choice," *Psychological Review*, vol. 79, pp. 281–290, 1972.

[16] H. A. Simon, "A behavioral model of rational choice," *Quarterly Journal of Economics*, vol. 69, no. 1, pp. 99–118, 1955.

[17] W. Thorngate, "Efficient decision heuristics," *Behavioral Science*, vol. 25, pp. 219–225, 1980.

[18] J. W. Payne and J. R. Bettman, "Adaptive strategy selection in decision making," *Journal of Experimental Psychology*, vol. 14, no. 3, pp. 534–552, 1988.

[19] N. S. Iyer, "A family of dominance filters for multiple criteria decision-making: Choosing the right filter for a decision situation," Ph.D. dissertation, The Ohio State University, 2001.