A High-Order Feature Synthesis and Selection Algorithm Applied to Insurance Risk Modelling

Author names

Author affiliations

Abstract: In many jurisdictions, automobile insurers have access to risk-sharing pools to which they can transfer some of their worst risks. Better selection of these risks in order to maximize profitability is the application we consider in this paper. To that end, different feature selection and modeling approaches are tested and compared. In particular, we introduce a flexible scoring model that is better suited to the application considered than the usual claims level. We also devise a feature synthesis and selection method that is robust to overfitting due to the use of a rank averaging technique. We show what should be the most suitable sorting criterion depending on pool regulations. Given the sequential structure of insurance data, we use a technique, similar to cross validation, but that is coherent with the sequential structure. We explain how software maturity level affects the information that is available for decision-making which in turn, impacts profitability.

Keywords: data mining; business intelligence; feature extraction; feature synthesis; feature selection; scoring models; bagging; cross validation; sequential cross validation; profit maximisation; automobile insurance; risk sharing pools; loss ratios.

Reference to this paper should be made as follows: ... NameA, InitialsA and NameB, InitialsB, (xxxx) 'A High-Order Feature Synthesis and Selection Algorithm Applied to Insurance Risk Modelling', *Int. J. of Busines Intelligence and Data Mining*, Vol. x, No. x, pp.xxx–xxx.

Biographical notes: Author bios.

1 Introduction

Most automobile insurers use underwriting criteria to decide who can obtain coverage with their company. For people with the poorest driving records, simply finding an insurer that will accept to offer a quote can be very hard. Since automobile insurance is mandatory in many jurisdictions, some legislators have devised *facilities* or *pools* which act as (re)insurers for these poor risks. Insurers can then accept these otherwise unsuitable risks and simply *cede* (i.e. transfer) them to the pool. In such cases, policy administration usually remains a duty of the insurer for which it keeps part of the insurance premium but the risk is borne by the pool, i.e. insurance claims incurred on ceded risks are paid by the pool. Losses in the pool

are then shared among insurers that operate in the jurisdiction.

Rules vary widely from one pool to another. For some, insurers are allowed to simply *choose*, within their book of business, a certain portion of their risks and cede these risks to the pool. From a data-mining perspective, this possibility of choice that is sometimes granted is paramount as this creates the need, sometimes overlooked by the insurers themselves, to maximise the profitability of the transactions with the pool, given the rules set forth by the legislator.

In this paper, we consider the pools of the two largest Canadian provinces, Québec and Ontario, with volumes of 2.8 and 9.6 billion Canadian dollars, respectively. We first describe the characteristics of Québec's pool, then consider those that prevail in Ontario and later (subsection 2.5) show that they lead to a different ranking criterion of the policies to be ceded.

Québec's automobile insurers are allowed to cede up to 10% of the volume of their book of business. Each insurer chooses the risks to cede to the pool. For those risks, the pool reimburses all claims. In return, the insurer must pay 75% of the premiums that were charged to the ceded risks. The remaining 25% are kept by the insurer to cover for administrative expenses and commissions related to the policy. Thus, insureds for which the mathematical expectation of the total claims is above 75% of the premium charged should be ceded by the insurer since we expect that claims will exceed what the pool charges to cover them.

In Ontario (*Ontario Risk Sharing Pool Procedures Manual*, 2006), up to 5% of the insured *units* can be ceded. A *unit* is defined as one year of coverage for one vehicle and is also referred to as a *car year*. Risks are not ceded in full, rather the insurer retains a 15% share of each ceded risk. The amount an insurer can charge for expenses may vary from one risk to another, depending on the premium structure that has been filed with the legislator, but are limited to 30% of the premium. Thus, in most cases, an insurer can transfer 85% of the risk by paying 59.5% (85%×(1-30%)) of the premium. Here, risks with an expected claims level above 70% of their premium should be ceded.

An important figure that is often used to evaluate the profitability of an insurer's operations is the *loss ratio*, defined as the ratio of claims to written premiums (during a given period). This ratio is available from insurers' financial statements. If an insurer has a high loss ratio, then other expenses must remain low for the insurer to remain profitable. Conversely, insurers often measure the *permissible loss ratio*, i.e. the loss ratio that, given fixed values for other expenses, would lead to targeted values for profit and return on equity. The loss ratio is directly tied to the pool optimisation problem described above: in the context of the Québec pool for instance, risks should be ceded if their expected loss ratio is above 75%. Thus, looking at a series of insurers financial results, one can readily see whether identifying risks to cede to the pool should be an easy task or not: insurers with global loss ratios close to or above 75% very likely have numerous risks that can be pinpointed, using state-of-art data-mining techniques, as having expected loss ratios above the 75% target. The goal of this paper is to introduce an effective scoring model to perform such a task.

The use of data mining techniques, including scoring models, for insurance risk estimation has been explored in the past, particularly in the areas of pure premium estimation (rate-making) (Dugas et al., 2003), fraud detection (Artís et al., 2002; Viaene et al., 2002) and underwriting (Apte et al., 1999). However, specific ap-

plications to risk-sharing pool optimisation has not, to the best of our knowledge, been previously investigated.

1.1 Importance of Feature Selection

Insurance databases typically include a large number of variables, with important redundancies between them. Many of these variables were included for purposes other than rate-making and are irrelevant from the risk-assessment perspective that is of interest here. For this reason, an initial feature selection stage is warranted in order to retain those features most likely to provide improvements in predictive accuracy. Limiting the number of features helps alleviate the curse of dimensionality, improve generalisation performance, and may also provide a better understanding of the underlying data-generating process.

Many feature selection algorithms have been proposed in order to build linear and/or nonlinear models. See (Wang and Fu, 2005) for an in-depth treatment of feature selection in the context of data mining and computational intelligence and (Liu et al., 2005) for a review of recent trends and challenges in feature selection.

According to the taxonomy set forth by Guyon and Elisseeff (2003), these fall in one of three categories: wrappers, embedded methods or filters. Wrappers (Kohavi and John, 1997) compare different sets of features on the basis of their predictive performance. One needs to specify how the space of all possible feature subsets should be searched. Greedy search strategies are often used and can be of two types: according to forward selection strategies, features are progressively selected and added to an initial empty set. Pursuit-type algorithms (Chen, 1995; Friedman and Tuckey, 1974; Mallat and Zhang, 1993; Vincent and Bengio, 2002) and the more recent LARS (Efron et al., 2004) algorithm fall in this category. Backward elimination strategies proceed the other way around: from an initial set of features including all variables, the least promising are progressively removed. Oh et al. (2004) proposed a hybrid genetic algorithms that alternate between both forward and backard selection until a convergence criterion is attained.

Embedded methods can either estimate or directly evaluate changes in the objective function in order to select features. Such methods include regularisation or shrinkage approaches for which the objective function includes a goodness-of-fit term to be maximised as well as a regularisation term that penalises complexity and is to be minimised. The regularisation term is often expressed as the $\mathcal{L}_p$-norm (usually, $p \in \{0, 1, 2\}$) of the model parameters. The Lasso (Tibshirani, 1996) is a popular algorithm that falls in that category. Finally, filters correspond to a feature selection preprocessing that is conducted independently of the learning machine to be used. Wang et al. (2007) propose a t-statistic based filter before a wrapper is used on a reduced set of features. Su and Hsiao (2009) propose a multiclass MTS algorithm that also combines a filter before a wrapper is used.

This paper proposes an approach, called *Basis Selection Regression*, that combines feature synthesis and feature selection. The feature selection stage of the algorithm falls within the category of wrappers with greedy forward search. The feature synthesis stage of the algorithm adds combinations of the most recently selected feature with other current features. Thus, the set of candidate features is dynamically increased, as features are selected. Here, we consider linear models but the Basis Selection Regression algorithm can equally be applied to nonlinear

models. Another innovation, from a practical viewpoint, is the use of the loss ratio, rather than the claims level, as the regression target.

1.2 Overview

This paper is organised as follows: in section 2 we detail the scoring models compared in this paper, including the feature selection methodology and the scoring criteria that are applicable depending on the pool regulations. In section 3 we explain the performance evaluation methodology, and give experimental results in section 4. In section 5 we examine reliability considerations that must be faced by models deployed in the field, and in section 6 we quantify one dimension of the data mining process maturity on the achievable financial benefits. Finally section 7 examines avenues for future research.

2 Models

The profit resulting from the decision of ceding a particular insured to the pool is the amount paid in claims for the insured's policy (and that the pool will pay instead of the insurer) minus what the insurer must pay to the pool, which corresponds to a premium p' that is a percentage of the annual premium rate p charged to the insured: $p' = \alpha p$ (in Québec, $\alpha = 0.75$). Thus, ceding a policy of duration d will yield a profit of $c - dp'$ where c is the total claim amount incurred over the coverage period. Our goal is to identify insureds for which transfer to the pool would be, on average, the most profitable.

All models considered here can be interpreted as *scoring models*: given an input profile $\mathbf{x}$, the model outputs a real-valued score quantifying the desirability of ceding to the pool the insured having the given profile. The objective, obviously, is to assign a higher score to profiles that are more likely to be profitable. In a batch implementation (see section 6 for alternatives), the scoring process proceeds as follows:

1. All profiles that are considered in a given batch (e.g. one month worth of new or renewed policies) are scored separately;

2. Profiles in the batch are *sorted* in decreasing order, as a function of the score;

3. Higher-scoring profiles are ceded up to a target limit, which is a function of pool regulations and insurer business objectives.

The models that we compare in the experimental section are described next. We follow with a description of two possible sorting criteria (used in step 2, above), both of which are based on the model score.

2.1 Frequency/Severity Approach

Usually, insurers estimate the expected claims level, conditional on input profile $\mathbf{x}$, assumed to be known at the start of a policy, through a two-staged approach

often referred to as *frequency/severity* modelling:

$$\begin{aligned} E[c|\mathbf{x}] &= E[c|y=0,\mathbf{x}]P(y=0|\mathbf{x}) + \\ & \quad E[c|y=1,\mathbf{x}]P(y=1|\mathbf{x}) \end{aligned}$$

where y is a binary variable identifying, for a given policy, the occurrence of an accident ($y = 1$) or not ($y = 0$). Since, by definition, $E[c|y = 0, \mathbf{x}] \equiv 0$, the above equation simplifies to the following:

$$E[c|\mathbf{x}] = E[c|y=1,\mathbf{x}]P(y=1|\mathbf{x}). \tag{1}$$

The first term, $E[c|y = 1, \mathbf{x}]$ (called *severity* factor), estimates the claims given that an accident occurred, while the second term, $P(y = 1|\mathbf{x})$ (called *frequency* factor), gives the probability that an accident occurs.

The severity part can be addressed as a simple regression problem; in the experiments, we considered *Weighted Partial Least Squares* (WPLS) regression (see Frank and Friedman (1993) and Srebro and Jaakkola (2003)). A WPLS model search for the multidimensional directions in the input space that explain the maximum multidimensional variance directions in the target space.

The frequency part corresponds to a two-class classification formulation. We make use of a generative model to estimate class-conditional probabilities $P(\mathbf{x}|y = k)$, $k = 0, 1$. Then, given prior class probabilities $P(y = k)$, $k = 0, 1$, Bayes' theorem is used to obtain posterior class probabilities:

$$\begin{aligned} P(y=k|\mathbf{x}) &= \frac{P(\mathbf{x}|y=k)P(y=k)}{P(\mathbf{x})} \qquad k=0,1 \\ P(\mathbf{x}) &= \sum_i P(\mathbf{x}|y=k)P(y=k). \end{aligned}$$

Prior probabilities are estimated by maximum likelihood, as the training set proportions of each class (the large quantity of data for insurance problems—several hundreds of thousands or millions of training examples—makes regularisation unnecessary). As a baseline, class-conditional densities can be modeled using multivariate Gaussian distributions. We first considered tied distributions whereby both class-conditional densities share the same covariance matrix. Within the context of classification, this approach leads to a linear decision boundary, and is therefore referred to as the *Linear Discriminant Analysis* (LDA). Relaxing the homoscedastic hypothesis, i.e., using different covariance matrices for each class, lead to a quadratic decision boundary, hence the term *Quadratic Discriminant Analysis* (QDA). Although we are not performing classification but conditional probability estimation, we shall retain both LDA and QDA labels for later references.

2.2 Basis Selection Regressors

As an alternative to the standard frequency/severity approach, we consider models that consist in a linear combination of *automatically-discovered features* to predict the loss ratio of each insured given its profile. Using the loss ratio as the target of our model rather than the claims level is motivated by the application we consider for which profitability depends on the loss ratio (see subsection 2.5).

Given the profile $\mathbf{x}$, we model the loss ratio for the duration of the policy as

$$\mathrm{BSR}(\mathbf{x};\boldsymbol{\beta}) = \sum_{i=1}^{N} \beta_i \phi_i(\mathbf{x}) + \epsilon, \tag{2}$$

where $\boldsymbol{\beta}$ is a vector of model parameters, $\phi_i(\cdot)$ are features extracted from the raw input profile, and ϵ is a noise term.

Given a fixed set of features, the training criterion targets the empirical *weighted loss ratio* of each policy in the training set, where the weight is given by the duration of the policy. Let $\mathbf{x}_j$ be the insured profile of training policy j, as observed when the *policy is written* (i.e. when a rate-making or pool transfer decision can be made from the insurer standpoint), d_j be the duration the policy j (fraction of year), p_j the premium amount and c_j the observed claim amount. Thus, for profile $\mathbf{x}_j$, the observed loss ratio is c_j/p_j and this value is used as the target, within the context of supervised learning. The model parameters are obtained by the standard ridge estimator (Hoerl and Kennard, 1970), which minimises a regularised squared error,

$$\hat{\boldsymbol{\beta}} = \arg\min_{\boldsymbol{\beta}} \left\{ \frac{\sum_j d_j (\mathrm{BSR}(\mathbf{x}_j;\boldsymbol{\beta}) - \frac{c_j}{p_j})^2}{\sum_j d_j} + \lambda \sum_i \beta_i^2 \right\}, \tag{3}$$

where the hyper-parameter λ can be determined empirically.

The features $\phi_i(\cdot)$ are selected using a well-known stepwise forward selection procedure, from a dictionary of possible transformations depending on the variable type.

Algorithm 1 shows the algorithm in pseudo-code. It starts with an initial set of features, Ψ, described below. Those features are *candidates* that can be retained in the final model, which has selected features Φ. The maximum number of features, N, is fixed *a priori*.[a] The algorithm iterates greedily to find, at each round, the best new feature to add to the selected set, in the sense of most reducing the mean-squared error on the training data, keeping fixed the features selected in previous iterations. After a feature ϕ_{best} has been found at a given iteration, it is removed from the set of candidates. Then, feature synthesis is performed by adding interaction (product) terms between ϕ_{best} and previously-selected features to the set of candidates.[b] To constrain model capacity, only interaction terms up to a maximum order M are allowed.[c]

For automobile insurance modelling, standard raw variables include the driver's age, sex, vehicle type, driving experience, accident history and geographical location. The initial set of features consists of an encoding of *individual raw variables*, determined by the variable type. In other words, each element of Ψ is a function of a single raw variable only; the stepwise algorithm synthesises higher-order features as low-order ones are selected.

The encoding performed by the initial features depends on the variable type, as follows:

[a] Previous experiments have shown that using $N = 75$ usually provides the best out-of-sample performance. This is the value we have used in our experiments below.

[b] The notation $\phi_1 \times \phi_2$ denotes a feature consisting of the product between features ϕ_1 and ϕ_2.

[c] We restricted interactions up to $M = 2$, for otherwise computational complexity quickly explodes and severe overfitting problems are likely to be encountered.

Algorithm 1 Stepwise feature selection with automatic higher-order feature synthesis

```
def StepwiseSelection(model-train, data):
Φ ⟵ ∅
Ψ ⟵ {φ_k}        # initial set of features
while |Φ| < N and |Ψ| > 0 do
  best ⟵ ∞
  φ_best ⟵ None
  # Find the best feature among remaining ones
  for φ in Ψ: do
    train-error ⟵ model-train(data, Φ̂)
    if train-error < best then
      best ⟵ train-error
      φ_best ⟵ φ
      # Add best feature to working set. Add candidate
      # interactions with previous features to dictionary
      Φ ⟵ Φ ∪ φ_best
      Ψ ⟵ Ψ − φ_best
      for φ in Φ: do
        φ̂ = φ × φ_best
        if order(φ̂) ≤ M then
          Ψ ⟵ Ψ ∪ φ̂
        end if
      end for
    end if
  end for
  Return Φ
end while
```

- **Continuous** variables (e.g. age) are standardised to have zero-mean and unit-standard deviation,

$$\phi_k(\mathbf{x}) = \frac{x_k - \hat{\mu}_k}{\hat{\sigma}_k},$$

where $\hat{\mu}_k$ and $\hat{\sigma}_k$ are the mean and standard deviation of raw variable k estimated on the training data.

- **Discrete unordered** variables (e.g. vehicle type) are encoded as one-hot (dummy variables), with one less variable than the number of levels. Suppose that variable k has ℓ levels, denoted $1, 2, \ldots, \ell$ for simplicity; we introduce $\ell - 1$ features defined as

$$\phi_{k,m}(\mathbf{x}) = I[x_k = m], \qquad m = 1, \ldots, \ell - 1,$$

where $I[\cdot]$ is the indicator function.

- **Discrete ordered** variables (e.g. number of accidents) are encoded in "thermometer form". Suppose that variable k has ℓ ordered levels, denoted $1, 2, \ldots, \ell$. We introduce $\ell - 1$ features defined as

$$\phi_{k,m}(\mathbf{x}) = I[x_k \geq m], \qquad m = 2, \ldots, \ell.$$

For discrete variables, each level in the one-hot or thermometer encoding is considered independent and separately added to the initial feature set Ψ.

It should be emphasised that although the current scoring model is linear, non-linear extensions (e.g. feed-forward neural networks) are easy to accommodate in the framework introduced above. An interesting middle ground is to include in the set Ψ features that are formed by the *kernel evaluation* between two raw variables, similarly to the Kernel Matching Pursuit algorithm (Vincent and Bengio, 2002).

For reference, the above model is referred to as the *Basis Selection Regressor*.

2.3 Linear Regressor

For comparison purposes, we experimented with a *Linear Regressor* model, but rather than simply use it on the raw input variables, we applied the model on all the initial features generated by the *Basis Selection Regressor* model of subsection 2.2, without performing any feature selection; this corresponds to the initial set Ψ in algorithm 1 and excludes higher-order features. As will be seen in section 3, with such a large number of features, it is crucial to adequately regularise the model, which was accomplished by penalising with a squared $\mathcal{L}_2$ norm penalty.

As for the Basis Selection Regressor, the linear regressor expresses a predicted loss ratio as a linear combination of features of the same form as eq. (2).

$$\mathrm{LR}(\mathbf{x}; \boldsymbol{\beta}) = \beta_0 + \sum_{i=1}^{K} \beta_i x_i + \epsilon, \tag{4}$$

where $\boldsymbol{\beta} = (\beta_0, \ldots, \beta_K)'$ and $\mathbf{x} = (x_1, \ldots, x_K)'$, and ϵ is a noise term.

The training procedure minimises the following objective:

$$\hat{\boldsymbol{\beta}} = \arg\min_{\boldsymbol{\beta}} \left\{ \sum_{j=1}^{P} d_j p_j \Big(\mathrm{LR}(\mathbf{x}_j, \boldsymbol{\beta}) - \frac{c_j}{d_j p_j} \Big)^2 + \lambda \sum_{i=0}^{K} \beta_i^2 \right\} \tag{5}$$

where j iterates over all the P training profiles, λ is the regularisation coefficient (weight decay), and $c_j/(d_j p_j)$ is the regression target corresponding to the loss ratio for profile j.

In addition to this linear regression using all features, we included in the model set to be compared an elementary pruning procedure aimed at eliminating coefficients with low statistical significance. For this purpose, the training phase is carried out in three steps:

1. Linear regression on *all the features*, as above;

2. Pruning the less informative features, according to the criterion given next;

3. Linear regression on the remaining features.

The pruning phase keeps a *fixed number* of the most significant features (the exact parameters are listed in section 4). We use the *t-ratio* of the regression coefficients as the discriminant factor. It is computed as (Greene, 2002):

$$t = \hat{\beta}_i / \sqrt{S_{ii}} \tag{6}$$

where S_{ii} are the elements on the diagonal of the S matrix, the asymptotic sampling variance of the estimator $\hat{\boldsymbol{\beta}}$,

$$S = \mathrm{Var}(\hat{\boldsymbol{\beta}}) = \hat{\sigma}^2 (X'X)^{-1} \tag{7}$$

and $\hat{\sigma}^2$ is an unbiased estimator of the noise variance

$$\hat{\sigma}^2 = \frac{1}{P-K} \sum_{j=1}^{P} \Big(\mathrm{LR}(\mathbf{x}_j, \hat{\boldsymbol{\beta}}) - \frac{c_j}{d_j p_j} \Big)^2. \tag{8}$$

2.4 Bagging Model

Finally, due to the high noise level in the data, we experimented with a *Bagging* model, which is known to often improve performance in similar situations (Breiman, 1994). The idea is quite simple: starting with a training set of P examples, we generate B new training sets (called *bags*) using bootstrap resampling (Efron, 1979). Then, we train the B bags using the chosen sub-model, and we combine the solutions by averaging the outputs (for regression) or by voting (for classification).

In our experiments, we applied the Bagging meta-model on the Linear Regressor model (with and without basis pruning).

2.5 Sorting Criteria

Assuming that, according to our models, more than the allowed 10% of the volume of business should be ceded, then we must rank and sort the policies in order to choose which will actually be ceded. How this should be done bears close ties with the well-known knapsack problem of combinatorial optimisation, e.g. Cormen et al. (2001). The knapsack problem considers the situation where a certain number of essentials, up to a maximum weight, are to be put in a bag and taken on a trip. There are different versions of the problem, leading to different algorithmic solutions. Since each automobile insurance policy is either ceded or not, this corresponds to the *0–1 version* which would require us to use a dynamic programming approach. But, since each individual policy is very small compared to the volume of business of an insurer, we revert to the *fractional knapsack* version allowing us to use a greedy algorithm according to which policies are ceded from the first in rank until we reach 10% of the insurer's volume. As an approximation, the last ceded policy can be assumed to only be partially ceded so that precisely 10% of the insurer's total volume ends up in the pool. This situation corresponds to the regulations of the Québec pool, which puts a constraint on the *total premium volume* that can be ceded.

According to the greedy solution to the fractional knapsack problem, essentials are sorted in decreasing order of their value-to-weight ratio so that the total value, for a given maximum weight, is maximised. In our case, the value is the profit associated to ceding a policy and that corresponds to the expected claims level less 75% of the premium charged. The weight associated to each policy is the proportion of the entire book volume it represents, i.e. its premium divided by total volume. Thus, we obtain

$$\begin{aligned} \frac{\text{value}}{\text{weight}} &= \frac{\text{expected claims level} - 75\%\,\text{premium}}{\text{premium / book volume}} \\ &= \text{book volume}(\text{expected loss ratio} - 75\%). \end{aligned}$$

Removing constants shared by all policies that do not affect ranking, we conclude that policies should be *sorted in decreasing order of their expected loss ratio.* Models of subsections 2.2, 2.3, and 2.4 provide expected loss ratios as outputs and these can be readily used for sorting. On the other hand, the frequency/severity approach of subsection 2.1 leads to models that output an expected claims level. Dividing the output by the premium level, we obtain the desired expected loss ratio.

Let us now consider the case of the Ontario pool where the limit is set to 5% of the *number of policies* insured. In that case, all policies can be considered as having equal weight. Here again, each policy's value is its expected cession profit, obtained through a slightly different formula. The end result is that in order to maximise total profit, policies must be *sorted in decreasing order of their expected cession profit.*

These arguments are better illustrated through the use of a simple numerical example. Consider a very small canadian insurer with 20 policies that generate a volume of $10,000 in annual premiums. Of the 20 policies, three have been identified as having expected loss ratios above 75% and could therefore be ceded with (expected) profit. Policy A, with annual premium of $1,000, has an expected loss ratio of 100%. Policies B and C, both with annual premiums of $500, have

expected loss ratios of 110% and 120%, respectively.

Let us first assume this insurer operates in Québec so that a maximum of $1,000 (10% of the volume) in premiums can be ceded. Expected profit of ceding a policy is obtained as the annual premium multiplied by the difference between the expected loss ratio and 75%. For policies A, B, and C, we thus have expected profits of $250, $175, and $225, respectively. Now consider sorting these policies in decreasing order of expected profit. Accordingly, policy A is ceded first and the $1000 ceding limit is instantly reached. Total expected profit is therefore $250. On the other hand, if policies are ranked in decreasing order of expected loss ratio, policy C is first ceded with expected profit of $225. But since policy C's annual premium is $500, there remains room for us to cede policy B, with expected profit of $175, for a total of $400 in expected profit. In that case, it is more profitable to rank policies in decreasing order of expected loss ratio ($400 total expected profit) than in decreasing order of expected profit ($250 total expected profit).

Consider now the case of Ontario where policies with an expected loss ratio above 70% should be ceded. Insurers are allowed to cede up to 5% of their policies (or units). Policies are not ceded in full: insurers retain 15% of the risk with the pool assuming the remaining 85%. In our simple example, the insurer is limited to ceding a single policy (5%×20 policies). With a threshold of 70% and a cession proportion of 85%, we obtain expected profits of $255, $170, and $212.50 for policies A, B, and C, respectively. In that case, it is more profitable to rank policies in decreasing order of expected profit ($255 total expected profit) than in decreasing order of expected loss ratio ($212.50 total expected profit).

This simple example thus illustrates the fact that, whereas policies should be sorted in decreasing order of expected loss ratio in Québec, they should be sorted in decreasing order of expected profit in Ontario. The impact of choosing one sorting criterion over the other, for the Québec pool, is examined in subsection 4.1.

3 Experimental Procedure and Minimum-Rank Basis Selection

Insurance data is affected by a certain level of nonstationarities: the average premium level tends to vary across time according to market conditions. To better evaluate model performance in this context, we relied on the technique of *sequential validation* (Gingras et al., 2002), which is an empirical testing procedure that aims to emulate the behaviour of a rational decision maker updating a model as well as possible across time.

Sequential validation (also known as prequential analysis, rolling window, or simulated out-of-sample testing) is inspired by the well-known technique of cross-validation (Stone, 1974; Hastie et al., 2001), and is appropriate when the elements of a data set cannot be permuted freely, such as is the case for sequential learning tasks. One can intuitively understand the procedure from the illustration in Figure 1. One trains an initial model from a starting subset $Train(0)$ of the data,[d] which is tested out-of-sample on a data subset $Test(0)$ immediately following the end of the training set. This test set is then added to the training set for the next iteration, a new model is trained and tested on a subsequent test set, and so forth. As the figure shows, at the end of the procedure, one obtains out-of-sample test results for a large

[d] "Starting" is meant in the temporal sense.

fraction of the original data set, all the while always testing with a model trained on "relatively recent data".

Other techniques similar to sequential validation, such as streaming ensemble algorithm (SEA) (Street and Kim, 2001), have been suggested for on-line learning purposes. Note that these later techniques update models instantly, as new data arrives. In that context, a model may only see each instance once. This differs from our off-line implementation of sequential validation whereby models are updated on a regular (here, monthly) basis, but not instantly.

The raw variables include all standard rate-making variables typically used for automobile insurance, including driving experience, sex of driver, vehicle type, accident history, and so forth. We performed sequential validation by using the *most recent 12 to 48 months* (depending on the model) of policy data to use as the training set, testing on the following month, and rolling forward by one month (i.e. adding the test month to the new training set, and deleting the oldest one). In our data set, a total of 117 raw variables were usable as inputs; after encoding of discrete variables (as outlined in subsection 2.2), 1726 variables were included in the initial set (Ψ in algorithm 1) for stepwise variable selection.

Given such a large set of variables and the repeated selection that occurs within the sequential validation procedure, caution is advisable lest overfitting proves seriously detrimental.[e] To this end, we adapt the basis selection regression algorithm of subsection 2.2 as we do not perform a complete selection of features at each training iteration, but rather emphasise features that have worked well in the past. As before, let N be the total number of features that should be kept in the final model. We proceed as follows:

- The first three years of data are used as a **warm-up phase**. Sequential validation is performed as usual, but the test performance results are discarded. In this phase, a total of 25 models are trained, and the identity and rank (from 0 to $N-1$) of selected features, for each of the 25 models, are recorded. The set of N features arising out of warm-up stage w is denoted by Φ_w.

- Next comes the **validation phase**, where model performance is evaluated as in Figure 1. Consider step s of sequential validation. We decompose this phase into three steps:

  i. *(Feature generation for current stage).* We apply algorithm 1 to find the top N features given the training set *Train*(s). Denote by Φ_s the set of features selected at this step.

  ii. *(Rank averaging).* We compute the average rank of all features that have been generated in both the warm-up phase and all previous validation phases. More specifically, for all features ϕ, we compute

$$\overline{\mathrm{Rnk}}_\phi = \frac{1}{W+s}\Big(\sum_{w=1}^{W} \mathrm{Rnk}_\phi(\Phi_w) + \sum_{s'=1}^{s} \mathrm{Rnk}_\phi(\Phi_{s'})\Big),$$

  where $\mathrm{Rnk}_\phi(X)$ denotes the rank of feature ϕ in set X (which can range from 0 to $|X|-1$), or $|X|$ if ϕ is not part of X. In words, this step

[e] A delicate balance must be struck between the size of the sliding training set used in sequential validation, and the level of nonstationarities encountered in the data—a manifestation of the classical bias-variance dilemma.

considers *all features* that have been generated so far, and establishes a "quality measure" based on the average rank that each feature gets across all prior and current training sets.

iii. *(Final selection).* We form the subset of $\overline{N}$ features having the *highest average rank*, and apply again algorithm 1 to that subset only to select N features,[f] without allowing the creation of new higher-order features within the algorithm. This final set of features is used for sequential validation stage s.

The purpose of this procedure is to ensure a certain stability within selected features as we proceed with the sequential validation. We consistently observed that simply applying plain feature selection from anew at each sequential validation stage s produces quite unstable features that likely overfit the current training set.

This selection procedure is called the Minimum-Rank Basis Selection Procedure (BSR-MRK).

4 Results

Our data was provided by a mid-sized Canadian automobile insurer and ranges from October 1999 until December 2007. The total data consists of nearly one million policy records and associated claim data. The out-of-sample results presented here are obtained on the last 36 distinct test months, and follow the experimental procedure set forth in section 3; for reference, over these 36 months, the total claims filed were C$97M, the total premium volume was C$160M, for an overall loss ratio of nearly 60%.

Table 1 shows a summary of the model performances. For each model, we present the loss-ratio and the profit for 2% and 5% cession percentages, and also the cession percentage (between 1% and 5%) that gives the maximum profit.

Here is the brief description of each model. In what follows, all hyperparameters were selected by cross-validation.

- FS-LDA: *Frequency/Severity* model with *LDA* model for the frequency part. Training set is 48 months long. Weight decay of 10^{-3} for the *LDA*.
- FS-QDA: *Frequency/Severity* model with *QDA* model for the frequency part. Training set is 48 months long. Weight decay of 10^{-6} for the *QDA*.
- FS-WPLS: *Frequency/Severity* model with *WPLS* model for the frequency part. Training set is 12 months long.
- BSR: *Basis Selection Regressor* model with $N = 75$. Training set is 12 months long.
- BSR-MRK: *Min-Rank Basis Selection* model with $N = 75$ and $\overline{N} = 100$. Training set is 12 months long.
- LR: *Linear Regressor* model with $\lambda = 2 \times 10^7$. Training set is 12 months long.

[f] In our experiments, $\overline{N}$ is fixed at 100.

Table 1 Summary of model performances. Both the loss-ratio and the proportional profit (measured in basis points) are provided, at three cession levels: 2%, 5%, and the best level for the model (indicated in the column %).

Model	**Ceding 2%**		**Ceding 5%**		**Best Performance**		
	LR	Profit	LR	Profit	%	LR	Profit
FS-LDA	0.833	16.5	0.846	48.4	5%	0.846	48.4
FS-QDA	0.747	-0.5	0.703	-23.7	1%	0.932	18.0
FS-WPLS	0.801	10.1	0.815	32.5	5%	0.815	32.5
BSR	0.930	36.0	0.832	41.2	3%	0.902	45.8
BSR-MRK	0.904	30.7	0.893	71.9	5%	0.893	71.9
LR	0.947	39.4	0.843	46.5	4%	0.880	52.1
BG-LR	0.911	32.1	0.852	51.0	5%	0.852	51.0
LR-P	0.963	42.5	0.840	45.3	4%	0.878	51.4
BG-LR-P	0.856	21.1	0.860	55.1	5%	0.860	55.1

- BG-LR: *Bagging* of $B = 5$ *Linear Regressor* model with $\lambda = 2 \times 10^7$. Training set is 12 months long.
- LR-P: *Linear Regressor with Pruning* model with $\lambda = 2 \times 10^7$. Keep best 500 features. Training set is 12 months long.
- BG-LR-P: *Bagging* of $B = 5$ *Linear Regressor with Pruning* model with $\lambda = 2 \times 10^7$. Keep best 250 features. Training set is 12 months long.

In Figure 2, we show plots of the two best performing models, in terms of maximum profitability: the min-rank basis selection model and the linear regressor with bagging and pruning. For both models, the proportional profit is plotted against the percentage of ceded insureds (between 0 and 10%). For percentages above 5%, the BSR-MRK outperforms all other candidate models, including the linear regressor with bagging and pruning. In our experiments, bagging and pruning have had little impact on the results, compared to those obtained by using the min-rank procedure.

4.1 Impact of Sorting Criterion

As emphasised in subsection 2.5, the appropriate sorting criterion (either decreasing order of predicted loss ratio or expected cession profit) depends on the regulatory context in which a specific pool operates. For the Québec risk-sharing pool, which is volume-constrained, the appropriate criterion is the loss ratio; this is the case that we are considering with the current data.

To illustrate the practical difference between the two criteria, we compared them under the Basis Selection Regression (BSR) model with $N = 75$ selected bases. Table 2 shows the out-of-sample model performance at various cession percentages, for both sorting criteria (averaged over 5000 bootstrap resamples of policies and

Table 2 Performance as a function of the percentage of ceded premium volume, for the loss ratio (left) and the predicted profit (right) sorting criteria. In each case are given the average loss ratio of the ceded insureds and the proportional profit, which is expressed as a fraction (in basis points) of the total written premium volume, where 100 basis points = 1% (and, in parenthesis, the 95% bootstrap confidence intervals). All results are obtained from 5000 bootstrap resamples.

	Sorted by Loss Ratio		Sorted by Predicted Profit	
Ceded	Loss Ratio	Profit (bps)	Loss Ratio	Profit (bps)
1%	0.996	35.6 (12.5 – 60.9)	0.922	33.9 (9.3 – 60.8)
3%	0.889	58.5 (23.2 – 95.6)	0.890	71.1 (32.0 – 113.0)
5%	0.884	90.4 (43.6 – 138.9)	0.852	79.3 (33.0 – 127.3)
7%	0.858	99.4 (47.2 – 153.6)	0.840	92.1 (37.9 – 147.9)
9%	0.835	98.8 (43.1 – 157.6)	0.812	77.9 (20.3 – 137.7)

claims obtained on 36 consecutive months (Davison and Hinkley, 1997)). The performance measures are (i) the loss ratio of the *ceded insureds*, and (ii) the net profit as a proportion of the total premium volume, expressed in basis points. Recall that, under the Québec pool rules, it is profitable to cede an insured whose loss ratio is greater than 75%.

Consistent with previous results, we note that the model is very profitable across a wide range of cession percentages under both sorting criteria. The loss ratio of ceded insureds—a direct measure of model performance—exceeds both the profitability threshold (75%) and the average book loss ratio over the period (60%).

The loss ratio criterion obtains better results than the predicted profit criterion, except at a cession percentage of 3% where both criteria perform (almost) equally well. This is due to the fact that, in the 1%-3% range, cessions made according to the predicted profit criterion outperform those made according to the loss ratio criterion, a peculiarity weakness of the trained model.

A closer inspection reveals that the loss ratio criterion performs slightly better than the predicted profit on this task, both in terms of loss ratio of ceded insureds and earned net profit. The confidence intervals on the loss ratio always excludes the 75% threshold, which allows rejecting the null hypothesis that the model cannot cede profitably at the $p = 0.05$ level.

These results are illustrated in graphical form in Figure 3 expressed as a function of the percentage of ceded insureds extending up to 10% (the transfer limit in the Québec pool), and separately show the performance under the two sorting criteria: by loss ratio and predicted profit.

To arrive at a more definitive assessment of the performance difference between the loss ratio and predicted profit sorting criteria, we compare them in a pairwise fashion in a bootstrap test. More specifically, for each bootstrap resample of the test-set profiles and claims (over the same 36-month test duration), we compute the loss-ratio/predicted-profit difference for the two performance measures: proportional profit and loss ratio of ceded risks. We then tally these results over 5000 bootstrap repetitions to obtain estimates of the distribution of performance differences. Figure 3 shows the performance difference according to the proportional profit measure (left) according to the loss ratio measure (right). Up to about 5% of ceded risks, both sorting criteria give approximately the same performance. Beyond

this level, an advantage emerges for the loss-ratio sorting criterion, with the difference statistically significant at the 90% level (under both performance measures) at high cession percentages.

5 Robust Choice of RSP Operation Points

As mentioned, model selection for RSP purposes is based on out-of-sample performance, using a strict sequential validation framework. In this section, we attempt to replicate an operational constraint of insurers by including a three-month gap between the end of the training set and the start of the test set. The reason for this gap is that it can take up to three months for automobile insurance claims to fully develop.[g] Thus, models must be trained with data at least three months old to avoid systematic risk underestimation. Since virtually all accidents are declared within a few days, frequency estimation could easily be performed using one month old data, within the context of the frequency/severity approach. On the other hand, effect on severity would be material since, as mentioned, up to three months may be necessary before the exact claim amount is known.

Although careful attention is put to ensure this out-of-sample performance evaluation process is free of estimation bias, it remains very noisy. Noise in profit estimation is mainly caused by the presence of a few very large claims; a way to measure this noise and derive confidence intervals for the profit, at different cession percentages, is to draw bootstrap resamples of the test set. For each sample, a profit curve is obtained. Statistics for the profit, at each cession percentage of interest, are obtained from the set of all curves.

Test set bootstrap resampling allows to better assess the robustness of performance of models across data sets. Of course, other factors must be accounted for so that a complete view of the range of performances that can be expected to occur is displayed. In particular, confidence intervals obtained through this process of test set bootstrap resampling assume stationarity of the underlying data generating process which, as mentioned before, is not supported by empirical evidence. Confidence intervals must therefore be observed with caution.

6 Robust Choice of Evaluation Framework

From time to time, insurance contracts may be modified, often at moments when insurance companies would not normally be allowed to make the decision of putting the contract in the risk-sharing pool: while contracts can be taken out of the pool at any moment, ceding contracts is permitted in some situations only, like the renewal or the initial inception of a policy contract. The information that is available in order to describe an insured's profile at the moment a model must evaluate the risk associated to a particular profile may differ from the information available at the time the decision to cede that profile is actually made. The extent by which profile information may differ between these two events depends on the delay between them which in turn, depends on the integration of the data mining

[g] In other words, the time it takes, on average, for the damage to occur to the car, filing the claim and fully assessing the value of the damage.

model within the information technology infrastructure of the insurance company, often referred to as the *process maturity level*.

From the way the data is recorded and then supplied to a modeller, and considering the internal rules of the risk-sharing pool, it is obvious that finding the right way of preparing this data to be fed to machine-learning algorithms is a non-trivial task. Any change to a contract may affect the final performance of a model, but the model has to learn as if it was making a decision prior to knowing about such changes and then consider future modifications to calculate performance. Moreover, our initial experimentations have shown that the exact way in which we view a contract and its different profiles (wherein every change to a contract starts a new profile) may affect learning algorithms by indirectly "leaking" information from the future in very subtle ways, and therefore exhibit greater performance than what could actually be achieved. Thus, the goal is to create a data representation of a profile that is as close as possible to reality while remaining relatively simple and avoiding any leakage of information from the future. This representation will vary according to the above mentioned process maturity level.

Underwriters of insurance companies generate information that is fed to what is known as a *transactional database*. This transactional database is modified in real time. Actuaries, often operating in a separate department, are usually provided with a monthly update of a processed version of this transactional database, the *analytical database*. Implementing cession models at the transactional level so that cession recommendations can be given in real time within the underwriting interface, an *on-line* implementation, is a much more involved project than simply having the model interact with the analytical database and output a series of recommendations on a monthly basis, a *batch* implementation. An interesting aspect here is that, by comparing the performances obtained under these two maturity level hypotheses, one can easily derive a business case for a business intelligence project that involves bringing the system to a higher maturity level, simply by using the estimated difference in the risk-sharing pool profitability as a measure of the expected benefits for the project.

Batch implementation is simulated by making the decision to cede a profile, at policy renewal, based on in information available *before* the renewal date, i.e., we consider an insured's profile before renewal in order to decide to cede the insured from renewal onward. The single most important value that changes at renewal and that has substantial impact on profitability is premium level. Premium changes occur for various reasons, including changes in the risk (e.g., relocation, new car, etc.) but also because of strategic and marketing concerns that depend on current market conditions. In particular, brokers and underwriters may be given room to offer discounts if they feel an insured might be tempted to obtain coverage elsewhere. Within the context of an on-line implementation, we use up-to-date information that reflects these changes.

In Figure 4, we illustrate the effect of process maturity level on profitability. We use a Basis Selection Regressor with Bagging ($B = 5$). The left-hand side plot is obtained using a scenario that assumes that any profile can be ceded and that up-to-date information is always available when the decision is to be made, i.e., an on-line implementation. On the right-hand side, we consider a *batch* implementation whereby fresh information is not available and the decision to cede has to be made even before we know whether a contract will be renewed, and this decision has to

stand for the full length of the contract. Whereas the on-line implementation leads to a maximum profit of 187 basis points i.e., 1.87% of premium volume, the batch implementation maximum profit is a mere 76 basis points. Thus, accounting for up-to-date information provides an additional 111 basis points in profits.

7 Conclusion

In this paper, we compared the performance of different feature selection and modelling approaches through their performance on the task of selecting risks to be ceded to automobile insurance risk-sharing pools. At low cession percentages (ca. 2%), linear regression models perform best. But as we consider higher cession percentages (5-10%), the proposed BSR-MRK approach outperforms all others. Furthermore, if one is allowed to select the cession percentage optimally (profit maximising), then BSR-MRK again provides the best results. We showed that the regulatory constraints imposed by the jurisdiction in which the pool operates affect the optimal sorting criterion, and this has practical consequences both in terms of loss ratio of ceded insureds and overall net profitability. Finally, we described how software maturity level affects the information that can be used for cession decision purposes. This in turn affects profitability. By measuring differences in profitability between software maturity levels, one can easily derive sound business cases for business intelligence projects targeted at bringing software to a higher maturity level.

As a prospect for future work, we wish to consider situations where an insurer is operating close to the transfer limit allowed by the pool. In some jurisdictions, pool regulations strictly control the moments where insureds can be ceded so that, in some cases, it may be advantageous to avoid transferring up to the limit, in order to reserve space for better risks that will be anticipated to come along in the future. Such a method could further enhance the profitability of the approach in strongly budget-constrained situations.

References

Apte, Chidanand, Edna Grossman, Edwin P. D. Pednault, Barry K. Rosen, Fateh A. Tipu and Brian White (1999), 'Probabilistic estimation-based data mining for discovering insurance risks', *IEEE Intelligent Systems* **14**(6), 49–58.

Artís, Manuel, Mercedes Ayuso and Guillén Montserrat (2002), 'Detection of automobile insurance fraud with discrete choice models and misclassified claims', *Journal of Risk & Insurance* **69**(3), 325–340.

Breiman, Leo (1994), 'Bagging predictors', *Machine Learning* **24**(2), 123–140.

Chen, S. (1995), Basis Pursuit, PhD thesis.

Cormen, Thomas H., Charles E. Leiserson, Ronald L. Rivest and Clifford Stein (2001), *Introduction to Algorithms*, second edn, MIT Press, Cambridge, MA.

Davison, A. C. and D. V. Hinkley (1997), *Bootstrap Methods and Their Application*, Cambridge Series in Statistical and Probabilistic Mathematics, Cambridge University Press, Cambridge, UK.

Dugas, Charles, Yoshua Bengio, Nicolas Chapados, Pascal Vincent, Germain Denoncourt and Christian Fournier (2003), Statistical learning algorithms applied to automobile insurance ratemaking, *in* A. F.Shapiro and L. C.Jain, eds, 'Intelligent and Other Computational Techniques in Insurance: Theory and Applications', Series on Innovative Intelligence, 6, World Scientific Publishing Company, pp. 137–197.

Efron, B., T. Hastie, I. Johnstone and R. Tibshirani (2004), 'Least angle regression', *The Annals of Statistics* **32**(2), 407–499.

Efron, Bradley (1979), 'Bootstrap methods: Another look at the jackknife', *Annals of Statistics* **7**(1), 1–26.

Frank, Ildiko and Jerome Friedman (1993), 'A statistical view of some chemometrics regression tools', *Technometrics* **35**(2), 109–148.

Friedman, J.H. and J.W. Tuckey (1974), 'A projection pursuit algorithm for exploratory data analysis', *IEEE Transactions on Computers* **C-23**(9).

Gingras, François, Yoshua Bengio and Claude Nadeau (2002), On out-of-sample statistics for time-series, Technical Report 2002s-51, CIRANO.

Greene, William H. (2002), *Econometric Analysis*, fifth edn, Prentice Hall, Englewood Cliffs, NJ.

Guyon, I. and A. Elisseeff (2003), 'An introduction to variable and feature selection', *Journal of Machine Learning Reasearch* **3**, 1157–1182.

Hastie, Trevor, Robert Tibshirani and Jerome Friedman (2001), *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*, Springer, Berlin, New York.

Hoerl, A.E. and R.W. Kennard (1970), 'Ridge regression: Biased estimation for nonorthogonal problems', *Technometrics* **12**(3), 55–67.

Kohavi, R. and G. John (1997), 'Wrappers for feature selection', *Artificial Intelligence* **97**(1–2), 273–324.

Liu, H., E.R. Dougherty, J.G Dy, K. Torkkola, E. Tuv, H. Peng, C. Ding, F. Long, M. Berens, L. Parsons, Z. Zhao, L. Yu and G. Forman (2005), 'Evolving feature selection', *IEEE Intelligent Systems* **20**(6), 64–76.

Mallat, S.G. and Z. Zhang (1993), 'Matching pursuits with time-frequency dictionaries', *IEEE Transactions on Signal Processing* **41**(12), 3397–3415.

Oh, I.-S., J.-S. Lee and B.-R. Moon (2004), 'Hybrid genetic algorithms for feature selection', *IEEE Transactions on Pattern Analysis and Machine Intelligence* **26**(11), 1424–1437.

Ontario Risk Sharing Pool Procedures Manual (2006), Technical report, Facility Association.

Srebro, N. and T Jaakkola (2003), Weighted low-rank approximations, *in* 'Proceedings of the Twentieth International Conference on Machine Learning (ICML-2003)', Washington D.C.

Stone, M. (1974), 'Cross-validatory choice and assessment of statistical predictions', *Journal of the Royal Statistical Society, B* **36**(1), 111–147.

Street, N. and Y. Kim (2001), A streaming ensemble technique (sea) for large-scale classification, *in* 'Proceedings of the 7th ACM SIGKDD international conference on knowledge discovery and data mining', ACM, pp. 377–382.

Su, C.-T. and Y.-H. Hsiao (2009), 'Multiclass MTS for simultaneous feature selection and classification', *IEEE Transactions on Knowledge and Data Engineering* **21**(2), 192–205.

Tibshirani, R. (1996), 'Regression shrinkage and selection via the lasso', *Journal of the Royal Statistical Society Series B* **58**, 267–288.

Viaene, Stijn, Richard A. Derrig, Bart Baesens and Guido Dedene (2002), 'A comparison of state-of-the-art classification techniques for expert automobile insurance claim fraud detection', *Journal of Risk & Insurance* **69**(3), 373–421.

Vincent, Pascal and Yoshua Bengio (2002), 'Kernel matching pursuit', *Machine Learning* **48**, 169–191.

Wang, L. and X. Fu (2005), *Data Mining with Computational Intelligence*, Springer.

Wang, L.P., F. Chu and W. Xie (2007), 'Accurate cancer classification using expressions of very few genes', *IEEE/ACM Transactions on Computational Biology and Bioinformatics* **4**(1), 40–53.

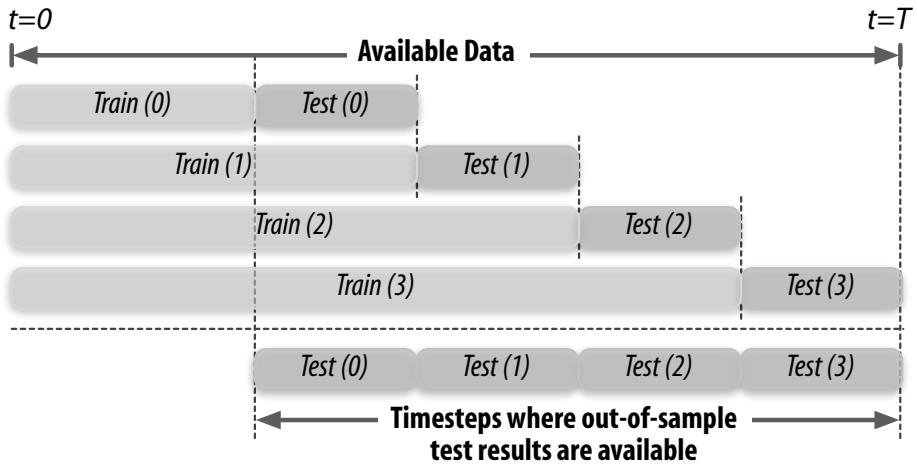

Figure 1 Illustration of the sequential validation procedure, where training and testing stages are interleaved through time.

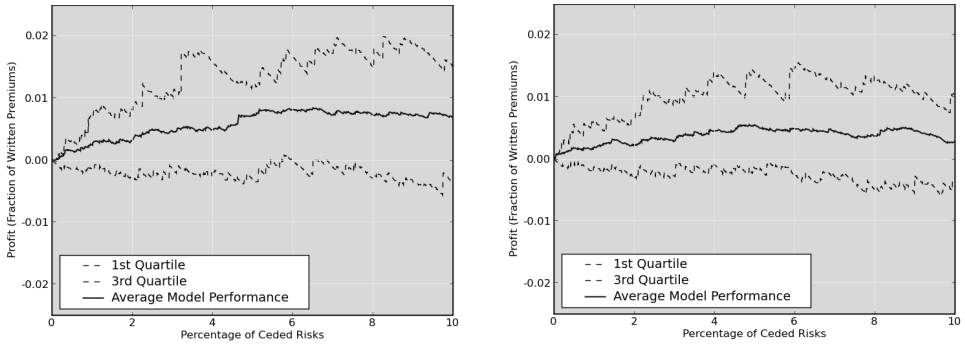

Figure 2 Proportional profit for the two most profitable models: min-rank basis selection regressor (left) and the linear regressor with bagging and pruning (right)

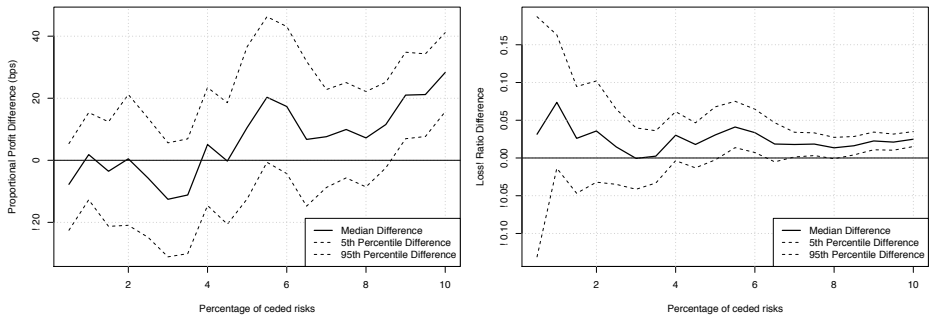

Figure 3 Difference between sorting by the loss-ratio and the predicted profit criteria. The difference in loss ratio (left) and the difference in predicted profit, expressed in basis points (right) are plotted as a function of the percentage of ceded risks. At a 90% confidence level, the loss-ratio criterion is significantly better that the predicted profit criterion for percentages above 8.5% (left) and 6.5% (right).

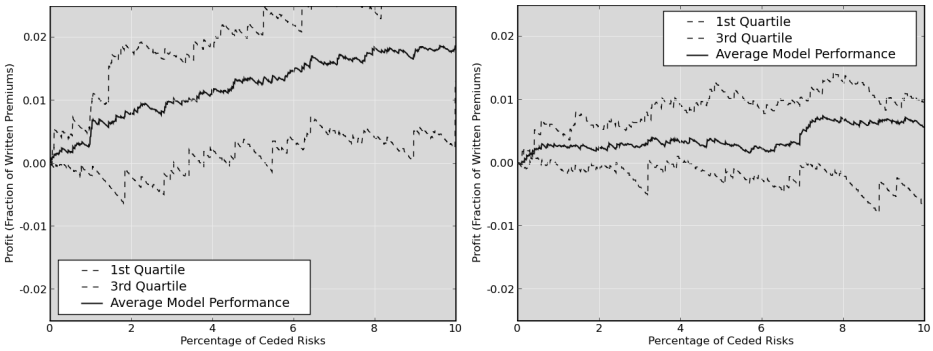

Figure 4 Left: online implementation assumes up-to-date information is available when a decision is to be made. Right: batch implementation uses information available prior to renewal in order to evaluate a profile.