Struct Multidisc Optim (2014) 50:523–535
DOI 10.1007/s00158-014-1105-z

RESEARCH PAPER

Optimal SVM parameter selection for non-separable and unbalanced datasets

Peng Jiang · Samy Missoum · Zhao Chen

Received: 1 November 2013 / Revised: 5 February 2014 / Accepted: 20 February 2014 / Published online: 8 July 2014

Abstract This article presents a study of three validation metrics used for the selection of optimal parameters of a support vector machine (SVM) classifier in the case of non-separable and unbalanced datasets. This situation is often encountered when the data is obtained experimentally or clinically. The three metrics selected in this work are the area under the ROC curve (AUC), accuracy, and balanced accuracy. These validation metrics are tested using computational data only, which enables the creation of fully separable sets of data. This way, non-separable datasets, representative of a real-world problem, can be created by projection onto a lower dimensional sub-space. The knowledge of the separable dataset, unknown in real-world problems, provides a reference to compare the three validation metrics using a quantity referred to as the "weighted likelihood". As an application example, the study investigates a classification model for hip fracture prediction. The data is obtained from a parameterized finite element model of a femur. The performance of the various validation metrics is studied for several levels of separability, ratios of unbalance, and training set sizes.

Keywords Non-separable and unbalanced datasets · Support vector machines · Cross validation · Validation metrics

P. Jiang · S. Missoum (✉)
Aerospace and Mechanical Engineering Department,
University of Arizona, Tucson, Arizona
e-mail: smissoum@ame.arizona.edu

Z. Chen
Mel and Enid Zuckerman College of Public Health,
University of Arizona, Tucson, Arizona

1 Introduction

In many areas of science and engineering, models are used to predict the responses of a system, the outcome of a physical process, or evaluate a risk. A model can be purely computational such as finite element or computational fluid dynamics simulations. It can also be solely constructed from experimental or clinical data. Regardless of how the model is built, the most important characteristic is its predictive ability. In other words, how is the model going to represent reality beyond the data that was used to construct it?

There exist several validation metrics to quantify the predictive ability of a model. Examples of metrics are the accuracy, balanced accuracy (Brodersen et al. 2010), Area Under the Receiver Operating Characteristic (ROC) curve (AUC) (Metz 1978; Fawcett 2006), Matthews correlation coefficient (Matthews 1975) and F-score (Rijsbergen 1979). These metrics are used to select the best parameters of a model through cross-validation (Kohavi 1995). Beyond the choice of optimal parameters, these metrics can also be used on a validation dataset that has not been involved in the training of the model. Although some of these metrics have gained wide acceptance, care must be taken in some situations. For instance, in the case of a binary classification problem (e.g., failure or safe), two types of difficulties can occur, especially if the model is constructed using physical experiments or clinical data. The first one is the non-separability of the data. This phenomenon suggests the existence of factors that are not accounted for as well as measurement errors (these two aspects are often simply viewed as "randomness"). The second problem stems from the fact that the data might be unbalanced. For instance, in the reliability or biomedical fields, there is often a much

smaller number of failure or unhealthy cases than safe or healthy cases.

The objective of this article is to test the performance of three well-known validation metrics for a classification problem where both issues of non-separability and unbalanced datasets occur. The three metrics are: AUC, basic accuracy, and balanced accuracy. These metrics are used to find the optimal parameters of a support vector machine (SVM) classifier.

In this work, computational data was used instead of experiments in order to have more control on the features of the datasets. In order to create non-separable cases, a set of separable data was created and then projected onto a sub-space thus leading to non-separable data. In addition, this approach provides an exact reference metric, referred to as "weighted likelihood", to compare the various validation metrics.

As part of an ongoing effort on the prediction of hip fracture, the test cases presented in this work are based on a finite element model of a femur. Given a failure criterion and a set of parameters (e.g., femoral head geometry), an SVM separating failed and safe samples is constructed. Several scenarios of non-separable and unbalanced datasets are studied.

The paper is organized as follows: A review of SVM for balanced and unbalanced data is presented in the background Section 2. The section also introduces the three validation metrics used in this paper: accuracy, balanced accuracy, and AUC. Section 3 presents the details of the parameter selection strategy in case of non-separable and unbalanced data. It also provides the derivation for a likelihood-based reference metric. Finally, Section 4 provides the results and conclusions based on various datasets with different levels of unbalance, non-separability, as well as sizes of training samples.

2 Background

2.1 SVM classification

In this paper, we are concerned with the predictive capability of a classification model. The classifier chosen in this work is referred to as an SVM (Cristianini and Shawe-Taylor 2000; Burges 1998). SVM is now a widely accepted machine learning technique that has been used in many applications (Basudhar et al. 2008; Yang 2010; Tay and Cao 2001; Basudhar and Missoum 2010; Konig et al. 2005).

An SVM is used to construct an explicit boundary that separates samples belonging to two classes labeled as $+1$ and -1. Given a set of N training samples $\mathbf{x}_i$ in a d-dimensional space, and the corresponding class labels, a linear SVM separation function is found through the solution of the following quadratic programming problem:

$$\begin{aligned} \min_{\mathbf{w},\boldsymbol{\xi},b} \quad & \tfrac{1}{2}\|\mathbf{w}\|^2 + C\sum_{i=1}^{N}\xi_i \\ s.t. \quad & y_i(\mathbf{w}\cdot\mathbf{x}_i - b) \geq 1-\xi_i \\ & \xi_i \geq 0, i = 1,\ldots,N \end{aligned} \tag{1}$$

where b is a scalar referred to as the bias, y_i are the classes, C is the cost coefficient, and ξ_i are slack variables which measure the degree of misclassification of each sample $\mathbf{x}_i$ in the case the data is non separable. SVM can be generalized to the nonlinear case by writing the dual problem and replacing the inner product by a kernel:

$$\begin{aligned} \min_{\boldsymbol{\lambda}} \quad & \frac{1}{2}\sum_{i,j}\lambda_i\lambda_j y_i y_j K(\mathbf{x}_i,\mathbf{x}_j) - \sum_{i=1}^{N}\lambda_i \\ s.t. \quad & \sum_{i=1}^{N}\lambda_i y_i = 0 \\ & 0 \leq \lambda_i \leq C, i = 1,\ldots,N \end{aligned} \tag{2}$$

where λ_i are Lagrange multipliers. The training samples for which the Lagrange multipliers are non-zero are referred to as the *support vectors*. The number of support vectors is usually much smaller than N, and therefore, only a small fraction of the samples affect the SVM equation.

The corresponding SVM boundary is given as:

$$s(\mathbf{x}) = b + \sum_{i=1}^{N}\lambda_i y_i K(\mathbf{x}_i,\mathbf{x}) = \mathbf{0} \tag{3}$$

The classification of any arbitrary point $\mathbf{x}_i$ is given by the sign of $s(\mathbf{x}_i)$. The kernel function K in (3) can have several forms, such as polynomial or Gaussian radial basis kernel, which is used in this article:

$$K(\mathbf{x}_i,\mathbf{x}) = \exp\left(-\gamma\,||\mathbf{x}_i-\mathbf{x}||^2\right), \gamma > 0 \tag{4}$$

where γ is the width parameter of the Gaussian kernel.

For some classification problems, especially when handling data collected for biomedical studies, the data is usually unbalanced. In other words, a class might be far more populated than the other one. It order to balance the data, Osuna and Vapnik (Osuna et al. 1997; Vapnik 1999) proposed using different cost coefficients (i.e., weights) for the different classes in the SVM formulation. The corresponding linear formulation is:

$$\begin{aligned} \min_{\mathbf{w},\boldsymbol{\xi},b} \quad & \tfrac{1}{2}\|\mathbf{w}\|^2 + C^+\sum_{i=1}^{N^+}\xi_i + C^-\sum_{i=1}^{N^-}\xi_i \\ s.t. \quad & y_i(\mathbf{w}\cdot\mathbf{x}_i - b) \geq 1-\xi_i \\ & \xi_i \geq 0, i = 1,\ldots,N \end{aligned} \tag{5}$$

where C^+ and C^- are cost coefficients for $+1$ and -1 class respectively. N^+ and N^- are number of samples from

+1 and −1 classes. The coefficients are typically chosen as (Chang and Lin 2011):

$$C^+ = C \times w^+$$
$$C^- = C \times w^- \quad (6)$$

where C is the common cost coefficient for both classes, w^+ and w^- are the weights for +1 and −1 class respectively. The weights are typically chosen as $w^+ = 1$ and $w^- = N^+/N^-$. This article uses the weighted formulation of SVM for all the results.

2.2 k-Fold cross validation

Cross validation is a commonly used technique to find the parameters of a model such as the cost coefficient and the width parameter for an SVM. In k-fold cross-validation, samples from both safe and failed classes in the training set are randomly divided into k subsets of equal size. Of all the k subsets, a single subset is used as validation samples for evaluating the model while the remaining $k - 1$ subsets are used as training samples. The cross-validation process is then repeated k times, with each of the k subsets used exactly once. The k results from the "folds" are averaged to produce a single estimation of model performance. In this article, 10-fold cross-validation is used (McLachlan et al. 2004; Kohavi 1995). Three validation metrics are presented below: accuracy, AUC, and balanced accuracy.

2.3 Commonly used validation metrics

2.3.1 Accuracy and balanced accuracy

For convenience, we introduce the following abbreviations: N^+ (number of "positive" samples), N^- (number of "negative" samples), TP (number of true positives or correctly classified positive samples), TN (number of true negatives or correctly classified negative samples), FP (number of false positives or misclassified negative samples), FN (number of false negatives or misclassified positive samples).

Accuracy is an intuitive and widely used criterion for evaluating a classifier. It works well if the number ofsamples in different classes are balanced. The criterion can be expressed as:

$$\text{Accuracy} = \frac{TP + TN}{N^+ + N^-} \quad (7)$$

Leave-one-out error is a validation metric based on accuracy. Some upper bounds of the leave-one-out error can be derived. These bounds include Jaakkola-Haussler bound (Jaakkola et al. 1999), radius-margin bound (Vapnik 1998), Opper-Winther bound (Opper and Winther 2000) as well as span bound (Vapnik and Chapelle 2000).

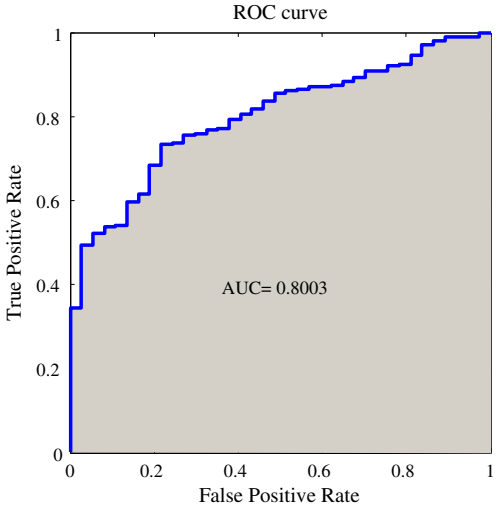


Fig. 1 An example of ROC curve and corresponding AUC

When the two classes are highly unbalanced, the performance of this measure may lead to acute "over-fitting" (see results in Section 4). In the case the data is not balanced, the balanced accuracy can be used (Brodersen et al. 2010):

$$\text{Balanced Accuracy} = \frac{1}{2}\left(\frac{TP}{N^+} + \frac{TN}{N^-}\right) \quad (8)$$

2.3.2 Area under ROC curve (AUC)

The ROC curve (Metz 1978) is a graphical representation of the relation between true and false positive predictions for all the possible decision thresholds. In the case of SVM classification, thresholds are defined by the SVM value. More specifically, for each threshold a True Positive Rate ($TPR = TP/(TP + FN)$) and a False Positive Rate ($FPR = FP/(FP + TN)$) are calculated. Graphed as coordinate pairs, these measures form the ROC curve. An example ROC curve is depicted in Fig. 1. Once the ROC curve constructed, the "area under the curve" (AUC) is used as a validation metric. A perfect AUC will be equal to one. It can be interpreted as the "probability that a classifier will rank a randomly selected positive sample higher than a randomly chosen negative one" (Fawcett 2006). An AUC value of 0.5 indicates no discriminative ability between samples from different classes, which would be equivalent to flipping a coin to make a decision.

2.4 Parameter selection strategy for SVM

The optimal parameters C and γ of the SVM Gaussian kernel are the maximizers of the cross-validation metrics

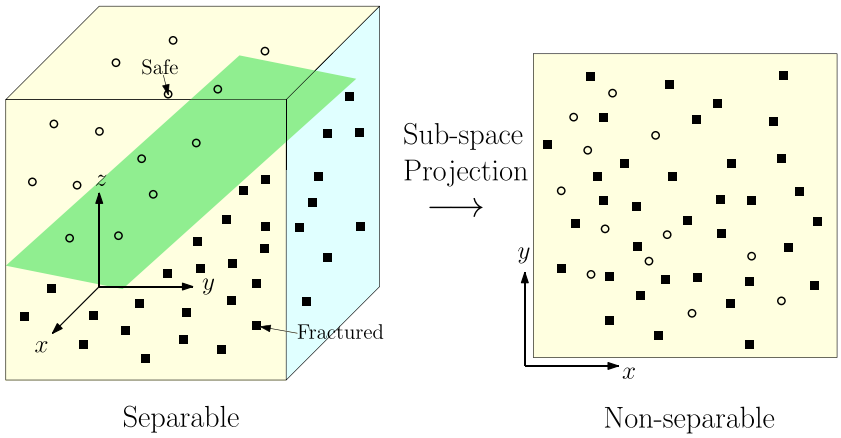


Fig. 2 Manufactured non-separable samples obtained by projection of a separable dataset in a higher dimensional space

described in the previous section. A typical approach consists of constructing a grid and choosing the maximizer out of the discrete set of points. Another approach is to use a global optimization method such as a Genetic Algorithm (Goldberg and Holland 1988) or DIRECT (Björkman and Holmström 1999). Typical ranges of parameters, as chosen in this work, are: $C \in [2^{-10}, 2^{17}]$ and $\gamma \in [2^{-25}, 2^{10}]$. Within these ranges, the SVM can be a hard or soft classifier and the decision boundary can go from a hyperplane to a highly non-linear hypersurface.

2.5 Confidence interval estimation

In order to obtain a confidence interval for the various validation metrics, bootstrapping can be used (Efron and Tibshirani 1997; Varian 2005). For a dataset of size n, bootstrapping works by uniformly selecting, with replacement, n data points from the pool. The validation metric can be recalculated from these bootstrap samples. This process is repeated for a large number of times to form a distribution of validation metric values. From this distribution, 95% or 99% confidence intervals can be empirically estimated.

3 Methodology

3.1 Manufactured non-separable cases

In many engineering or biomedical problems, the data is not separable. This stems from the fact that the data is usually studied in a finite dimensional space which does not account for all the factors that might influence an outcome. For instance, when studying hip fracture data from a cohort of patients, the results are typically reported in a space made of parameters such as age, weight, bone mineral density, etc. Even in the case when this space is high dimensional, the data might still not be separable as the number of dimensions used might still be a fraction of the actual number of factors involved in the occurrence of hip fracture. In other words, non-separable data can be seen as the projection of otherwise separable data onto a space of lower dimensionality. Figure 2 depicts an example in a three dimensional space where the data is separable (separation represented by a plane) and the corresponding projection on a two dimensional space where the data is no longer separable.

Based on these observations, this article proposes to manufacture non-separable cases by projecting the data from a separable space onto its sub-space. The manufactured dataset will exhibit the same type of non-separability encountered in experimental or clinical database. Using this approach, the normally unknown separable case is available and enables the derivation of a "reference" quantity to compare the various validation metrics.

In general, another origin of non-separability also stems from errors in measurements whereby the values of the parameters are not known exactly. By itself, this will contribute to non-separability. Without any loss of generality and for the sake of clarity, this difference will be considered immaterial. Alternatively, the reader can assume that there is no uncertainty on the measured data.

3.1.1 Weighted likelihood

This section introduces a metric which enables the comparison between classifiers constructed with non-separable data. This comparison is made possible by using information from the, usually unknown, model constructed from separable data. The proposed metric is based on the following idea: the non-separable case will produce misclassification. Because of the availability of the classifier with separable data (an approximation of the Bayes classifier (Murty and Devi 2011), it is then possible to find the probability P_i that a misclassified sample $\mathbf{x}_i$ belongs to its predicted class.

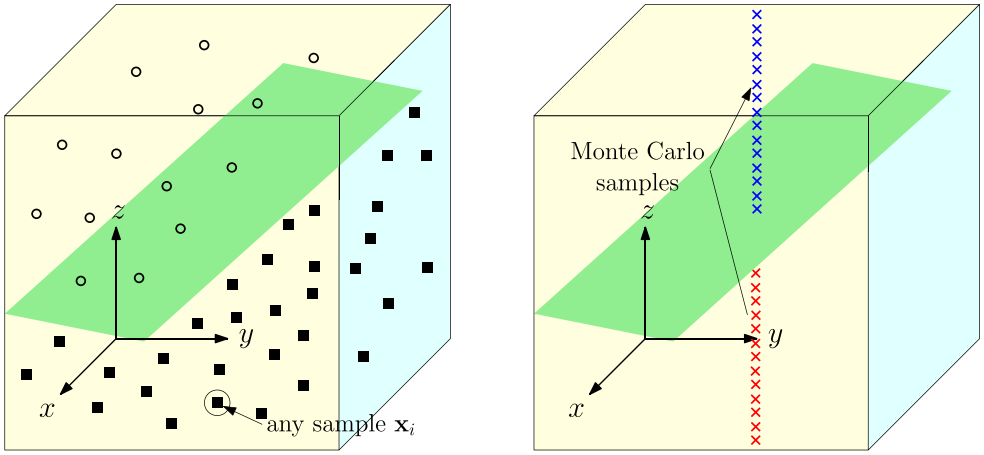


Fig. 3 Calculation of the probability of belonging to a class for any sample $\mathbf{x}_i$ using Monte Carlo simulation. The sampling is performed based on the variables that were removed to obtain non-separable data

Gathering and averaging this information for all the misclassified samples, one can form a "weighted likelihood" defined as:

$$\overline{\mathcal{L}}_w = \frac{1}{N_{misc}} \sum_{i=1}^{N_{misc}} w_i \log(P_i) \tag{9}$$

$$P_i = \begin{cases} P(-1|\mathbf{x}_i) & \text{if } s(\mathbf{x}_i) < 0 \text{ and } s(\mathbf{x}_i) y_i < 0 \\ P(+1|\mathbf{x}_i) & \text{if } s(\mathbf{x}_i) > 0 \text{ and } s(\mathbf{x}_i) y_i < 0 \end{cases},$$

$$w_i = \begin{cases} 1 & \text{if } y_i = +1 \\ \frac{N^+}{N^-} & \text{if } y_i = -1 \end{cases}$$

where N_{misc} is the number of misclassified samples by the SVM constructed in the sub-space, y_i is the actual class of sample $\mathbf{x}_i$ and w_i is the weight of for sample $\mathbf{x}_i$. The weights are used for the case where the data is unbalanced.

The probability P_i can be efficiently obtained using Monte-Carlo simulations using the SVM from the separable case. Figure 3 provides an example of the methodology in a three dimensional space. For every data point, Monte-Carlo samples are generated along the dimension that is removed to generate a non-separable dataset. In the case of an SVM where the space is split into positive and negative regions, the probability of belonging to the +1 class for any sample $\mathbf{x}_i$ can be calculated as:

$$P(+1|\mathbf{x}_i) = \frac{N_i^{MC+}}{N^{MC}}, i = 1, \ldots, N \tag{10}$$

where $\mathbf{x}_i$ is the i^{th} sample in the sub-space, N^{MC} is the number of Monte Carlo samples, and N_i^{MC+} represents the number of Monte Carlo samples that are predicted as positive by the SVM constructed with the separable dataset. The corresponding probability of belonging to the -1 class can be calculated as:

$$P(-1|\mathbf{x}_i) = 1 - P(+1|\mathbf{x}_i), i = 1, \ldots, N \tag{11}$$

That is, $\overline{\mathcal{L}}_w$ is a measure of how correct is the classifier in the sub-space compared to the one with separable samples. The larger algebraic value of the weighted likelihood, the better is the SVM in comparison to the actual separable SVM.

It is also possible to obtain a reference value of the weighted likelihood. This is done by increasing the number of samples until the weighted likelihood converges. The converged value of the likelihood is then considered as the reference value.

4 Results

This section provides results for various SVM classifiers trained and tested using different validation metrics on various dataset configurations. As described in the methodology section, non-separable datasets are generated by projection of a separable case in higher dimension. Sample sets used in this section have different sizes, levels of separability as well as levels of unbalance. The scores for the three validation metrics and the weighted likelihood $\overline{\mathcal{L}}_w$ are provided. The SVM model is constructed using a training set which will be used for cross-validation. All the validation metrics are evaluated on a different test set that was not used in the training process. Each result is provided with a corresponding 95% confidence interval.

The following notations are used in this section:

- $\overline{\mathcal{L}}_w$: weighted likelihood.
- Ref. $\overline{\mathcal{L}}_w$: reference value of weighted likelihood (see Section 3).
- AUC: area under ROC curve.
- Acc: accuracy.
- $Bacc$: balanced accuracy.

As a test case, we consider the problem of hip fracture prediction. SVM is used as a classifier between fractured and healthy individuals. The data is obtained from a fully parameterized finite element model as described in the following section.

Table 1 Parameters implemented in the finite element model of a femur

Region	Name	Parameter
	Geometric Parameters	
Neck	Outer diameter	*d_neck*
	Thickness of the cortical bone	*t_neck*
Intertrochanter	Outer width	*w_trochanter*
	Thickness of the cortical bone	*t_trochanter*
Shaft	Outer diameter	*d_shaft*
	Thickness of the cortical bone	*t_shaft*
	Neck-shaft angle	*alpha*
	Other Parameters	
	Weight	*weight*
	Young's modulus	$E_{Cortical}$
	Poisson ratio	v

Table 2 Distribution of 3 parameters for sample generation

Parameter	Distribution
E_{Cortical}	$N(17.80, 2.10)$ (GPa)
T_{Neck}	$N(1.58, 0.26)$ (mm)
weight	$N(63.96, 15.90)$ (Kg)

4.1 Fully parameterized finite element model of a femur

A fully parameterized finite element model of a femur is constructed in ANSYS using ANSYS Parametric Design Language (APDL) (ANSYS 2011). The model parameters are listed in Table 1 and depicted in Fig. 4a. In addition, Fig. 4b depicts an example of contour of principal strain.

4.2 Sample generation and failure criterion

The data used in the experiments are obtained by sampling three variables: the Young's modulus of the cortical bone (E_{Cortical}), the thickness of cortical bone around the neck (T_{Neck}) and the weight of the individual. Each variable follows a normal distribution with means and standard deviations provided in Table 2. A total of 2000 samples were drawn and evaluated through the finite element model from which stress and strain information can be obtained. In this work, failure (i.e., fracture) is assessed using the maximum principal strain both in tension and compression (Bayraktar et al. 2004). Other measures could be used (Doblaré and García J 2003), however the choice of the measure does not remove any of the generality of the conclusions of the article. The thresholds chosen for the maximum principal strains are 1.04% in compression and 0.73% in tension (Grassi et al. 2012). Based on this failure criterion, samples can be classified into safe (+1) or failed (−1) classes.

In the three dimensional space, the data is separable because the output of the finite element model is deterministic (Fig. 5). Projection of the three-dimensional samples onto the E_{Cortical} and T_{Neck} plane leads to non-separable samples as shown in Fig. 5.

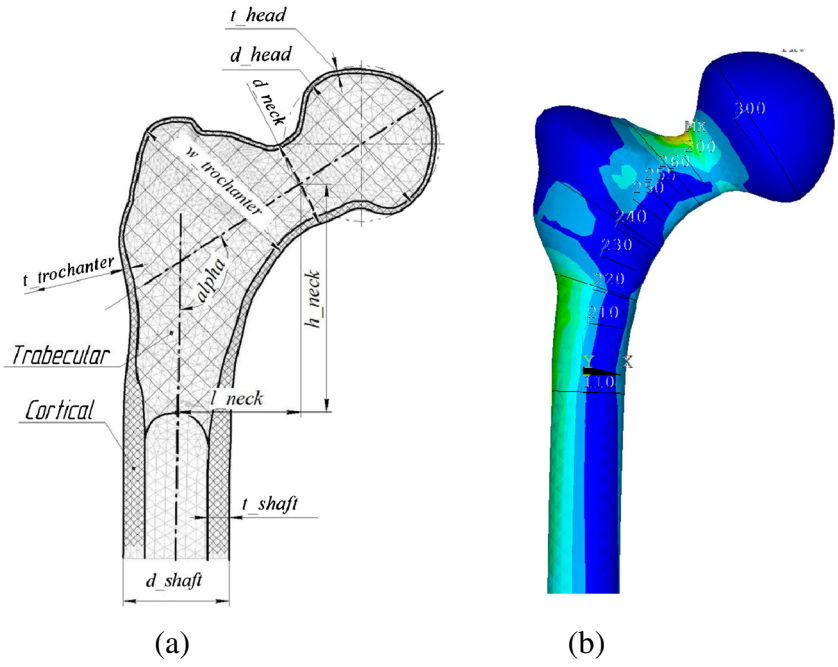


Fig. 4 Fully parameterized finite element model of a femur: **a** Parameters. **b** Contour of principal strain. (Max principal strain is around the neck.)

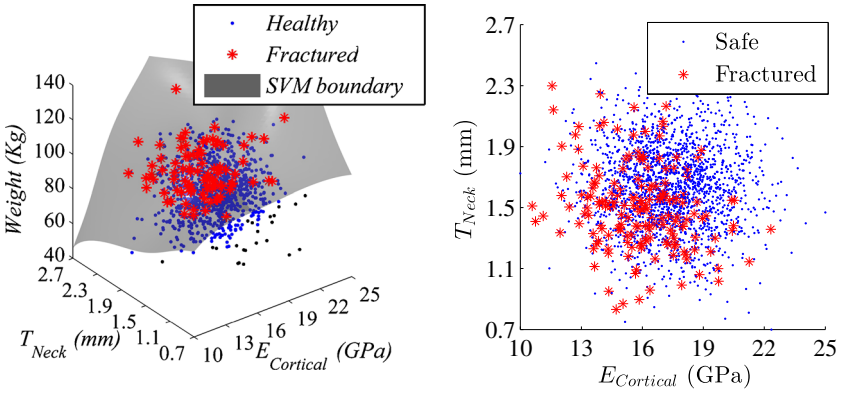


Fig. 5 3 dimensional SVM with separable data (left). Projection onto the (E_{Cortical} and T_{Neck}) space (right)

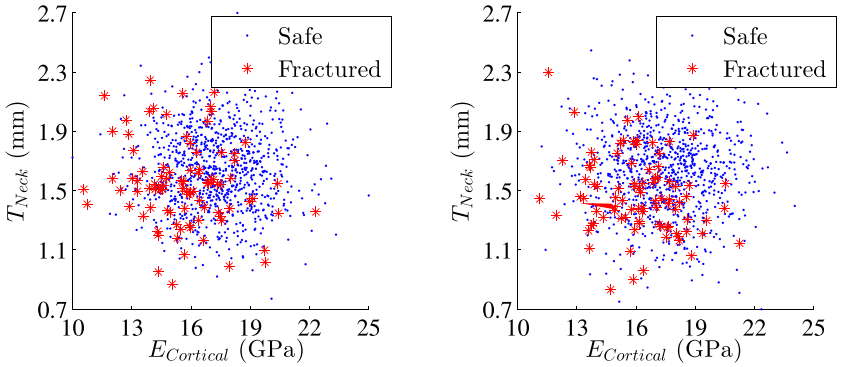


Fig. 6 Examples of training (**a**) and validation (**b**) sample sets

4.3 Parameter selection using *AUC*, *Acc*, and *Bacc*.

Based on the two-dimensional non-separable samples, the three cross-validation metrics can be used for parameter selection as well as validation of the selected model. For this purpose, the data is randomly split into training and validation sets of equal size as shown in Fig. 6. Parameters of the SVM are selected using k-fold cross-validation based on the training set. The performance of the SVM is carried out on the validation set.

SVMs with highest scores from cross-validation as well as the training sample set are depicted in Fig. 7. The scores for the grid used in the selection of the parameters (C, γ) and the corresponding optima are shown in Fig. 8. It is

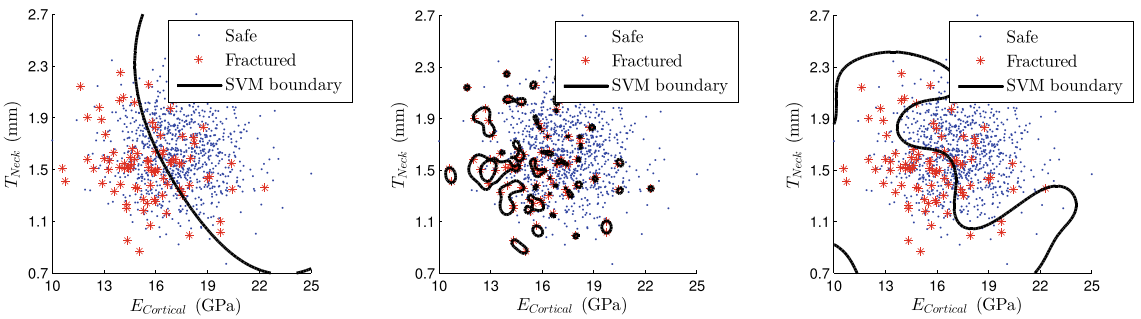
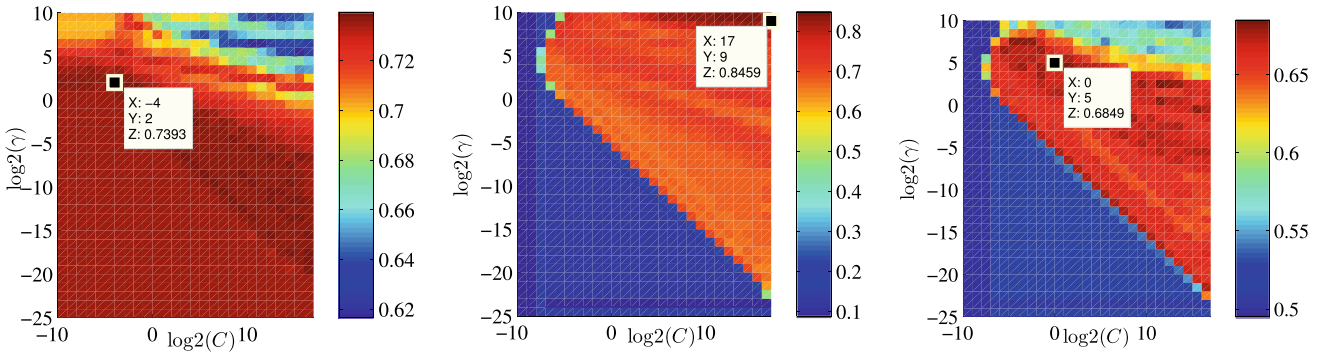


Fig. 7 Training samples and SVMs selected based on different validation metrics



(a) Score of AUC from grid search **(b)** Score of accuracy from grid search **(c)** Score of balanced accuracy from grid search

Fig. 8 Maps of validation metrics on a grid along with the corresponding maxima

Table 3 Validation metrics (scores) and 95% confidence intervals

SVM selected	Score [95% CI]				Difference from
using:	*AUC*	*Acc*	*Bacc*	$\overline{\mathcal{L}}_w$ [95% CI]	Ref. $\overline{\mathcal{L}}_w$ *
AUC	0.73 [0.68 0.78]	0.68 [0.65 0.71]	0.65 [0.60 0.71]	-3.02 [-4.03 -2.06]	28.78%
Acc	0.58 [0.51 0.64]	0.85 [0.83 0.87]	0.55 [0.51 0.60]	-10.64 [-13.30 -8.16]	353.40%
Bacc	0.67 [0.61 0.73]	0.70 [0.67 0.73]	0.63 [0.57 0.69]	-3.90 [-5.11 -2.77]	66.08%

Reference $\overline{\mathcal{L}}_w$=-2.35 [-3.02, -1.78]

noteworthy that the SVM selected using the basic accuracy metric leads to "over-fitting" and will have a poor predictive capability.

Table 3 provides the results for the weighted likelihood which is compared to the reference value described in Section 3. Figure 9 depicts the convergence of the weighted likelihood as a function of the number of training samples which is used to obtain the reference likelihood value.

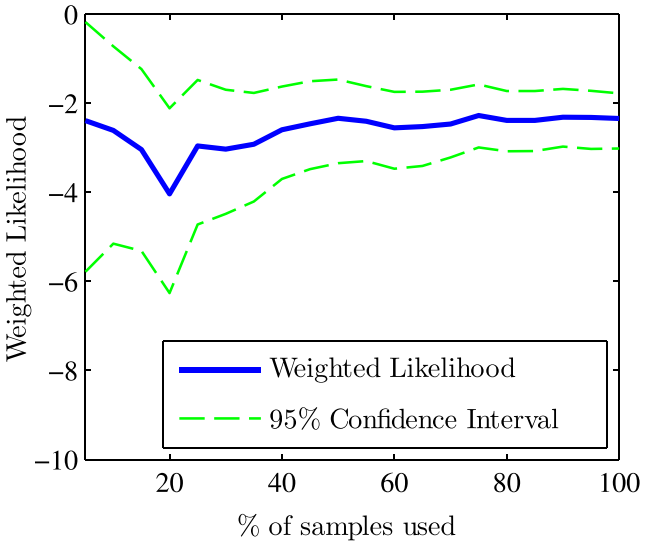


Fig. 9 Evolution of the weighted likelihood $\overline{\mathcal{L}}_w$ and its 95% confidence interval as a function of the number of training samples. The reference $\overline{\mathcal{L}}_w$ is the value at convergence

From this point on, this article will not consider accuracy (*Acc*) metric since it is not suitable (and this is a well-known problem) in the case of unbalanced data.

4.4 Influence of separability, unbalance, and training sample size

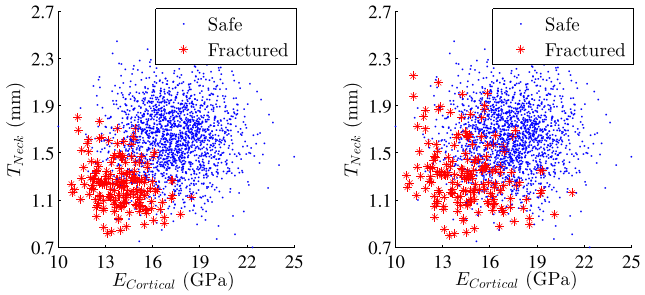
This section extends the previous study to different levels of separability, unbalance, and sizes of training sample sets.

4.4.1 Level of separability

In order to create configurations with different levels of separability, the spread of fractured samples in sub-space of E_{Cortical} and T_{Neck} was modified. For this purpose, an isoprobabilist transformation between the original normal distributions and new Weibull distributions was used:

$$F(\mathbf{x}_i|a,b) = \Phi\left(\frac{\mathbf{x}_i - \mu_i}{\sigma_i}\right), i = 1,2 \tag{12}$$

where $F(\mathbf{x}_i|a,b)$ is the cumulative distribution function of Weibull distribution with parameter a and b. Φ is the



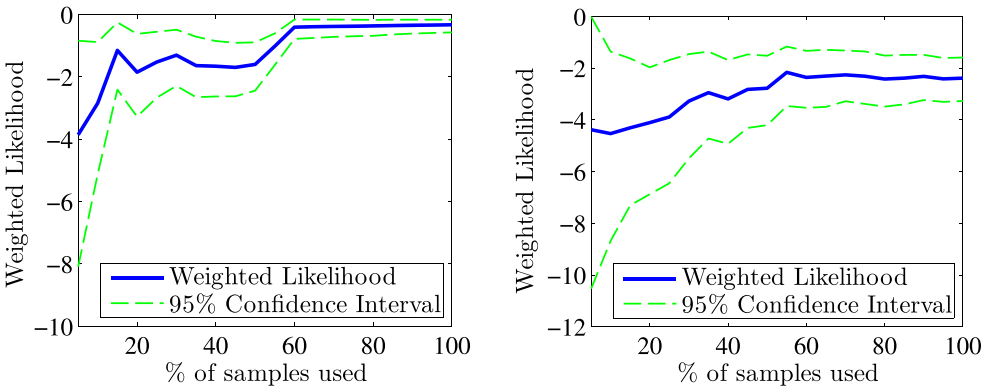
(a) Configuration 1 **(b)** Configuration 2

Fig. 10 Different sample configurations with two levels of separability

cumulative distribution function of the standard normal distribution. μ_i and σ_i are the empirical mean and standard deviation of variables E_{Cortical} and T_{Neck} separately. The parameters of the Weibull distribution were modified to control the spread.

In addition to the original distribution (Fig. 5), two new levels of separability are introduced (Fig. 10 a and b). These three levels of separability will be referred to as Configuration 1,2 and 3. Configuration 3 refers to the original case without transformation.

Figure 11 shows the evolution of weighted likelihood as well as its 95% confidence interval for both sample configurations 1 and 2. The evolution is depicted as a



(a) Configuration 1. **(b)** Configuration 2.

Fig. 11 Evolution of $\overline{\mathcal{L}}_w$ and its 95% confidence interval for the two separability levels as a function of number of training samples

Table 4 Validation metrics and 95% confidence intervals for level of separability "Configuration 1"

SVM selected	Score [95% CI]				Difference from
using:	*AUC*	*Acc*	*Bacc*	$\overline{\mathcal{L}}_w$ [95% CI]	Ref. $\overline{\mathcal{L}}_w$ *
AUC	0.97 [0.96 0.98]	0.89 [0.87 0.91]	0.93 [0.91 0.95]	-0.35 [-0.69 -0.16]	6.05%
Bacc	0.97 [0.96 0.98]	0.83 [0.81 0.85]	0.90 [0.88 0.92]	-0.72 [-1.50 -0.20]	118.67%

* Reference $\overline{\mathcal{L}}_w$=-0.33 [-0.57, -0.17]

Table 5 Validation metrics and 95% confidence intervals for level of separability "Configuration 2"

SVM selected	Score [95% CI]				Difference from
using:	*AUC*	*Acc*	*Bacc*	$\overline{\mathcal{L}}_w$ [95% CI]	Ref. $\overline{\mathcal{L}}_w$ *
AUC	0.92 [0.88 0.94]	0.82 [0.79 0.84]	0.83 [0.79 0.87]	-2.49 [-3.78 -1.31]	4.21%
Bacc	0.91 [0.88 0.94]	0.84 [0.82 0.86]	0.84 [0.80 0.88]	-2.92 [-4.51 -1.54]	22.33%

*Reference $\overline{\mathcal{L}}_w$=-2.39 [-3.26, -1.58]

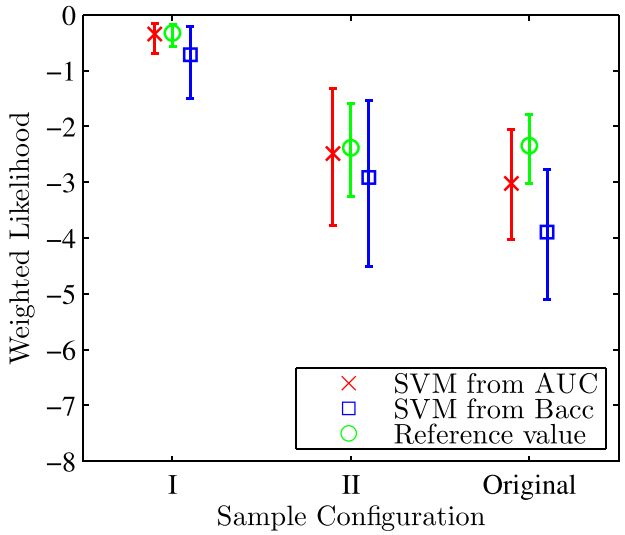


Fig. 12 Comparison of weighted likelihood between the three levels of separability

function of the number of training samples. Reference values of weighted likelihood for sample configuration 1 & 2 as well as their 95% confidence intervals are given in Tables 4 and 5.

Results on sample separability configuration 1 and 2 are listed in Table 4 and 5 separately.

Figure 12 shows that as the samples become more separable, weighted likelihood from the SVM selected based on AUC is closer to the reference value. In addition, its 95% confidence interval is smaller than the SVM selected based on balanced accuracy.

4.4.2 Level of unbalance

This section studies the change of weighted likelihood and its 95% confidence interval with different levels of unbalance by varying ratios between safe (+1) and failed (−1) classes (Fig. 13). Validation metrics (scores) and relative

Fig. 13 For separability level "Configuration 2", study of the influence of unbalance by changing the ratio between safe and failed samples

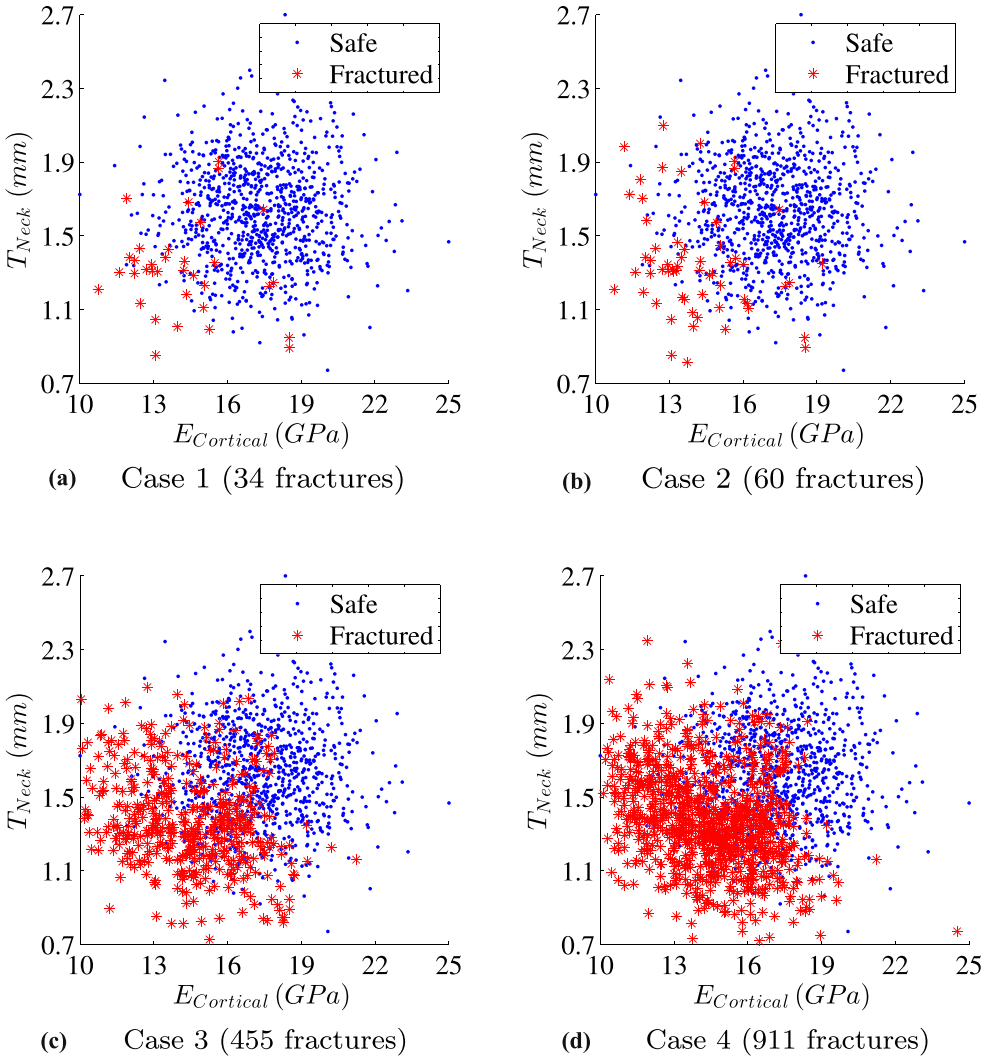


Table 6 Performances of SVMs selected using different validation metrics for four levels of unbalance

	SVM selected using:	Score [95% CI] *AUC*	*Bacc*	$\overline{\mathcal{L}}_w$	Difference from Ref. $\overline{\mathcal{L}}_w$ *
Case 1	*AUC*	0.90 [0.87 0.93]	0.84 [0.77 0.90]	-3.40 [-6.17 -1.14]	16.80%
	Bacc	0.90 [0.87 0.94]	0.80 [0.72 0.87]	-5.30 [-8.92 -2.30]	82.20%
Case 2	*AUC*	0.90 [0.86 0.94]	0.81 [0.75 0.86]	-3.95 [-6.08 -2.16]	55.33%
	Bacc	0.90 [0.86 0.94]	0.83 [0.78 0.88]	-4.06 [-6.46 -1.92]	59.76%
Case 3	*AUC*	0.90 [0.89 0.92]	0.83 [0.81 0.85]	-1.75 [-2.12 -1.37]	20.76%
	Bacc	0.90 [0.89 0.92]	0.83 [0.80 0.85]	-1.78 [-2.15 -1.45]	23.28%
Case 4	*AUC*	0.91 [0.90 0.92]	0.8 [0.82 0.85]	-1.36 [-1.58 -1.15]	10.31%
	Bacc	0.89 [0.87 0.91]	0.84 [0.82 0.86]	-1.38 [-1.60 -1.15]	11.52%

[1] Reference $\overline{\mathcal{L}}_w$ for Case 1 is -2.91 [-4.63, -1.39].
[2] Reference $\overline{\mathcal{L}}_w$ for Case 2 is -2.54 [-3.79, -1.50].
[3] Reference $\overline{\mathcal{L}}_w$ for Case 3 is -1.45 [-1.67, -1.24].
[4] Reference $\overline{\mathcal{L}}_w$ for Case 4 is -1.23 [-1.36, -1.12]

differences of weighted likelihood to the reference value are listed in Table 6.

Figure 14 shows that as the ratio between safe and failed classes grows larger, the 95% confidence interval of weighted likelihood becomes wider. The use of AUC provides better results than the balanced accuracy: the weighted likelihood is closer to the reference value and also is associated with a tighter confidence interval.

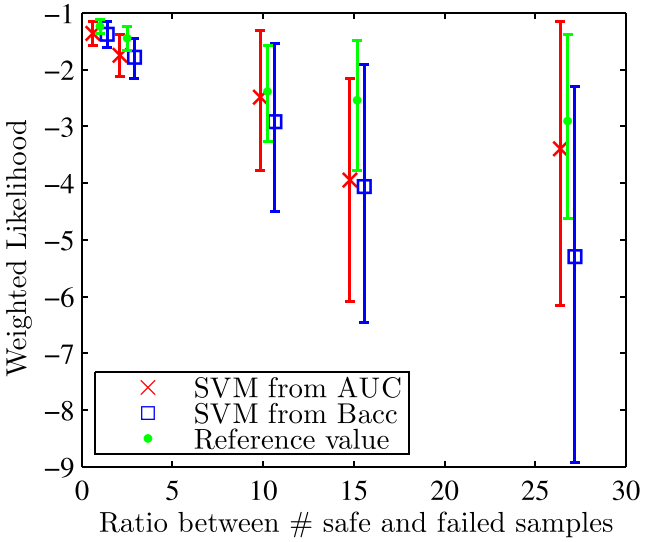


Fig. 14 Weighted likelihood and its 95% confidence interval for different levels of unbalance

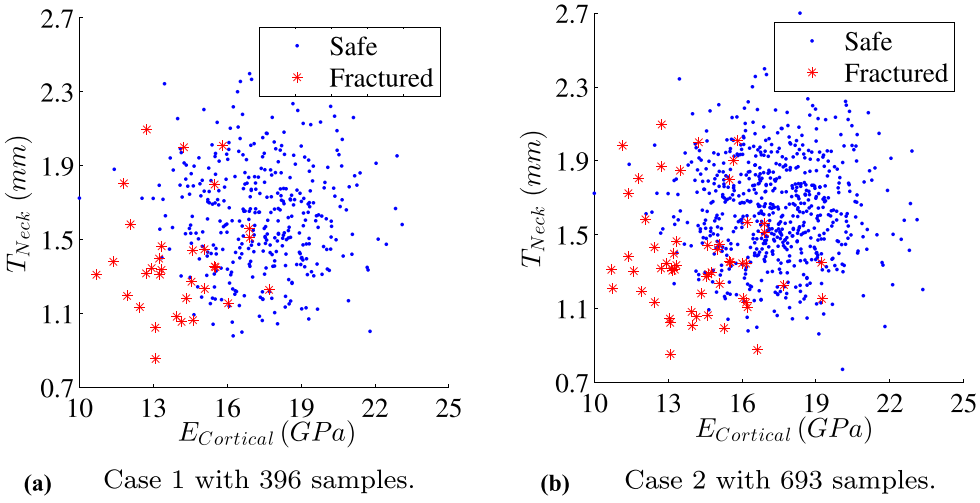
4.4.3 Number of samples

This section studies the influence of the number of samples. The ratio between failure and safe samples is kept constant. Three cases with different sizes of training samples are created as shown in Fig. 15. Case 1 uses 40% of training samples, Case 2 uses 70% of training samples and Case 3 contains all samples available in the training set as shown in Fig. 10b. The size of the test set is constant.

Results of weighted likelihood and relative difference from the reference value are provided in Table 7. As the size of training samples increases, SVMs selected based on AUC demonstrate again a better performance than the balanced accuracy (see Fig. 16).

5 Conclusion

This article compared three commonly used validation metrics for the selection of optimal SVM parameters in the case of non-separable and unbalanced data. A systematic study with different levels of separability and levels of unbalance as well as sizes of training samples, were presented. The datasets used were created from a finite element model for the prediction of hip fracture. The results show the advantage of the AUC metric, mostly for case with large degrees of unbalance and non-separability. The next steps of this study will involve higher dimensional problems along with the use of actual clinical data.



(a) Case 1 with 396 samples. **(b)** Case 2 with 693 samples.

Fig. 15 Different sizes of training samples used for parameter selection

Table 7 Performances of SVMs selected using different metrics on validation set for 3 cases with various sizes of training sets

	SVM selected using:	Score [95% CI]			Difference from Ref. $\overline{\mathcal{L}}_w$ *
		AUC	*Bacc*	$\overline{\mathcal{L}}_w$	
Case 1	*AUC*	0.91 [0.88 0.93]	0.83 [0.78 0.87]	-3.55 [-5.30 -2.02]	48.89%
	Bacc	0.84 [0.78 0.89]	0.75 [0.70 0.80]	-3.71 [-5.68 -2.00]	55.60%
Case 2	*AUC*	0.91 [0.88 0.94]	0.83 [0.79 0.87]	-3.26 [-4.91 -1.83]	36.75%
	Bacc	0.85 [0.80 0.90]	0.76 [0.71 0.81]	-3.51 [-5.37 -1.93]	47.35%
Case 3	*AUC*	0.91 [0.88 0.94]	0.84 [0.80 0.88]	-2.49 [-3.86 -1.31]	4.35%
	Bacc	0.91 [0.88 0.94]	0.83 [0.79 0.87]	-2.86 [-4.37 -1.53]	20.06%

[1] Reference $\overline{\mathcal{L}}_w$=-2.39 [-3.26, -1.58]

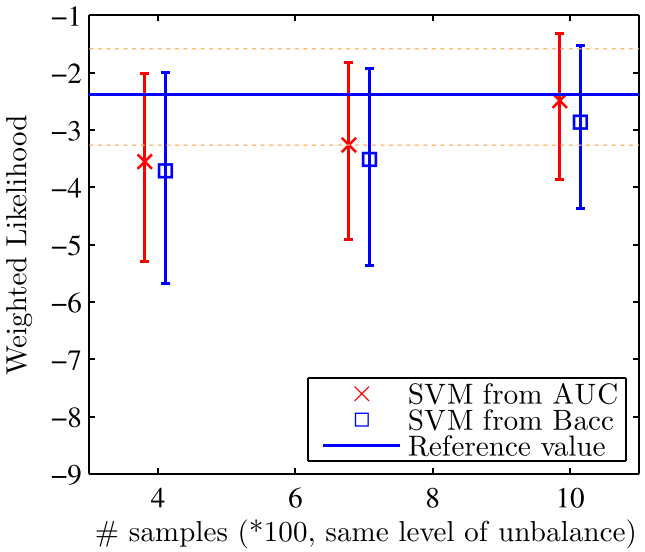


Fig. 16 Weighted likelihood and its 95% confidence interval for different sizes of training samples

Acknowledgments The support of the National Science Foundation (award CMMI-1029257) and National Institutes of Health (grant NIAMS 1R21AR060811) are gratefully acknowledged.

References

ANSYS (2011) ANSYS Parametric Design Language Guide, ANSYS Inc

Basudhar A, Missoum S (2010) An improved adaptive sampling scheme for the construction of explicit boundaries. Struct Multidiscip Optim 42(4):517–529

Basudhar A, Missoum S, Harrison Sanchez A (2008) Limit state function identification using Support Vector Machines for discontinuous responses and disjoint failure domains. Probabilistic Eng Mech 23(1):1–11

Bayraktar H, Morgan E, Niebur G, Morris G, Wong E, Keaveny T (2004) Comparison of the elastic and yield properties of human femoral trabecular and cortical bone tissue. J Biomech 37(1):27–35

Björkman M, Holmström K (1999) Global optimization using the direct algorithm in matlab

Brodersen K, Ong CS, Stephan K, Buhmann J (2010) The balanced accuracy and its posterior distribution. In: 2010 20th International Conference on Pattern Recognition (ICPR), pp 3121–3124, doi:10.1109/ICPR.2010.764

Burges CJ (1998) A tutorial on support vector machines for pattern recognition. Data Min Knowl Discov 2(2):121–167
Chang CC, Lin CJ (2011) LIBSVM: a library for support vector machines. ACM Trans Intell Syst Technol 2:1–27
Cristianini N, Shawe-Taylor J (2000) An introduction to support vector machines and other kernel-based learning methods. Cambridge university press
Doblaré M, García J (2003) On the modelling bone tissue fracture and healing of the bone tissue. Acta Científica Venezolana 54(1):58–75
Efron B, Tibshirani R (1997) Improvements on cross-validation: the 632+ bootstrap method. J Am Stat Assoc 92(438):548–560
Fawcett T (2006) An introduction to roc analysis. Pattern recogn Lett 27(8):861–874
Goldberg DE, Holland JH (1988) Genetic algorithms and machine learning. Mach Learn 3(2):95–99
Grassi L, Schileo E, Taddei F, Zani L, Juszczyk M, Cristofolini L, Viceconti M (2012) Accuracy of finite element predictions in sideways load configurations for the proximal human femur. J Biomech 45(2):394–399. doi:10.1016/j.jbiomech.2011.10.019
Jaakkola T, Haussler D et al (1999) Exploiting generative models in discriminative classifiers. Adv Neural Inf Proces Syst: 487–493
Kohavi R (1995) A study of cross-validation and bootstrap for accuracy estimation and model selection. International joint Conference on artificial intelligence, Lawrence Erlbaum Associates Ltd, vol 14, pp 1137-1145
Konig I, Malley J, Pajevic S, Weimar C, Diener H, Ziegler A (2005) Tutorial in biostatistics patient-centered prognosis using learning machines
Matthews BW (1975) Comparison of the predicted and observed secondary structure of t4 phage lysozyme. Biochimica et biophysica acta 405(2):442
McLachlan GJ, Do KA, Ambroise C (2004) Analyzing microarray gene expression data, vol 422. Wiley-Interscience
Metz CE (1978) Basic principles of ROC analysis. Seminars in nuclear medicine, Elsevier, vol8, pp 283-298
Murty MN, Devi VS (2011) Pattern recognition, An algorithmic approach. Springer
Opper M, Winther O (2000) Gaussian processes for classification: mean-field algorithms. Neural Comput 12(11):2655–2684
Osuna E, Freund R, Girosi F (1997) Support vector machines. Training and applications
Rijsbergen CJV (1979) Information Retrieval, 2nd edn. Butterworth-Heinemann, Newton
Tay FE, Cao L (2001) Application of support vector machines in financial time series forecasting. Omega 29(4):309–317
Vapnik V (1999) The nature of statistical learning theory. springer
Vapnik V, Chapelle O (2000) Bounds on error expectation for support vector machines. Neural Comput 12(9):2013–2036
Vapnik VN (1998) Statistical learning theory. Wiley
Varian H (2005) Bootstrap tutorial. Math J 9(4):768–775
Yang ZR (2010) Machine learning approaches to bioinformatics, vol 4. World Scientific Publishing Company Incorporated