HI-DID-IT, a HIgh-DImensional Data Integration Toolbox for clinical applications

Anneleen Daemen*[1,2], Olivier Gevaert[1,2], Fabian Ojeda[1], Johan A K Suykens[1]and Bart De Moor[1]

[1]Department of Electrical Engineering (ESAT-SCD), Katholieke Universiteit Leuven, Kasteelpark Arenberg 10, Leuven, Belgium
[2]The first two authors contributed equally to this article

Email: Anneleen Daemen - anneleen.daemen@esat.kuleuven.be;

*Corresponding author

Abstract

Background: Due to the popularity of microarray technology, classification models have mainly been developed on transcriptomics data. However, recent technological developments have introduced other omics types. Fusion of various types of genome-wide data in the patient domain has therefore become an inevitable requirement to improve prognosis, diagnosis and therapy response prediction.

Results: HI-DID-IT (HIgh-DImensional Data Integration Toolbox) is a prediction toolbox in Matlab through which multiple high-dimensional data sources can be integrated. The Least Squares Support Vector Machine is used for modeling classification and regression problems. Different integration schemes, feature selection strategies and the optimization of the weights assigned to the different data sets are implemented.

Conclusions: This toolbox can be used for model training, model testing as well as for comparison with other methodological approaches in benchmarking studies. Although HI-DID-IT was developed with clinical data in mind, our toolbox is not limited to this setting. HI-DID-IT can be applied to any problem for which multiple data sets for a set of observations are available. The toolbox is freely available under the GNU GPL licence at http://homes.esat.kuleuven.be/~bioiuser/HIDIDIT/index.php

Background

Since the cost of health care for age-related diseases such as cancer is significantly increasing due to the aging population, *clinical decision support (CDS)* systems are essential to improve cost-effectiveness of the overall health care system. The aim of decision support systems is in general to automate decisions based on domain knowledge and training data because problems become increasingly complex, humans are poorly equipped to deal with complexity, and they may have difficulties to objectively act under stress, time pressure and in high risk situations. Decision support improves reliability of decisions and reduces labor-intensive procedures by automatically classifying new data. Due to juridical issues, a trend is seen in medical applications towards *clinical-aided systems* to assist clinicians in their decision making. Such systems offer a set of alternatives or execute a selection if the clinician approves [1]. This contributes towards the trend of P4 medicine - predictive, personalized, preventive and participatory - with increasing empowerment of the patient, driven by new technologies and the systems approach to medicine.
A decade ago, CDS was defined as software designed to aid in clinical decision making, by matching characteristics of patients to a computerized knowledge base, to obtain patient-specific recommendations for clinicians. A systematic review has shown that computer-based systems mainly enhanced clinical performance for drug dosing and preventive care, while the results for diagnostics were not convincingly on a larger scale [2]. CDS systems targeting diagnostic assistance are therefore urgently needed. In the field of cancer, for example, the aim is to improve speed, accuracy and reliability of diagnostic and prognostic tools, pattern recognition, follow-up and screening compared to routine clinical management. Moreover, clinicians are overwhelmed by a *tsunami* of data for each single patient [3]. Clinical data on patient history, tumor characteristics, laboratory analysis, ultrasound parameters, or environmental factors are accessible but do not provide enough information to reliably predict patient prognoses. For a decade, microarray technology was used in research with as aim to improve decision support. Especially within the domain of cancer, it has earned a prominent place with several gene signatures currently being validated in clinical trials [4]. Besides transcriptomics data, information can also be unraveled from the genome, proteome and metabolome level. Improvements of these technologies add a new dimension to our ability to obtain a more comprehensive and holistic understanding of the complex biology of cancer. A key point in systems biology and a challenge for bioinformatics is therefore the integration of different levels of biological information and an efficient translation of cancer genome information into molecular markers and therapeutics for use in clinical practice. It is likely that different views on the genome contain distinct, partly independent and complementary information due to processes such as alternative splicing and post-translational

modifications. However, these large volumes of data combined with factors such as time, cost and personalized treatment are complicating clinical decision making. Although clinicians prefer interpretable CDS systems [5], the shift towards genome-wide data requires non-traditional computational approaches to be introduced in clinical practice.

We therefore investigated how multiple high-throughput data sets gathered from the same patients can be combined, in order to exploit the strength of each data set and to efficiently improve clinical decision support [6] [7] [8]. Our starting point was to fuse distinct experimental data sets into one mathematical model for the development of more homogeneous classifiers. Data integration occurred within the patient domain at another level as described so far in the literature. Instead of merging data sets or analyzing them in turn, we opted for an actual integration of the measurements in multiple data sets using an intermediate integration approach. Variables from different omics data sets were treated equally such that the most relevant features from each data set were selected and combined in a machine learning-based model. For the purpose of data integration, kernel methods are ideally suited. Kernel methods are a powerful class of methods for pattern analysis and have become a standard tool in data analysis, computational statistics, and machine learning applications [9] [10]. Data is embedded into a high-dimensional vector space in which the data entities are represented through a set of pairwise comparisons, called the kernel matrix K. The ability of kernel methods to deal with complexly structured data makes them ideally positioned for heterogeneous data integration. As kernel method for supervised classification, we made use of the Least Squares Support Vector Machine (LS-SVM), which is easy and fast to solve for high-dimensional data [11] [12]. It forms a linear discriminant boundary in the embedded space with maximum distance between data entities of the two considered classes. Depending on the applied kernel function, this corresponds to a linear or non-linear discriminant function in the original space.

We started with the integration of clinical and microarray data through which prognosis prediction for breast cancer patients could be improved [6]. Similar conclusions were drawn for lung cancer by others [13], with models including both microarray data and clinical variables outperforming in most cases those without clinical variables. However, due to the availability of much more high-throughput data sources, we extended our integration approach [7] [8]. For two case studies with transcriptomics and proteomics data for 36 rectal cancer patients and genomics and transcriptomics data for 50 patients with prostate cancer, the use of information from different levels in the central dogma of molecular biology improved classification performance and increased predictive power. Here, we present our framework that we expanded compared to the results presented in [8] from a methodological perspective to the training and

testing of models with clinical applicability. The framework is available as a software implementation for data integration in Matlab.

Results

Two-class and regression problems

While only classification problems were considered in our previous study [8], the presented toolbox has been extended to problems with a continuous outcome of interest (that is, regression problems). For breast cancer, for example, lymph node involvement is the most important prognostic factor. In most studies to date, a two-class problem was considered to distinguish patients with positive lymph nodes from patients with no affected lymph nodes. However, useful information is lost when binarizing a continuous variable. When the exact number of positive lymph nodes and examined nodes would be available, the lymph node ratio could be defined as the number of lymph nodes found to be positive divided by the total number of examined lymph nodes [14]. The lymph node ratio better reflects the severity of lymph node involvement and allows biologically important features to be detected more reliably. Specific to classification problems, data sets may be skewed in favor of one class such that the contribution of false negative and false positive errors to the optimization criterion is not balanced. For such cases, an adapted version of the LS-SVM is available in the toolbox in which a different weight is given to positive and negative samples in order to take the unbalance into account [15].

Integration schemes

Three ways exist to learn simultaneously from multiple data sets. They differ in the stage of the model building process at which integration occurs and are referred to as early, intermediate and late integration [16]. Figure 1 gives a global overview of these three methods in case of two available data sets. Early integration refers to integration at database level. Data sets are concatenated before a model is built, allowing dependencies between features from different data sets to be modeled. Early integration in combination with a non-linear kernel function is thus required when combinations of features from different data sets are expected to be meaningful. With intermediate integration, data sets are treated as separate entities and combination occurs at kernel level. Local features are first created by applying a kernel function to each data set separately, to extract relationships between variables within a data set while ignoring relationships between different types of variables. The corresponding kernel matrices are added or combined via a weighted sum (that is, $K = \sum_{m=1}^{M} \mu_m K_m$). Positive semidefiniteness of the linear

combination of kernel matrices is guaranteed by constraining the weights of the kernels to be non-negative. Intermediate integration corresponds to linearly combining the local features to create global features that are incorporated in the final model. The decision boundary is thus constructed in an explicit heterogeneous feature space. In the late integration approach, models are built on each of the data sets separately before combining their predictions with a decision function, possibly weighted with μ_m. This approach assumes complete independence between the data sets and requires the combination of multiple hypotheses to obtain a single prediction per patient. Although the intermediate integration approach was previously chosen in [8], all three integration options are available in the toolbox with the intermediate approach as default.

We also compared the three approaches on the data sets described in [8]. For prostate cancer, the outcomes grade, stage, metastasis and recurrence were predicted with the transcriptomics and genomics data set. For rectal cancer, transcriptomics and proteomics data were available at two time points during treatment and were combined at each time point separately for the prediction of tumor regression, lymph node stage and the circumferential resection margin (CRM). For both intermediate and late integration, weights μ_m were set equal while the regularization parameter γ was optimized with a grid search. The area under the ROC curve (AUC) values obtained with the different integration strategies for all combinations of number of genes and number of proteins or copy number variations, ranging from 5 to 35 in steps of 5, were compared using the one-sided paired-sampled t-test. Intermediate integration outperformed late integration for 8 out of 10 cases with p-values ranging from 2.76e-17 to 0.179 (median p-value = 1.18e-5), while for the other 2 cases the performance of late integration was not significantly better (p-values = 0.2192 and 0.3467). Intermediate integration was better than early integration in 6 cases (p-values 2.62e-30 to 3.82e-8, median = 1.29e-11). For three of the remaining cases (that is, CRM at the second time point, grade and recurrence), early integration performed significantly better. This may be explained by the discrepancy in predictive information contained in the individual data sets, which causes the majority of features to be selected from the most informative data set. For intermediate and late integration with equal weights assigned to both data sets, on the other hand, a less informative data set can deteriorate performance.

Weight optimization

The choice of the weights μ_m assigned to the kernel matrices or decision functions is important and depends on the specific application. In the default setting, the weights are set equal. However, as one data set may contain more information than other sets, the weights can be learned from the data or specified

beforehand. When the data sets or kernel matrices are normalized, the optimized weights reflect the relative importance of the different data sets. Optimization of the weights assigned to the individual data sets is crucial when dealing with many data sets of which only several are relevant. A redundant data set or a data set with much noise will receive a weight close to zero and its influence on the outcome is kept small. However, when the number of data sets is limited and most of them are reliable and relevant to the problem at hand, a trade-off needs to be made between performance and computational burden (for example, extra required cross-validation loops). Previous studies have shown that only in case of redundant or noisy data sets, optimization of weights leads to a better performance [17]. Wainer [18] formally proved that when all optimized weights are uniformly distributed on the interval [0.25; 0.75], use of equal weights barely changes performance. When the data sets (or kernel matrices) are standardized, this holds when no data set is completely redundant. This is often true for patient-related data, that is, all data sets are relevant with no data set containing sufficient information for perfect classification. In that case, equal weights can be opted for. An advantage of equal weights is robustness against outliers that are not representative for validation data and may lead to a reduction in accuracy. Moreover, optimizing weights reduces the degree of freedom available for selecting other parameters [18]. In the toolbox, weights can both be optimized or set equal. Optimization occurs with a grid search as explained in the section on tuning parameter optimization.

Feature selection

Not only the importance of the data sets with respect to each other can be determined, but selection of relevant features within each data set has also become a necessity due to the increase in large data sets. Dimensionality reduction reduces the number of parameters that needs to be estimated from the data, increases the reliability of the parameter estimates for a fixed sample size, and may alleviate the curse of dimensionality when the number of samples is limited. On the other hand, when many features are included, classifiers become more complex such that noise in the training set may be modeled. Overfitting with an overestimate of the generalization performance can partly be corrected by selecting a small feature subset [19]. Although dimensionality reduction introduces an additional layer of complexity, this is sound in biomedical applications due to the typical redundancy in high-dimensional data sets. Prediction accuracy may improve by excluding redundant and non-discriminatory features. Moreover, a limited number of genes and molecules are involved in cancer and knowledge about more relevant candidate features can improve insight into the underlying processes that generated the data. When a signature is validated, a potentially cheaper diagnostic test can also be developed due to the small number of

measurements that have to be done.

As feature extraction methods result in new variables that lack a clear physical interpretation, only feature selection methods were considered in our toolbox. The aim of feature selection is two-fold: obtaining interpretable features with biological relevance, and a robust classifier with high generalization power that is insensitive to outliers. In the domain of machine learning, a wealth of feature selection techniques has been developed. Choice of the feature selection technique, however, remains a widely discussed topic as no consensus technique exists with an optimal performance on all data sets [20]. We opted for univariate feature selection because in cancer research sample sizes are often limited and univariate feature selection methods have shown to lead to good and stable performances across many cancer types with in many cases consistently better results than multivariate approaches [21].

The presented toolbox allows building a model on a predefined number of features, selected with the Wilcoxon rank sum test for classification or the Spearman correlation coefficient for regression. To keep the sample to feature ratio sufficiently large, we suggest to include not more features than the number of samples on which the optimal number of features is determined [22]. Secondly, the number of features can be optimized according to a specified criterion. Due to the clinical advantage of models based on a small set of features, the number of features is subsequently calculated at which performance has dropped with a predefined percentage with respect to performance at the optimal number of features. When the aim of a study is to obtain a set of relevant predictive features, cross-validation should be applied to the data sets because the selected features may highly vary with the specific set of training samples. The features that are most frequently selected in the different iterations are likely to be more relevant candidate features. The feature selection method needs to be specified by the user as no feature selection is performed by default.

Kernel functions

Any symmetric, positive semidefinite function is a valid kernel function. If the kernel function, and thus the feature space, is chosen suitably, the embedded data are linearly separable and any learning algorithm can easily be applied in the embedded space. However, no formal proof of optimality exists for the use of one kernel function above an other. For each data set $\{x^i\}_{i=1}^N$ of N samples with feature vectors $x^i \in \mathbb{R}^p$, we previously used the linear kernel function ${x^i}^T x^j$ between samples i and j. Besides this kernel function, two other functions $k(x^i, x^j)$ that are most frequently employed in classification problems have been made available in the toolbox: the polynomial kernel $({x^i}^T x^j + \tau)^d$ with - as kernel parameters - the intercept constant $\tau \in \mathbb{R}^+$ and degree $d \in \mathbb{N}$, and the radial basis function (RBF) $\exp\big(-\|x^i - x^j\|_2^2/\sigma^2\big)$ with

$\sigma \in \mathbb{R}^+$ representing the width of a Gaussian distribution centered on the data points. $\| \|_2^2$ denotes the squared Euclidean norm. The polynomial kernel corresponds to a feature space spanned by all products of d variables at the most. This kernel results in a quadratic separating surface in the input space for $d = 2$, and it represents the cubic kernel for $d = 3$. When the RBF kernel function is used, the discriminant boundary corresponds to a sum of Gaussian distributions. Large σ values result in smooth Gaussians, and therefore a smooth or almost linear decision boundary. When σ is decreased, the Gaussians become more peaked around the data points, leading to more complex decision boundaries.

These kernel functions all correspond to a different transformation of the data, meaning that they extract a specific type of information from the data. The specific choice of the kernel function therefore depends on the distribution of the data entities. The normalized version of a kernel function equals $k(x^i, x^j)/\sqrt{k(x^i, x^i)k(x^j, x^j)}$. Normalization projects all data points onto the unit sphere in such a manner that the elements in the kernel matrix are bounded. In the toolbox, not only the linear, polynomial and RBF kernel are provided but also their normalized version. Normalizing is required when combining multiple data sets to guarantee the same order of magnitude for the kernel matrices. The normalized linear kernel function is used by default.

Tuning parameter optimization

Using all data for both model building and testing results in an optimistically biased estimate and should be avoided [23]. Ideally, when the number of samples is sufficiently large, data can be divided into a training set for model building, a validation set for model selection and a test set on which the final model is tested. This limits the modeling of noise in the training data and reduces the influence of outliers on model building. In most situations, however, the number of samples is limited and no samples can be completely left untouched. In that case, some form of resampling is required for performing the statistical analyses. A distinction is made between holdout, v-fold cross-validation (CV) and leave-one-out (LOO). In the holdout method, data are split into a training and test set. Stratification to outcome ensures that the relative proportion of outcomes sampled in both training and test set is similar to the original proportion in the full data set. The option of stratification is left to the user. Models are built on the training set and their performance is assessed on the test set. Due to dependence of test performance on the choice of training and test samples and to obtain a better estimate of the prediction performance, the split of the data should be repeated a sufficient number of times, referred to as randomization. The optimal model is the one that in general performs best for all test sets. Due to the overlap between test sets in different

runs, v-fold CV is another possibility. The collection of N samples is split into v folds, possibly stratified to outcome. Each fold with N/v samples is left out once and used as test set. Classifier performance is assessed as the overall performance obtained on all v folds. The leave-one-out method is a special instance of v-fold CV with v equal to the number of samples. With LOO, one sample is left out for testing in each iteration and the procedure of training and testing is repeated N times. The toolbox allows using all data at once for the optimization of the parameters, or to use a certain type of resampling to reduce the risk of overfitting. The specific choice of the estimation method depends on the sample size, with LOO for small data sets and v-fold CV or randomized splits of data into a training and test part for larger data sets. The default setting is 5-fold (for a sample size < 50) or 10-fold CV with stratification of the samples to outcome. When randomization without further specifications is selected by the user, 10 stratified randomizations with 70% of the data used for training are performed. A set of independent observations can also be provided on which the trained model is validated.

The parameters to be optimized are the number of features, the weights μ_m assigned to the different data sets, the regularization parameter γ of the LS-SVM and the parameters of each non-linear kernel function (that is, intercept constant τ and degree d for the polynomial kernel, width σ for the RBF kernel). A good generalization performance can be achieved by carefully tuning the algorithm and its parameters [24]. Within each performance loop, an internal LOO or 5-fold CV strategy is therefore applied to the training data for the optimization of all parameters using an exhaustive grid search. With r the number of parameters to be optimized, an r-dimensional grid of different values for those parameters is defined. For the selection of γ, 40 default values are considered, ranging from 10^{-4} to 10^{6} on a logarithmic scale. In case of a non-linear kernel function, the grid is expanded with the additional tuning parameters τ, d and σ. We made the kernel parameters τ and d of the polynomial kernel vary on a linear scale, with 10 default values for τ and d varying from 1 to 5. For the RBF kernel, width σ causes a trade-off between data fit and complexity of the decision surface, with larger values implying a smoother decision surface. By scaling the width of the RBF kernel with the square root of the number of variables, less nonlinearities are incorporated into the model in case of high-dimensional data, contributing to a reduced risk of overfitting. We therefore take into account the dimensionality of the data for the range of σ, with 10 default values defined on an exponentially growing scale as suggested by Hsu and colleagues [24]. In each iteration of the internal LOO or 5-fold CV, feature selection is performed where necessary and models are built for all possible combinations of parameters on the defined r-dimensional grid. The optimal set of parameters is chosen according to the selected optimization criterion. Contrary to [8], the parameters γ, τ and σ are

fine-tuned in a second phase around their optimal values with the other parameters fixed to their optimal value. For this fine-tuned grid, the same number of parameter values are considered as for the first grid. In case independent test data are available, the LS-SVM classifier is reconstructed on the complete training set with use of the optimal parameters before testing.

Optimization criteria

As the receiver operating characteristics (ROC) curve summarizes the performance of a classifier by showing the true positive rate (sensitivity) versus false positive rate (1-specificity) when varying the discrimination threshold, the area under the ROC curve is an evaluation measure that is independent of the operating point. We therefore considered the AUC in [8] as criterion for parameter optimization, in order to show the advantage of data integration over models built on one data set or a combination of less data sets. In addition, when disease-specific requirements such as high sensitivity or high specificity are known in a study, it is better to optimize the model parameters with respect to these requirements. With our toolbox, the parameters can be optimized according to different criteria, with the choice of the criterion depending on the problem at hand. For classification problems, the AUC is most often maximized, although it is also possible to minimize the balanced error rate or the classification error, or to maximize sensitivity or sum of sensitivity and specificity. For regression problems, the parameters are chosen corresponding to the model with the lowest mean squared error.

Conclusions

A single layer of molecular information, also called an omics layer, may not be sufficient to explain the complete network of molecules underlying a disease. As we previously showed that the use of an integration framework on data sources from different levels in the central dogma can improve decision support in cancer, we present here, in companion with our work in [8], a toolbox for the integration of experimental data from multiple such levels. Our tool can be used in three different ways. Models can be trained in a randomized setting, with v-fold cross-validation or by using leave-one-out when the number of samples is limited, in order to obtain a classification or regression model with the most stable features. Secondly, the obtained LS-SVM models can be validated on independent data. This application allows models to be thoroughly tested before being used in clinical trials. Finally, by using the training option for performance assessment, comparison with other methodologies in benchmarking studies is possible.

Availability and requirements

HI-DID-IT, a software implementation for data integration in Matlab, is freely available under the GNU GPL licence at http://homes.esat.kuleuven.be/~bioiuser/HIDIDIT/index.php

Abbreviations

AUC, area under the ROC curve; CDS, clinical decision support; CRM, circumferential margin involvement; CV, cross-validation; LOO, leave-one-out; LS-SVM, Least Squares Support Vector Machine; RBF, radial basis function; ROC, receiver operating characteristic.

Competing interests

The authors declare that they have no competing interests.

Authors' contributions

AD, OG, and FO implemented the toolbox. AD drafted the manuscript. OG, JS, and BDM helped drafting the manuscript. All authors read and approved the final manuscript.

Acknowledgements

AD is research assistant of the Fund for Scientific Research - Flanders (FWO-Vlaanderen). OG is also supported by FWO. BDM is a full professor and JS a professor at the Katholieke Universiteit Leuven, Belgium. This work is partially supported by: **1.** Research Council KUL: GOA AMBioRICS, CoE EF/05/007 SymBioSys, PROMETA, several PhD/postdoc & fellow grants. **2.** Flemish Government: **a.** FWO: PhD/postdoc grants, projects G.0241.04 (Functional Genomics), G.0499.04 (Statistics), G.0318.05 (subfunctionalization), G.0302.07 (SVM/Kernel), research communities (ICCoS, ANMMM, MLDM); **b.** IWT: PhD Grants, GBOU-McKnow-E (Knowledge management algorithms), GBOU-ANA (biosensors), TAD-BioScope-IT, Silicos; SBO-BioFrame, SBO-MoKa, TBM-Endometriosis. **3.** Belgian Federal Science Policy Office: IUAP P6/25 (BioMaGNet, Bioinformatics and Modeling: from Genomes to Networks, 2007-2011). **4.** EU-RTD: ERNSI: European Research Network on System Identification; FP6-NoE Biopattern; FP6-IP e-Tumours, FP6-MC-EST Bioptrain, FP6-STREP Strokemap.

References

1. Bates DW, Kuperman GJ, Wang S, Gandhi T, Kittler A, Volk L, Spurr C, Khorasani R, Tanasijevic M, Middleton B: **Ten commandments for effective clinical decision support: making the practice of evidence-based medicine a reality.** *J Am Med Inform Assoc* 2003, **10**:523–530.
2. Hunt DL, Haynes RB, Hanna SE, Smith K: **Effects of computer-based clinical decision support systems on physician performance and patient outcomes.** *J Am Med Assoc* 1998, **280**:1339–1346.
3. Mathew JP, Taylor BS, Bader GD, Pyarajan S, Antoniotti M, Chinnaiyan AM, Sander C, Burakoff SJ, Mishra B: **From bytes to bedside: data integration and computational biology for translational cancer research.** *PLoS Comput Biol* 2007, **3**:e12.
4. Cardoso F, van't Veer L, Rutgers E, Loi S, Mook S, Piccart-Gebhart MJ: **Clinical application of the 70-gene profile: the MINDACT trial.** *J Clin Oncol* 2008, **26**:729–735.
5. Wyatt JC, Altman DG: **Prognostic models: clinically useful or quickly forgotten?** *Br Med J* 1995, **311**:1539–1541.
6. Daemen A, Gevaert O, De Moor B: **Integration of clinical and microarray data with kernel methods.** In *Conf Proc IEEE Eng Med Biol Soc* 2007:5411–5415.
7. Daemen A, Gevaert O, De Bie T, Debucquoy A, Machiels JP, De Moor B, Haustermans K: **Integrating microarray and proteomics data to predict the response on cetuximab in patients with rectal cancer.** In *Pacific Symposium on Biocomputing* 2008:166–177.
8. Daemen A, Gevaert O, Ojeda F, Debucquoy A, Suykens JAK, Sempoux C, Machiels JP, Haustermans K, Moor BD: **A kernel-based integration of genome-wide data for clinical decision support.** *Genome Med* 2009, **1**:39.
9. Schölkopf B, Smola AJ: *Learning with kernels.* Cambridge, MA: MIT Press 2002.
10. Shawe-Taylor J, Cristianini N: *Kernel methods for pattern analysis.* Cambridge: Cambridge University Press 2004.
11. Suykens JAK, Vandewalle J: **Least Squares Support Vector Machine classifiers.** *Neural Processing Letters* 1999, **9**:293–300.
12. Suykens JAK, Van Gestel T, De Brabanter J, De Moor B, Vandewalle J: *Least Squares Support Vector Machines.* Singapore: World Scientific 2002.
13. Shedden K, Taylor JM, Enkemann SA, Tsao MS, Yeatman TJ, Gerald WL, Eschrich S, Jurisica I, Giordano TJ, Misek DE, Chang AC, Zhu CQ, Strumpf D, Hanash S, Shepherd FA, Ding K, Seymour L, Naoki K, Pennell N, Weir B, *et al*: **Gene expression-based survival prediction in lung adenocarcinoma: a multi-site, blinded validation study.** *Nat Med* 2008, **14**:822–827.
14. Vinh-Hung V, Verkooijen HM, Fioretta G, Neyroud-Caspar I, Rapiti E, Vlastos G, Deglise C, Usel M, Lutz JM, Bouchardy C: **Lymph node ratio as an alternative to pN staging in node-positive breast cancer.** *J Clin Oncol* 2009, **27**:1062–1068.
15. Cawley GC: **Leave-one-out cross-validation based model selection criteria for weighted LS-SVMs.** In *Proceedings of the International Joint Conference on Neural Networks (IJCNN)* 2006:1661–1668.
16. Pavlidis P, Weston J, Cai J, Grundy WN: **Learning gene functional classifications from multiple data types.** *J Comput Biol* 2002, **9**:401–411.
17. Lanckriet GRG, De Bie T, Cristianini N, Jordan MI, Noble WS: **A statistical framework for genomic data fusion.** *Bioinformatics* 2004, **20**:2626–2635.
18. Wainer H: **Estimating coefficients in linear models: it don't make no nevermind.** *Psychological Bulletin* 1976, **83**:213–217.
19. Sima C, Dougherty ER: **The peaking phenomenon in the presence of feature selection.** *Pattern Recognit Lett* 2008, **29**:1667–1674.
20. Saeys Y, Inza I, Larrañaga P: **A review of feature selection techniques in bioinformatics.** *Bioinformatics* 2007, **23**:2507–2517.
21. Lai C, Reinders MJT, van't Veer LJ, Wessels LFA: **A comparison of univariate and multivariate gene selection techniques for classification of cancer datasets.** *Bioinformatics* 2006, **7**:235–244.

22. Li W, Yang Y: **How many genes are needed for a discriminant microarray data analysis.** In *Methods of Microarray Data Analysis.* Edited by Lin SM, Johnson KF, Kluwer Academic 2002:137–150.

23. Jain AK, Duin RPW, Mao J: **Statistical pattern recognition: a review.** *IEEE Trans Pattern Anal Mach Intell* 2000, **22**:4–37.

24. Hsu CW, Chang CC, Lin CJ: **A practical guide to support vector classification. Posted at http://www.csie.ntu.edu.tw/~cjlin/papers/guide/guide.pdf.**

Figure legends

Figure 1 - Three integration schemes to learn from multiple data sets

The data sets are represented as matrices with rows corresponding to patients and columns corresponding to features. With early integration, an LS-SVM model is trained on the kernel matrix, computed from the concatenated data set. In intermediate integration, a kernel matrix is computed for each data set separately and an LS-SVM is trained on the sum of the kernel matrices. In late integration, LS-SVMs are trained separately for each data set, and a decision function results in a single outcome per patient.

Additional file legends

Additional file 1

HIDIDIT.zip: a software implementation for data integration in Matlab.