Computers in Biology and Medicine 107 (2019) 302–322

Contents lists available at ScienceDirect

Computers in Biology and Medicine

journal homepage: www.elsevier.com/locate/compbiomed

Clinical intelligence: New machine learning techniques for predicting clinical drug response

Turki Turki[a,*], Jason T.L. Wang[b,**]

[a] King Abdulaziz University, Department of Computer Science, Jeddah, 21589, Saudi Arabia
[b] New Jersey Institute of Technology, Bioinformatics Program and Department of Computer Science, Newark, NJ, 07102-1982, USA

ARTICLE INFO

Keywords:
Transfer learning
Machine learning
Drug discovery
Drug sensitivity
Applications in biology and medicine

ABSTRACT

Predicting the response, or sensitivity, of a clinical drug to a specific cancer type is an important research problem. By predicting the clinical drug response correctly, clinicians are able to understand patient-to-patient differences in drug sensitivity outcomes, which in turn results in lesser time spent and lower cost associated with identifying effective drug candidates. Although technological advances in high-throughput drug screening in cells led to the generation of a substantial amount of relevant data, the analysis of such data would be a challenging task. There is a critical need for advanced machine learning (ML) algorithms to generate accurate predictions of clinical drug response. A major goal of this work is to provide advanced ML tools to data analysts, who would in turn build prediction calculators to be incorporated into intelligent clinical decision support systems. Such innovative tools could be used to enhance patient-care, among other uses. To achieve this goal, we develop new ML techniques, including a transfer learning approach coupled with or without a boosting technique. Experimental results on real clinical data pertaining to breast cancer, multiple myeloma, and triple-negative cancer patients demonstrate the effectiveness and superiority of the proposed approaches compared to baseline approaches, including existing transfer learning methods.

1. Introduction

Machine learning (ML) techniques have been successfully applied to solve many real biological problems [1–8]. Specifically, ML techniques are pertinent to improving the prediction performance or making accurate predictions for a given task [9]. ML methods provide promising solutions in artificial intelligence when applied to clinical informatics, and could improve the cancer drug discovery process in the coming years. Fig. 1 reports the increasing number of ML publications in the field of clinical informatics since 2014, to demonstrate the avid interest in finding solutions that incorporate ML methods.

The primary goal of cancer research is to discover the most effective treatment for each cancer patient, where each patient responds differently to a specific treatment due to (1) external factors, such as use of tobacco products and unhealthy diet; and (2) internal factors, such as cancer cell heterogeneity and immune conditions. As the number of cancer patients worldwide increases every year, correctly predicting the sensitivity (i.e., responding) or resistance (non-responding) of a cancer to a specific drug (also called predicting the clinical drug response) will be of significant interest to clinicians and care-givers [10].

Data-driven approaches employing machine learning for cancer drug sensitivity prediction are principally categorized into three groups: the supervised approach, transfer learning approach, and ranking approach. Several supervised approaches to cancer drug sensitivity prediction have been developed. Supervised approaches perform drug sensitivity prediction via the use of labeled clinical data. For example, Turki et al. [10] proposed a supervised approach modelling the drug sensitivity prediction as a link prediction problem. First, the proposed approach takes the data of cancer patients as the input expression. Second, a feature learning technique is applied to the expression data to generate new feature representation of the expression data. Third, feature selection and instance selection are performed via statistical leverage scores and active learning to generate data with fewer features and fewer examples. Finally, a machine learning algorithm is applied to the reduced data to generate drug sensitivity predictions. Geeleher et al. [11] proposed a supervised approach that takes microarray data from cancer cell lines as input training data and microarray data from cancer clinical trials as testing data. The training data and testing data are then processed and homogenized. A machine learning algorithm is applied to the training data, to obtain a model.

* Corresponding author.
** Corresponding author.
E-mail addresses: tturki@kau.edu.sa (T. Turki), wangj@njit.edu (J.T.L. Wang).

https://doi.org/10.1016/j.compbiomed.2018.12.017
Received 13 July 2018; Received in revised form 29 December 2018; Accepted 31 December 2018

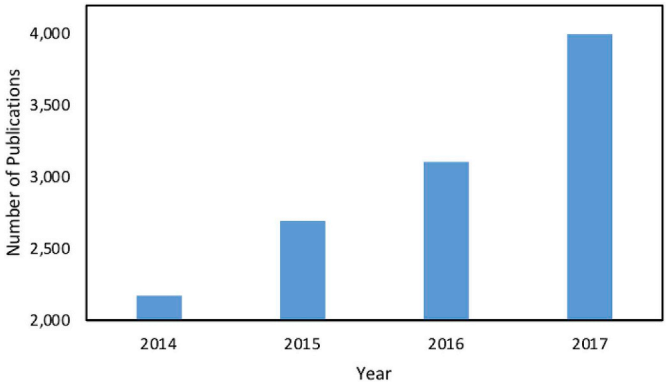

Fig. 1. The number of publications (from 2014 to 2017) based on the search keywords with the terms Drug Sensitivity or Clinical Informatics or Drug Discovery and the term Machine Learning. Publication statistics were acquired from Google Scholar.

The obtained model is applied to the testing data, to generate in-vivo drug sensitivity predictions. Majumder et al. [12] proposed CANScript, a supervised approach to perform drug sensitivity prediction. CANScript processes data related to colorectal cancer (CRC) as well as head and neck squamous cell carcinoma (HNSCC) tumors of 109 patients. Then, a machine learning algorithm is applied to the 109 patients' data, to obtain a model. Finally, the obtained model is applied to the testing data of 55 cancer patients, to generate drug sensitivity predictions. Basu et al. proposed a supervised approach employing a weighted version of elastic net [13] for cancer drug sensitivity prediction. Other supervised approaches have also been proposed for the task of cancer drug sensitivity prediction [14,15].

Transfer learning approaches design computational techniques employing auxiliary data from a related task B to improve the prediction performance or to make accurate predictions in a target task A [1,16–18]. Turki et al. [1] proposed transfer learning approaches to improve drug sensitivity prediction in multiple myeloma patients. These approaches were designed to employ auxiliary data of a related task, such as drug sensitivity prediction of breast cancer patients, to improve prediction accuracy in the task of predicting drug sensitivity of multiple myeloma patients. Several in-vitro datasets of related tasks have been utilized to build models and perform in-vivo drug sensitivity predictions of multiple myeloma patients. Recently Turki et al. [18] proposed another transfer learning approach for cancer drug sensitivity prediction using Procrustes analysis and mean shift. The goal of this approach was to change the representation of auxiliary data of a related task to a representation closer to data in the target task. Then, a machine learning algorithm is applied to the combination of auxiliary data of the new representation and target training data, to obtain a model used to generate predictions on the target test set.

The work described here differs from Refs. [1,18] in three ways. (1) We present two new transfer learning approaches, where the first approach includes a transfer mechanism that incorporates a boosting technique to improve the transferring mechanism by means of excluding some auxiliary data that is not relevant for the target task. The second approach employs only the transfer mechanism of the first approach, without the boosting technique. (2) We include additional performance testing results under the transfer learning and domain adaptation settings, using only in-vivo clinical data to train models and generate predictions. (3) We evaluate existing transfer learning algorithms [19,20], and compare the same with the proposed transfer learning approaches.

Ranking approaches employ a data-driven technique to identify the most effective drugs for a given cancer. Costello et al. [21] assessed the performance of many data-driven techniques for the task of ranking the most effective drugs for each breast cancer cell line. The training data consisted of 53 breast cancer cell lines, where each breast cancer cell line was associated with 28 drugs ordered in precedence of the most effective drug (ranked first) to the most ineffective (ranked last). The goal was to find a data-driven technique capable of correctly ranking drugs for each cell line in the test set of 18 breast cancer cell lines. All prediction algorithms were evaluated against the ground truth using a weighted probabilistic c-index (wpc-index) and Spearman correlations. The best performing technique employed a supervised algorithm utilizing a new feature representation coupled with a probabilistic non-linear regression. The second-best performing technique utilized random forest regression. The remaining prediction algorithms generate predictions that were not statistically significant.

Although supervised algorithms are designed for superior accuracy of their predictions, these algorithms require a sizeable set of negative and positive training examples, that in turn is associated with higher costs of cancer drug sensitivity screening. Ranking approaches face the similar challenge of high costs associated with cancer drug screening, that in this case is owing to each cancer cell line being screened against several drug compounds. Moreover, obtaining labeled clinical data of patients might introduce ethical issues. In contrast to supervised algorithms, transfer learning approaches can employ auxiliary data from related tasks to improve the prediction performance or to make accurate predictions in a target task of cancer drug sensitivity prediction. However, this process requires the development of computational techniques that adopt successful knowledge transfer mechanisms.

Contributions. The main contributions of this paper are as follows.

- We propose new transfer learning approaches for the clinical informatics domain, enabling state-of-the-art machine learning algorithms to achieve high performance results from several real clinical datasets pertaining to patients of multiple myeloma, triple-negative breast cancer, and breast cancer.
- The proposed approaches adopt modified versions of boosting and advanced transfer learning algorithms, that are the first to be applied in the clinical informatics domain [19,22].
- Unlike previous works, we evaluate the proposed approaches using several performance measures against baselines, including existing transfer learning algorithms, such as TrAdaBoost and CORAL-SVM [19,20], based solely on in-vivo data.
- We perform an empirical study to demonstrate the predictive accuracy of the proposed transfer learning approaches. Experimental results show the effectiveness and superior performance of the proposed approaches, when compared to the baseline approaches.

Organization. The rest of this paper is organized as follows. Section 2 reviews existing research related to this paper. Section 3 describes in detail the proposed transfer learning approaches. Section 4 presents experimental results, comparing the proposed approaches against the baselines. Section 5 provides a discussion of the results. Section 6 concludes the paper and suggests directions of future work.

2. Related work

There are two main bodies of research related to our work — correlation alignment for unsupervised domain adaptation (CORAL), particularly CORAL-SVM, and boosting for transfer learning (TrAdaBoost)

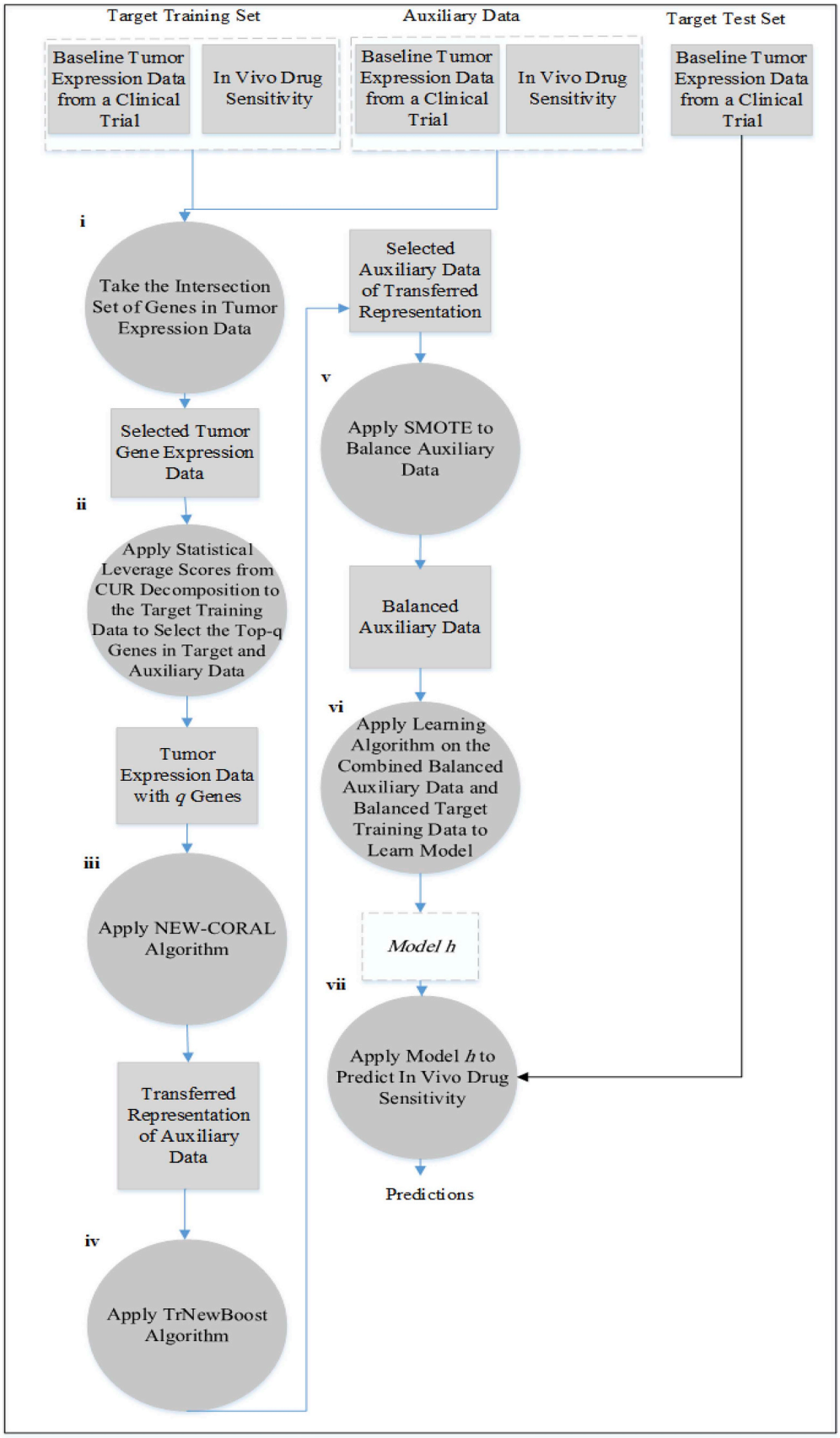

Fig. 2. The first transfer learning approach for predicting drug sensitivity using tumor expression data.

[19,20].

2.1. Correlation alignment for unsupervised domain adaptation (CORAL)

Sun et al. [20] proposed an unsupervised algorithm, called CORAL, for transferring knowledge from a source task to a target task. Algorithm 1 illustrates how the CORAL algorithm works. For a given auxiliary dataset D_{Src} consisting of m examples and target dataset D_{Tar} consisting of q examples, where the examples in the source and target tasks have the same number of features (line 1), the covariance of the source dataset, denoted $Cov(D_{Src})$, is calculated and added by the identity matrix of n columns (i.e., I_n) as shown in line 2. Similarly, the covariance of the target dataset, denoted $Cov(D_{Tar})$, is calculated and added by the identity matrix of n columns (i.e., I_n). Here, n in lines 2 and 3 refers to the number of features. In line 4, the auxiliary dataset D_{Src} is whitened. Then, the whitened auxiliary dataset is re-colored with the covariance of the target distribution (see line 5), to obtain the transformed auxiliary dataset $D^*_{Src} \in R^{m \times n}$ that has a closer representation to the unlabeled target dataset D_{Tar}.

Algorithm 1. CORAL for Unsupervised Domain Adaptation.

1: **CORAL**$(D_{Src} = \{x_{src_i}\}_{i=1}^{m}, D_{Tar} = \{x_{tar_i}\}_{i=1}^{q})$
2: $C_{Src} \leftarrow Cov(D_{Src}) + I_n$
3: $C_{Tar} \leftarrow Cov(D_{Tar}) + I_n$
4: $D_{Src} \leftarrow D_{Src} * C_{Src}^{-\frac{1}{2}}$
5: $D^*_{Src} \leftarrow D_{Src} * C_{Tar}^{\frac{1}{2}}$

2.2. Correlation alignment coupled with SVM (CORAL-SVM)

Algorithm 2 shows how to apply CORAL to support vector machines (SVM). In line 1, CORAL-SVM receives the auxiliary dataset D_{SrcX}, labels of the auxiliary data D_{SrcY}, and the unlabeled target test dataset $D_{Ta\ rX'}$. Lines 2–5 are the same as lines 2–5 in Algorithm 1, with the difference that we use the unlabeled target test dataset instead of the unlabeled target dataset. Line 6 aims to multiply the original auxiliary dataset by the new transformed auxiliary dataset and the transpose of the original auxiliary dataset. Line 7 stores the labels of the auxiliary data into Y. Line 8 changes the representation of the target test data, to get transformed target test data. In line 9, the new auxiliary dataset X along with the labels in Y are provided as input to support vector machines with a linear kernel, to obtain model h. Then, model h is applied to the target test data, to generate predictions stored in P.

Algorithm 2. Correlation Alignment Coupled with SVM (CORAL-SVM).

1: **CORAL-SVM**$(D_{SrcX}, D_{SrcY}, D_{TarX'})$
2: $C_{SrcX} \leftarrow Cov(D_{SrcX}) + I_n$
3: $C_{TarX'} \leftarrow Cov(D_{TarX}) + I_n$
4: $D_{SrcX} \leftarrow D_{SrcX} * C_{SrcX}^{-\frac{1}{2}}$
5: $D^*_{SrcX} \leftarrow D_{SrcX} * C_{TarX'}^{\frac{1}{2}}$
6: $X \leftarrow D_{SrcX} D^*_{SrcX} D^T_{SrcX}$
7: $Y \leftarrow D_{SrcY}$
8: $X_{test} \leftarrow D_{SrcX} D^*_{SrcX} D^T_{TarX'}$
9: $h \leftarrow$ SVM-LEARN(X,Y)
10: $P \leftarrow h(X^T_{test})$

2.3. Transfer AdaBoost (TrAdaBoost)

Dai et al. [19] proposed TrAdaBoost, which is a modified version of AdaBoost adapted for the transfer learning scenario. Algorithm 3 outlines the working mechanism of TrAdaBoost. In line 1, the TrAdaBoost algorithm receives auxiliary and target data in D consisting of q examples that are in the same feature space. In lines 2–3, the auxiliary and target data are assigned equal weights. The 'for' loop in lines 5–14 runs for t times utilizing the auxiliary data that are close to the target data as follows. In line 6, all weighted examples are normalized, to create a probability distribution stored in P_t where t denotes the tth iteration. Then at line 7, D and P_t are provided as inputs to a machine learning algorithm, to obtain model h_t. In lines 8–9, the error of the target training examples is calculated. In lines 10–11, the target weight β_t in the tth iteration and the source weight β are calculated. In lines 12–13, the weight vector is updated as follows. For the auxiliary data $\{(x_{src_i}, y_{src_i})\}_{i=1}^{m}$, if the classifier h_t incorrectly predicts the ith source example x_{src_i}, then this indicates that the source example would be different from the target data. Therefore, its weight is decreased in the next iteration (i.e., the $t+1$ iteration). On the other hand, if the classifier h_t correctly classifies the ith source example, then it means that the ith source example would have a closer representation to the target data. Hence, we increase its weight. For the target data $\{(x_{tar_i}, y_{tar_i})\}_{i=m+1}^{q}$, the weight updating mechanism works like AdaBoost. Specifically, if the classifier h_t incorrectly predicts the ith target example $x_{t\ a\ r_i}$, then its weight is increased during the next iteration. Otherwise, its weight is decreased. This process is repeated for additional t-1 times. For a given target test example $x'_{t\ a\ r_i}$ a prediction is made based on the classifiers from iteration $t = \lceil \frac{L}{2} \rceil$ to iteration $t = L$. In this study $L = 5$. (see lines 15–16).

Algorithm 3. TrAdaBoost.

1: **TrAdaBoost**$(D = \{(x_{src_i}, y_{src_i})\}_{i=1}^{m} \cup \{(x_{tar_i}, y_{tar_i})\}_{i=m+1}^{q})$
2: **for** $i = 1$ **to** q **do**
3: $W_1(i) \leftarrow 1$
4: **end for**
5: **for** $t = 1$ **to** L **do**
6: $P_t \leftarrow \frac{W_t}{\sum_{i=1}^{q} W_t(i)}$
7: $h_t \leftarrow$ fit classifier $h_t \in H$ providing it D and weights P_t
8: Calculate the error of h_t on target training examples:
9: $\epsilon_t \leftarrow \sum_{i=m+1}^{q} \frac{W_{tar_t}(i) I[h_t(x_{tar_i}) \neq y_{tar_i}]}{\sum_{i=m+1}^{q} W_{tar_t}(i)}$
10: $\beta_t \leftarrow \frac{\epsilon_t}{(1-\epsilon_t)}$
11: $\beta \leftarrow \frac{1}{(1+\sqrt{\frac{2 \ln n}{N}})}$
12: Update the new weight vector:
13:
$$W_{t+1}(i) \leftarrow \begin{cases} W_t(i)\beta^{|h_t(x_{src_i}) - y_{src_i}|} & 1 \leq i \leq m \\ W_t(i)\beta_t^{-|h_t(x_{tar_i}) - y_{tar_i}|} & m+1 \leq i \leq q \end{cases}$$
14: **end for**
15: Prediction for a test example x'_{tar_i} is given by:
16:
$$H(x'_{tar_i}) \leftarrow \begin{cases} +1, & \prod_{t=\lceil \frac{L}{2} \rceil}^{L} \beta_t^{-h_t(x'_{tar_i})} \geq \prod_{t=\lceil \frac{L}{2} \rceil}^{L} \beta_t^{-\frac{1}{2}} \\ -1, & \text{otherwise} \end{cases}$$

3. Our proposed approaches

3.1. The first transfer learning approach (PT1)

Fig. 2 shows the flowchart of the first transfer learning approach, which takes the following input: target training set D_{Tar} consisting of tumor samples $x_{tar_i} \in R^l$ and the corresponding drug sensitivity labels $y_{tar_i} \in$ {"sensitive", "resistant"}, auxiliary data set D_{Src} consisting of tumor samples $x_{src_j} \in R^d$ and the corresponding drug sensitivity labels $y_{src_j} \in$ {"sensitive", "resistant"} of a related task, and a target test set D'_{Tar} consisting of unseen tumor samples. The first proposed approach, named the NEW-CORAL algorithm, consists of 7 steps, explained as follows. (i) We take the intersection of genes in the target training set and auxiliary data set, assuming n is the number of intersected genes between the target training set and auxiliary data set consisting of l and d genes, respectively. (ii) The dimensionality of the target training data is reduced by applying the statistical leverage scores concept from CUR matrix decomposition. Genes with high statistical leverage scores are significant, as explained [10,18]. Hence, the top-q genes in the target training set are selected based on the top-q statistical leverage scores. The q genes in the auxiliary data set are selected using the same q genes in the target training set. (iii) We provide the target training set and auxiliary data set of q genes as input to the NEW-CORAL algorithm. Lines 2–5 of the NEW-CORAL algorithm are the same as Lines 2–5 of the CORAL algorithm in Section 2.1, where we obtain a transformed representation of auxiliary data set D^*_{SrcX} closer to the target training data.

Algorithm 4. NEW-CORAL.

1: **NEW-CORAL**(D_{SrcX},D_{SrcY},D_{TarX},D_{TarY})
2: $C_{SrcX} \leftarrow Cov(D_{SrcX}) + I_n$
3: $C_{TarX} \leftarrow Cov(D_{TarX}) + I_n$
4: $D_{SrcX} \leftarrow D_{SrcX} * C_{SrcX}^{-\frac{1}{2}}$
5: $D^*_{SrcX} \leftarrow D_{SrcX} * C_{TarX}^{\frac{1}{2}}$
6: For each example $x_{src_i} \in D^*_{SrcX}$, its corresponding label $y_{src_i} \in D_{SrcY}$ equals

$$y^*_{src_i} \leftarrow y_{tar_{j*}} \text{ where } j^* \leftarrow \underset{j \in \{1,2,\ldots,|D_{TarX}|\}}{\operatorname{argmin}} ||x_{src_i} - x_{tar_j}||$$

7: **return**(D^*_{SrcX},Y^*_{Src})

For each auxiliary example $x_{src_i} \in D^*_{SrcX}$, we update its label by assigning the label $y_{tar_{j*}}$ of the closest target example $x_{tar_{j*}}$ to the auxiliary example x_{src_i}, where closeness is measured using Euclidean distance (see line 6). In line 7, we return the updated auxiliary data of new representation. (iv) To select auxiliary data of acceptable quality, we apply the TrNewBoost algorithm (see Algorithm 5), that works as follows. In line 1, TrNewBoost accepts as input the combination of target training data and auxiliary data of new representation. In lines 2–3, the target training data and auxiliary data are stored in and D_{Src}, respectively. Lines 4–6 assign equal weights to the examples in the target training set . In lines 7–15, we focus to a greater extent on the target training examples x_{tar_i} in in the $(t+1)$th iteration by increasing its weights, if these examples are predicted incorrectly in the tth iteration, while focusing to a lower degree to the target training examples in the $(t+1)$th iteration by decreasing its weights, if these examples are predicted correctly in the tth iteration. In particular, we invoke a machine learning algorithm by providing the algorithm with and corresponding weights W_{Tar}, to generate a classifier h_t in iteration t. Lines 9–10 perform predictions on using h_t and calculate the errors obtained by predicting the examples in using W_{Tar}.

Algorithm 5. TrNewBoost.

1: **TrNewBoost**($D = \{(x_{tar_i}, y_{tar_i})\}_{i=1}^{m} \cup \{(x_{src_i}, y_{src_i})\}_{i=m+1}^{q}$)
2: $D_{Tar} = \{(x_{tar_i}, y_{tar_i})\}_{i=1}^{m}$
3: $D_{Src} = \{(x_{src_i}, y_{src_i})\}_{i=m+1}^{q}$
4: **for** $i = 1$ **to** m **do**
5: $W_{Tar_1}(x_{tar_i}) \leftarrow \frac{1}{m}$
6: **end for**
7: **for** $t = 1$ **to** T **do**
8: $h_t \leftarrow$ fit classifier $h_t \in H$ on D_{Tar} with W_{Tar_t}
9: $E_t \leftarrow \{x_{tar_i} | h_t(x_{tar_i}) \neq y_{tar_i}\}$
10: $\varepsilon_t = \sum_{x_{tar_i} \in E_t} W_{Tar_t}(x_{tar_i})$
11: $\alpha_t \leftarrow \frac{1}{2} \log \frac{1-\epsilon_t}{\epsilon_t}$
12: **for** $i = 1$ **to** m **do**
13: $W_{Tar_{t+1}}(x_{tar_i}) \leftarrow \frac{W_{Tar_t}(x_{tar_i}) exp(-\alpha_t h_t(x_{tar_i}) y_{tar_i})}{Z_t}$
14: **end for**
15: **end for**
16: **for** $i = 1$ **to** $|D_{Src}|$ **do**
17: $F_{Src_i} \leftarrow \sum_{t=1}^{T} \alpha_t h_t(x_{src_i})$
18: $F^*_{Src_i} \leftarrow sign(F_{Src_i})$
19: **end for**
20: $D_{Src}{}^* \leftarrow \{(x_{src_i}, y_{src_i}) | F^*_{Src_i} = y_{src_i}\}$
21: **return**($D_{Src}{}^*$)

In line 11, the weight of classifier h_t in the tth iteration is calculated. (In this study, there are 11 iterations in total, i.e., $T = 11$.) For each auxiliary example x_{src_i} in D_{Src}, we generate predictions using the weighted majority vote of the classifiers h_t, $1 \leq t \leq T$ (see lines 16–19). If the auxiliary example x_{src_i} is correctly predicted, then we store x_{src_i} and its label y_{src_i} in D^*_{Src} as this auxiliary example is likely to be close to the examples in the target training set D_{Tar} (see line 20). If the auxiliary example x_{src_i} is incorrectly predicted, then we ignore it as this example is likely to be distant from the examples in the target training set D_{Tar}. Line 21 returns selected auxiliary examples of new representation stored in D^*_{Src}. (v) We balance the examples in D^*_{Src} by utilizing SMOTE [23], generating additional synthetic target examples for the auxiliary examples with the minority class label as follows:

$$x_{new} = x_{tar_i} + \left(x_r - x_{tar_i}\right) \lambda. \quad (1)$$

where x_r denotes the target example closest to x_{tar_i}, for r greater than 1 and less than the number of examples in D_{Tar} having the same class label of the minority class examples in D^*_{Src} and $\lambda \in$ (0.009,...,0.09). We continue generating additional minority examples of the auxiliary data using equation (1) until the minority class and majority class have the same number of examples. (vi) We balance the target training data by undersampling the majority class examples in the target training set, and then providing the balanced auxiliary data of new representation and the balanced target training data as input to a machine learning algorithm, to obtain model h. (vii) We select q genes from the target test set using the same q genes selected from the target training set. Then, we apply model h to the target test set of the q genes, to generate predictions.

3.2. The second transfer learning approach (PT2)

Fig. 3 shows the flowchart of the second proposed transfer learning approach (PT2), consisting of 6 steps. The first three steps i-iii in PT2 are the same as steps i-iii in PT1. The only difference between the second transfer learning approach (PT2) and the first transfer learning approach (PT1) is that PT2 does not utilize the TrNewBoost algorithm.

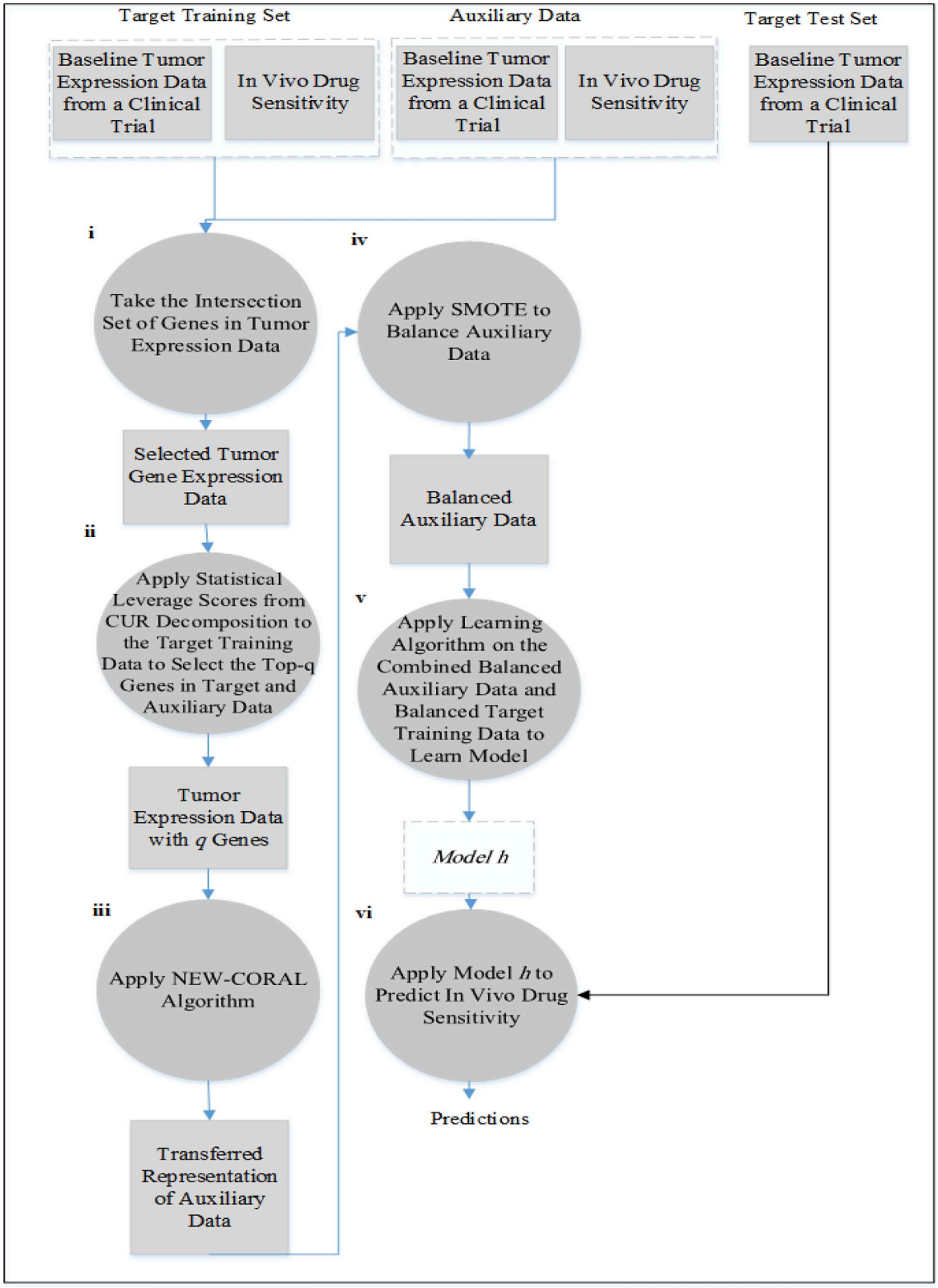

Fig. 3. The second transfer learning approach for predicting drug sensitivity using tumor expression data.

That is, after step iii, at step iv PT2 skips the TrNewBoost step in PT1, while balancing the auxiliary data of new representation obtained from the NEW-CORAL algorithm by utilizing SMOTE to generate additional synthetic examples for minority class examples of auxiliary data as explained in Section 3.1. (v) We balance the target training data via undersampling the majority class examples, and then providing the balanced target training data and balanced auxiliary data as input to a machine learning algorithm, to obtain model h. (vi) We select q genes from the target test set using the same q genes selected from the target training set. Then, model h is applied to the target test set of the q genes, to generate in vivo drug sensitivity predictions.

4. Experiments and results

In this section, we first describe the datasets used in this study. Then, we present our experimental methodology. Finally, we compare the proposed approaches against several baselines using different performance measures.

4.1. Datasets

4.1.1. In-vivo breast cancer data

This dataset consisted of 24 samples of breast cancer patients, 6538 genes (i.e., features), and a column vector of labels. The dataset was obtained using the Affymetrix Human Genome U95 Version 2 Array platform generated at Baylor College of Medicine (Houston, TX, USA, GSE6434). Out of the 24 breast cancer patients, 10 were “sensitive” to docetaxel, while the remaining patients were “resistant” to docetaxel [24–26]. That is, the drug docetaxel effectively reduced tumor size in 10 patients while docetaxel did not significantly reduce tumor size in the remaining 14 patients. The dataset is accessible from NCBI's Gene Expression Omnibus (GEO) [27] with the accession number GSE6434 and available at https://www.ncbi.nlm.nih.gov/geo/query/acc.cgi?acc=GSE6434 [28]. The data used in our study were downloaded from http://genemed.uchicago.edu/~pgeeleher/cgpPrediction/ [11].

4.1.2. In-vivo multiple myeloma data

This dataset consisted of 169 samples of multiple myeloma patients, 9114 genes, and a column vector of labels. The dataset was obtained using the Affymetrix Human Genome U133A Array platform generated at Millennium Pharmaceuticals (Cambridge, MA, USA, GSE9782). 85 out of the 169 multiple myeloma patients were designated as “Responder” to bortezomib [29–31]. That is, the drug bortezomib worked effectively in reducing tumor size in 85 patients. The remaining 84 multiple myeloma patients were designated as “Non-Responder” to bortezomib. In other words, the drug bortezomib was ineffective in reducing tumor size in these 84 patients. The dataset is accessible from NCBI's GEO with the accession number GSE9782 and available at https://www.ncbi.nlm.nih.gov/geo/query/acc.cgi?acc=GSE9782 [32]. The data used in our study were downloaded from http://genemed.uchicago.edu/~pgeeleher/cgpPrediction/.

4.1.3. In-vivo triple-negative breast cancer data

This dataset consisted of 24 samples of triple-negative breast cancer patients, 9620 genes, and a column vector of labels. The dataset was obtained using the Affymetrix GeneChip Human Genome U133 Plus 2.0 platform. 15 out of the 24 patients were categorized as “Responder” to cisplatin [30] while the remaining 9 patients were categorized as “Non-Responder” to cisplatin. In other words, the drug cisplatin effectively treated 15 triple-negative breast cancer patients while cisplatin was ineffective for the remaining 9 patients. The dataset is accessible from ArrayExpress [33–35] with the accession number E-GEOD-18864 and available at https://www.ebi.ac.uk/arrayexpress/experiments/E-GEOD-18864/ [33–35]. The data used in our study were downloaded from http://genemed.uchicago.edu/~pgeeleher/cgpPrediction/.

4.2. Experimental methodology

In this study, we used several state-of-the-art machine learning algorithms, including support vector machines (SVM) [36], xgboost [37], DeepBoost [38], and decision trees [39], where decision trees were employed in TrAdaBoost and TrNewBoost. We compared the proposed transfer learning approaches against three baseline approaches, described below.

1) First Baseline (BL1)

This baseline provides the combination of the auxiliary data and target training data as input to a machine learning algorithm, to obtain a model. Then, the obtained model is applied on the target test set, to generate predictions.

2) Second Baseline (BL2)

This baseline employs the CORAL-SVM algorithm, described in Section 2.2, to obtain a new representation of the auxiliary data closer to the target test set. Then, the auxiliary data of the new representation are provided as input to a machine learning algorithm, to obtain a model. Finally, the obtained model is applied on the target test set, to generate predictions.

3) Third Baseline (BL3)

This baseline utilizes TrAdaBoost, described in Section 2.3, and provides the target training set and auxiliary data closer to the target training data as input to a machine learning algorithm, to obtain a model. Then, the obtained model is applied on the target test set, to generate predictions. Table 1 summarizes the prediction algorithms used in our study.

We calculated several performance measures [40] based on the confusion matrix in Table 2 as follows:

Table 1
Summary of the prediction algorithms used in our study.

Abbreviation	Prediction Algorithm
PT1 + SVM	The first proposed transfer learning approach using support vector machines with a linear kernel
PT1 + XGB	The first proposed transfer learning approach using xgboost
PT1 + DB	The first proposed transfer learning approach using DeepBoost
PT2 + SVM	The second proposed transfer learning approach using support vector machines with a linear kernel
PT2 + XGB	The second proposed transfer learning approach using xgboost
PT2 + DB	The second proposed transfer learning approach using DeepBoost
BL1 + SVM	The first baseline approach using support vector machines with a linear kernel
BL1 + XGB	The first baseline approach using xgboost
BL1 + DB	The first baseline approach using DeepBoost
BL2	The second baseline approach using CORAL-SVM
BL3	The third baseline approach using TrAdaBoost

Table 2
Two-class confusion matrix.

	Predicted Positive	Predicted Negative
Actual Positive	TP (True Positive)	FN (False Negative)
Actual Negative	FP (False Positive)	TN (True Negative)

- Accuracy (ACC) $= \frac{TP + TN}{TP + TN + FP + FN}$.
- Area Under Curve (AUC) $= 0.5 \times (\text{Sensitivity} + \text{Specificity})$, where Sensitivity $= \frac{TP}{TP + FN}$ and Specificity $= \frac{TN}{TN + FP}$
- G− Mean $= \sqrt{\text{Precision} \times \text{Recall}}$ where Precision $= \frac{TP}{TP + FP}$ and Recall $= \frac{TP}{TP + FN}$

To evaluate the prediction algorithms, each dataset was randomly split into five folds, where different folds were assigned to the training and test sets. Specifically, for each run $j \in \{1,2,3,4,5\}$, fold j was assigned to the test set while the remaining folds were assigned to the training set. In each run, the above performance measures, including accuracy (ACC), area under curve (AUC) and G-Mean, were calculated and the performance results on each testing fold were recorded. Then, the mean values of ACC, AUC and G-Mean respectively over all five runs were calculated, denoted MACC, MAUC and MGM respectively, which represent the performance results obtained from 5-fold cross-validation. In addition, the standard deviation (SD) of all five testing folds was recorded. The software used in our work included SVM with a linear kernel [36], xgboost [41], DeepBoost [42], decision trees [39] and rCUR [43] where rCUR was employed in our proposed transfer learning approaches for selecting the top-q genes as described in Section 3.1. We performed all experiments using R.

4.3. Experimental results

In this section, we compare the proposed transfer learning approaches against the baselines, reporting 5-fold cross-validation results based on several clinical datasets related to different cancer types.

4.3.1. Predicting in-vivo drug sensitivity of breast cancer patients

4.3.1.1. Utilization of in-vivo triple-negative breast cancer data. Table 3 presents the performance results of the prediction algorithms studied in this work. As illustrated in Table 3, the machine learning algorithms employing the second proposed transfer learning approach (PT2) generate better performance results when compared to the three baselines. Specifically, the prediction algorithms PT2+XGB and PT2+DB achieve the highest MACCs of 0.710. When performance measures for imbalanced classification are considered, PT2+SVM outperforms all other prediction algorithms, yielding the highest MAUC value (i.e., 0.758) and the highest MGM value (i.e., 0.698). These results show the superior performance of the machine learning algorithms utilizing the PT2 approach.

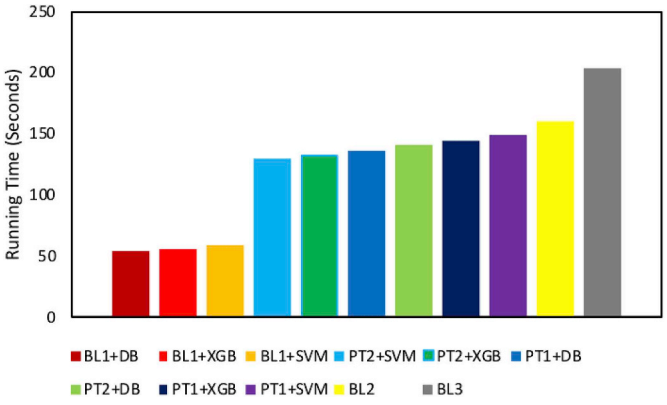

Fig. 4. Comparison of the total running times (in seconds) of the prediction algorithms on the target test data pertaining to in-vivo breast cancer patients. The auxiliary data are related to in-vivo triple-negative breast cancer patients. The prediction algorithms are ranked from left to right where the leftmost algorithm is fastest and the rightmost algorithm is slowest.

Table 3
Performance results of the studied prediction algorithms obtained from 5-fold cross-validation on clinical data pertaining to in-vivo breast cancer patients. The auxiliary data are related to in-vivo triple negative breast cancer patients. The highest MACC, MAUC, and MGM values are shown in bold. MACC denotes the mean accuracy. MAUC denotes the mean area under curve. MGM denotes the mean G-Mean value. SD is the standard deviation calculated based on the five runs of 5-fold cross-validation.

	MACC	SD	MAUC	SD	MGM	SD
PT1+SVM	0.700	0.200	0.708	0.224	0.598	0.368
PT1+XGB	0.590	0.124	0.458	0.144	0.230	0.316
PT1+DB	0.630	0.156	0.625	0.250	0.567	0.344
PT2+SVM	0.660	0.240	**0.758**	0.145	**0.698**	0.190
PT2+XGB	**0.710**	0.102	0.541	0.138	0.256	0.354
PT2+DB	**0.710**	0.102	0.683	0.205	0.593	0.349
BL1+SVM	0.700	0.200	0.725	0.231	0.620	0.379
BL1+XGB	0.630	0.156	0.616	0.184	0.404	0.386
BL1+DB	0.670	0.299	0.616	0.357	0.504	0.472
BL2	0.366	0.209	0.383	0.298	0.256	0.354
BL3	0.540	0.219	0.500	0.117	0.115	0.258

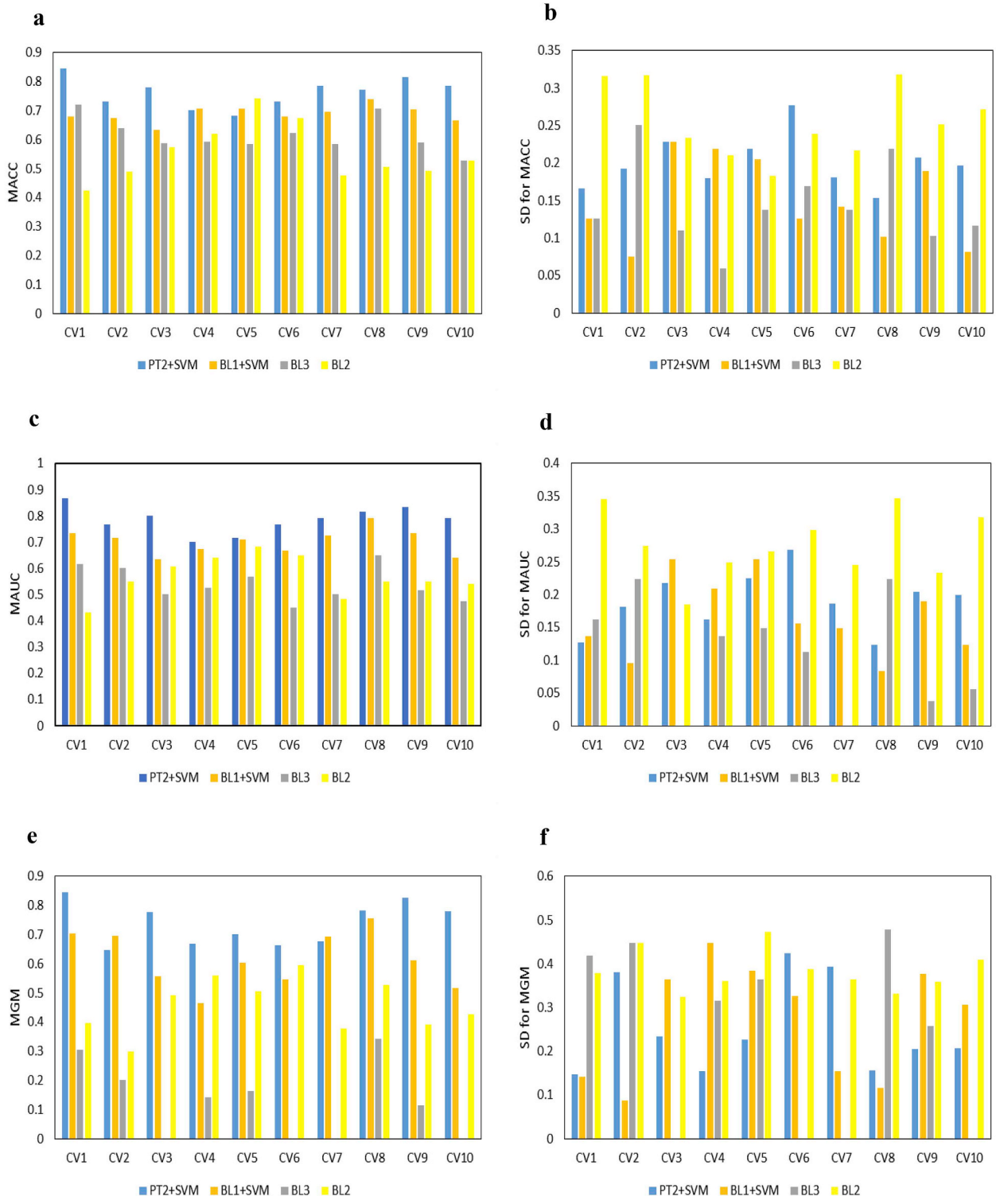

Fig. 5. Performance results of the top-performing prediction algorithms, applied to the target test data pertaining to in-vivo breast cancer patients. The auxiliary data are related to in-vivo triple-negative breast cancer patients. These results are obtained by running CV 10 times. CV*i* is the *i*th cross-validation. MACC is mean accuracy. MAUC is mean AUC. MGM is mean G-Mean. SD is standard deviation.

Table 4
P-values and average rankings between all pairs of the top-performing prediction algorithms in Fig. 5c on the clinical data pertaining to in-vivo breast cancer patients based on the Friedman test. The auxiliary data are related to in-vivo triple negative breast cancer patients. The lower rank a prediction algorithm has, the better performance that algorithm achieves. The difference between the two algorithms in a pair with $p < 0.001$ is considered highly statistically significant. The difference between the two algorithms in a pair with $p < 0.05$ is considered statistically significant.

Prediction Algorithm A	Prediction Algorithm B	Average Rankings	*P*-Value
PT2 + SVM	BL1 + SVM	(1, 2)	83×10^{-3}
PT2 + SVM	BL2	(1, 3.4)	32×10^{-6}
PT2 + SVM	BL3	(1, 3.6)	67×10^{-7}
BL1 + SVM	BL2	(2, 3.4)	15×10^{-3}
BL1 + SVM	BL3	(2, 3.6)	55×10^{-4}
BL2	BL3	(3.4, 3.6)	72×10^{-2}

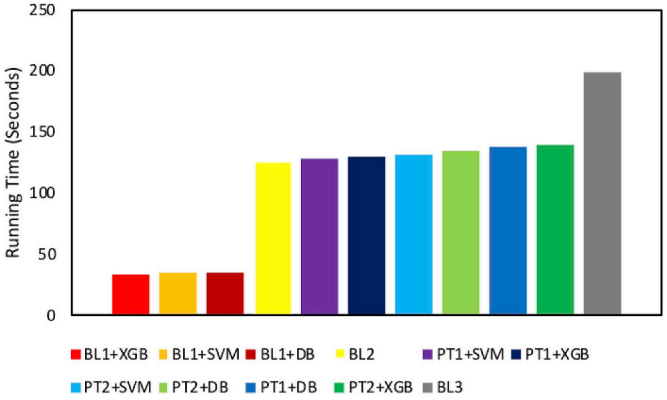

Fig. 6. Comparison of the total running times (in seconds) of the prediction algorithms on the target test data pertaining to in-vivo breast cancer patients. The auxiliary data are related to in-vivo multiple myeloma patients. The prediction algorithms are ranked from left to right where the leftmost prediction algorithm is fastest and the rightmost algorithm is slowest.

Fig. 4 shows the total running times of the prediction algorithms spent in the 5-fold cross-validation procedure. The results reported in Fig. 4 indicate that the prediction algorithms employing the first baseline approach (BL1) were faster than all other algorithms. The prediction algorithms employing the second and third baselines (i.e., BL2 and BL3) were slower than the prediction algorithms employing our proposed transfer learning approaches.

To understand the stability of the reported results we run cross-validation (CV) on the top-performing prediction algorithms 10 times. Fig. 5a–f shows the performance results obtained from the 10 runs of cross-validation. It can be seen from Fig. 5a, c, and e that PT2 + SVM employing our proposed PT2 approach outperforms the baselines consistently in terms of MACC, MAUC and MGM. These results clearly demonstrate the predictive superiority of PT2 + SVM.

To assess statistical significance, we use the non-parametric Friedman test [44–50]. Table 4 presents the average rankings of all pairs of the top-performing prediction algorithms in Fig. 5c and the corresponding *p*-values, calculated based on MAUC. It can be seen from Table 4 that PT2 + SVM outperforms BL2 and BL3, and the differences are statistically significant ($p < 0.001$). BL1 + SVM is also significantly better than BL2 and BL3 ($p < 0.05$). The difference between PT2 + SVM and BL1 + SVM is not significant ($p = 0.083$).

4.3.1.2. Utilization of in-vivo multiple myeloma data. Table 5 presents performance results obtained by running 5-fold cross-validation on clinical data pertaining to in-vivo breast cancer patients, where the auxiliary data are related to in-vivo multiple myeloma patients. As illustrated in Table 5, the prediction algorithm PT2 + DB employing the proposed PT2 approach outperforms the baseline algorithms. Specifically, PT2 + DB yields the highest MACC of 0.746, the highest MAUC of 0.708, and the highest MGM of 0.563.

Table 5
Performance results of the studied prediction algorithms obtained from 5-fold cross-validation on clinical data pertaining to in-vivo breast cancer patients. The auxiliary data are related to in-vivo multiple myeloma patients. The highest MACC, MAUC, and MGM values are shown in bold. MACC denotes the mean accuracy. MAUC denotes the mean area under curve. MGM denotes the mean G-Mean value. SD is the standard deviation calculated based on the five runs of 5-fold cross-validation.

	MACC	SD	MAUC	SD	MGM	SD
PT1 + SVM	0.740	0.224	0.683	0.302	0.526	0.486
PT1 + XGB	0.620	0.210	0.550	0.273	0.326	0.447
PT1 + DB	0.686	0.232	0.641	0.288	0.478	0.462
PT2 + SVM	0.626	0.146	0.583	0.220	0.394	0.372
PT2 + XGB	0.666	0.286	0.633	0.325	0.504	0.472
PT2 + DB	**0.746**	0.284	**0.708**	0.330	**0.563**	0.519
BL1 + SVM	0.600	0.169	0.600	0.231	0.420	0.392
BL1 + XGB	0.633	0.110	0.633	0.126	0.492	0.292
BL1 + DB	0.633	0.110	0.650	0.149	0.514	0.315
BL2	0.593	0.153	0.466	0.074	0.000	0.000
BL3	0.593	0.153	0.525	0.136	0.141	0.316

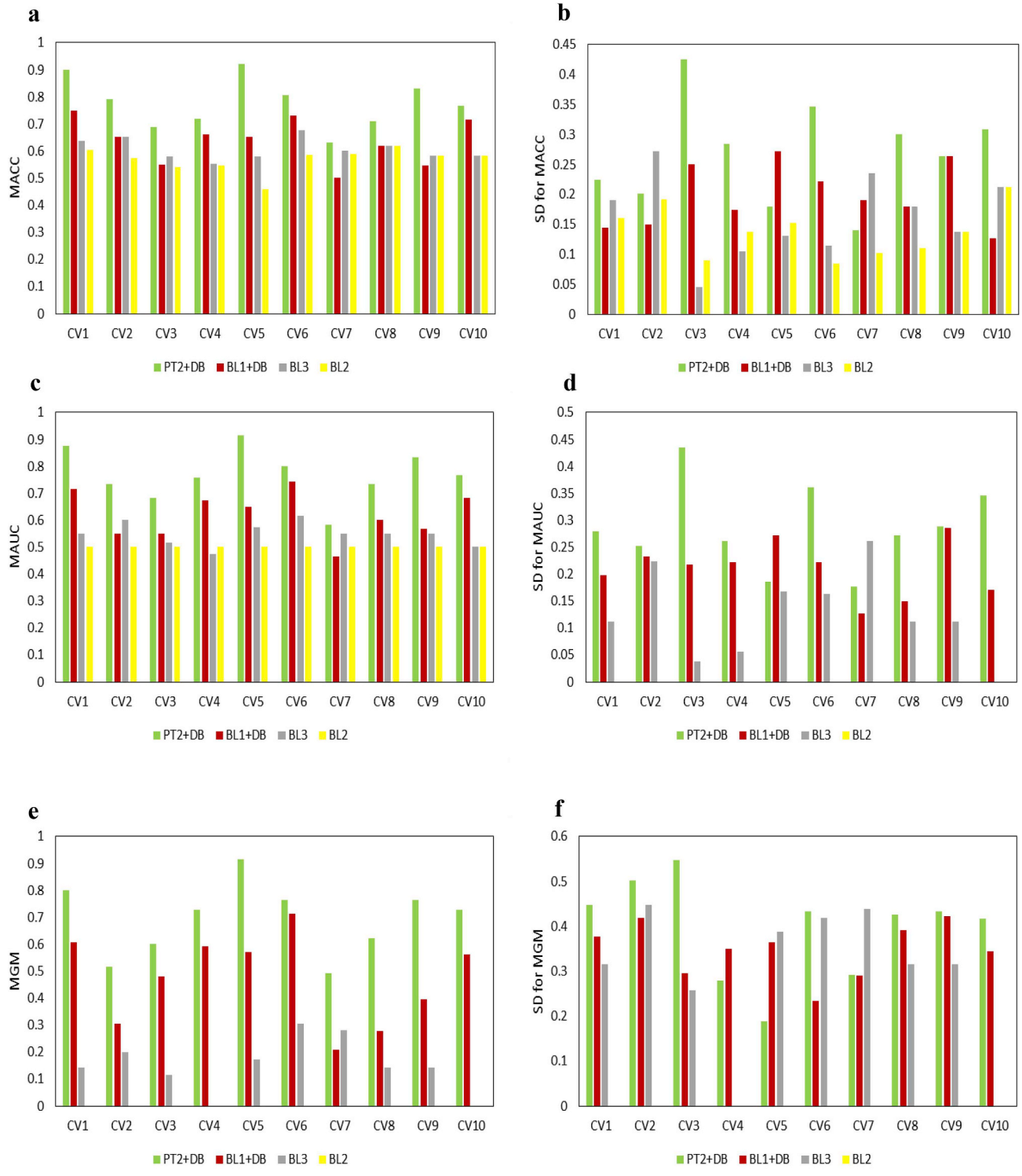

Fig. 7. Performance results of the top-performing prediction algorithms, applied to the target test data pertaining to in-vivo breast cancer patients. The auxiliary data are related to in-vivo multiple myeloma patients. These results are obtained by running CV 10 times. CV*i* is the *i*th cross-validation. MACC is mean accuracy. MAUC is mean AUC. MGM is mean G-Mean. SD is standard deviation.

Table 6
P-values and average rankings of the Friedman test between all pairs of the top-performing prediction algorithms in Fig. 7c on clinical data pertaining to in-vivo breast cancer patients. The auxiliary data are related to in-vivo multiple myeloma patients. The lower rank a prediction algorithm has, the better performance that algorithm achieves. The difference between the two algorithms in a pair with $p < 0.001$ is considered highly statistically significant. The difference between the two algorithms in a pair with $p < 0.05$ is considered statistically significant.

Prediction Algorithm A	Prediction Algorithm B	Average Rankings	*P*-Value
PT2 + DB	BL1 + DB	(1, 2.3)	24×10^{-3}
PT2 + DB	BL2	(1, 3.75)	19×10^{-7}
PT2 + DB	BL3	(1, 2.95)	73×10^{-5}
BL1 + DB	BL2	(2.3, 3.75)	12×10^{-3}
BL1 + DB	BL3	(2.3, 2.95)	26×10^{-2}
BL2	BL3	(3.75, 2.95)	16×10^{-2}

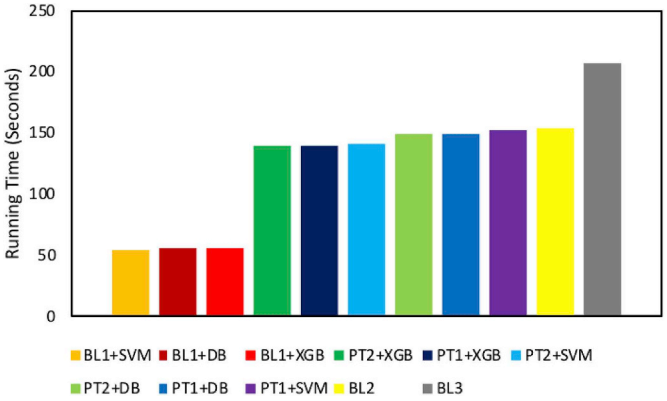

Fig. 8. Comparison of the total running times (in seconds) of the prediction algorithms on the target test data pertaining to in-vivo triple-negative breast cancer patients. The auxiliary data are related to in-vivo breast cancer patients. The prediction algorithms are ranked from left to right where the leftmost prediction algorithm is fastest and the rightmost algorithm is slowest.

Fig. 6 shows the total running times of the prediction algorithms spent in the 5-fold cross-validation procedure. It can be seen from Fig. 6 that the prediction algorithms employing the first baseline approach (BL1) and BL2 were faster than the other prediction algorithms. The prediction algorithm employing the third baseline (BL3) was slower than our prediction algorithms employing the proposed transfer learning approaches.

Fig. 7 presents the performance results of the top-performing prediction algorithms obtained by using 10 runs of cross-validation (CV). These results show that PT2 + DB consistently beats the baseline approaches. Table 6 shows the average rankings and the corresponding *p*-values, calculated based on MAUC, between all pairs of the top-performing prediction algorithms in Fig. 7c according to the Friedman test. It can be seen from Table 6 that our prediction algorithm PT2 + DB outperforms BL2 and BL3 with statistical significance ($p < 0.001$). PT2 + DB also beats BL1 + DB with statistical significance ($p < 0.05$). BL1 + DB is significantly better than BL2 ($p < 0.05$). However, the difference between BL1 + DB and BL3 is not significant ($p > 0.05$). These results show that PT2 + DB significantly outperforms all the other methods.

4.3.2. Predicting in-vivo drug sensitivity of triple-negative breast cancer patients

4.3.2.1. Utilization of in-vivo breast cancer data. Table 7 presents performance results of the prediction algorithms obtained by running 5-fold cross-validation on clinical data pertaining to in-vivo triple-negative breast cancer patients. The auxiliary data are related to in-vivo breast cancer patients. As illustrated in Table 7, PT1 + XGB and PT2 + XGB employing our proposed transfer learning approaches yield the highest MACC of 0.750, the highest MAUC of 0.791, and the highest MGM of 0.780. These results clearly demonstrate the predictive capabilities of the proposed approaches. Fig. 8 shows the running times of the prediction algorithms. It can be seen from Fig. 8 that the prediction algorithms employing the first baseline approach (BL1) were

Table 7
Performance results of the studied prediction algorithms obtained by utilizing 5-fold cross-validation on clinical data pertaining to in-vivo triple-negative breast cancer patients. The auxiliary data are related to in-vivo breast cancer patients. The highest MACC, MAUC, and MGM values are shown in bold. MACC denotes the mean accuracy. MAUC denotes the mean area under curve. MGM denotes the mean G-Mean value. SD is the standard deviation calculated based on the five runs of 5-fold cross-validation.

	MACC	SD	MAUC	SD	MGM	SD
PT1 + SVM	0.458	0.058	0.555	0.078	0.422	0.069
PT1 + XGB	**0.750**	0.000	**0.791**	0.058	**0.780**	0.050
PT1 + DB	0.625	0.294	0.722	0.157	0.652	0.255
PT2 + SVM	0.458	0.058	0.555	0.078	0.422	0.069
PT2 + XGB	**0.750**	0.000	**0.791**	0.058	**0.780**	0.050
PT2 + DB	0.625	0.294	0.722	0.157	0.652	0.255
BL1 + SVM	0.500	0.235	0.555	0.314	0.517	0.322
BL1 + XGB	0.583	0.000	0.541	0.058	0.524	0.074
BL1 + DB	0.416	0.235	0.541	0.058	0.288	0.408
BL2	0.416	0.235	0.375	0.294	0.263	0.372
BL3	0.500	0.117	0.513	0.137	0.470	0.194

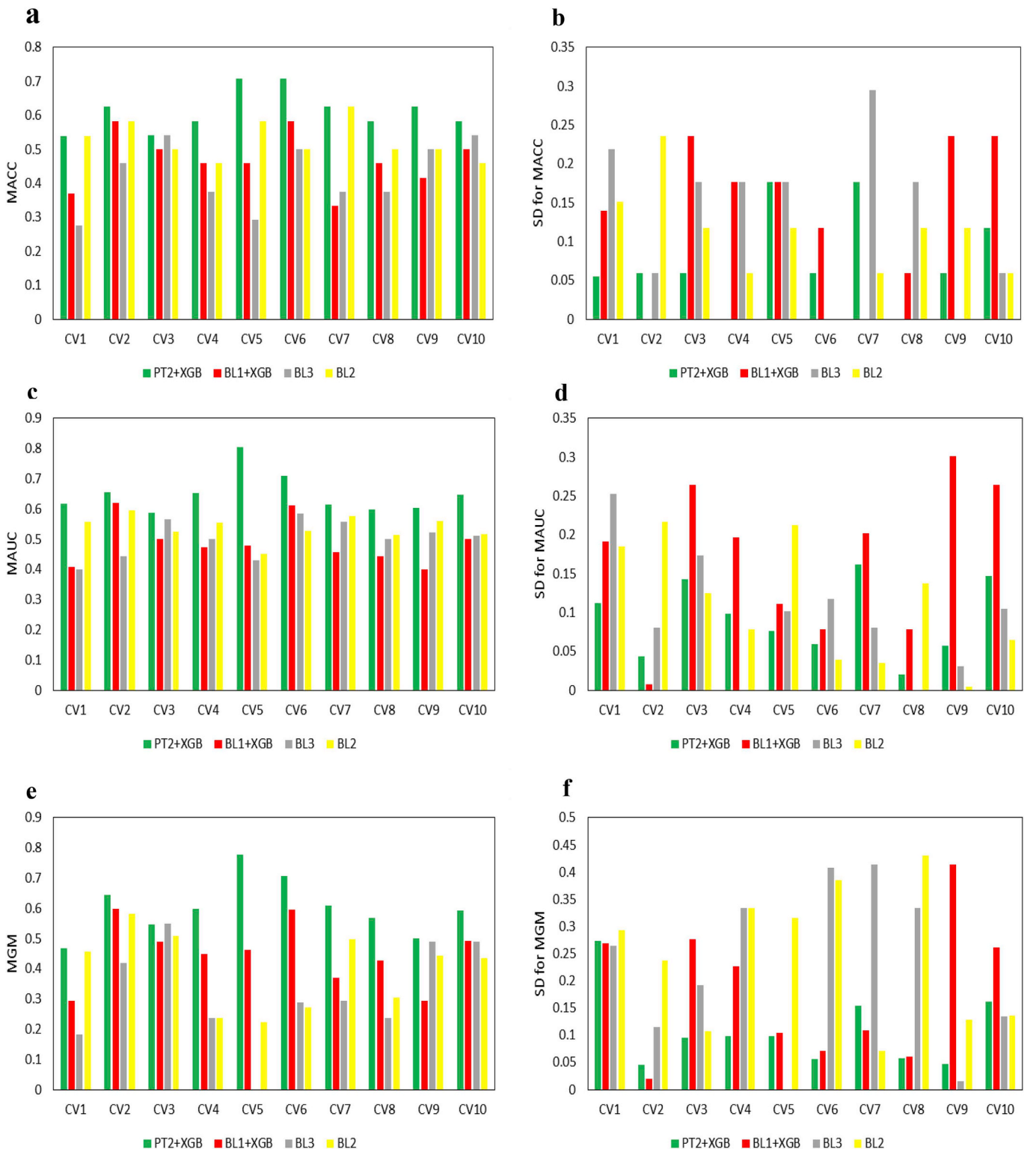

Fig. 9. Performance results of the top-performing prediction algorithms, applied to the target test data pertaining to in-vivo triple-negative breast cancer patients. The auxiliary data are related to in-vivo breast cancer patients. These results are obtained by running CV 10 times. CV*i* is the *i*th cross-validation. MACC is mean accuracy. MAUC is mean AUC. MGM is mean G-Mean. SD is standard deviation.

Table 8
P-values and average rankings of the Friedman test between all pairs of the top-performing prediction algorithms in Fig. 9c on clinical data pertaining to in-vivo triple-negative breast cancer patients. The auxiliary data are related to in-vivo breast cancer patients. The lower rank a prediction algorithm has, the better performance that algorithm achieves. The difference between the two algorithms in a pair with $p < 0.001$) is considered highly statistically significant. The difference between the two algorithms in a pair with $p < 0.05$ is considered statistically significant.

Prediction Algorithm A	Prediction Algorithm B	Average Rankings	*P*-Value
PT2 + XGB	BL1 + XGB	(1, 3.3)	67×10^{-6}
PT2 + XGB	BL2	(1, 2.5)	93×10^{-4}
PT2 + XGB	BL3	(1, 3.2)	13×10^{-5}
BL1 + XGB	BL2	(3.3, 2.5)	16×10^{-2}
BL1 + XGB	BL3	(3.3, 3.2)	86×10^{-2}
BL2	BL3	(2.5, 3.2)	22×10^{-2}

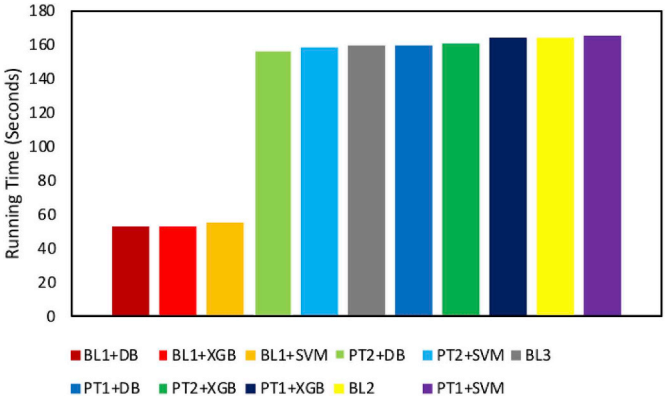

Fig. 10. Comparison of the total running times (in seconds) of the prediction algorithms on the target test data pertaining to in-vivo triple-negative breast cancer patients. The auxiliary data are related to in-vivo multiple myeloma patients. The prediction algorithms are ranked from left to right where the leftmost prediction algorithm is fastest and the rightmost algorithm is slowest.

faster than the other prediction algorithms. The prediction algorithms employing our proposed transfer learning approaches were faster than the prediction algorithms employing BL2 and BL3.

Fig. 9 presents the performance results of the top-performing prediction algorithms obtained by utilizing 10 runs of cross-validation (CV). The figure shows that PT2 + XGB consistently beats the baseline approaches. Table 8 reports the average rankings and the corresponding *p*-values, calculated based on MAUC, between all pairs of the top-performing prediction algorithms in Fig. 9c according to the Friedman test. The table shows that PT2 + XGB significantly outperforms the baseline approaches.

4.3.2.2. Utilization of in-vivo multiple myeloma data. Table 9 reports the performance results of the prediction algorithms obtained by running 5-fold cross-validation on clinical data pertaining to in-vivo triple-negative breast cancer patients. The auxiliary data are related to in-vivo multiple myeloma patients. As illustrated by the results in Table 9, PT2 + DB and PT1 + DB yield the highest MACC of 0.625 (a tie with BL2), the highest MAUC of 0.569, and the highest MGM of 0.543. These results demonstrate the suitability of the proposed transfer learning approaches PT1 and PT2. Fig. 10 displays the total running times of the studied prediction algorithms. The prediction algorithms employing the first baseline approach (BL1) were faster than the other algorithms. PT2 + DB and PT2 + SVM were faster than BL3 and BL2.

Fig. 11 presents the performance results of the top-performing prediction algorithms obtained from 10 runs of cross-validation (CV). The figure shows that PT2 + DB consistently outperforms the baselines, demonstrating the predictive capability of PT2 + DB. Moreover, the *p*-values in Table 10 indicate that PT2 + DB is significantly better than the baselines. These results demonstrate the superior and stable performance achieved by PT2 + DB.

Table 9
Performance results of the studied prediction algorithms obtained from 5-fold cross-validation on clinical data pertaining to in-vivo triple-negative breast cancer patients. The auxiliary data are related to in-vivo multiple myeloma patients. The highest MACC, MAUC, and MGM values are highlighted in bold. MACC denotes the mean accuracy. MAUC denotes the mean area under curve. MGM denotes the mean G-Mean value. SD is the standard deviation calculated based on the five runs of 5-fold cross-validation.

	MACC	SD	MAUC	SD	MGM	SD
PT1 + SVM	0.458	0.058	0.541	0.176	0.288	0.408
PT1 + XGB	0.583	0.000	0.541	0.058	0.524	0.074
PT1 + DB	**0.625**	0.058	**0.569**	0.019	**0.543**	0.048
PT2 + SVM	0.458	0.058	0.541	0.176	0.288	0.408
PT2 + XGB	0.583	0.000	0.541	0.058	0.524	0.074
PT2 + DB	**0.625**	0.058	**0.569**	0.019	**0.543**	0.048
BL1 + SVM	0.416	0.000	0.513	0.137	0.439	0.044
BL1 + XGB	0.458	0.058	0.500	0.000	0.235	0.333
BL1 + DB	0.583	0.117	0.527	0.039	0.254	0.360
BL2	**0.625**	0.176	0.500	0.000	0.000	0.000
BL3	0.416	0.117	0.472	0.039	0.192	0.272

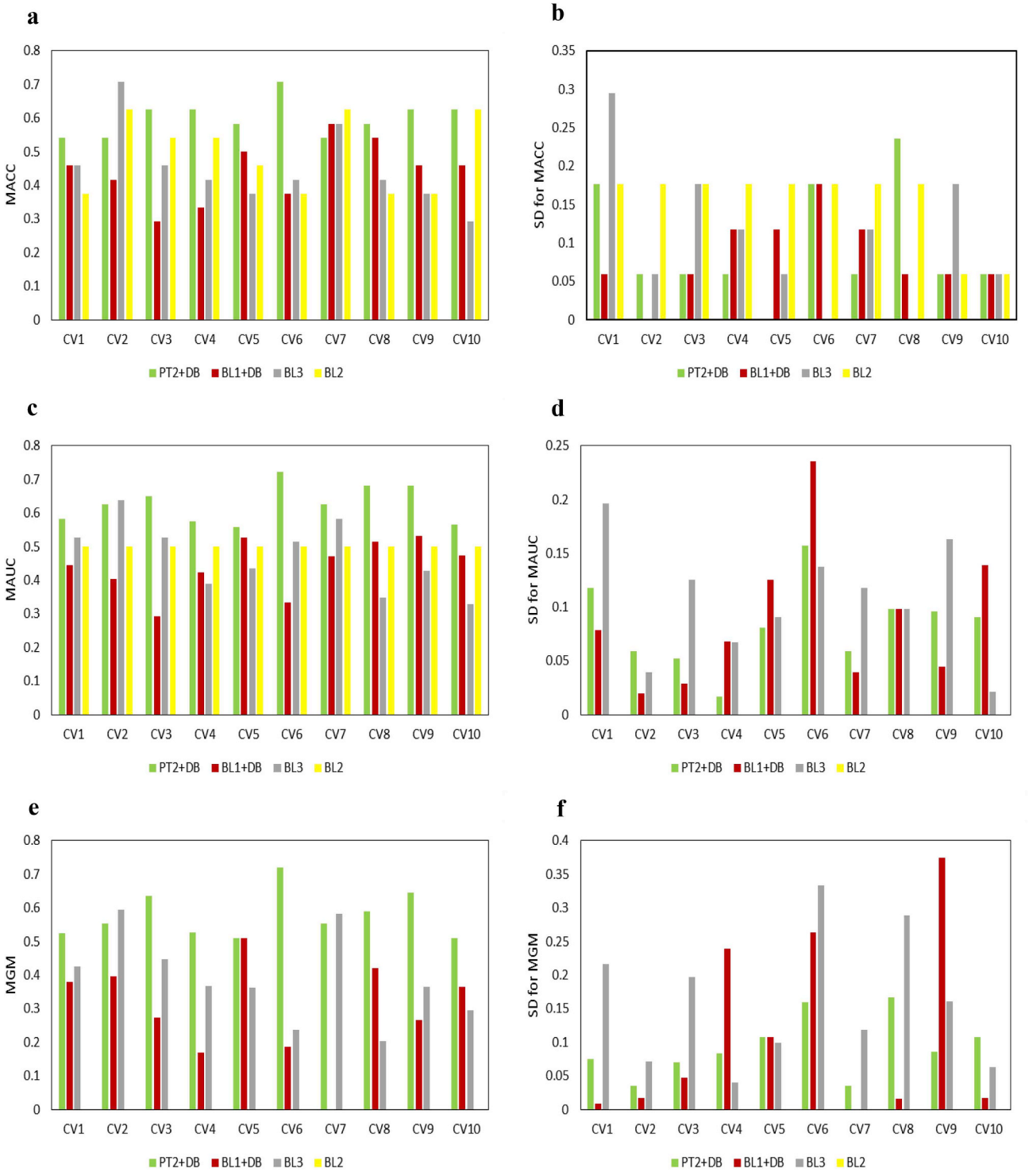

Fig. 11. Performance results of the top-performing prediction algorithms, applied to the target test data pertaining to in-vivo triple-negative breast cancer patients. The auxiliary data are related to in-vivo multiple myeloma patients. These results are obtained by running CV 10 times. CV*i* is the *i*th cross-validation. MACC is mean accuracy. MAUC is mean AUC. MGM is mean G-Mean. SD is standard deviation.

Table 10
P-values and average rankings of the Friedman test between all pairs of the top-performing prediction algorithms in Fig. 11c on clinical data pertaining to in-vivo triple-negative breast cancer patients. The auxiliary data are related to in-vivo multiple myeloma patients. The lower rank a prediction algorithm has, the better performance that algorithm achieves. The difference between the two algorithms in a pair with $p < 0.001$ is considered highly statistically significant. The difference between the two algorithms in a pair with $p < 0.05$ is considered statistically significant.

Prediction Algorithm A	Prediction Algorithm B	Average Rankings	*P*-Value
PT2 + DB	BL1 + DB	(1.1, 3.2)	27×10^{-5}
PT2 + DB	BL2	(1.1, 2.8)	32×10^{-4}
PT2 + DB	BL3	(1.1, 2.9)	18×10^{-4}
BL1 + DB	BL2	(3.2, 2.8)	48×10^{-2}
BL1 + DB	BL3	(3.2, 2.9)	60×10^{-2}
BL2	BL3	(2.8, 2.9)	86×10^{-2}

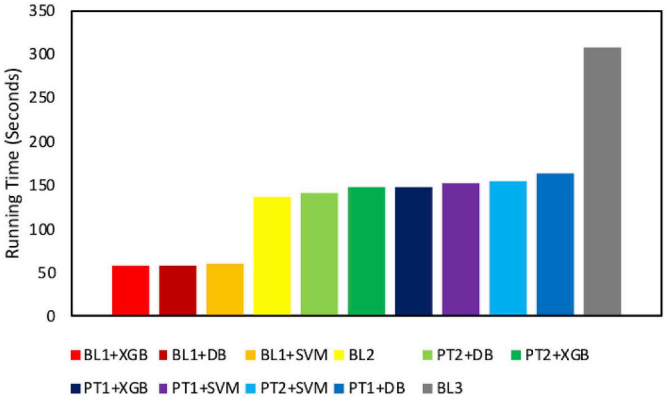

Fig. 12. Comparison of the total running times (in seconds) of the prediction algorithms on the target test data pertaining to in-vivo multiple myeloma patients. The auxiliary data are related to in-vivo triple-negative breast cancer patients. The prediction algorithms are ranked from left to right where the leftmost prediction algorithm is fastest and the rightmost algorithm is slowest.

Table 11
Performance results of the studied prediction algorithms obtained from 5-fold cross-validation on clinical data pertaining to in-vivo multiple myeloma patients. The auxiliary data are related to in-vivo triple-negative breast cancer patients. The highest MACC, MAUC, and MGM values are highlighted in bold. MACC denotes the mean accuracy. MAUC denotes the mean area under curve. MGM denotes the mean G-Mean value. SD is the standard deviation calculated based on the five runs of 5-fold cross-validation.

	MACC	SD	MAUC	SD	MGM	SD
PT1 + SVM	0.602	0.074	0.606	0.073	0.591	0.081
PT1 + XGB	0.644	0.073	0.644	0.074	0.640	0.076
PT1 + DB	0.585	0.090	0.589	0.091	0.568	0.099
PT2 + SVM	0.589	0.094	0.590	0.092	0.578	0.102
PT2 + XGB	**0.651**	0.048	0.650	0.046	0.641	0.045
PT2 + DB	0.550	0.094	0.551	0.099	0.534	0.105
BL1 + SVM	0.596	0.066	0.597	0.066	0.585	0.068
BL1 + XGB	0.585	0.089	0.585	0.088	0.582	0.086
BL1 + DB	**0.651**	0.073	**0.655**	0.069	**0.642**	0.066
BL2	0.490	0.069	0.488	0.063	0.057	0.181
BL3	0.531	0.076	0.535	0.067	0.165	0.271

4.3.3. Predicting in-vivo drug sensitivity of multiple myeloma patients

4.3.3.1. Utilization of in-vivo triple-negative breast cancer data. Table 11 presents performance results of the prediction algorithms obtained by running 5-fold cross-validation on clinical data pertaining to in-vivo multiple myeloma patients, where the auxiliary data are related to in-vivo triple-negative breast cancer patients. As illustrated in Table 11, PT2 + XGB and BL1 + DB tie on yielding the highest MACC of 0.651. For MAUC and MGM, BL1 + DB performs slightly better than PT2 + XGB, yielding the highest MAUC of 0.655 and the highest MGM of 0.642. The results displayed in Fig. 12 show that the prediction algorithms employing the first baseline approach (BL1) were faster than the other algorithms. The prediction algorithm employing BL3 was the slowest.

Fig. 13 presents the performance results of the top-performing prediction algorithms obtained by running cross-validation (CV) 10 times. As illustrated in Fig. 13a–f, our prediction algorithm PT2 + XGB consistently outperforms the baselines including BL1 + DB, BL2, and BL3. The *p*-values in Table 12 indicate that PT2 + XGB significantly

Table 12
P-values and average rankings of the Friedman test between all pairs of the top-performing prediction algorithms in Fig. 13c on clinical data pertaining to in-vivo multiple myeloma patients. The auxiliary data are related to in-vivo triple-negative breast cancer patients. The lower rank a prediction algorithm has, the better performance that algorithm achieves. The difference between the two algorithms in a pair with $p < 0.001$ is considered highly statistically significant. The difference between the two algorithms in a pair with $p < 0.05$ is considered statistically significant.

Prediction Algorithm A	Prediction Algorithm B	Average Rankings	*P*-Value
PT2 + XGB	BL1 + DB	(1.1, 1.9)	16×10^{-2}
PT2 + XGB	BL2	(1.1, 3.8)	29×10^{-7}
PT2 + XGB	BL3	(1.1, 3.2)	27×10^{-5}
BL1 + DB	BL2	(1.9, 3.8)	99×10^{-7}
BL1 + DB	BL3	(1.9, 3.2)	24×10^{-3}
BL2	BL3	(3.8, 3.2)	29×10^{-2}

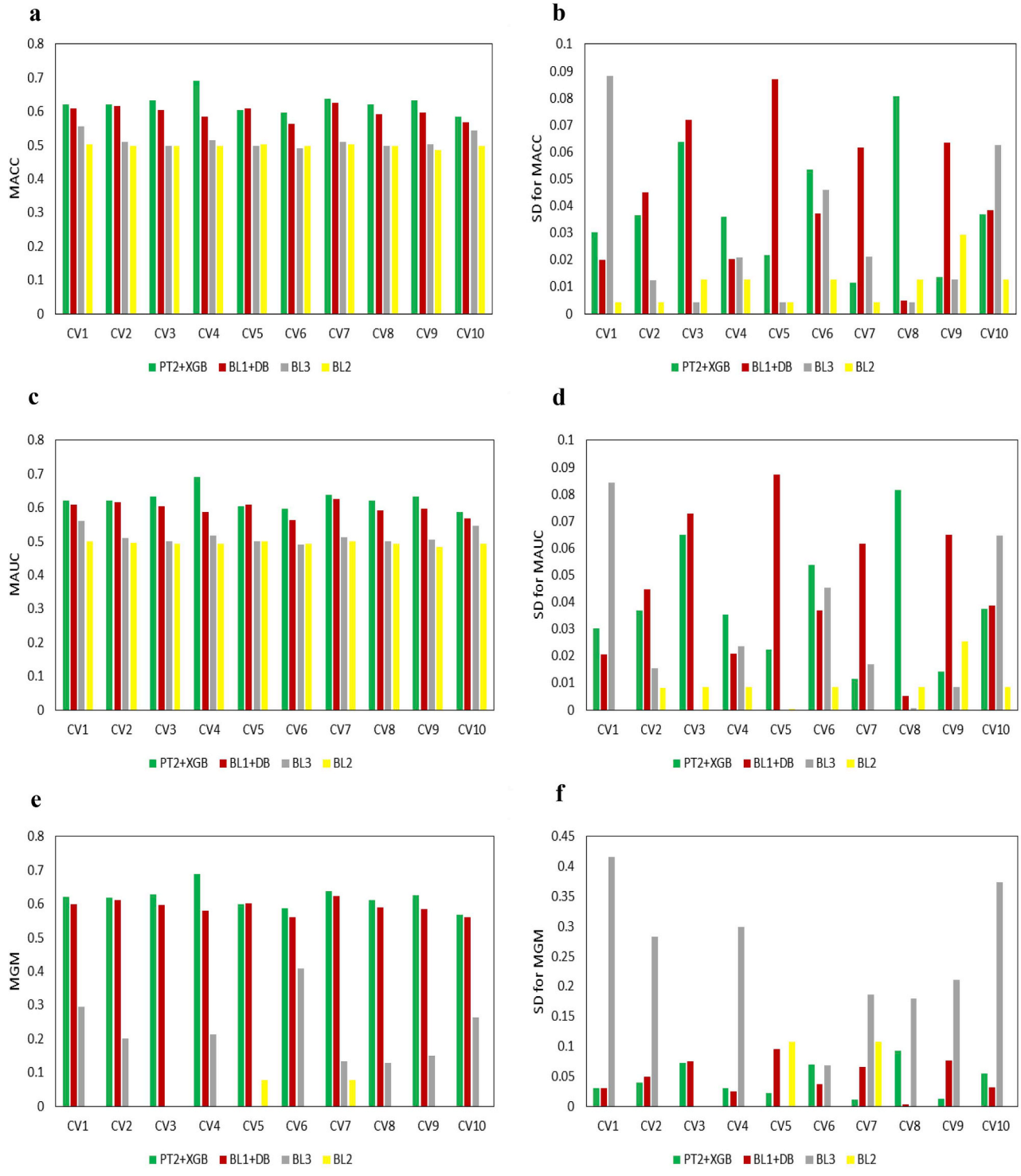

Fig. 13. Performance results of the top-performing prediction algorithms, applied to the target test data pertaining to in-vivo multiple myeloma patients. The auxiliary data are related to in-vivo triple-negative breast cancer patients. These results are obtained by running CV 10 times. CV*i* is the *i*th cross-validation. MACC is mean accuracy. MAUC is mean AUC. MGM is mean G-Mean. SD is standard deviation.

Table 13
Performance results of the studied prediction algorithms obtained from 5-fold cross-validation on clinical data pertaining to in-vivo multiple myeloma patients. The auxiliary data are related to in-vivo breast cancer patients. The highest MACC, MAUC, and MGM values are shown in bold. MACC denotes the mean accuracy. MAUC denotes the mean area under curve. MGM denotes the mean G-Mean value. SD is the standard deviation calculated based on the five runs of 5-fold cross-validation.

	MACC	SD	MAUC	SD	MGM	SD
PT1 + SVM	0.632	0.097	0.631	0.097	0.628	0.099
PT1 + XGB	0.609	0.072	0.609	0.072	0.605	0.072
PT1 + DB	0.604	0.045	0.604	0.045	0.595	0.046
PT2 + SVM	**0.638**	0.046	**0.637**	0.045	**0.634**	0.043
PT2 + XGB	0.549	0.060	0.551	0.060	0.542	0.062
PT2 + DB	0.557	0.113	0.558	0.114	0.548	0.107
BL1 + SVM	0.602	0.080	0.601	0.080	0.595	0.082
BL1 + XGB	0.608	0.115	0.609	0.115	0.604	0.119
BL1 + DB	0.621	0.064	0.621	0.062	0.618	0.060
BL2	0.497	0.024	0.494	0.013	0.000	0.000
BL3	0.520	0.055	0.524	0.046	0.254	0.259

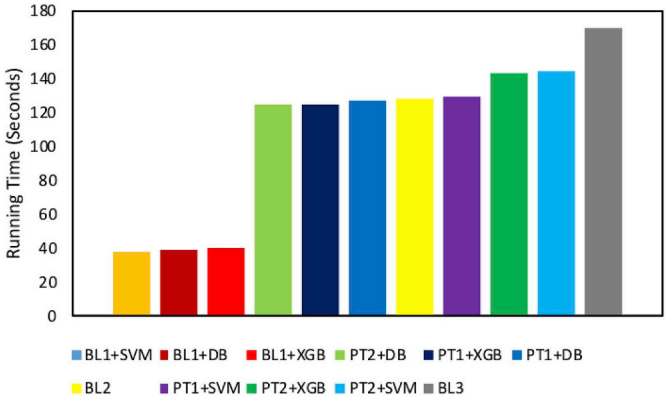

Fig. 14. Comparison of the total running times (in seconds) of the prediction algorithms on the target test data pertaining to in-vivo multiple myeloma patients. The auxiliary data are related to in-vivo breast cancer patients. The prediction algorithms are ranked from left to right where the leftmost prediction algorithm is fastest and the rightmost algorithm is slowest.

Table 14
P-values and average rankings of the Friedman test between all pairs of the top-performing prediction algorithms in Fig. 15c on clinical data pertaining to in-vivo multiple myeloma patients. The auxiliary data are related to in-vivo breast cancer patients. The lower rank a prediction algorithm has, the better performance that algorithm achieves. The difference between the two algorithms in a pair with $p < 0.001$ is considered highly statistically significant. The difference between the two algorithms in a pair with $p < 0.05$ is considered statistically significant.

Prediction Algorithm A	Prediction Algorithm B	Average Rankings	*P*-Value
PT1 + SVM	PT2 + SVM	(1.5, 1.5)	10×10^{-1}
PT1 + SVM	BL1 + DB	(1.5, 3.3)	10×10^{-3}
PT1 + SVM	BL2	(1.5, 4.65)	83×10^{-7}
PT1 + SVM	BL3	(1.5, 4.05)	31×10^{-5}
PT2 + SVM	BL1 + DB	(1.5, 3.3)	10×10^{-3}
PT2 + SVM	BL2	(1.5, 4.65)	83×10^{-7}
PT2 + SVM	BL3	(1.5, 4.05)	31×10^{-5}
BL1 + DB	BL2	(3.3, 4.65)	56×10^{-3}
BL1 + DB	BL3	(3.3, 4.05)	28×10^{-2}
BL2	BL3	(4.65, 4.05)	39×10^{-2}

outperforms BL2 and BL3 ($p < 0.001$). BL1 + DB also significantly outperforms BL2 and BL3 ($p < 0.05$). The difference between PT2 + XGB and BL1 + DB is not significant ($p = 0.16$).

4.3.3.2. Utilization of in-vivo breast cancer data. Table 13 presents performance results of the prediction algorithms obtained by running 5-fold cross-validation on clinical data pertaining to in-vivo multiple myeloma patients, where the auxiliary data are related to in-vivo breast cancer patients. As illustrated in Table 13, PT2 + SVM outperforms all baselines, yielding the highest MACC of 0.638, the highest MAUC of 0.637 and the highest MGM of 0.634. Fig. 14 shows the total running times of the prediction algorithms. The prediction algorithms employing the first baseline approach (BL1) were faster than the other algorithms. BL3 was the slowest among all prediction algorithms.

As shown in Fig. 15 based on 10 runs of cross-validation (CV), the prediction algorithms PT1 + SVM and PT2 + SVM consistently outperform the baselines including BL1 + DB, BL2, and BL3. Moreover, the *p*-values in Table 14 indicate that PT1 + SVM and PT2 + SVM are significantly better than BL1 + DB, BL2, and BL3. These results demonstrate the superior and stable performance of PT1 + SVM and PT2 + SVM.

5. Discussion

Our first proposed transfer learning approach (PT1) aims to incorporate auxiliary data of a transformed representation closer to the target training set into the input of a machine learning algorithm, to obtain a prediction model. If the transformed auxiliary data are not closer to the target training set, then PT1 abstains from incorporating the transformed auxiliary data into the target training data. Consequently, only the target training data are provided as input to a machine learning algorithm, to obtain a model for generating predictions on the target test set. The component that is responsible for either selecting or not selecting (i.e., abstaining) an example from the transferred auxiliary data is TrNewBoost, which stands for transfer with abstention boosted. The second transfer learning approach (PT2) performs all steps of PT1 except that PT2 does not invoke the TrNewBoost procedure.

Predicting the drug response to a specific cancer type remains a challenging task. Making accurate cancer drug sensitivity predictions would increase the chances of identifying effective cancer drugs, that in turn facilitates clinicians administer better personalized treatments, leading to increased survival rates for cancer patients [51–61]. In this

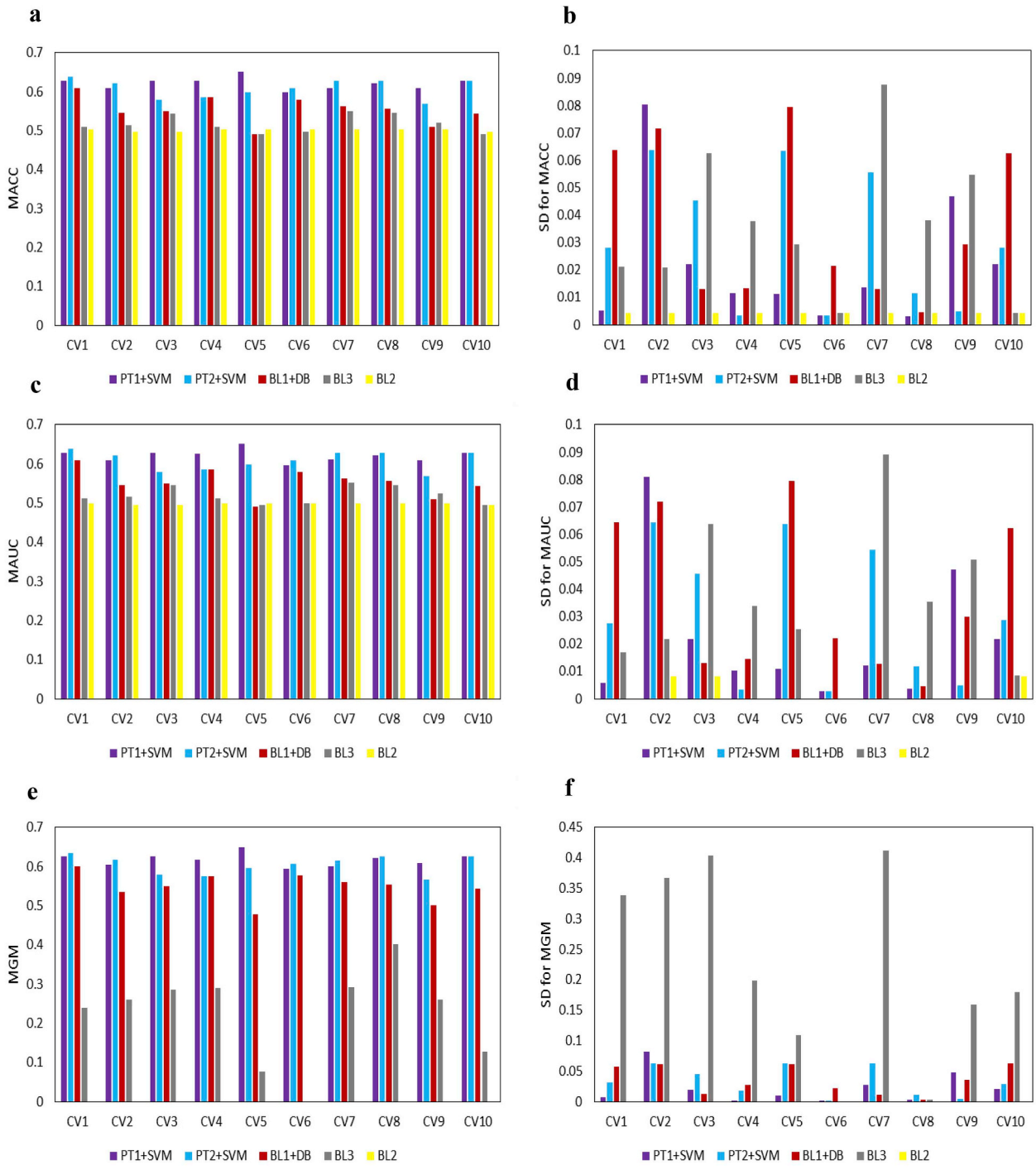

Fig. 15. Performance results of the top-performing prediction algorithms, applied to the target test data pertaining to in-vivo multiple myeloma patients. The auxiliary data are related to in-vivo breast cancer patients. These results are obtained by running CV 10 times. CV*i* is the *i*th cross-validation. MACC is mean accuracy. MAUC is mean AUC. MGM is mean G-Mean. SD is standard deviation.

study, we aim to improve cancer drug sensitivity prediction by leveraging advanced machine learning techniques, that consist of a transfer learning approach combined with or without a boosting technique. Extensive experimental results show that our prediction algorithms outperform the existing transfer learning approaches employing CORAL-SVM and TrAdaBoost.

For each prediction task, we ran cross-validation 10 times for top-performing prediction algorithms, where each run had different samples in the corresponding folds. This type of assessment is important to examine the stability of the prediction algorithms. According to the *p*-values obtained from the non-parametric Friedman test, our proposed prediction algorithms significantly outperform the baselines BL2 and BL3 that employ the existing CORAL-SVM and TrAdaBoost algorithms respectively.

The efficiency of the prediction algorithms was evaluated based on running time. According to the resulting running times of the prediction algorithms, the prediction algorithms employing the first baseline approach (BL1) were faster than the other prediction algorithms. This is owing to BL1 merely combining the auxiliary data from a related task with the training data of a target task without using any transfer mechanism. On the other hand, all the other approaches (i.e., PT1, PT2, BL2, and BL3) adopted transfer mechanisms, and hence were slower than BL1.

It is worth noting that, in the baseline approach BL2, the auxiliary data were transformed using a transfer mechanism that incorporates unlabeled target test data. Then, only the transformed auxiliary data were provided as input to a machine learning algorithm, to obtain a model for generating predictions on the transferred target test data. This type of transfer learning belongs to domain adaptation [62–64]. Thus, BL2 differs from the other transfer learning approaches that also utilized target training data to build the models for generating predictions on the target test data.

In this work, all approaches were compared under transfer learning settings, where we evaluated the performance of all approaches using several real clinical datasets pertaining to patients with breast cancer, triple-negative breast cancer, and multiple myeloma. Our experimental results demonstrated the suitability of the transfer mechanisms employed in the proposed approaches. Our results also demonstrated the superiority of the proposed transfer learning approaches over the existing methods including BL2 and BL3. To the best of our knowledge, this is the first study to compare different transfer learning approaches, including state-of-the-art methods such as BL2 and BL3, in the clinical informatics domain.

6. Conclusion and future work

We propose two new transfer learning approaches to accurately predict the clinical drug response of patients with different cancer types, including breast cancer, triple-negative breast cancer, and multiple myeloma. The first proposed approach works by transferring knowledge of auxiliary data from a related task to a target task of drug sensitivity prediction. The second approach includes an abstention mechanism, whose aim is to abstain from including those tumor samples that do not have closer representation to the target task. These transfer learning approaches are designed to improve predictive performance. Extensive experimental results based on several real clinical datasets show that our approaches outperform existing methods and generate superior statistically significant results.

In future work, we plan to (1) apply the proposed transfer learning approaches to multi-class classification problems as opposed to the binary classification problems studied here; (2) incorporate the proposed transfer learning approaches with deep leaning algorithms to tackle different real-world biomedical problems; (3) evaluate the performance of the proposed transfer approaches against baselines used in this work in different problem domains; and (4) extend the techniques proposed here to analyze data available in National Cancer Institute's DREAM Drug Sensitivity Prediction Challenge [65,66].

Conflicts of interest

None declared.

Acknowledgements

This project was funded by the Deanship of Scientific Research (DSR) at King Abdulaziz University, Jeddah, under grant no. G-121-611-39. The authors, therefore, acknowledge with thanks DSR for technical and financial support.

References

[1] T. Turki, Z. Wei, J.T.L. Wang, Transfer learning approaches to improve drug sensitivity prediction in multiple myeloma patients, IEEE Access 5 (2017) 7381–7393.
[2] J. Ker, L. Wang, J. Rao, T. Lim, Deep learning applications in medical image analysis, IEEE Access 6 (2018) 9375–9389.
[3] M. Tan, Prediction of anti-cancer drug response by kernelized multi-task learning, Artif. Intell. Med. 73 (2016) 70–77.
[4] N.P. Singh, R.S. Bapi, P. Vinod, Machine learning models to predict the progression from early to late stages of papillary renal cell carcinoma, Comput. Biol. Med. 100 (2018) 92–99.
[5] Y. Chen, Y. Luo, W. Huang, D. Hu, R.-q. Zheng, S.-z. Cong, F.-k. Meng, H. Yang, H.-j. Lin, Y. Sun, Machine-learning-based classification of real-time tissue elastography for hepatic fibrosis in patients with chronic hepatitis B, Comput. Biol. Med. 89 (2017) 18–23.
[6] K.R. Bisaso, G.T. Anguzu, S.A. Karungi, A. Kiragga, B. Castelnuovo, A survey of machine learning applications in HIV clinical research and care, Comput. Biol. Med. 91 (2017) 366–371.
[7] P. Fergus, M. Selvaraj, C. Chalmers, Machine learning ensemble modelling to classify caesarean section and vaginal delivery types using Cardiotocography traces, Comput. Biol. Med. 93 (2018) 7–16.
[8] B. Gholami, T.S. Phan, W.M. Haddad, A. Cason, J. Mullis, L. Price, J.M. Bailey, Replicating human expertise of mechanical ventilation waveform analysis in detecting patient-ventilator cycling asynchrony using machine learning, Comput. Biol. Med. 97 (2018) 137–144.
[9] M. Mohri, A. Rostamizadeh, A. Talwalkar, Foundations of Machine Learning, MIT press, Cambridge, MA, 2012.
[10] T. Turki, Z. Wei, A link prediction approach to cancer drug sensitivity prediction, BMC Syst. Biol. 11 (2017) 94.
[11] P. Geeleher, N.J. Cox, R.S. Huang, Clinical drug response can be predicted using baseline gene expression levels and in vitro drug sensitivity in cell lines, Genome Biol. 15 (2014) R47-R47.
[12] B. Majumder, U. Baraneedharan, S. Thiyagarajan, P. Radhakrishnan, H. Narasimhan, M. Dhandapani, N. Brijwani, D.D. Pinto, A. Prasath, B.U. Shanthappa, Predicting clinical response to anticancer drugs using an ex vivo platform that captures tumour heterogeneity, Nat. Commun. 6 (2015).
[13] A. Basu, R. Mitra, H. Liu, S.L. Schreiber, P.A. Clemons, RWEN: response-weighted elastic net for prediction of chemosensitivity of cancer cell lines, Bioinformatics 1 (2018) 8.
[14] T. Turki, Z. Wei, Learning approaches to improve prediction of drug sensitivity in breast cancer patients, 2016 38th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC), 2016, pp. 3314–3320.
[15] M.P. Menden, F. Iorio, M. Garnett, U. McDermott, C.H. Benes, P.J. Ballester, J. Saez-Rodriguez, Machine learning prediction of cancer cell sensitivity to drugs based on genomic and chemical properties, Publ. Library Sci. (PLoS) One 8 (2013) e61318.
[16] K. Weiss, T.M. Khoshgoftaar, D. Wang, A survey of transfer learning, J. Big Data 3 (2016) 1–40.
[17] S.J. Pan, Q. Yang, A survey on transfer learning, IEEE Trans. Knowl. Data Eng. 22 (2010) 1345–1359.
[18] T. Turki, Z. Wei, J.T.L. Wang, A transfer learning approach via procrustes analysis and mean shift for cancer drug sensitivity prediction, J. Bioinf. Comput. Biol. 16 (2018) 1840014.
[19] W. Dai, Q. Yang, G.-R. Xue, Y. Yu, Boosting for transfer learning, Proceedings of the 24th International Conference on Machine Learning, ACM, 2007, pp. 193–200.
[20] B. Sun, J. Feng, K. Saenko, Correlation Alignment for Unsupervised Domain Adaptation, Domain Adaptation in Computer Vision Applications, Springer, 2017, pp. 153–171.
[21] J.C. Costello, L.M. Heiser, E. Georgii, M. Gonen, M.P. Menden, N.J. Wang, M. Bansal, M. Ammad-ud-din, P. Hintsanen, S.A. Khan, J.-P. Mpindi, O. Kallioniemi, A. Honkela, T. Aittokallio, K. Wennerberg, N.D. Community, J.J. Collins, D. Gallahan, D. Singer, J. Saez-Rodriguez, S. Kaski, J.W. Gray, G. Stolovitzky, A community effort to assess and improve drug sensitivity prediction algorithms, Nat. Biotechnol. 32 (2014) 1202–1212.
[22] Y. Freund, R.E. Schapire, A decision-theoretic generalization of on-line learning and an application to boosting, The Proceedings of the Second European Conference on

Computational Learning Theory, Springer, 1995, pp. 23–37.
[23] T. Turki, J.T.L. Wang, Reverse engineering gene regulatory networks using sampling and boosting techniques, in: P. Perner (Ed.), Machine Learning and Data Mining in Pattern Recognition: 13th International Conference, MLDM 2017, New York, NY, USA, July 15-20, 2017, Proceedings, Springer International Publishing, 2017, pp. 63–77.
[24] H. Joensuu, P.-L. Kellokumpu-Lehtinen, P. Bono, T. Alanko, V. Kataja, R. Asola, T. Utriainen, R. Kokko, A. Hemminki, M. Tarkkanen, T. Turpeenniemi-Hujanen, S. Jyrkkiö, M. Flander, L. Helle, S. Ingalsuo, K. Johansson, A.-S. Jääskeläinen, M. Pajunen, M. Rauhala, J. Kaleva-Kerola, T. Salminen, M. Leinonen, I. Elomaa, J. Isola, Adjuvant docetaxel or vinorelbine with or without trastuzumab for breast cancer, N. Engl. J. Med. 354 (2006) 809–820.
[25] A. Awada, J. Albanell, P.A. Canney, L.Y. Dirix, T. Gil, F. Cardoso, P. Gascon, M.J. Piccart, J. Baselga, Bortezomib/docetaxel combination therapy in patients with anthracycline-pretreated advanced/metastatic breast cancer: a phase I/II dose-escalation study, Br. J. Canc. 98 (2008).
[26] J.C. Chang, E.C. Wooten, A. Tsimelzon, S.G. Hilsenbeck, M.C. Gutierrez, R. Elledge, S. Mohsin, C.K. Osborne, G.C. Chamness, D.C. Allred, P. O'Connell, Gene expression profiling for the prediction of therapeutic response to docetaxel in patients with breast cancer, Lancet 362 (2003) 362–369.
[27] R. Edgar, M. Domrachev, A.E. Lash, Gene Expression Omnibus: NCBI gene expression and hybridization array data repository, Nucleic Acids Res. 30 (2002) 207–210.
[28] J.C. Chang, E.C. Wooten, A. Tsimelzon, S.G. Hilsenbeck, M.C. Gutierrez, Y.-L. Tham, M. Kalidas, R. Elledge, S. Mohsin, C.K. Osborne, G.C. Chamness, D.C. Allred, M.T. Lewis, H. Wong, P. O'Connell, Patterns of resistance and incomplete response to docetaxel by gene expression profiling in breast cancer patients, J. Clin. Oncol. 23 (2005) 1169–1177.
[29] A. Paramore, S. Frantz, Bortezomib, Nat. Rev. Drug Discov. 2 (2003) 611–612.
[30] C. Li, R. Li, J.R. Grandis, D.E. Johnson, Bortezomib induces apoptosis via Bim and Bik up-regulation and synergizes with cisplatin in the killing of head and neck squamous cell carcinoma cells, Mol. Canc. Therapeut. 7 (2008) 1647–1655.
[31] T.J. Lynch, D. Fenton, V. Hirsh, D. Bodkin, E.L. Middleman, A. Chiappori, B. Halmos, R. Favis, H. Liu, W.L. Trepicchio, A randomized phase 2 study of erlotinib alone and in combination with bortezomib in previously treated advanced non-small cell lung cancer, J. Thorac. Oncol. 4 (2009) 1002–1009.
[32] G. Mulligan, C. Mitsiades, B. Bryant, F. Zhan, W.J. Chng, S. Roels, E. Koenig, A. Fergus, Y. Huang, P. Richardson, W.L. Trepicchio, A. Broyl, P. Sonneveld, J.D. Shaughnessy, P. Leif Bergsagel, D. Schenkein, D.-L. Esseltine, A. Boral, K.C. Anderson, Gene expression profiling and correlation with outcome in clinical trials of the proteasome inhibitor bortezomib, Blood 109 (2007) 3177–3188.
[33] D.P. Silver, A.L. Richardson, A.C. Eklund, Z.C. Wang, Z. Szallasi, Q. Li, N. Juul, C.-O. Leong, D. Calogrias, A. Buraimoh, Efficacy of neoadjuvant Cisplatin in triple-negative breast cancer, J. Clin. Oncol. 28 (2010) 1145.
[34] N. Juul, Z. Szallasi, A.C. Eklund, Q. Li, R.A. Burrell, M. Gerlinger, V. Valero, E. Andreopoulou, F.J. Esteva, W.F. Symmans, Assessment of an RNA interference screen-derived mitotic and ceramide pathway metagene as a predictor of response to neoadjuvant paclitaxel for primary triple-negative breast cancer: a retrospective analysis of five clinical trials, Lancet Oncol. 11 (2010) 358–365.
[35] Y. Li, L. Zou, Q. Li, B. Haibe-Kains, R. Tian, Y. Li, C. Desmedt, C. Sotiriou, Z. Szallasi, J.D. Iglehart, Amplification of LAPTM4B and YWHAZ contributes to chemotherapy resistance and recurrence of breast cancer, Nat. Med. 16 (2010) 214.
[36] C.-C. Chang, C.-J. Lin, LIBSVM: a library for support vector machines, ACM Trans. Intell. Syst. Technol. (TIST) 2 (2011) 27.
[37] T. Chen, C. Guestrin, Xgboost: a scalable tree boosting system, Proceedings of the 22nd Acm Sigkdd International Conference on Knowledge Discovery and Data Mining, ACM, 2016, pp. 785–794.
[38] C. Cortes, M. Mohri, U. Syed, Deep Boosting, International Conference on Machine Learning, International Conference on Machine Learning, 2014, pp. 1179–1187.
[39] L. Breiman, Classification and Regression Trees, Routledge, 2017.
[40] N. Japkowicz, M. Shah, Evaluating Learning Algorithms: a Classification Perspective, Cambridge University Press, New York City, NY, 2011.
[41] T. Chen, T. He, M. Benesty, Xgboost: Extreme Gradient Boosting, (2015) R package version 0.3-3.
[42] D. Marcous, Y. Sandbank, Deepboost: Deep Boosting Ensemble Modeling, R Package Version 0.1.6.
[43] A. Bodor, I. Csabai, M.W. Mahoney, N. Solymosi, rCUR: an R package for CUR matrix decomposition, BioMed Central (BMC) Bioinf. 13 (2012) 103.
[44] L. Shang, Z. Lu, H. Li, Neural responding machine for short-text conversation, Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics (ACL), 2015, pp. 1577–1586.
[45] D. Brzezinski, J. Stefanowski, Reacting to different types of concept drift: the accuracy updated ensemble algorithm, IEEE Trans. Neural Netw. Learn. Syst. 25 (2014) 81–94.
[46] B. Calvo, G. Santafé Rodrigo, scmamp: Statistical comparison of multiple algorithms in multiple problems, R J. 8/1 (2016) Aug. 2016.
[47] S. García, A. Fernández, J. Luengo, F. Herrera, Advanced nonparametric tests for multiple comparisons in the design of experiments in computational intelligence and data mining: experimental analysis of power, Inf. Sci. 180 (2010) 2044–2064.
[48] J. Demšar, Statistical comparisons of classifiers over multiple data sets, J. Mach. Learn. Res. 7 (2006) 1–30.
[49] I. Rodríguez-Fdez, A. Canosa, M. Mucientes, A. Bugarín, STAC: a web platform for the comparison of algorithms using statistical tests, Fuzzy Systems (FUZZ-IEEE), 2015 IEEE International Conference on, IEEE, 2015, pp. 1–8.
[50] D.C. Howell, Fundamental Statistics for the Behavioral Sciences, Wadsworth Cengage Learning, 2010.
[51] H. Liu, Y. Zhao, L. Zhang, X. Chen, Anti-cancer drug response prediction using neighbor-based collaborative filtering with global effect removal, Mol. Ther. Nucleic Acids 13 (2018) 303–311.
[52] R. Rahman, K. Matlock, S. Ghosh, R. Pal, Heterogeneity aware random forest for drug sensitivity prediction, Sci. Rep. 7 (2017) 11347.
[53] A. Honkela, M. Das, A. Nieminen, O. Dikmen, S. Kaski, Efficient differentially private learning improves drug sensitivity prediction, Biol. Direct 13 (2018) 1.
[54] C. Suphavilai, D. Bertrand, N. Nagarajan, Predicting cancer drug response using a recommender system, Bioinformatics 34 (22) (2018) 3907–3914.
[55] J. Barretina, G. Caponigro, N. Stransky, K. Venkatesan, A.A. Margolin, S. Kim, C.J. Wilson, J. Lehár, G.V. Kryukov, D. Sonkin, The Cancer Cell Line Encyclopedia enables predictive modelling of anticancer drug sensitivity, Nature 483 (2012) 603.
[56] A. Kamb, S. Wee, C. Lengauer, Why is cancer drug discovery so difficult? Nat. Rev. Drug Discov. 6 (2007) 115–120.
[57] F. Iorio, T.A. Knijnenburg, D.J. Vis, G.R. Bignell, M.P. Menden, M. Schubert, N. Aben, E. Gonçalves, S. Barthorpe, H. Lightfoot, A landscape of pharmacogenomic interactions in cancer, Cell 166 (2016) 740–754.
[58] R. Rahman, J. Otridge, R. Pal, IntegratedMRF: random forest-based framework for integrating prediction from different data types, Bioinformatics 33 (2017) 1407–1410.
[59] Q. Wan, R. Pal, An ensemble based top performing approach for NCI-DREAM drug sensitivity prediction challenge, PLoS One 9 (2014) e101183.
[60] Y. Chang, H. Park, H.-J. Yang, S. Lee, K.-Y. Lee, T.S. Kim, J. Jung, J.-M. Shin, Cancer drug response profile scan (CDRscan): a deep learning model that predicts drug effectiveness from cancer genomic signature, Sci. Rep. 8 (2018) 8857.
[61] K. Matlock, C. De Niz, R. Rahman, S. Ghosh, R. Pal, Investigation of model stacking for drug sensitivity prediction, BMC Bioinf. 19 (2018) 71.
[62] S. Sun, H. Shi, Y. Wu, A survey of multi-source domain adaptation, Inf. Fusion 24 (2015) 84–92.
[63] Y. Mansour, M. Mohri, A. Rostamizadeh, Domain adaptation with multiple sources, Adv. Neural Inf. Process. Syst. (2009) 1041–1048.
[64] Y. Mansour, M. Mohri, A. Rostamizadeh, Domain adaptation: learning bounds and algorithms, 22nd Conference on Learning Theory, COLT, 2009.
[65] Y. Liu, F.W. Crawford, Estimating dose-specific cell division and apoptosis rates from chemo-sensitivity experiments, Sci. Rep. 8 (2018) 2705.
[66] N. Fujita, S. Mizuarai, K. Murakami, K. Nakai, Biomarker discovery by integrated joint non-negative matrix factorization and pathway signature analyses, Sci. Rep. 8 (2018) 9743.

Turki Turki received the B.S. degree in computer science from King Abdulaziz University, the M.S. degree in computer science from NYU.POLY, and the Ph.D. degree in computer science from the New Jersey Institute of Technology. He is currently an assistant professor with the Department of Computer Science, King Abdulaziz University, Saudi Arabia. His research interests include algorithms, machine learning, data mining, big data analytics, sustainable computing, health informatics, bioinformatics, computational biology, and social networks. His works have been published in journals such as BMC Genomics, BMC Systems Biology, IEEE Access, BioMed Research International, Journal of Bioinformatics and Computational Biology, Computers in Biology and Medicine, and Genes. He was presented with the distinction award from the deanship of scientific research at the King Abdulaziz University. He is supported by King Abdulaziz University and is currently working on several biomedicine related projects. Dr. Turki has served on the program committees of several international conferences. Also, he is an editorial board member of Computers in Biology and Medicine and Sustainable Computing: Informatics and Systems.

Jason T. L. Wang received the Ph.D. degree in computer science from the Courant Institute of Mathematical Sciences at New York University. He is a Professor of Computer Science at New Jersey Institute of Technology, and Director of the University's Data and Knowledge Engineering Laboratory. Dr. Wang's research interests include data science and computational biomedicine. He has published 9 books and over 150 refereed journal and conference papers in these areas. Dr. Wang has served on the program committees of over 200 national and international conferences, and on the editorial boards of several journals including ACM Transactions on Knowledge Discovery from Data (TKDD).