mClass: Cancer Type Classification with Somatic Point Mutation Data

Md Abid Hasan(✉) and Stefano Lonardi(✉)

University of California Riverside, 900 University Ave, Riverside, CA 92507, USA
mhasa006@ucr.edu, stelo@cs.ucr.edu

Abstract. Cancer is a complex disease associated with abnormal DNA mutations. Not all tumors are cancerous and not all cancers are the same. Correct cancer type diagnosis can indicate the most effective drug therapy and increase survival rate. At the molecular level, it has been shown that cancer type classification can be carried out from the analysis of somatic point mutation. However, the high dimensionality and sparsity of genomic mutation data, coupled with its small sample size has been a hindrance in accurate classification of cancer. We address these problems by introducing a novel classification method called mClass that accounts for the sparsity of the data. mClass is a feature selection method that ranks genes based on their similarity across samples and employs their normalized mutual information to determine the set of genes that provide optimal classification accuracy. Experimental results on TCGA datasets show that mClass significantly improves testing accuracy compared to DeepGene, which is the state-of-the-art in cancer-type classification based on somatic mutation data. In addition, when compared with other cancer gene prediction tools, the set of genes selected by mClass contains the highest number of genes in top 100 genes listed in the Cancer Gene Census. mClass is available at https://github.com/mdahasan/mClass.

Keywords: Cancer classification · Somatic point mutation
Genetic variation

1 Introduction

Cancer is a complex disease that results from an accumulation of DNA mutations and epigenetic modifications in somatic cells. Remarkable scientific progress has shed light on almost every biological aspect of this disease. Despite this progress, cancer is still one of the most challenging disease of our time with an increasing numbers of new cases and resulting in 14.6% of all human death each year [1]. Not all tumors are cancerous and not all cancers are the same. There is no single test that can diagnose cancer type with perfect accuracy. The diagnosis process requires careful examination and extensive testing to determine whether a person has cancer and which type. Traditional cancer diagnosis method involves lab tests, genetic tests, tumor biopsies, etc. The effective differentiation of cancers

M. Blanchette and A. Ouangraoua (Eds.): RECOMB-CG 2018, LNBI 11183, pp. 131–145, 2018.
https://doi.org/10.1007/978-3-030-00834-5_7

with similar histopathological appearance can indicate the most effective drug treatment and increase survival rates (see, e.g., [2,6,8,9]).

Technological advancements in sequencing technologies has resulted in a dramatic increase in the quantity and quality of sequencing data related to cancer, now available in databases such as The Cancer Genome Atlas [4] and the International Cancer Genome Consortium [3]. These vast repositories provide genomic data from thousands of patients across different cancer subtypes [5]. The abundance of this data has enabled researchers to devise new statistical approaches for the accurate identification of cancer types and subtypes. Cancer classification methods use gene expression data and/or somatic point mutation such as copy number variation, translocations and small insertions and deletions. Several methods have been proposed to accurately predict cancer types and subtypes (see, e.g., [2,11–13]). The classification of cancer based on the somatic point mutation data can be challenging because of the high dimensionality and sparsity of the data. In cancer patients only a few genes are mutated with high frequency, while most of the genes have a low rate of mutation [10].

The literature on cancer classification methods is extensive. For instance, in [7] the authors proposed a pan-cancer classification method based on gene expression data. They used over nine thousand samples for 31 cancer types to train a method in which a genetic algorithm carries out the gene selection and a nearest neighbor method is used as a classifier.

The authors of [23] proposed to find discriminatory gene sets by measuring the relevance of individual genes using mean and standard deviation of each sample to the class centroid. In [24] the authors introduced new scoring functions to design a stable gene selection method. Their method scores genes based on the assumption that discriminatory genes have different mean values across different classes, small intra-class variation and relatively large inter-class variation.

The authors of [14] combined the clustering gene selection with statistical tests such as T-test and F-test and the gene selection method proposed in [23] to deal the high dimensionality in gene expression data. Genes are assigned to clusters if they are close to the centroids after applying k-means clustering.

In [2], the authors proposed a deep neural network for the classification of multiple cancer types from somatic point mutation data, called DeepGene. To the best of our knowledge, DeepGene is the state-of-the-art for multiple cancer classifications using somatic point mutation data. DeepGene clusters genes based on mutation occurrence and uses a sparse representation to index non-zero elements. The data is then fed into a fully connected deep neural network that learns specific cancer types.

In this paper, we address the shortcomings of existing methods dealing with the sparsity and high-dimensionality of somatic point mutation data by proposing an efficient feature selection method based on information theory. A logistic regression model demonstrates the effectiveness of our approach for cancer type classification. Although in a medical setting the task of predicting cancer type from somatic point mutation data might not be practical, here we investigate the fundamental question on whether somatic point mutation data has sufficient discriminative power to allow for cancer type classification.

2 Methods

Given m individuals affected by cancer, the input to our feature selection method is composed of the class labels, i.e., the cancer type for the m individuals, and the mutation frequency of all genes for the m individuals. Selected features are then fed into a classifier as described below.

Let n be the number of human genes for which somatic point mutation data is available. Let $C \in \{1 \ldots l\}^m$ be the vector containing the class labels where l is the number of cancer types, and let $G \in \{0 \ldots k\}^{m \times n}, k \in \mathbb{N}$ be the matrix representing the number of mutations observed in each gene (i.e., $G(i,j) = k$ if gene i has k mutations in sample j).

The significance of a gene being involved in a particular type of cancer depend on its mutation frequency. Genes with higher mutations are expected to be more relevant for the causation of cancer [16]. In our method, we disregard genes that contain less than $t\%$ mutations across all samples. This filtering step removes non-significant genes from further consideration thus reduce the adverse impact of the data sparsity. Our feature selection model has two steps. First, we cluster genes based on their pairwise similarity. Then, we rank genes using a normalized mutual information criterion [15].

2.1 Gene Clustering

Grouping similar genes into clusters allows our method to identify and eliminate redundant genes within a cluster without compromising the efficiency of the feature selection. The reduction of data also reduces the complexity of downstream steps. Since G is a sparse matrix, we use the cosine similarity because of its good mathematical properties on sparse vectors. Given two n-dimensional vectors X and Y the cosine similarity is defined as

$$s(X,Y) = \frac{\sum_{i=1}^{n} X_i Y_i}{\sqrt{\sum_{i=1}^{n} X_i^2}\sqrt{\sum_{i=1}^{n} Y_i^2}}$$

where X_i and Y_i are the i-th components of vector X and Y. Gene p is assigned to the cluster of gene q if the cosine similarity between row vectors $G[:,p]$ and $G[:,q]$ is higher than a predefined threshold e. According to this procedure, it is possible that the same gene could end up in multiple clusters. To select unique genes out of these clusters, we rank the genes based on mutation count and mutual information with the class label within the cluster as described next.

2.2 Normalized Mutual Information

Our gene selection method relies on an information theoretic measure that evaluates the predictive ability of each gene. Let X be a discrete random variable where each event $x \in X$ occurs with probability $p(x)$. The *entropy* $H(X)$ of variable X is the sum of the information content of each discrete event weighted by the individual event probability, that is $H(X) = -\sum_{x \in X} p(x) \log_2 p(x)$.

Given two discrete random variables X and Y with joint probability $p(x, y)$ and marginal probabilities $p(x)$ and $p(y)$, the *conditional entropy* of variable Y conditioned on variable X is defined as $H(Y|X) = \sum_{x \in X, y \in Y} p(x, y) \log_2(p(x)/p(x, y))$. Similarly, $H(X|Y) = \sum_{x \in X, y \in Y} p(x, y) \log_2(p(y)/p(x, y))$. We have that $H(Y|X) = H(Y)$ iff X and Y are independent random variables. The *mutual information* $I(X, Y)$ is the gain of information about random variable X due to additional information from random variable Y, that is

$$I(X, Y) = H(X) - H(X|Y) = \sum_{x \in X} \sum_{y \in Y} p(x, y) \log_2 \left(\frac{p(x, y)}{p(x)p(y)} \right)$$

Given a set F of features (the set of genes in G in this case) and class variables C, the feature selection based on mutual information finds a subset $S \subset F$ such that the mutual information $I(C, S)$ is maximized. In order to achieve that goal we use the Normalized Mutual Information based Feature Selection (NMIFS) technique. NMIFS is a heuristic algorithm that selects one feature at a time. NMIFS differs from other mutual information based feature selection technique such as MIFS [17], MIFS-U [18] and mRMR [19] in that it does not depend on the parameter used to control the redundancy penalization. Also NMIFS does not assume that the random variables have uniform probability distribution.

Given features $f_i \in F - S$ and $f_s \in S$ we express the mutual information as

$$I(f_i, f_s) = H(f_i) - H(f_i|f_s) = H(f_s) - H(f_s|f_i) \tag{1}$$

where $H(f_i)$ and $H(f_s)$ are the entropies and $H(f_i|f_s)$ and $H(f_s|f_i)$ are conditional entropies.

The mutual information $I(f_i, f_s)$ is non-negative, and attains its maximum at $\min\{H(f_i), H(f_s)\}$. We can define the normalized mutual information between f_i and f_s as

$$\text{norm}I(f_i, f_s) = \frac{I(f_i, f_s)}{\min\{H(f_i), H(f_s)\}} \tag{2}$$

The *average normalized mutual information* is a measure of redundancy between f_i and $f_s \in S$ for $s = 1, \ldots, |S|$ and it defined as

$$\frac{1}{|S|} \sum_{f_s \in S} \text{norm}I(f_i, f_s)$$

where $|S|$ is the cardinality of subset S. Our gene selection criterion selects a gene $f_i \in F - S$ that maximizes

$$J(C, f_i) = I(C, f_i) - \frac{1}{|S|} \sum_{f_s \in S} \text{norm}I(f_i, f_s) \tag{3}$$

where $I(C, f_i)$ is the mutual information between feature f_i and class variable C.

2.3 Feature Selection

A sketch of mClass' algorithm is shown as Algorithm 1. The algorithm first determines the number of mutations of each gene from the input matrix G. Then it computes the cosine similarity between all pairs of genes that have a mutation percentage across all sample of at least $t\%$. Genes are assigned to the same cluster when their similarity exceeds threshold e. The process assigns each gene to one or more clusters. The top v genes from each clusters are selected into a representative list R'.

Next, mClass collects the unique set of genes U from the representative set R'. It then calculates the mutual information between all features/genes $f_i \in U$ and the class variable C. To calculate Eqs. (2) and (3) mClass discretizes the gene mutation values into d equal-width bins. The gene $\hat{f_i}$ which has the maximum mutual information with the class variable C is selected as the first feature in S (S is the final set of ranked genes). That gene is then removed from U. For all the other genes in U mClass first calculates the normalized mutual information between all pair of genes in U and S using Eq. (2). A gene $f_i \in U$ is selected when it maximizes Eq. (3). The gene is then added to S and removed from U. This process is repeated until all genes are given a rank in the ordered set S. Instead of deciding on a predefined number of features *a priori* to be used in the classifier, we select a variable number of genes in S based on their ability to classify the data.

2.4 Cancer Type Classifier

As said, we employ a logistic regression (LG) multi-class classifier for a given number of genes in the ranked set S. The linear model describes the probabilities describing the possible outcome of a single trial using logistic function. Here we use a One-vs-Rest (OvR) for the multi-class classification implementation with L_2 regularization. For the binary case, the L_2-regularized logistic regression optimizes the following cost function

$$\text{minimize}_w \sum_{x,y} \log(1 + \exp(-w^T x.y)) + \lambda w^T w) \tag{4}$$

The objective is to find the feature weights (w) that minimizes the cost function in Eq. (4). Here x is the feature vector (genes) and y is the class label. The hyper-parameter λ used to control the strength of regularization was left as the default value (as defined by `scikit-learn`). As said, the classifier is fed the genes in S incrementally. To determine the final set of features we select genes based on their ability to accurately classify the dataset. The model decomposes the optimization problem in Eq. (4) in a OvR fashion so that the binary classifier can be trained on all classes.

3 Experimental Results

In this section, we describe the experimental setup, i.e., datasets and the parameters used in the feature selection and classification, as well as other implemen-

```
Data: Gene mutation data $G \in \{0, k\}^{m \times n}$, similarity measure threshold $e$,
      mutation count threshold $t$, discretization value $d$, $v$, class variable $C$
Result: Ordered set of genes $S$
set $R \leftarrow \emptyset$;
for each gene $f_i \in G$ do
    if number of mutation of $f_i > t$ then
        $R \leftarrow R \cup \{f_i\}$;
    end
end
set $CL \leftarrow \emptyset$;
for each gene $f_i \in R$ do
    create a new cluster in $CL$ for $f_i$;
    for each gene $f_j \in R$, $j \neq i$ do
        if cosine similarity $s(f_i, f_j) > e$ then
            assign $f_i$ and $f_j$ to same cluster in $CL$
        end
    end
end
set $R' \leftarrow \emptyset$;
for each cluster $cl \in CL$ do
    set $R' \leftarrow R' \cup \{$top $v$ genes in $cl\}$
end
collect unique genes $U \leftarrow set(R')$;
discretize gene mutation values in $d$ equal-width bins;
select the first feature $\hat{f}_i = \text{argmax}_{f_i \in U}\{I(C; f_i)\}$ ;
set $U \leftarrow U - \{\hat{f}_i\}$;
set $S \leftarrow \{\hat{f}_i\}$;
for each gene $f_i$ in $U$ do
    calculate $I(f_i; f_s)$ for all pairs $(f_i, f_s)$ with $f_i \in U$ and $f_s \in S$;
    select feature $f_i \in U$ that maximizes $J$ in Equation (3);
    set $U \leftarrow U - \{f_i\}$;
    set $S \leftarrow S \cup \{f_i\}$;
end
return ordered set $S$;
```

Algorithm 1. mClass feature selection algorithm

tation details. Data preprocessing, feature selection and classification evaluation steps were implemented in Python. All tested classifiers are available from the Python package `scikit-learn`.

3.1 Datasets

We used two cancer datasets to test mClass. The first dataset is a twelve-type cancer dataset from The Cancer Genome Atlas (TCGA) [4]. The dataset was assembled by selecting the genes across all samples for all cancer types that contain mutations. Table 1 shows the basic statistics of each cancer type. Observe that the number of samples and the number of mutations varies significantly

Table 1. Sample and mutation statistics for the twelve-type cancer dataset

Cancer type	Number of samples	Number of mutations
ACC	90	18,272
BLCA	130	37,948
BRCA	982	83,360
CESC	194	45,293
HNSC	279	49,264
KIRP	161	13,640
LGG	286	9,228
LUAD	230	68,270
PAAD	150	30,123
PRAD	332	11,802
STAD	289	130,050
UCS	57	10,129
Total	3,180	507,379

across cancer types. After removing samples that have less than five mutations across all genes, the dataset contained 3,151 samples and 23,236 genes. The second dataset from TCGA contains four cancer types, namely COAD, SKCM, LAML and KIRC. It contains 1,043 samples with a total of 363,285 mutations across 25,286 genes. Details about this dataset and the corresponding experimental results are discussed in Sect. 3.5.

3.2 Parameters

mClass' feature selection uses four parameters: the similarity measure threshold e for the clustering step, the minimum mutation count threshold t to eliminate non-informative genes, the number v of top genes selected from each cluster and the number of bins d used for discretizing gene mutation values (see Algorithm 1).

In our experiments, parameter t was set to 1 which has the effect of disregarding genes with less that 1% mutation across the samples. As said, the pairwise gene similarity is calculated using the cosine similarity measure and genes are assigned into same cluster if the similarity between them is greater than the similarity threshold e. The algorithm then selects the top v% genes from each cluster for gene ranking step. The values for e, t, v and d were selected experimentally based on ability of the method to accurately classify the datasets using the selected number of features. For instance, Table 2 shows the classification accuracy of mClass+LG (mClass's feature selection followed by logistic regression) on the twelve-type cancer dataset, for various choices of e. Based on this analysis, we selected $e = 0.55$. Similarly, we tested the values of v in the range 5%–25%, and we obtained the highest classification accuracy with $v = 10\%$.

Table 2. Classification accuracy of mClass+LG as a function of similarity threshold e on the twelve-type cancer dataset

Similarity threshold (e)	Classification Accuracy
0.50	0.708
0.55	**0.718**
0.60	0.715
0.65	0.715
0.70	0.715
0.75	0.715

Table 3. Ten-fold cross validation accuracy for mClass+LG and DeepGene (three configurations) on the twelve-type cancer dataset

Method	Cross-validation Accuracy
DeepGene (CGF + ISR)	0.655
DeepGene (CGF)	0.638
DeepGene (ISR)	0.649
mClass+LG	**0.675**

A similar experimental analysis (not shown) indicated that $d = 5$ was the optimal choice for these datasets. Incidentally, the same value of d was used in [22].

3.3 Evaluation Metrics and Comparison with DeepGene

We have used the evaluation metrics introduced in [2] to compare the results. All evaluation experiments were performed by randomly selecting 90% of the input data as training data and 10% of the input as testing data. We compared the ten-fold cross validation accuracy of mClass+LG (mClass's feature selection followed by logistic regression) and testing accuracy against state-of-the-art DeepGene [2].

As said, mClass selects the optimal number of features in a forward selection fashion. We compared mClass' cross-validation results with DeepGene, which employs a convolutional neural network (CNN) as the classifier. The performance of DeepGene was calculated in three different configuration: clustered gene filter and indexed sparsity reduction, only cluster gene filter and only indexed sparsity reduction.

The ten-fold cross-validation results between mClass and three configuration of DeepGene on the twelve-type cancer dataset is shown in Table 3. Observe that the classification accuracy of mClass outperformed all three configurations of DeepGene proposed in [2]. The classification accuracy of mClass is more than 3% higher than the best configuration of DeepGene.

We also compared the testing accuracy of mClass with (i) the best configuration of DeepGene and (ii) LG on the full dataset (i.e., no feature selection).

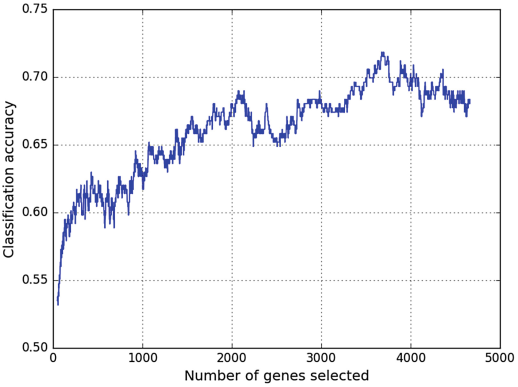

Fig. 1. Classification accuracies as a function of the number of feature (genes) selected

Table 4. Testing accuracies of mClass+LG, DeepGene and LG on full dataset (twelve-type cancer dataset)

Method	Classification accuracy
Full dataset (no feature selection)	0.677
DeepGene (CGF+ISR)	0.655
mClass+LG	**0.718**

The logistic regression classifier in mClass uses balanced weights to counter the imbalance in the number of samples in the dataset. Using the forward feature selection technique described in Algorithm 1, the testing accuracy of the classifier was measured by adding ranked gene one at a time. Figure 1 shows the progression of forward feature selection. mClass obtains the best testing accuracy $(TP + TN)/(TP + TN + FP + FN)$ of 0.718 using a collection of top 3,676 genes which is 9.6% higher than the accuracy obtained by the best configuration of DeepGene with an average precision $TP/(TP + FP)$ of 0.74, recall $TP+(TP+FN)$ of 0.718 and F-Score $(2 \times precision \times recall)/(precision + recall)$ of 0.711 as shown in Table 5. Figure 2 illustrates the confusion matrix for the twelve-type cancer dataset. Observe that with mClass + LG, false positives rate is highest for BRCA while BLCA has the highest rate of false negatives. Table 4 summarizes the testing accuracy of these three methods.

3.4 Testing Other Classifiers

As said, mClass+LG uses a logistic regression as the classifier for the cancer classification datasets. We have tested the classification accuracies of other classifiers following mClass' feature selection. We employed Support Vector Machine (SVM) both with the linear and RBF kernel, k-nearest neighbor (KNN), Naive Bayes and Random Forest. All the classifiers were available from the Python package `scikit-learn`.

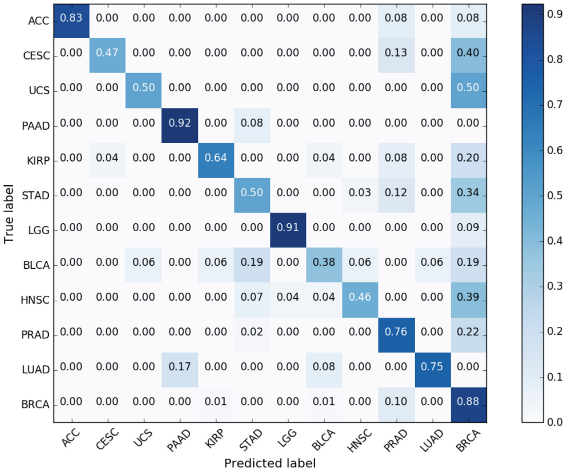

Fig. 2. Normalized confusion matrix for the twelve-type cancer dataset

Table 5. Classification results on twelve-type cancer dataset

Cancer type	Precision	Recall	F-Score	Support
ACC	1.00	0.83	0.91	12
CESC	0.88	0.47	0.61	15
UCS	0.33	0.50	0.40	2
PAAD	0.80	0.92	0.86	13
KIRP	0.89	0.64	0.74	25
STAD	0.70	0.50	0.58	32
LGG	0.95	0.91	0.93	23
BLCA	0.67	0.38	0.48	16
HNSC	0.81	0.46	0.59	28
PRAD	0.68	0.78	0.73	50
LUAD	0.90	0.75	0.82	12
BRCA	0.62	0.88	0.71	88
Average/Total	0.74	0.72	0.71	316

To classify the data using SVM with the RBF kernel, we optimized the parameter C and γ using 10-fold cross validation (keeping other parameters to default). The highest accuracy was obtained with $C = 2e^2$ and $\gamma = 2e^{-5}$. We have used the same parameter C for the linear kernel version of the SVM. The classification with KNN employed Euclidean distance and Pearson correlation coefficient. The 10-fold cross validation showed an optimal accuracy of 0.316 for Euclidean distance using a threshold of 3 and an accuracy of 0.436 with the Pearson correlation coefficient using a neighborhood size of 4. The ensemble

Random Forest classifier's employed a maximum of 1,000 trees in the forest. We set the minimum number of samples required to split an internal node to 9. All other parameters were set to default.

The performance of the various classifier is shown in Fig. 3. The experimental results show a significant advantage of LG over all other classifiers. mClass+LG achieves (i) a 9.6% testing classification improvement over the best configuration of DeepGene (ii) a 24.6% improvement over the linear kernel SVM, (iii) a 29.6% improvement over the RBF kernel SVM, (iv) a 106.9% improvement over KNN with Euclidean distance, (v) a 64.6% improvement over the KNN with Pearson correlation coefficient, (vi) a 83.6% improvement over Naive Bayes and (vii) a 30.3% improvement over Random Forest.

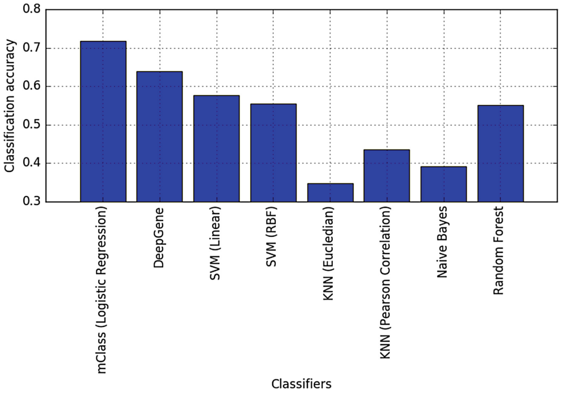

Fig. 3. Classification accuracy of mClass+LG, DeepGene and other classifiers applied to the features selected by mClass

3.5 Experimental Results on the Four-Type Dataset

As mentioned above, we used a second dataset consisting four type of cancers, namely COAD, SKCM, LAML and KIRC. After removing genes with less than 1% mutations across all samples, the dimension of the dataset was reduced to 1043 × 25286. The dataset contains 154 samples for COAD, 345 samples for SKCM, 158 samples for LAML and 386 samples for KIRC. Total number of mutations in this dataset is 363,285. We used the same parameter values for e, t, v and d as in the previous experiment. The 10-fold cross-validation peaked with an accuracy score of 89.5% with 1,132 genes. For testing accuracy, the dataset was divided into training and testing dataset of size 698 (67%) and 345 (33%), respectively. Using 1,132 features, mClass+LG achieves an accuracy of 87.5% on this dataset. Table 6 shows the average precision and f1-score for each class in this dataset. Figure 4 shows the normalized confusion matrix for our classifier. We could not compare the performance of mClass with DeepGene on

Table 6. Testing accuracies on the four-type cancer dataset using mClass

Cancer Type	Precision	Recall	F-score	Support
COAD	0.93	0.75	0.83	51
SKCM	0.97	0.88	0.92	101
LAML	0.65	0.98	0.79	56
KIRC	0.94	0.88	0.91	137
Avg/Total	0.90	0.87	0.88	345

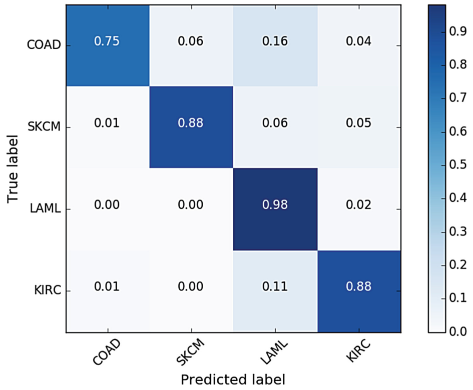

Fig. 4. Normalized confusion matrix for the four-type cancer dataset

this second dataset because, according to the authors, the data pre-processing code necessary to feed the training model for DNN is not available anymore.

3.6 Comparisons of Predicted Genes

We compared the genes selected by mClass+LG using the 12-types dataset with genes from Cancer Gene Census (CGC). At the time of writing the CGC database contains 719 genes. About 90% of these genes contain somatic mutations, 20% contain germline mutation and 10% contain both types of mutations. We compared mClass' selected genes against the selection carried out by Mutsig 2.0, Mutsig CV [20], MutationAccessor [21] and Muffin [16]. These latter methods predicts cancer genes by analyzing cancer somatic mutation data from 18 types of cancer. We examined the top 100, 500 and 1000 genes produced by these methods, and counted how many of these genes were annotated in the CGC database.

Figure 5 shows these counts for mClass, Mutsig 2.0, Mutsig CV, MutationAccessor and Muffin. Observe that for the top 100 genes, mClass identifies about 50% more CGC genes than MutSig 2.0, MutSig CV and MutationAccessor. mClass identifies more CGC genes than Mutsig 2.0, Mutsig CV and MutationAccessor for the 500 and 1000 case. However, mClass falls short by 18% and 14%

than Muffin in identifying CGC genes in top 500 and top 1000 genes. Although the purpose of mClass was not identifying driver genes, it is remarkable that the top ranked genes selected by mClass contains a large proportion of cancer driver genes.

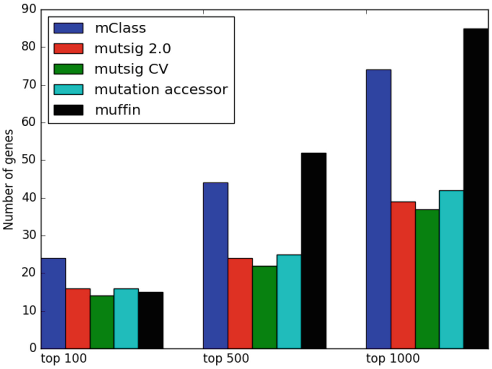

Fig. 5. Number of CGC genes produced by mClass, Mutsig 2.0, Mutsig CV, MutationAccessor and Muffin in their top 100, 500 and 1000 selection

4 Conclusions

In this paper we proposed a gene selection method based on clustering and normalized mutual information to rank genes for multiple cancer classification using somatic point mutation data. A logistic regression classifier in an one-vs-rest configuration is applied for multiple cancer classification using the selected genes. Experimental results on two TCGA datasets shows significant improvements in classification accuracy. We also showed that our feature selection method ranked genes that match CGC-annotated genes. Moreover, the model can be extended by including other genomic data that could further improve the overall classification performance. For instance, one could use mutation signature associated with specific cancer types to improve the overall accuracy.

Funding. This work was supported in part by the US National Science Foundation [IOS-1543963, IIS-1526742].

References

1. Stewart, B., Wild, C.P.: World Cancer Report 2014. World Health Organization (2015)
2. Yuchen, Y., Yi, S., et al.: DeepGene: an advanced cancer type classifier based on deep learning and somatic point mutations. BMC Bioinf. **17**(Suppl 17), 243–303 (2016)

3. Zhang, J., Baran, J., et al.: International cancer genome consortium data portal - a one-stop shop for cancer genomics data. Database: J. Biol. Databases Curation **2011**, bar026 (2015). https://doi.org/10.1093/database/bar026
4. Tomczak, K., Czerwinska, P., Wiznerowicz, M.: The cancer genome atlas (TCGA): an immeasurable source of knowledge. Contemp. Oncol. **19**(1A), A68–A77 (2011)
5. Amar, D., Izraeli, S., et al.: Utilizing somatic mutation data from numerous studies for cancer research: proof of concept and applications. Oncogene **36**, 3375–3383 (2017)
6. Golub, T.R., Slonim, D.K., et al.: Molecular classification of cancer: class discovery and class prediction by gene expression monitoring. Science **286**(5439), 531–537 (1999)
7. Yuanyuan, L., Kang, K., et al.: A comprehensive genomic pan-cancer classification using The Cancer Genome Atlas gene expression data. BMC Genomics **18**, 508 (2017)
8. Long, D.L.: Tumor heterogeneity and personalized medicine. New Engl. J. Med. **366**(10), 956–957 (2012)
9. Gudeman, J., Jozwiakowski, M., Chollet, J., Randell, M.: Potential risks of pharmacy compounding. Drugs R D **13**(1), 1–8 (2013)
10. Michael, S.L., Stojanov, P., Craig, H.M.: Discovery and saturation analysis of cancer genes across 21 tumor types. Nature **505**, 495–501 (2014)
11. Browne, R.P., McNicholas, P.D., Sparling, M.D.: Model-based learning using a mixture of mixtures of gaussian and uniform distributions. IEEE Trans. Pattern Anal. Mach. Intell. **34**(4), 814–817 (2012)
12. Chicco, D., Sadowski, P., Baldi, P.: Deep autoencoder neural networks for gene ontology annotation prediction. In: Proceedings of ACM Conference in Bioinformatics and Computational Biology, 2014, pp. 533–540. Newport Beach: Health Informatics (2014)
13. Chow, C.K., Zhu, H., Lacy, J., et al.: A cooperative feature gene extraction algorithm that combines classification and clustering. In: IEEE International Conference on Bioinformatics and Biomedicine Workshop (BIBMW), pp. 197–202 (2009)
14. Cai, Z., Xu, L., Shi, Y., et al.: Using gene clustering to identify discriminatory genes with higher classification accuracy. In: IEEE Symposium on Bioinformatics and BioEngineering (BIBE), pp. 235–242. Arlington (2006)
15. Pablo, A.E., Michel, T., Claudio, A.P., Jacek, M.Z.: Normalized mutual information feature selection. IEEE Trans. Neural Netw. **20**(2), 189–201 (2009)
16. Cho, A., Shim, J.E., Kim, E., et al.: MUFFIN: cancer gene discovery via network analysis of somatic mutation data. Genome Biol. **17**, 129 (2016)
17. Battiti, R.: Using mutual information for selection features in supervised neural network. IEEE Trans. Neural Netw. **5**(4), 537–550 (1994)
18. Kwak, N., Choi, C.H.: Input feature selection for classification problems. IEEE Trans. Neural Netw. **13**(1), 143–159 (2002)
19. Peng, H., Long, F., Ding, C.: Feature selection based on mutual information: criteria of max-dependency, max-relevance and min-redundancy. IEEE Trans. Pattern Anal. Mach. Intell. **27**(8), 1226–1238 (2005)
20. Lawrence, M.S., Stojanov, P., Polak, P., et al.: Mutational heterogeneity in cancer and the search for new cancer-associated genes. Nature **499**(7454), 214–218 (2013)
21. Reva, B., Antipin, Y., Sander, C.: Predicting the functional impact of protein mutations: application to cancer genomics. Nucleic Acids Res. **39**(17), e118 (2011)
22. Carter, H.: Cancer-specific high-throughput annotation of somatic mutations: computational prediction of driver missense mutations. Cancer Res. **69**(176), 6660–6667 (2009)

23. Ji-Hoon, C., Dongkwon, L., Jin Hyun, P., In-Beum, L.: New gene selection method for classification of cancer subtypes considering within-class variation. FEBS Lett. **551**, 3–7 (2003)
24. Kun, Y., Zhipeng, C., Jianzhong, L., Gouhui, L.: A stable gene selection in microarray data analysis. BMC Bioinf. **7**, 228 (2006)