USING GENE EXPRESSION DATA TO PREDICT CLINICAL INFORMATION IN SEVEN HUMAN CANCERS

Nathan Abell
Department of Genetics
Stanford University School of Medicine

Abstract

The expansive heterogeneity of known cancers share one key property - genetic and transcriptomic abnormalities. Discovering and quantifying the landscape of genomic abnormalities is a long-standing goal of the cancer research community. To this end, the Genomic Data Commons (GDC) has aggregated and standardized tens of thousands of experimental datasets from dozens of human cancers. Here, we describe our efforts to construct a pipeline for the prediction of a specific clinical feature (including a range of quantitative and categorical responses) on all available samples for a given cancer type. We then apply this pipeline to a set of features in seven human tissues, and obtain very accurate classifiers for categorical clinical features, with comparably poor performance for quantitative features.

INTRODUCTION AND BACKGROUND

Cancer, beneath its heterogeneity, is a fundamentally genetic disease. In many contexts, cancers arise from genetic disregulation of cell growth circuits due to somatic mutation, which can have downstream effects on many processes including gene expression. [1, 2] Accordingly, significant efforts have been devoted to identifying predictive gene expression signatures of human cancers, to both (a) understand carcinogenic molecular drivers and (b) assist in diagnosis and sub-classification in a clinical context. These efforts have been fruitful, identifying molecular subtypes of many cancers that correlate well to clinical outcomes and other disease phenotypes. In particular, gene expression has an unsurprisingly clear effect in many cancers, and is highly predictive of numerous morphological, cellular, and outcome phenotypes. However, many of these studies over the past 5-10 years have only had limited datasets (relative to what is presently available), with intensive analysis of individual cancer profiles using predictive and biomarker-identifying unsupervised approaches [3–6]

For this reason, we are interested in gene expression as predictive of human cancer traits. The most significant modern effort to scale-up this approach, in terms of tissue diversity and sample size, are consortium efforts like The Cancer Genome Atlas (TCGA) and Therapeutically Applicable Research To Generate Effective Treatments (TARGET), both of which have generated massive, petabyte-scale datasets. [9, 10] To integrate and standardize findings across research consortia, the NIH-sponsored Genomic Data Commons (GDC: https://gdc-portal.nci.nih.gov) was established, particularly to link clinical and genetic datasets. [11] This integrative approach has enabled previously impossible studies, in particular the comparison of the genetic drivers of cancer across tissue types, beyond their simple identification. [7, 8]

Our ultimate goal, then, is to design a pipeline capable of obtaining, processing, and regressing/predicting outcomes for any specific cancer type. Crucially, this pipeline must be sufficiently modular that it can be automatically run for each cancer type. Here, we present the details of the pipeline in its current state, and preliminary results for two of the largest GDC cancer types - kidney and breast.

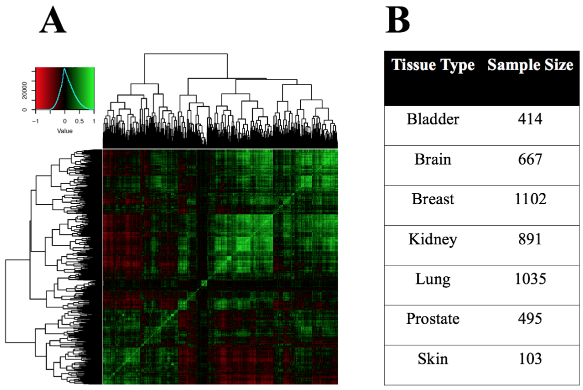
DATASET AND CLINICAL FEATURES

The GDC, collectively, gene expression data for 29 tissue/cancer types (e.g. brain) from 39 distinct projects (e.g. TCGA-lung). In the vast majority of cases, there is matched clinical and biospecimen data for each tumor, though the attached clinical features vary substantially between tissues. In this study, we restricted our focus to seven specific human cancers for practical reasons: bladder, brain, breast, kidney, lung, prostate, and skin. However, the described procedures could be applied to any tissue represented in the GDC database.

We built a suite of functions for downloading all data of a given tissue using the GDC API through the R languagae. [12] We downloaded RNA-sequencing data from the GDC Data Portal for our selected tissues, retrieving the read counts per gene as quantified by HTSeq. [13] Gene expression counts are notoriously skewed, with relatively small numbers of genes having very large expression counts. For this reason, normalizing expression data by the sum of all reads is subject to additional variability of there are large upper outliers. Thus, we normalized the data by dividing all values of a given dataset by its 75th percentile value, following GDC recommendation. Notably, this is a within-sample normalization, and does not involve sharing information (e.g. the mean) between individual gene expression profiles.

Thus, we obtain (for each tissue) a n by p matrix, where n is the sample number and p is the number of genes (60484 in all tissues), where each sample corresponds to a vector of 60484 real numbers. In Figure 1A, we show a hierarchically clustered heatmap (using average linkage and the hclust function in R) of the sample Pearson correlation matrix for one representative tissue (lung cancer).

There are clear differences between different subsets of tumors, and clear clusters of similar gene expression profiles. This suggests that many of the underlying gene expression measurements are highly correlated, which is strongly expected for biological and experimental reasons. This



Tissue Type	Sample Size
Bladder	414
Brain	667
Breast	1102
Kidney	891
Lung	1035
Prostate	495
Skin	103

Figure 1: A: Representative Clustered Sample Correlation Matrix Heatmap - Lung Cancers; B: Sample Size for Each Tissue

presents an immediate problem, as many predictors are almost perfectly co-linear. Reducing the large set of genes to a representative set of variables is a crucial first task.

Next, we assessed the available clinical data attached to each sample, and removed those samples for which there was insufficient or missing clinical information. This required substantial pre-processing in a somewhat tissue-specific manner, as we quickly learned that the standardization efforts of the GDC were clearly more oriented towards the gene expression measurements, rather than the clinical information. The final input datasets contained the sample counts shown in Figure 1B.

After subsetting the potential input samples, we selected a set of outcomes for each tissue we would like to predict from the gene expression matrix. For each tissue, there are a range of quantitative (e.g. age at initial pathologic diagnosis), binomial (e.g. progesterone receptor positive/negative), and multinomial (disease subtype) responses. In Figure 2, we show a representative set of the kinds of features of interest for kidney and breast cancers.

This reveals another issue with the dataset: many categorical variables do not have many samples in some categories. Thus, going forward, we made two modifications: first, for any binomial or multinomial prediction, we removed all classes that did not contain at least 15% of the data; second, for all questions of tumor stage, we converted the Stage I-IV labels to simple "Low" and "High", where stages I and II were low, and III and IV were high. Overall, we used some features in all tissues (e.g. gender, age at diagnosis, disease subtype) and some in a tissue-specific way (e.g. progesterone receptor status, breast specific). After this preparation, the next step was to construct a pipeline that could accept a tissue-feature pair and output a set of fitted models and predictions.

METHODS

We begin, before any other step, by separating each tissue into a 70/30 training/validation split. We will operate only on the training set (using cross validation within the training set to fit models), and evaluate the final models on the validation set.

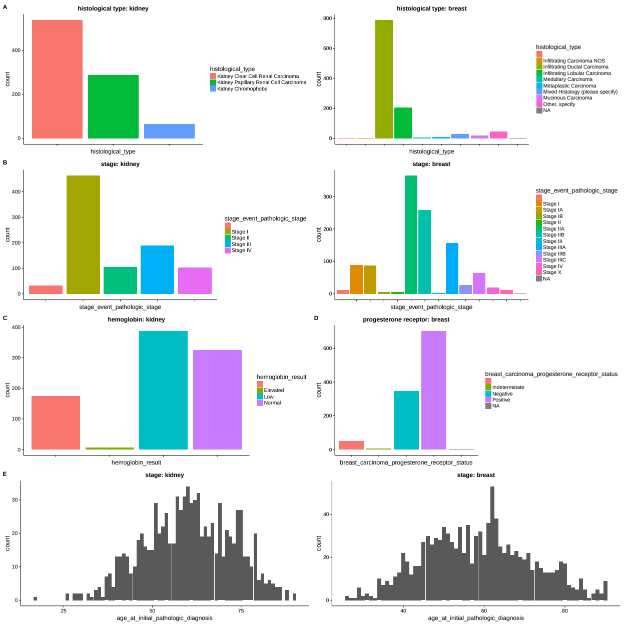


Figure 2: Distribution of Several Clinical Attributes in Kidney (Left) and Breast Cancer (Right)

Next, since we have set up a large set of predictors and a comparably small sample size (n < p), we expect we need to somehow reduce the number of predictors in each dataset. Given the strong observed correlation between many features, we would like a reduced feature set that captures the important axes of variation, and deals appropriately with correlated predictors. I considered principal component regression and ridge regression, but discarded those since they would not actually remove any individual predictor - the dimensions of the lower-dimensional space would still be linear combinations of many, perhaps most, of the predictors.

Thus, for a given tissue-feature pair, I applied lasso regression using the glmnet package and extracted the list of all predictors with non-zero coefficients based on the lambda with the lowest mis-classification rate or RMSE based on cross-validation. [14, 15] For the quantitative variables, the application is direct using standard linear regression. For categorical variables, we model the prediction for a multinomial model (or binomial, as a sub-case) as:

$$Pr(class = c|X = x) = \frac{exp(\beta_{0k} + \beta_k^T x}{\sum_{i=1}^{K} exp(\beta_{0i}|\beta_i^T x})$$

The sparse β that minimizes the error for the lasso is then:

$$\beta = min(||y = \sum_{j=1}^{J} X_j \beta_j||_2^2 + \lambda \sum_{j=1}^{J} ||\beta_{1_j}||_2)$$

So, to extract a feature set and thus a model, all that remains is to tune the λ parameter. I used 10-fold cross-validation to select a value of lambda, and see what our model looks like after doing so.

In Figure 3, we show an overview of our approach as applied to each tissue-feature pair. After splitting off the validation data, we normalize as described previously within each sample, then select non-zero features using the lasso. Taking that restricted subset, we then fit a set of models, automatically tune them using cross-validation, and test the resulting model fit on the held-out validation data to obtain either RMSE and R-squared values, for quantitative variables, or accuracy rates and AUC (as appropriate) for categorical variables.

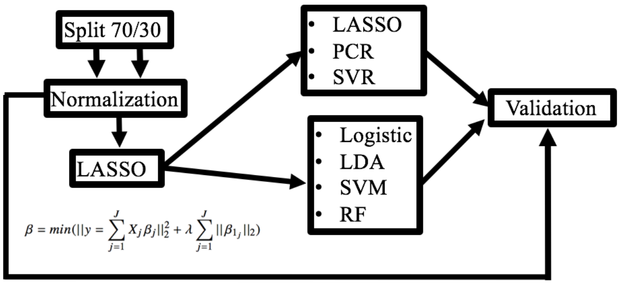

Split 70/30
Normalization
LASSO
$\beta = min(||y = \sum_{j=1}^{J} X_j \beta_j||_2^2 + \lambda \sum_{j=1}^{J} ||\beta_{1_j}||_2)$
• LASSO
• PCR
• SVR
• Logistic
• LDA
• SVM
• RF
Validation

Figure 3: Statistical Approach

Now that we have a substantially reduced feature space, we used the reduced expression matrix containing only predictors with non-zero lasso coefficients as input to a set of classifiers. For quantitative variables, we attempted to predict the given outcome using (i) the coefficients from the original lasso fit, (ii) principal component regression, and (iii) support vector regression using a Gaussian kernel. These were chosen to represent a range of model complexity, from direct linear regression to the nonlinear SVMs. For categorical attributes, we used (i) the coefficients from the original lasso fit, (ii) linear discriminant analysis, (iii) support vector classification with a Gaussian kernel, and (iv) random forests. Again, this represents a mix of linear and non-linear, more and less complex classifiers. When specific hyper-parameter tuning was necessary, a grid search was applied using the tune() function in R. Now, we go into some **brief** detail about the non-lasso-based models we applied (since the lasso-based models simply apply the linear regression coefficients obtained in the feature selection process with no modification).

Principal Component Regression

We implemented PCR using the pls package in R, and used all principal components for prediction (since this was after variable selection, so each PC was a linear combination of the very small, lasso-selected variable subset). [16]

PCR is a standard linear regression, except we use the principal component decomposition of the data variance-covariance matrix instead of the data directly when estimating the regression coefficients. Formally, for the standard linear model, coefficients are estimated by $\beta = (X^T X)^{-1} X^T Y$. For PCR, instead of using $X^T X$ directly, we instead get the principal component decomposition, set that equal to a new matrix, and plug in

$$\begin{aligned} \hat{\beta} &= (X^T X)^{-1} X^T Y \\ X^T X &= PDP^T \\ A^T A &= PDP^T \\ \hat{\beta_{PC}} &= (A^T A)^{-1} A^T Y \end{aligned}$$

where D is a diagonal matrix of eigenvalues and P is the eigenvector matrix. This is equivalent to projecting all samples to PC-space, and then regressing Y on all the principal components.

Support Vector Regression and Classification

To implement support vector machines, we used the e1071 package in R, which applies the common LibSVM package available in many languages. [17, 18] For categorical variables we used "C-classification" and for quantitative variables we used "epsilon-regression". Formally, C-classification SVMs solve the following optimization problem:

$$\begin{aligned} min_{w,b,\varepsilon} \frac{1}{2} w^T w + C \sum_{i}^{l} \varepsilon_i \\ s.t.\ y_i(w^T \phi(x_i) + b) \geq 1 - \varepsilon_i, \\ \varepsilon_i \geq 0, i = 1, ..., l \end{aligned}$$

Epsilon-regression solves a very similar optimization problem, which is formally:

$$\begin{aligned} min_{w,b,\varepsilon,\varepsilon*} \frac{1}{2} w^T w + C \sum_{i}^{l} \varepsilon_i + C \sum_{i}^{l} \varepsilon_i * \\ s.t.\ w^T \phi(x_i) + b - z_i \leq \epsilon + \varepsilon_i, \\ -w^T \phi(x_i) - b + z_i \leq \epsilon + \varepsilon_i *, \\ \varepsilon_i, \varepsilon_i * \geq 0, i = 1, ..., l \end{aligned}$$

For each model, fitting was performed using the grid-search function tune in LibSVM, which found optimal values for C and epsilon as appropriate and automatically used those values for validation prediction.

Linear Discriminant Analysis

We used the lda function in the R package MASS (Modern Applied Statistics with S). [19] As implemented, LDA assumes that the conitional probability of observing a data point given class, $p(X|Y = 0)$ or $p(X|Y = 1)$, follows a normal distribution with different mean vectors and covariance matrices. Then, after fitting the parameters of the distributions for the two classes, we compute the log likelihood ratio for the probability of membership in class 1 over class 2, and assign the data point accordingly (whether it is above or below some threshold). Prior probabilities were incorporated into the ratio by the lda function, and were the proportions of each class label in the training data.

Random Forests

Finally, we used the randomForest package to perform random forest classification only (though RFs can be applied to regression problems too). [20] Briefly, the algorithm constructs a large number of trees, each trained on a different randomly sub-sampled (with replacement) subset of the training data. The number of trees is a tunable parameter, but here, we simply set the number to 500 (the recommended default by the software authors). Then, for a new data point, all trees are used to predict the class of the data point and the class is determined by a majority vote or some function of all individual tree predictions (possibly incorporating weights). The software implements a variety of strategies to optimize eventual accuracy, and these shape properties like which points are sub-sampled at each tree, how the trees are weighted, and so on.

RESULTS AND DISCUSSION

For each tissue-feature pair, we computed the regularization path for the fitting parameter lambda, and chose the lambda that produced the least complex model with the lowest error. Across tissue-feature pairs, this selected between 7 and 158 genes out of over 60000. For some of the categorical features, to qualitatively evaluate the effect of feature reduction, we constructed principal component biplots before (60000 features and after (7-158 features). In Figure 4, we show a representative example for one specific analysis: breast cancer disease subtype, specifically differentiating between ductal carcinomas and lobular carcinomas. We first show the lasso regularization path, by log(lambda), with the dashed lines representing the lambda with the lowest mis-classification rate, and the lambda with the least complex model with an error within one standard error of the lowest mis-classification rate. Then, using the latter value for lambda, we construct PC plots, showing an increased separation after feature selection, though dramatically fewer genes are used. This is all a good sign.

Through analyzing the results from the lasso fit (so, not using the validation data yet), we quickly realized two facts: first, some categorical properties were highly predictable across tissues. For example, all disease subtypes (including the example in Figure 4) had fitted cross-validation misclassification rates of less than 10%. Second, quantitative responses showed very high cross-validated mean square error rates - all time-based responses (like age at diagnosis) had an RMSE of at least five years, and the regularization paths did not look very clean (i.e. there was not a rapid decrease from right to left in the analogous plot shown in Figure 4A). This leads to a suspicion, that turns out to be correct, that categorical validations will perform well, while quantitative validations will not.

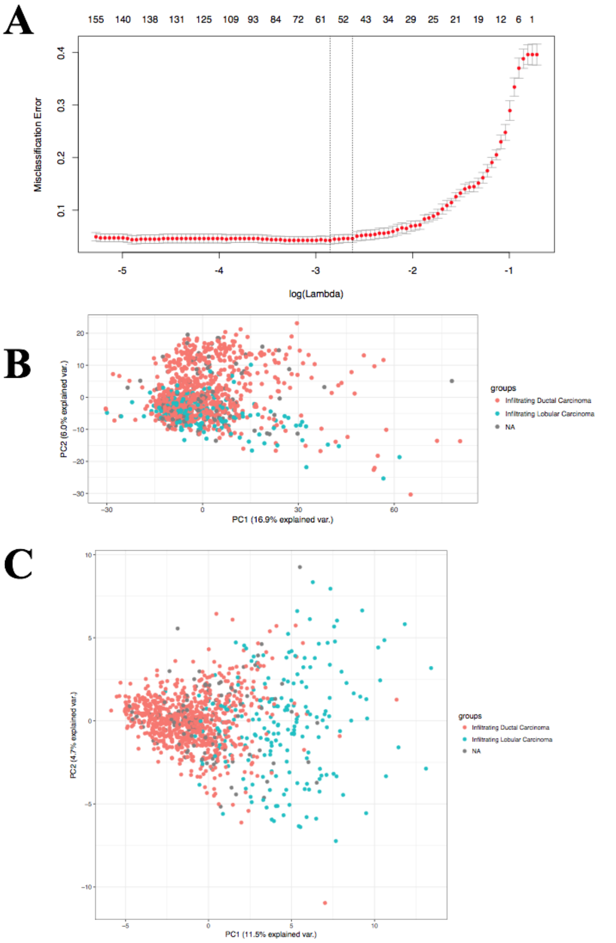
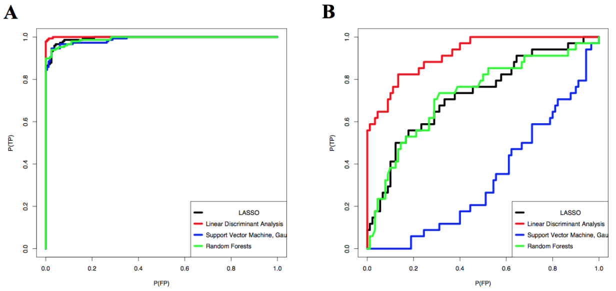


Figure 4: A: Lasso Regularization Path for Breast Cancer Subtype; B: PC Biplot Before Lasso Feature Selection; C: PC Biplot After Lasso Feature Selection. Points in B and C are colored by tumor subtype.

After obtaining the feature subsets for each tissue-feature comparison, we proceeded to auto-fit each model described in Methods (automating hyperparameter selection) and then predicting the feature in the validation set. For quantitative responses, we then computed RMSE and R^2 values comparing actual and predicted responses. For categorical responses, we computed precision, recall, accuracy, and AUC values.

Given the somewhat large number of fit models, and our space constraints in this report, we report the validation statistics for the top five quantitative tissue-feature pairs, where the pairs are sorted by maximum R^2 (since different quantitative features have different ranges/scales). We also report one of the better and one of the mediocre categorical tissue-feature pairs. (Note: not shown here, we also fit models for gender as a "positive control" while building the pipeline. For all tissues, gender was perfectly separable based on fewer than 10 genes, which is strongly expected given the known biology.) As is clear from Figure 5C, most of the quantative variables performed quite poorly, usually being wrong (in the best cases!) by years, making them functionally useless. While it is interesting that certain tissues tend to be in the top over others (like brain and breast), we feel very few conclusions can be drawn from these regressions.



	Accuracy	Precision	Recall	AUC
Binomial Lasso	0.89	0.89	0.89	0.91
LDA	0.93	0.93	0.93	0.98
SVM	0.86	0.86	0.86	0.91
RF	0.87	0.89	0.87	0.93

	Accuracy	Precision	Recall	AUC
Binomial Lasso	0.73	0.77	0.72	0.72
LDA	0.86	0.85	0.85	0.92
SVM	0.49	0.49	0.48	0.33
RF	0.76	0.76	0.76	0.73

C

Tissue	Feature	Lasso RMSE	Lasso R^2	PCR RMSE	PCR R^2	SVR RMSE	SVR R^2
brain	days_to_birth	4425.949	0.304	5144.902	0.26	4435.015	0.302
breast	days_to_birth	4159.087	0.233	4213.312	0.242	4032.109	0.277
breast	age_at_initial_pathologic_diagnosis (years)	11.209	0.234	14.535	0.115	11.21	0.243
kidney	age_at_initial_pathologic_diagnosis (years)	11.943	0.061	22.514	0.018	11.467	0.143
brain	days_to_death	998.506	0.097	1064.324	0.093	998.666	0.142

Figure 5: A: ROC Plot for Progesterone Receptor Status in Breast Cancer, with associated statistics; B: ROC Plot for Stage in Bladder Cancer, with associated statistics; C: Best Quantitative Prediction Results

However, within the categorical classifiers, there was substantial variation in predictability across tissue-feature pairs, and also between models within the same tissue-feature pair. As representative examples of this diversity, Figures 5A and 5B show ROC plots for two tissue-feature pairs, progesterone receptor status in breast cancer (positive/negative) and tumor stage in bladder cancer (low/high). Clearly, the progesterone receptor status is highly predictable from gene expression, regardless of the underlying model chosen, while not only is bladder stage less predictable overall, it is very model-sensitive, with LDA significantly outperforming the binomial lasso and random forests, which themselves outperform the SVM. We also note that we performed, but did not show here, disease subtype for all seven tissues (part of which was shown in Figure 4, for breast). In all of those binomial or multinomial models, accuracy rates were above 0.5, reflecting a very strong ability to classify subtype by gene expression. Instead, we selected two other binomial analyses that represent part of the range of what we observed with categorical variables.

CONCLUSIONS AND FUTURE DIRECTIONS

Our general conclusion, based on these results, is that (i) gene expression, as such, may not contain much information about things like age, or at least not enough to be useful in this context, and (ii) responses that are more "molecular" in nature tend to be better classified than responses that are more "anatomical" in nature. Put differently, responses like tumor subtype or progesterone receptor status are generally obtained by a pathologist doing some experimental procedure to a tissue and obtaining a specific signal, while properties like tumor stage encompass a variety of subjective inputs like tumor size, the overall health of the patient, metastasis, and so on. Thus, "molecular" responses may be "closer" to the prediction data (gene expression), since they are obtained from experiments on cells, while the "gross" responses are a bit more distant (though, as we can see from Figure 5B, still predictable!).

Going forward, there are a large number of additions that could improve both the statistical approach and the final results. Most simplistically, adding more tissues through the pipeline would broaden the scope of the work. Additionally, there exist other data types in the GDC, that typically assign some value (for copy number, or methylation status) to each base pair in the genome. An interesting extension could be to associated each gene in the expression dataset with some value that represents its overall copy number/methylation status/other regulatory property, derived from the scores of the base pairs at that transcripts' origin. Then, additional layers of data could be integrated with the current approach. Finally, analyzing the intersection and overlap between specific gene subsets selected by lasso could help to connect the created models to some underlying molecular biology, as opposed to classification or regression - that is a substantially more difficult task, as the annotations of biological functions on many genes are notoriously noisy/inaccurate, but would be an interesting perspective to add.

ACKNOWLEDGEMENTS

We thank other students in the Departments of Genetics and Chemical and Systems Biology for discussion and ideas, particularly directing us to the GDC datasets and suggesting some initial analyses to perform (especially to look at progesterone receptors). This analysis was performed on personal computers, or on Stanford University computing clusters. All figures and documentation were prepared solely by Nathan Abell.

REFERENCES

[1] D Hanahan, RA Weinberg *Hallmarks of Cancer: The Next Generation*. Cell. 2011;144(5):646-74.

[2] IR Watson, K Takahashi, PA Futreal, L Chin . *Emerging patterns of somatic mutations in cancer*. Nat Rev Genet. 2013;14(10):703-18.

[3] K Kourou, TP Exarchos, KP Exarchos, MV Karamouzis, DI Fotiadis. *Machine learning applications in cancer prognosis and prediction*. Comput Struct Biotechnol J. 2015;13:8-17.

[4] RG Verhaak, KA Hoadley, E Purdom et al. *Integrated genomic analysis identifies clinically relevant subtypes of glioblastoma characterized by abnormalities in PDGFRA, IDH1, EGFR, and NF1.*. Cancer Cell. 2010;17(1):98-110.

[5] S Zheng, AD Cherniack, N Dewal, et al. *Comprehensive Pan-Genomic Characterization of Adrenocortical Carcinoma*. Cancer Cell. 2016;30(2):363.

[6] *Genomic Classification of Cutaneous Melanoma*. Cell. 2015;161(7):1681-96.

[7] KA Hoadley, C Yau, DM Wolf, et al. *Multiplatform analysis of 12 cancer types reveals molecular classification within and across tissues of origin.*. Cell. 2014;158(4):929-44.

[8] A Prat, B Adamo, C Fan, et al. *Genomic Analyses across Six Cancer Types Identify Basal-like Breast Cancer as a Unique Molecular Entity*. Sci Rep. 2015;5:8179

[9] *https://cancergenome.nih.gov*

[10] *https://ocg.cancer.gov/programs/target*

[11] https://gdc-portal.nci.nih.gov/

[12] R Core Team *R: A Language and Environment for Statistical Computing* R Foundation for Statistical Computing, Vienna, Austria, 2016.

[13] S Anders, PT Pyl, W Huber *HTSeq–a Python framework to work with high-throughput sequencing data.* Bioinformatics. 2015;31(2):166-9.

[14] R Tibshirani. *Regression shrinkage and selection via the lasso*. J. Royal. Statist. Soc B., 1996;58(1),267-288.

[15] J Friedman, T Hastie, R Tibshirani *Regularization Paths for Generalized Linear Models via Coordinate Descent.* Coordinate Descent. Journal of Statistical Software, 2010;33(1),1-22.

[16] BH Mevik, R Wehrens. *The pls Package: Principal Component and Partial Least Squares Regression in R* Jour. of Stat. Soft. 2010;18(2):1-24.

[17] D Meyer, E Dimitriadou, K Hornik, A Weiingessel, F Leisch, C-C Chang, C-C Lin. *e1071: Misc Functions of the Department of Statistics, Probability Theory Group* https://cran.r-project.org/web/packages/e1071/index.html

[18] C-C Chang, C-J Lin. *LIBSVM: a library for support vector machines*. ACM Transactions on Intelligent Systems and Technology, 2001;2(27):1-27.

[19] WN Venables, BD Ripley. *Modern Applied Statistics with S.* Fourth Edition. Springer, New York. ISBN 0-387-95457-0. 2004.

[20] A Liaw, M Wiener. *Classification and Regression by randomForest.* R News 2002;2(3):18-22.