Prediction Of Protein Levels In Cells [1]

M. Kemmerling

June 20, 2016

Abstract

This article presents linear regression and support vector regression as techniques to predict protein expressions. Various feature selection techniques and, in the latter case, kernels are examined with respect to their influence on prediction performance. Both linear regression with prior feature selection on the Lasso, as well as support vector regression with a radial basis function kernel prove to be useful prediction techniques. Including gene data in the feature space yields very limited success. Datasets from different tissues prove to be not directly compatible.

1 Introduction

In recent times, there has been a huge increase in available biological data due to high-throughput omics technologies. This includes measurements of genes, proteins, microRNAs and metabolites. The availability of such data enables the possibility of applying machine learning techniques in order to better understand the relationships between these different molecules. The development of computer models of single cells is a current area of research and a deeper understanding of the inner workings of cells could potentially lead to the development of (better) models. Eventually, sufficiently sophisticated models could help with the diagnosis and treatment of diseases.

In particular, this article focuses on how protein levels in cells can be predicted based on available data. The problem is approached by utilising regression techniques such as linear and support vector regression. Additionally, feature selection methods are investigated to narrow down the set of relevant features in the hope of improving prediction accuracy.

The methods investigated in this paper are applied on datasets provided by "The Cancer Genome Atlas" (TCGA) research network, a project dedicated to cataloguing genetic modifications related to cancer [11]. The main focus will be on the BRCA (breast invasive carcinoma) dataset, but datasets from different organs and tissues will be investigated as well to see whether the results are consistent across different datasets and how including them may influence prediction accuracy.

Implementations of machine learning techniques such as linear regression and support vector regression are taken from the scikit-learn Python package. [8]

The remainder of this paper is structured as described in the following. In the next section, the techniques applied in this paper, such as linear regression, support vector regression and various feature selection techniques will be discussed. The experiments section will contain descriptions of several experimental setups meant to measure the performance of different methods and how they compare. This will be followed by a presentation of the obtained results as well as a discussion thereof.

2 Methods

The task of predicting protein levels requires models to be able to produce continuous, real-valued output variables from input variables of the same nature. Thus, regression techniques are well suited for this problem and will be the main focus in this paper.

2.1 Linear Regression

Linear regression is a statistical approach for building a linear model which describes the relationship between a dependent variable and a set of independent variables, also called features. In this article, the focus will be on multiple linear regression, which deals with multiple features as opposed to simple linear regression, where only one feature is used to build the model. The dependent variable is generally denoted by y, while the features are denoted by X.

Ordinary least squares linear regression aims to minimise the residual sum of squares between the predicted and observed dependent variables by fitting a linear model where each feature is assigned a coefficient, or weight β_i, determining its contribution to the predicted variable. [6]

[1] This thesis was prepared in partial fulfillment of the requirements for the Degree of Bachelor of Science in Knowledge Engineering, Maastricht University, supervisor: Rachel Cavill

This results in a model of the form

$$y = x_0\beta_0 + x_1\beta_1 + ... + x_n\beta_n + \epsilon = X\beta + \epsilon \quad (1)$$

with minimisation objective

$$\hat{\beta} = \min_{\beta} ||X\beta - y|| \quad (2)$$

2.2 Support Vector Regression (SVR)

While support vector machines are mainly known for solving classification problems, the technique can be extended to work with regression problems.

Contrary to the classification algorithm, regression requires the use of a loss function. While a variety of different loss functions exist, the most common one is the ε-insensitive loss function proposed by Vapnik [1].

$$\mathcal{L}(y, g(x)) = \begin{cases} 0 & |y - g(x)| \leq \varepsilon \\ |y - g(x)| - \varepsilon & otherwise \end{cases} \quad (3)$$

where x_i are the input, y_i the target vectors and $g(x)$ the function to be fit.

The algorithm disregards all errors that lie within this ε-insensitive tube and hence only deviations larger than ε contribute to the cost. As a consequence, the produced model depends only on a subset of the training data, which are called *support vectors*.

With the help of the kernel trick, nonlinear regression can be performed by using a mapping φ to map the training vectors into a higher dimensional space in which the regression problem can be solved in a linear fashion. [9]

A selection of kernels $K(x, y) = \langle \varphi(x), \varphi(y) \rangle$ used in this paper is given below.

Polynomial kernel
The polynomial kernel function is given by

$$K(x, y) = (x \times y + 1)^d \quad (4)$$

where d determines the degree of the polynomial.

Radial basis function kernel
The radial basis function (RBF) kernel function is given by

$$K(x, y) = e^{-\frac{||x-y||^2}{2\sigma^2}} \quad (5)$$

where σ is a free parameter.

2.3 Feature selection

Only selecting a subset of all the available features to include in the prediction model can often improve performance. Typical causes for this phenomenon are erratic data and multicollinearity among the features. [5]

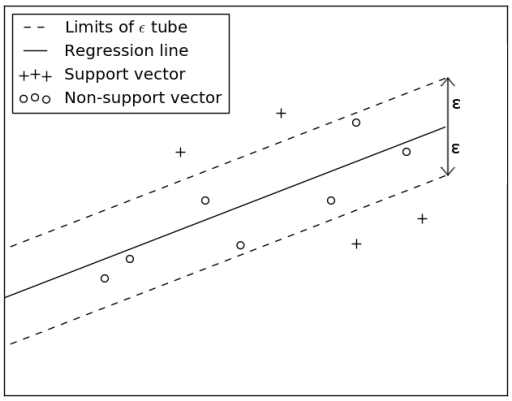

Figure 1: Principle of support vector regression, showing support and non-support vectors as well as the ε-tube. Figure adapted from [3].

Rather than selecting a group of features to use for all predictions, a separate group is selected for each protein, since different sets of features will be relevant for different proteins.

Conceivably, manual feature selection based on prior biological knowledge may be useful. Approaches to be examined in this paper include using the gene which codes for a particular protein as a feature, as well as selecting all genes from all pathways that contain the coding gene.

Besides manual selection, automatic feature selection processes that do not require any prior knowledge are often employed. Following is a description of selected automatic feature selection techniques.

Univariate feature selection (UFS) A straight forward approach to feature selection is to select the best apparent features based on some univariate statistical test.

To construct a ranking of all features, a sequential approach to building a regression model is applied. Initially, all possible models consisting of only one feature are built and the best one is selected as M_1 for the next step. Now, M_1 is extended to a two-feature model by again trying all remaining features and selecting the best one as M_2. This process is repeated until a model M_n is selected, where n is the total amount of features.

The choice of the best additional feature in each step is determined by the use of an F-test which assesses the significance of improvement of a model M_i with respect to its predecessor M_{i-1}. [13]

After the ranking is completed, a set of low performing features is discarded.

Lasso The *least absolute shrinkage and selection operator* (or Lasso) tries to add features one by one, but penalises the addition of extra features. As a consequence, coefficients of features not improving the model enough to overcome the given penalty are set to zero. [10]

In contrast to linear regression, the objective function is

$$\hat{\beta} = \min_{\beta} \frac{1}{2n} ||X\beta - y||_2^2 + \alpha||\beta||_1 \quad (6)$$

where n is the number of samples and α the penalty constant.

Only those features that have nonzero coefficients are selected from the constructed model to be used in further analysis.

Recursive feature elimination (RFE) The recursive feature elimination technique recursively selects decreasing sets of features based on a model which assigns weights to features. Initially, the model is trained on the whole set of features. The resulting weights are then used to prune features by discarding the ones with the smallest absolute weights. This process is repeated until a prespecified number of features remains.

To find the number of features that will have the best performance, this technique can simply be applied in a cross-validation loop. [4]

3 Experiments

In general, the data to be used in prediction experiments is evenly divided into training and test set.

The main measure of prediction accuracy used throughout this article is the coefficient of determination, denoted by R^2, which describes the proportion of variance that can be explained by the model. [6] A value $R^2 = 1$ implies that the regression function fits the data perfectly, while $R^2 = 0$ indicates that the model is unable to predict any of the variance of the dependent variable. When evaluating the model on unseen data, negative values are possible as well.

3.1 Prediction on the BRCA dataset

In order to assess which method is best suited to predict protein expressions, both linear regression and support vector regression are tested on the BRCA dataset, which is the biggest dataset offered by the TCGA network to date. It consists of 938 samples and 215 features in the form of proteins. This data is evenly split into a training set on which the model is built, and a test set which is used to evaluate the model performance on unseen data.

Multiple SVR models with different kernels, including linear, radial basis function (RBF), and second & third degree polynomial will be tested.

When using a linear kernel, the algorithm will often fail to converge. To avoid this, the number of iterations is restricted to a maximum of 500 for this experiment.

3.2 Comparison of feature selection techniques

To measure the effects of different feature selection techniques, separate prediction runs for each of the techniques (univariate feature selection, recursive feature elimination and the Lasso) as well as one without any feature selection are executed.

As part of the experiment, the best 10% of features is selected from the resulting ranking of the univariate selection strategy.

Recursive feature elimination is performed in a 3-fold cross-validation loop with a step size of three, i.e. each iteration three features are removed.

3.3 Coding genes as features

Preliminary experiments have shown that simply providing all available gene data to the prediction algorithms does not improve prediction accuracy. A possibly more sensible approach is to only include genes coding for the relevant protein. These genes are taken from a mapping provided on the TCGA website.

To examine whether including these genes improves the prediction results, three separate models are built and tested. The first uses only the selected genes as features, the second only the proteins, and the third includes both the selected genes and proteins.

Unfortunately, not all samples in the dataset contain both protein and gene data. Thus, there are 441 samples with both types of data left.

3.4 Pathway genes as features

Rather than just including the coding gene as a feature, additionally using genes in the same pathway as the aforementioned coding gene might yield better performing models. These genes interact with each other and thus might be relevant features. Since a gene may be present in more than one pathway, all genes from all pathways which include the coding gene are used as additional features.

The pathways themselves are provided by the *Max Planck Institute for Molecular Genetics*. [7] Using this and the mapping mentioned in the previous section, a list containing all the relevant genes for each protein is compiled. The given pathways may include genes for which no data is available in this experimental setting and consequently can not be used in the prediction.

In order to measure the effect these additional features exhibit, one model with both the usual protein features and the genes is built, as well as one using only

the protein features to serve as a baseline in the comparison. These two models are built separately with support vector regression using the RBF kernel and linear regression with prior feature selection through the Lasso.

3.5 Different datasets

To assess whether prediction performance can be improved by increasing the number of samples and to determine if datasets from different tissues and cancer types are compatible, additional datasets were taken into consideration. They were selected based on having a comparatively high number of samples and are comprised of: kidney renal clear cell carcinoma (KIRC) with 478 samples, prostate adenocarcinoma (PRAD) with 352 samples, head and neck squamous cell carcinoma (HNSC) with 357 samples, and uterine corpus endometrial carcinoma (UCEC) with 440 samples. As a comparison, BRCA, the main dataset used throughout this article, consists of 938 samples.

The first part of the experiment consists of simply training on a combination of each of the above with the BRCA dataset while recording any changes in performance.

For the second part, models trained on one single dataset are tested on a different single dataset. To mitigate the effects of differing sample sizes, a number of samples equal to the smallest dataset size is randomly sampled from each dataset.

Support vector regression with the RBF kernel was used for both parts of the experiment. If data for a certain protein was not available in any of the datasets, it was omitted from all models constructed in this experiment to ensure that results are comparable.

4 Results

4.1 Prediction on the BRCA dataset

Figure 2 shows a series of box-and-whiskers plots for, from left to right, linear regression, support vector regression with a linear kernel, a second degree polynomial kernel, a third degree polynomial kernel and a radial basis function kernel.

The plots are standard box-and-whiskers plots as proposed by Tukey [2], with the central line as the median, and the box spanning between the 25th and 75th percentile of the data. The whiskers end at the lowest (highest) data within 1.5 the interquartile range of the lower (upper) quartile. Data above or below the whiskers are considered outliers and marked by a +.

The different models have means:

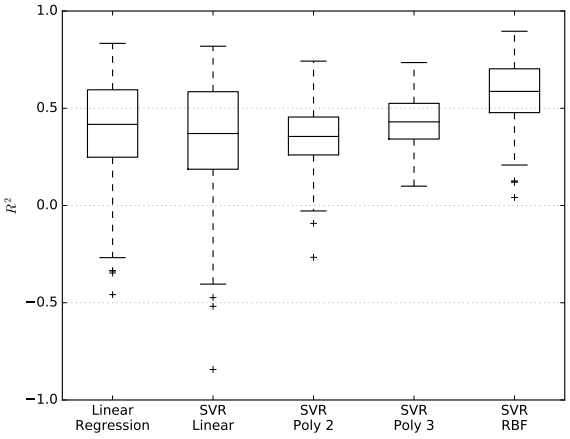

Figure 2: Prediction using linear regression and support vector regression with linear, second & third degree polynomial and RBF kernel

Model	Mean R^2
Linear Regression	0.398
SVR Linear	0.346
SVR Poly 2nd	0.354
SVR Poly 3rd	0.430
SVR RBF	0.577

4.2 Comparison of feature selection techniques

Figures 3-5 show three scatter plots corresponding to the different feature selection techniques. The horizontal axis refers to the R^2 performances with prior feature selection, while the vertical axis refers to the performances without prior feature selection. A reference line $y = x$ was added to the graphs. Points below this line show a better performance using the respective feature selection technique, while points above it signify a better performance without any feature selection.

The R^2 means of the respective techniques are given by:

Technique	Mean R^2
No feature selection	0.398
Lasso	0.560
UFS	0.504
RFE	0.495

4.3 Coding genes as features

A comparison of prediction using only the genes, only the proteins and their combination is given in figure 6. The depicted box-and-whiskers plot follows the same conventions as described in section 4.1.

The means of the respective models are given by:

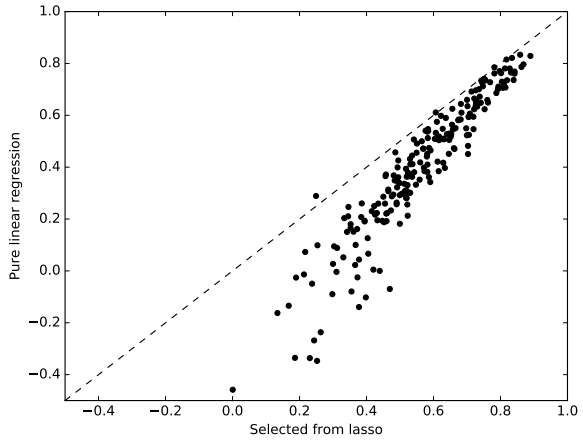

Figure 3: Feature selection: Lasso

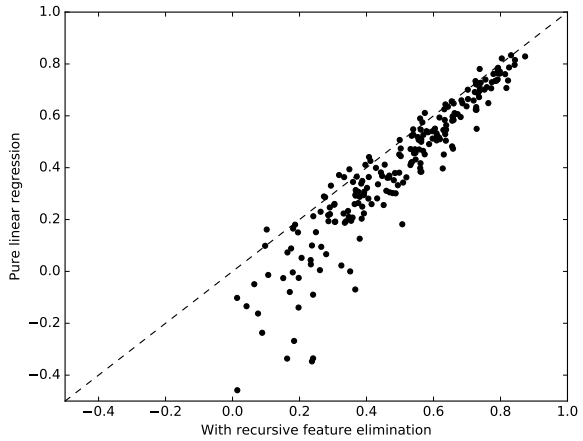

Figure 4: Feature selection: Recursive feature elimination

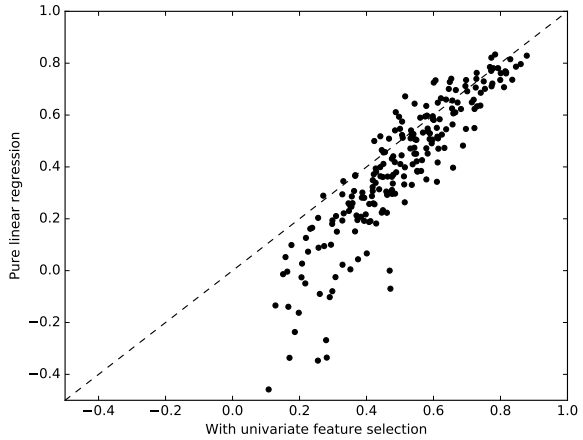

Figure 5: Feature selection: Univariate, F-test

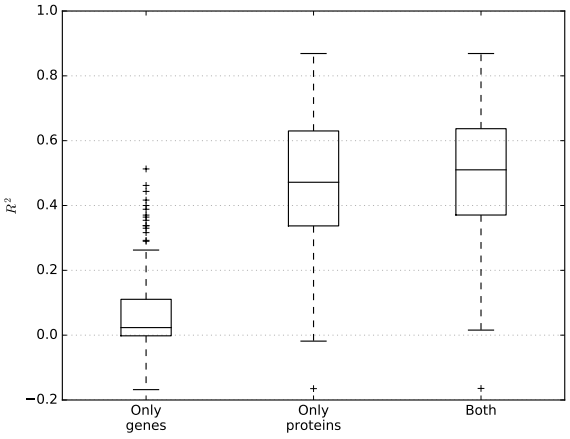

Figure 6: Prediction performance as R^2 using only the coding genes, only the proteins, and both.

Model	Mean R^2
Only genes	0.069
Only proteins	0.473
Both	0.501

4.4 Pathway genes as features

The results after including the genes from the relevant pathways are captured in figure 7. The box-and-whisker plots, from left to right, show the results of linear regression with the usual protein features, linear regression with the additional gene features, and the next two similarly for support vector regression.

The mean R^2's, including additional results using *only* the genes are as follows:

Model	LR	SVR
Only genes	-0.126	-0.185
Only proteins	0.467	0.498
Both	0.454	0.469

4.5 Different datasets

Building models on a combination of BRCA and each of the other datasets leads to performances as depicted in figure 8.

The means of the depicted box-and-whiskers plots are given by:

Dataset(s)	Mean R^2
BRCA	0.580
BRCA+KIRC	0.567
BRCA+PRAD	0.563
BRCA+UCEC	0.565
BRCA+HNSC	0.565

Testing models on one dataset after training them on a different one results in universally negative R^2 values as can be seen in table 1.

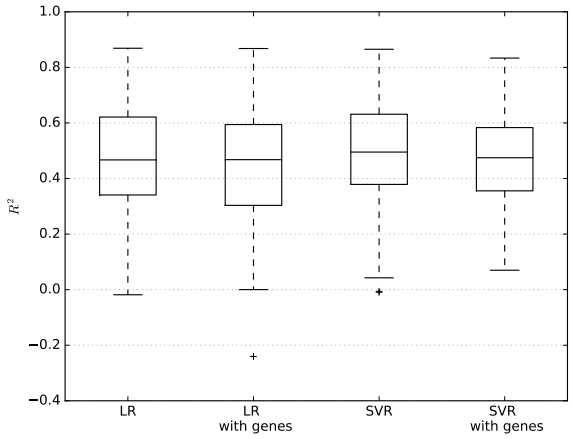

Figure 7: Prediction performance after including pathway genes in linear regression and SVR

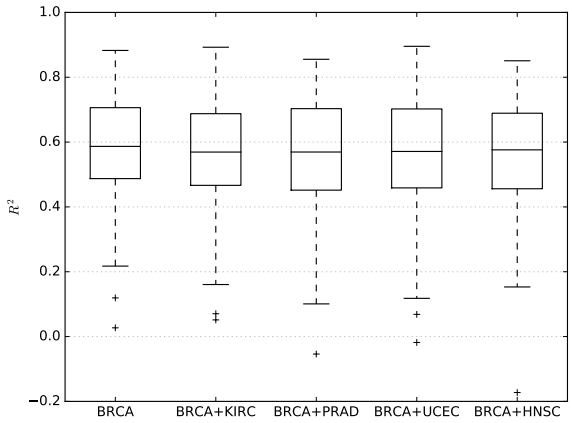

Figure 8: Prediction performances (of combinations) of different datasets.

trained on	BRCA	KIRC	PRAD	HNSC	UCEC
tested on					
BRCA		-1.07	-0.68	-0.86	-0.45
KIRC	-0.97		-1.00	-0.90	-0.89
PRAD	-0.73	-1.51		-1.41	-1.13
HNSC	-0.98	-0.96	-1.05		-0.95
UCEC	-0.62	-1.31	-1.14	-1.27	

Table 1: R^2 of SVR trained exclusively on one dataset and tested on another.

5 Discussion

5.1 Prediction on the BRCA dataset

The results of this prediction experiment can be divided into two groups, linear regression and support vector regression with a variety of kernels. In the latter group, the RBF kernel appears to be the best performing one in terms of mean R^2 (0.577).

While linear regression can not compare to the best SVR model, it still has the third highest mean R^2 out of all the models. It does, however, have the second highest variance, which might be undesirable, since there will be a comparatively big difference between proteins which are well predicted and those that are not.

Concluding, in this setup, support vector regression with the RBF kernel appears to be the method of choice. It remains to be seen whether feature selection will change this result.

5.2 Comparison of feature selection techniques

Irrespective of the specific technique, feature selection seems to most improve the predictions of those proteins which have a comparatively low prediction accuracy. In more technical terms, the correlation coefficient ρ between the R^2 without feature selection X and the improvement Y using the respective technique is negative, indicating that proteins with low initial prediction performance benefit the most from feature selection.

Technique	ρ_{XY}
Lasso	-0.879
UFS	-0.819
RFE	-0.711

With all techniques, the majority of predictions benefit from prior feature selection, but the exact amount varies between techniques. The univariate strategy may appear to be the least suitable one, since the number of models that perform worse than without feature selection is the highest in comparison with the other two techniques and further, the models that do perform worse do so to a higher extent than with the other techniques. However, looking at the mean R^2, univariate feature selection is slightly ahead of RFE. Both univariate feature selection and recursive feature elimination can not compare to the Lasso, which almost exclusively improves performance.

Indeed, performing Tukey's honest significant difference test [12] to examine which feature selection techniques yield significantly different results from each other, the only pair that does not show any significantly different results is UFS and RFE. This is illustrated in figure 9, where only the confidence intervals of UFS and RFE overlap.

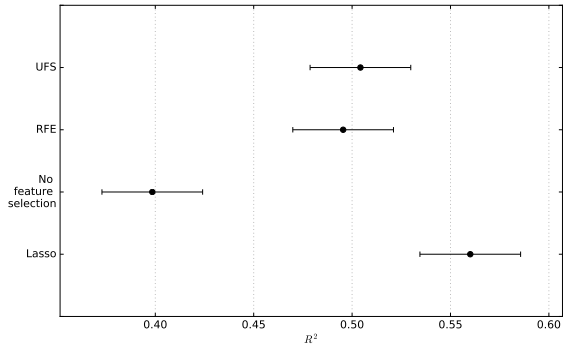

Figure 9: 95% confidence interval plot for the R^2 means of feature selection techniques. Intervals that do not overlap indicate significant differences.

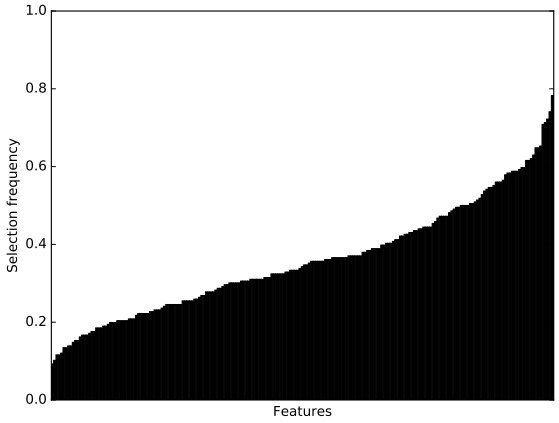

Figure 10: Number of times each of the 216 features is selected using the Lasso.

It is notable that there are proteins which defy the general trend by for instance yielding better results using univariate feature selection than with the Lasso. This could serve as motivation to determine which technique works best for each protein individually and build more tailored models. However, it is questionable whether this rather small improvement in accuracy justifies the loss of simplicity in the analysis.

Applying feature selection presents an opportunity to examine if some features are generally more useful than others by simply counting how often each feature is selected. For feature selection via the Lasso, this is depicted in figure 10, where each feature is represented as a bar, with height corresponding to the number of times it has been selected as given on the vertical axis. Clearly, this data is not uniformly distributed. Some features are selected roughly 80% of the time, while others are only chosen about 10% of the time. Thus, the data does show that not all features are equally useful.

In principle, univariate feature selection and the Lasso can also be used with SVR. Recursive feature elimination relies on the weights assigned to each feature to perform the selection, which makes this technique unsuitable for non-linear kernels. The two remaining feature selection methods result in a performance drop when applied prior to SVR. Consequentially, feature selection will only be used in combination with linear regression for the remainder of this article.

Concluding, feature selection has a significant impact on linear regression prediction performance, as can be seen when looking to the results of the previous section, where SVR with the RBF kernel was clearly the most suitable technique. While linear regression was lagging far behind without feature selection, with prior application of the Lasso, it performs only marginally worse than SVR.

5.3 Coding genes as features

Including only the genes coding for a protein as features results in relatively poorly performing models on average, with a mean R^2 of 0.069. However, figure 6 shows a large amount of positive outliers, which suggests that in some models the genes have significantly more predictive power than in the average case.

Based on this observation, one would hope using the gene expressions in addition to the protein expressions would lead to some improvement in performance. Looking at the plots of the protein-only and combined models, there is indeed a slight improvement with respective R^2 means of 0.473 and 0.501. To examine the significance of this improvement, a paired t-test is applied, resulting in a t-statistic of 6.215 on 214 degrees of freedom and thus a p-value < 0.0001. Therefore, there is strong evidence that, on average, including genes does lead to better models. Of course, while statistically significant, the improvement is still small. A 95% confidence interval for the true mean difference is given by $0.0286 \pm 0.0090 = (0.0195, 0.0376)$.

5.4 Pathway genes as features

As can be seen in figure 7, the inclusion of gene data from the relevant pathways does, on average, not lead to improvements using neither linear regression nor support vector regression. While it may seem counter-intuitive that providing more information decreases performance, this is very well possible, as it is the rationale behind using feature selection as well as not using all the available gene data in the first place. Looking at the mean R^2 of the models, it is no surprise that including the genes does not improve the performance, as the models using only the genes fail to yield a positive R^2.

A paired t-test on the linear regression values yields a t-statistic of 2.250 on 151 degrees of freedom, corresponding to a p-value of 0.026. Hence, choosing a significance level of $\alpha = 0.05$, the means of the models with and without the pathway genes are significantly different. A 95% confidence interval for the true mean difference is given by $0.0127 \pm 0.0111 = (0.0016, 0.0238)$. Unsurprisingly, there is even stronger evidence that including the pathway genes in the SVR models decreases prediction performance, with a t-statistic of 5.507, a p-value < 0.0001 and a 95% confidence interval for the true mean difference $0.0287 \pm 0.0103 = (0.0184, 0.0390)$.

It is notable that, while neither prediction method benefits from the added features, support vector regression appears to have greater trouble coping with them.

Despite the overall negative results in this experiment, the general trend is not a universal one. Even though most of the models suffer from the addition of the gene data, some proteins can be better predicted when including this data. Consequently, similar to section 5.2, it might again be possible to create more tailored models, which again presents a trade-off with regards to simplicity.

5.5 Different datasets

A model built on only data from one type of tissue and cancer appears to be unsuitable to predict protein levels of datasets from different conditions, as can be seen in table 1, which shows negative R^2 values for every possible combination. However, when the model is trained on both kinds of data, there is only a minor performance loss compared to training and testing on a single dataset, suggesting that a more general model is built in this case. In fact, the performance drop after including data from a different dataset is not as clear as it appears, as can be seen by constructing a series of 95% confidence intervals for the true mean difference between the BRCA model and each combination model.

Dataset	Confidence interval
BRCA+KIRC	(-0.0191, 0.0444)
BRCA+PRAD	(-0.0159, 0.0494)
BRCA+UCEC	(-0.0173, 0.0475)
BRCA+HNSC	(-0.0176, 0.0469)

Since the confidence intervals are neither fully negative nor positive, statistically speaking, it can not be concluded whether the single dataset or a combination yields a better performance.

Thus, the data suggests that predicting values from a dataset that was not part of the training data will not yield particularly useful results. However, one single model can be used to successfully predict protein levels from different datasets as long as both datasets were used in the training of the model.

6 Conclusion

It is clear from the various experiments that support vector regression using a radial basis function kernel is the most promising technique examined in this article. However, despite its different nature, linear regression with prior feature selection through the Lasso does not lag far behind the aforementioned approach.

Regardless of the technique, protein expressions were by far the most useful features. Gene expressions do appear to have some predictive power, but, at least with the chosen techniques, were not able to influence performance in any meaningful way.

Datasets from different tissues and cancer types were not directly compatible, but a model trained on (a part) of both datasets is able to produce useful predictions. A wider array of datasets from different tissues, disease types, and perhaps even healthy patients is needed to determine whether model performance deteriorates when data from too many sources is used to build the model.

Further research could be directed at using other available information such as microRNA to perform the prediction and eventually to build models which do not rely on other proteins' expression levels.

7 Acknowledgements

The results shown here are in whole or part based upon data generated by the TCGA Research Network: `http://cancergenome.nih.gov/`.

References

[1] Drucker, H, Burges, C J C, Kaufman, L, Smola, A, and Vapnik, V (1996). Support vector regression machines. *Advances in neural information processing systems*, Vol. 9, pp. 155–161.

[2] Frigge, Michael, Hoaglin, David C, and Iglewicz, Boris (1989). Some Implementations of the Boxplot. *The American Statistician*, Vol. 43, No. 1, pp. 50–54.

[3] Gilardi, Nicolas and Bengio, Samy (2001). Local machine learning models for spatial data analysis. *Journal of Geographic Information and Decision Analysis*, Vol. 4, p. 2000.

[4] Guyon, Isabelle, Weston, Jason, Barnhill, Stephen, and Vapnik, Vladimir (2002). Gene selection for cancer classification using support vector machines. *Machine learning*, Vol. 46, No. 1-3, pp. 389–422.

[5] Hocking, R. R. (1976). A biometrics invited paper. the analysis and selection of variables in linear regression. *Biometrics*, Vol. 32, No. 1, pp. 1–49.

[6] John O. Rawlings, David A. Dickey, Sastry G. Pantula (1998). *Applied Regression Analysis: A Research Tool*, p. 210. Springer, second edition.

[7] Max Planck Institute for Molecular Genomics (2016). ConsensusPathDB data access. *http://consensuspathdb.org/*.

[8] Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., and Duchesnay, E. (2011). Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, Vol. 12, pp. 2825–2830.

[9] Smola, Alex J. and Schölkopf, Bernhard (2004). A tutorial on support vector regression. *Statistics and Computing*, Vol. 14, No. 3, pp. 199–222.

[10] Tibshirani, Robert (1996). Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society. Series B (Methodological)*, Vol. 58, No. 1, pp. 267–288.

[11] Tomczak, Katarzyna, Czerwińska, Patrycja, and Wiznerowicz, Maciej (2015). The Cancer Genome Atlas (TCGA): an immeasurable source of knowledge. *Contemporary oncology (Poznań, Poland)*, Vol. 19, No. 1A, pp. A68–77.

[12] Tukey, John W. (1949). Comparing individual means in the analysis of variance. *Biometrics*, Vol. 5, No. 2, pp. 99–114.

[13] Zhang, Jin and Wang, Xueren (1997). Selecting the best regression equation via the p-value of f-test. *Metrika*, Vol. 46, No. 1, pp. 33–40.