Big Data in Finance: Assignment 3.

Prepared and submitted by Alexander Romanenko

1. Introduction

The purpose of this report is to demonstrate and assess potential use of machine learning techniques and data analytics algorithms in Finance for the purpose of prediction and classification of daily returns of industry portfolios. The report starts with analysis of the given dataset and goes on to describe the methods, which will be used for predictions. In particular, it discusses potential applications of Linear Regression, LASSO, Ridge Regression and Elastic Net. The discussion moves on to methods, such as Linear Discriminant Analysis, Quadratic Discriminant Analysis, Logistic Regression and K- Nearest Neighbour. The report also shows and assess the results of the analysis for the chosen techniques. Finally, the report provides suggestion for potential areas of future research in the area and concludes with a summary of the analysis.

2. Dataset Analysis

The dataset to be used for analysis contains daily returns for 49 industry portfolios over 500 days for the period between 02/01/2015 and 31/01/2016.

In theory, we would expect to see strong relationships between certain industries, for example Banks and Insurers. This could be explained by the fact that these industries are affected by similar internal and external factors. We can confirm our expectations by looking at 6 strongest correlations between the groups.

ind1	ind2	corr
Fin	Banks	0.939146
Mach	ElcEq	0.897937
LabEq	BusSv	0.891708
Insur	Fin	0.887761
Fin	BusSv	0.887725
Insur	Banks	0.871056

The results are not surprising as we would expect strong correlation between 'Trading' (Fin), 'Banks', 'Insurance' (Insur) and 'Business Services' (BusSv) industries. Similarly, there is a strong contemporaneous relationship between 'Machinery' (Mach) vs 'Electrical Equipment' (ElcEq) and 'Measuring and Control Equipment' (LabEq) vs 'Business Services' since they are affected by similar macro-economic factors. It is important to note that 'Business Services' contain various range of services, such as advertising, industrial, cleaning and building management, medical equipment, computer equipment, research, development and testing labs, etc. Therefore, this industry is highly correlated with both 'Trading' and 'Measuring and Control Equipment'.

ind1	ind2	corr
Gold	Banks	-0.04607
Gold	Clths	-0.01579
Gold	Insur	0.011199
Gold	Meals	0.011382
Gold	Hlth	0.021727
Gold	Txtls	0.024815

When looking at industries, which have weakest contemporaneous relationship, we can clearly see that 'Precious Metals' (Gold) industry is present in every one of the top six combinations. This is not surprising as precious metals are commonly used as a hedge against market falls as they are less impacted by macro-economic factors. In fact, prices of precious metals normally go up during market downturns as investors looking for a 'safe haven' assets (Bresiger, 2017).

3. Method

3.1 Linear Regression (OLS, Ridge Regression, LASSO and Elastic Net)

We will start predictions by analysing performance of various regression methods, in particular Ordinary Least Squares (OLS), Least Absolute Shrinkage and Selection Operator (LASSO) and Ridge Regression. In addition, the use and results for 'Elastic Net' method will be discussed.

Ordinary Least Squares (OLS) is a very commonly used method for estimating unknown parameters in a linear regression. The goal is to minimise the sum of squares of the differences between estimate and the actual values.

LASSO and Ridge are extensions of OLS method. Ridge Regression uses tuning parameter λ to impose a shrinkage penalty on regression coefficients, whereas LASSO reduces some of the coefficients to 0 and therefore removing these parameters from a model (En.wikipedia.org, 2017). LASSO's aim is to enhance accuracy and interpretability of models through variable selection and regularisation (En.wikipedia.org, 2017). Elastic Net approach is essentially a combination of Ridge and LASSO techniques, where the method finds ridge regression coefficients for each parameter and then applies LASSO type shrinkage to a few of them.

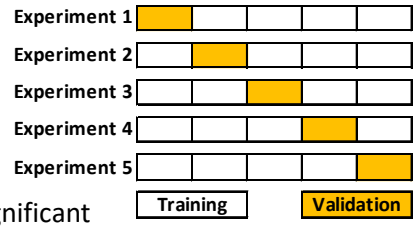
Performance Measurement. The performance comparison will be based on the Root Mean Squared Error values (RMSE) and Residual Sum of Squares (RSS) (also known as Sum of Squared Errors (SSE)). Once these measures are calculated they will be compared against a historic mean, which is used as our performance benchmark.

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2} \qquad RSS = \sum_{i=1}^{n}(y_i - \hat{y}_i)^2$$

Window size and high dimensionality. The data is represented as a time-series and therefore it is beneficial to use 'rolling window' approach rather than a 'static' one (Inoue, Jin and Rossi, 2017). The window size is selected to be 80. One of the reasons for this decision is the fact that it is the requirement for the report. However, we believe that the chosen window size is also justified taking into account high-dimensionality of the data. As we know, the data has 49 dimensions and therefore choosing rolling window below 49 will deny use of regression as we will be unable to calculate least squares effectively (James et al., 2015). Having window size close to 49 also presents potential issues as it leaves us with smaller degree of freedom and therefore there is a risk of over-fitting. Similarly, we do not want to have too big of a window (>100) as first of all we only have 500 data entries and secondly it might include past information, which might not necessarily be important currently. As a result, we believe that the window size of 80 is appropriate here. Finally, it is important to mention that window size is a hyperparameter here, therefore one should seek caution when selecting window sizes based on a performance of models.

Performance Testing. There are two popular ways for back-testing of time-series predictions: 'in sample' (IS) and 'out of sample' (OOS). With 'in sample' a model trained on n-size sample (in our case rolling window size) and test against values within this training sample. With 'out of sample' approach a model is trained on n-size sample and being tested using n+1 data (i.e. 'new' data). This approach is more accurate as it removes bias comparing to 'in sample' back-testing. OLS will be assessed using both 'in-sample' and 'out of sample' methods. Ridge regression and LASSO will be assessed using 'out of sample' methods.

Validation and Hyper-Parameter tuning. Cross validation has become an essential validation and testing technique in machine learning. Normally it is performed by splitting the dataset into n number of folds, the model is then being trained on n-1 folds and being tested on the remaining fold. This process is repeated so every single fold is used for validation. Key benefit of cross-validation comparing to standard test/validation split is avoidance of loss of significant modelling or testing capability (En.wikipedia.org, 2017).



Unfortunately, the approach discussed above is not suitable for time-series validation for a number of reasons. First of all, if there are some patterns emerging in period 3 and evident in period 4 and 5, these patterns will not be 'confirmed' if validating on period 1 or 2. Secondly, the nature of time-series means that we shouldn't have 'gaps' in our data when training a model. As a result, the 'forward chaining' approach is preferred where we train our model using past data to predict forward-looking events.

We use a variation of this approach during predictions for LASSO, Ridge and Elastic Net, where for every window, we start training the model on first 50 elements and validate it on 51st, then we train the model on first 51 elements of the window and validate it on 52nd, and so on until we reach 80- our window size. During the cross-validation we are testing various λ- shrinkage penalty parameter. The λ that gives us the lowest RMSE on average is then selected and is used for predicting of the element w+1, which follows the given window w. This process is repeated for all windows in the dataset.

LASSO parameters. As it has been discussed in the 'Validation' sub-section above, for every window an optimal λ will be selected. This λ in turn will shrink some of the model parameters to 0. For every window, we will store the 'remaining' parameters in order to identify which parameters are more useful and whether this make sense. This exercise will be only conducted for 'Business Services' industry as the process is very time consuming and requires a lot of processing power.

3.2 Logistic Regression and Machine Learning Techniques (LDA, QDA and KNN)

In previous section, we looked at prediction techniques, where we attempted to forecast a daily change for an industry portfolio. However, it is fair to say that in some instances, we do not need a precise forecast, but rather a classification of whether a stock (or other financial instrument) will appreciate or depreciate in price. In this section, we look at various techniques which deal with classification techniques. The dataset will introduce a new column 'dir' (for *direction*) with 1 for 'up' and 0 for 'down'.

Logistic Regression is a regression model which is used for predicting categorical values and has a similar form to a linear regression: $y= \beta_0 + \beta_1 * x_1 + \dots + \beta_n * x_n + \varepsilon$

Linear Discriminant Analysis (LDA) and Quadratic Discriminant Analysis (QDA) are methods, which are used to find a linear combination of features or separate the data into classes or events. Both methods assume that the observations of each class are drawn from a Gaussian distribution and

predictions are performed by plugging parameters estimates into Bayes theorem. Key difference between the two methods is that QDA assumes that each class has its own covariance matrix (James et al., 2015). There are a number of advantages of using LDA and QDA over logistic regression. First of all, these models are more stable when the classes are well-separated. Secondly, there is additional stability in situations with small n and normal distribution of X's in each of the classes (James et al., 2015).

K- nearest neighbour (KNN) is a non-parametric method for classification and regression. The new data is classified by the class which is most common among its k nearest neighbours.

Industry Choice. In order to simplify the examples for use of the technique and interpretation of the results, we will focus only on one industry- Business Services.

Performance Measurement. The performance of the ML techniques will be assessed using confusion matrix, which shows values for predicted against actual classifications in a tabular form. In addition, accuracy measurement will be introduced, which will show percentage of correctly classified entries.

Window Size. In order for the logistic regression algorithm to converge, the window size has been increased to 120. There will be no window used for other ML techniques.

Validation and Hyper-Parameter Tuning. Validation will be performed by splitting the dataset into a training set (400 entries) and a validation set (124 entries).

In the case of KNN, k is a hyper-parameter. In order to select the best k, we will split the dataset into a training set (350 entries), test set (90 entries) and validation set (84 entries). The model will be trained using the training set and tested using k from 1 to 200 on the test set. The k, which produces the highest accuracy will be selected to be used on the validation set.

Lag Impact. So far, we have used only lag of 1 day in our analysis. Here we will create a new dataframe, which will include additional 20 days of lag for 'Business Services' industry together with 1 day lag for all other industries (as shown on the right). By having the values in this way, will enable efficient use of logistic regression. This dataframe will be used to analyse whether the lags could play an important role in predicting/classifying an outcome.

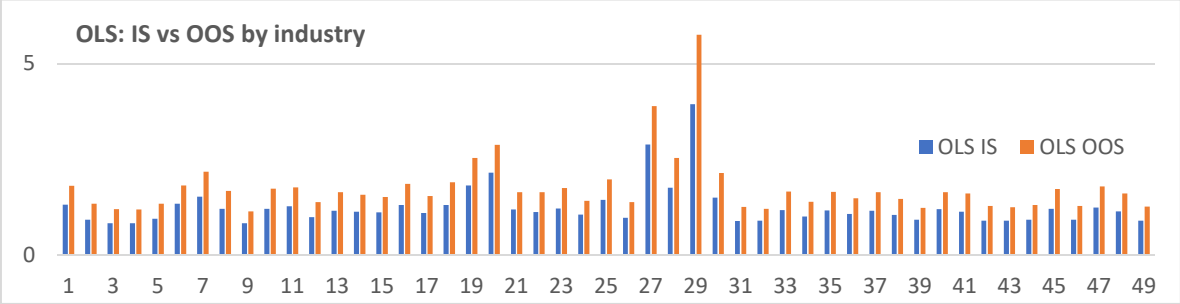
	BusSv	Other	…	Lag1	Lag2	Lag 3	…
20150102	-0.38	-0.65	…	N/A	N/A	N/A	…
20150105	-1.46	-1.52	…	-0.38	N/A	N/A	…
20150106	-0.93	-0.97	…	-1.46	-0.38	N/A	…
20150107	1.4	0.8	…	-0.93	-1.46	-0.38	…
20150108	1.92	1.41	…	1.4	-0.93	-1.46	…

4. Results

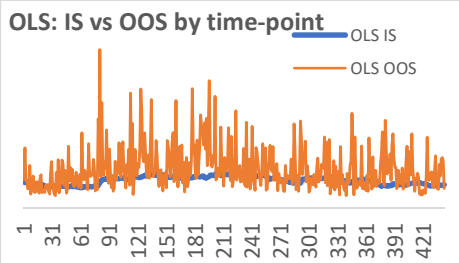
4.1 Linear Regression (OLS, Ridge Regression, LASSO and Elastic Net)

OLS: 'In sample' vs 'Out of sample'

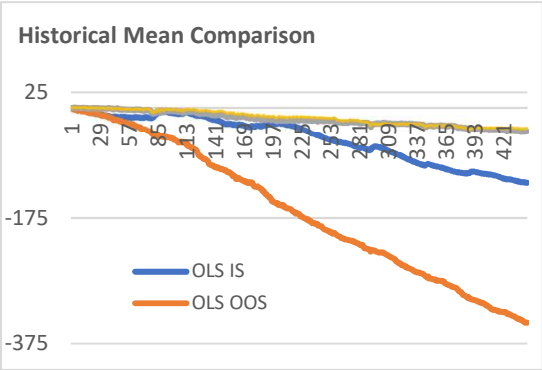
The graphs show that 'in sample' (IS) predictions are much more accurate with practically every single industry average RMSE is lower comparing to 'out of sample' (OOS) results. This is not surprising as the 'in sample' predictions are made on the data, which is used to train the model, whereas 'out of sample' predictions are made on the 'new' data (for full breakdown of results see Appendices).

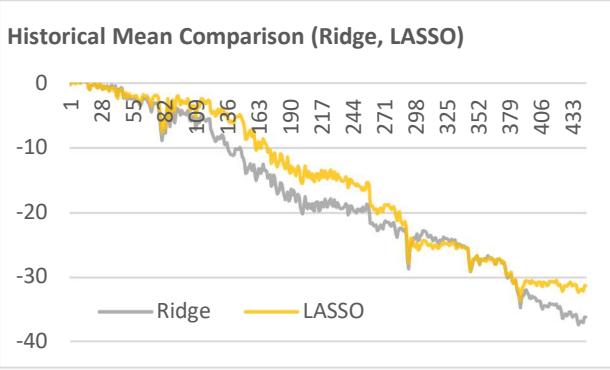


It is interesting to note that OOS results are more sporadic comparing to IS across the time period. Occasionally OOS does better than IS, however its general trend and average is clearly above the IS results.



Historical Mean comparison. As discussed in the 'Method' section the performance of the models is assessed by comparing against Historical Mean RMSE. The difference is shown as a cumulative RMSE differential. Direction of the graph demonstrates whether a prediction technique outperforms (up) or underperforms (down) comparing to Historical Mean RMSE.





We can see that none of the 4 techniques are outperforming Historical Mean (HM) with OOS doing particularly poorly. It is important to note that even though both Ridge and LASSO have underperformed comparing to HM, the overall RMSE cumulative difference is not very high with averages of -0.0814 and -0.0707 respectively.

It has been noted in research papers that in general LASSO should perform better (Chinco, Clark-Joseph and Ye, n.d.), however due to the high-dimensionality of the data with relatively small n, the LASSO's performance is on par with Ridge regression (En.wikipedia.org, 2017). Elastic Net approach overcomes this problem by adding a quadratic part to the shrinkage penalty. Unfortunately, due to the long processing time requirement for Elastic Net, this analysis has been unable to produce results for all industries, however the results for Business Services industry have been produced and they clearly show improvement in accuracy and a reduction of the average RMSE for this industry. As a future point of research, it will be interesting to see if performance of Elastic Net is consistently better across other industries.

	Ridge	Lasso	Elastic Net
RMSE	0.843894	0.8517176	0.7733384

LASSO Parameters. Summary of the most commonly used parameters by LASSO approach are somewhat inconclusive. These parameters are: 'Medical Equipment', 'Household', 'Healthcare', 'Tobacco and Cigarettes', 'Toys' and 'Clothes' (for full details see Appendices). Although one can attempt to find links between these industries and 'Business Services' industry, we would expect to see here industries closer related to business, such as 'Trading, Banks, Insurance, etc. Therefore, we could conclude that perhaps the industries which are used in predictions are not directly related to Business Services. This conclusion is somewhat echoed in the research paper by A. Chinco et. al (2017) where the authors suggest that the choice of parameters is not dictated by economic forces.

Industry Comparison. By analysing performance of prediction techniques from industries perspective, we can see that 'Soda', 'Beer' and 'Household' industries scored lowest RMSE values across most of the prediction techniques. This is not surprising as these industries represent items or services, which are not strongly affected by macroeconomic factors. In fact, by conducting volatility analysis of all 49 industries- these 3 industry again come up at the bottom of the list (for full volatility results see Appendices). This somewhat explains the lower RMSE results: less volatility means there are less unexpected spikes/ falls and therefore the average prediction is low.

4.2 Logistic Regression and Machine Learning Techniques (LDA, QDA and KNN)

Accuracy results. Similarly, to the section above, the in-sample testing produces the highest results with a particularly impressive 89.12% for QDA. However out-of-sample results for Log Regression, LDA and QDA are very close to 50%, which means that similar results could be achieved by flipping a coin. Results for KNN are slightly better with 57.02% accuracy.

	Log Reg	LDA	QDA	KNN
in-sample	60.50%	60.88%	89.12%	62.01%
out-of-sample	49.50%	51.61%	52.40%	57.02%

It is interesting to note that when looking at confusion matrices for LDA and QDA, we can see that the models guessed correctly that the market will go up 56.9% (37/(28+37)) and 58.1% (36/(26+36)) of times for LDA and QDA respectively.

LDA	actual	
pred	0	1
0	27	32
1	28	37

QDA	actual	
pred	0	1
0	29	33
1	26	36

KNN's initial test came up with the optimal k, which was was 80 with accuracy 63.3%. This parameter produced accuracy of 57.02% on the validation set.

Lag Impact result. By running the logistic model on the data with additional 20 lags we encountered higher IS accuracy rate of 64.16% (vs 60.5% with original data). The accuracy rate was even higher if taking into account only instances where the model predicted positive return with accuracy rate of 65.3% (198/(198+105)). No OOS results were produced as the dataframe had too many dimensions (70) for OOS calculations to be done accurately.

Log Reg	actual	
pred	0	1
0	126	76
1	105	198

The regression analysis identified lag of 15 (working) days as a significant parameter, which is a very interesting find. It is difficult to explain the anomaly without further analysis, but it confirms our initial assumptions that additional lags will provide useful information to be used in prediction. It is important to note that the lag structures will be different for different industries. These lag structures will result from regular external events, such as issuance or economic performance indicators, company's quarterly results, impact of holidays, seasons, etc.

5. Research Considerations and Suggestions for Future Work

It is important to stress again that we only had 500 entries of data and therefore it would be interesting to see how the accuracy values and RMSE will change when using a larger dataset. Similarly the available data represented EOD values, which might be skewed by end of day auctions/ spikes in trading. It will be interesting to see more mid-day data.

As discussed, the heavy processing requirement for some of the research limited ability to apply the data for all industries. This is something which should be considered to be done in the future. We have also discussed importance of choosing correct window sizes, therefore in future research there should be fine-tuning introduced for this hyper-parameter.

Finally, in addition to quantitative data, an effort should be made to be able to introduce some qualitative information as input into models. One such potential area could be the use of Natural Language Processing to analyse how news reports affect price movements.

6. Conclusion

This report has discussed and assessed various machine learning techniques, which could be used in prediction and classification of industry returns. Results demonstrated that Ridge Regression and LASSO have performed only slightly worse than the benchmark with the Elastic Net approach showing signs of outperforming it. Similarly KNN method showed decent performance, which should be re-tested with more data.

7. References

Bresiger, G. (2017). Precious Metals Funds: A Golden Opportunity?. [online] Investopedia. Available at: http://www.investopedia.com/articles/mutualfund/09/precious-metals-funds.asp [Accessed 17 Apr. 2017].

Chinco, A., Clark-Joseph, A. and Ye, M. (n.d.). Sparse Signals in the Cross-Section of Returns. SSRN Electronic Journal.

En.wikipedia.org. (2017). Cross-validation (statistics). [online] Available at: https://en.wikipedia.org/wiki/Cross-validation_(statistics) [Accessed 15 Apr. 2017].

En.wikipedia.org. (2017). Elastic net regularization. [online] Available at: https://en.wikipedia.org/wiki/Elastic_net_regularization [Accessed 18 Apr. 2017].

En.wikipedia.org. (2017). Lasso (statistics). [online] Available at: https://en.wikipedia.org/wiki/Lasso_(statistics) [Accessed 19 Apr. 2017].

En.wikipedia.org. (2017). Tikhonov regularization. [online] Available at: https://en.wikipedia.org/wiki/Tikhonov_regularization [Accessed 19 Apr. 2017].

Inoue, A., Jin, L. and Rossi, B. (2017). Rolling window selection for out-of-sample forecasting with time-varying parameters. Journal of Econometrics, 196(1), pp.55-67.

James, G., Witten, D., Hastie, T. and Tibshirani, R. (2015). An introduction to statistical learning. 1st ed.

8. Appendices

8.1 RMSE results

Count	Industry	Volatility	Historical Mean	OLS IS	OLS OOS	Ridge	LASSO	Elastic Net	OLS IS	OLS OOS	Ridge	LASSO
1	Agric	1.325618	0.952412	1.318027	1.811154	1.038488	1.014381		-0.36562	-0.85874	-0.08608	-0.06197
2	Food	0.974985	0.744796	0.923024	1.346272	0.80681	0.817566		-0.17823	-0.60148	-0.06201	-0.07277
3	Soda	0.888038	0.670284	0.832588	1.204753	0.731091	0.807669		-0.1623	-0.53447	-0.06081	-0.13738
4	Beer	0.87664	0.666091	0.837344	1.193012	0.727027	0.75424		-0.17125	-0.52692	-0.06094	-0.08815
5	Smoke	1.000896	0.7298	0.948809	1.345555	0.784246	0.830745		-0.21901	-0.61575	-0.05445	-0.10095
6	Toys	1.437001	1.049608	1.33945	1.820493	1.110214	1.115041		-0.28984	-0.77089	-0.06061	-0.06543
7	Fun	1.532602	1.147471	1.526052	2.173623	1.251381	1.20016		-0.37858	-1.02615	-0.10391	-0.05269
8	Books	1.200179	0.953869	1.211691	1.677426	1.037074	1.00722		-0.25782	-0.72356	-0.08321	-0.05335
9	Hshld	0.860234	0.648526	0.832993	1.141813	0.714857	0.763948		-0.18447	-0.49329	-0.06633	-0.11542
10	Clths	1.246318	0.975172	1.206918	1.736137	1.047767	1.053639		-0.23175	-0.76097	-0.07259	-0.07847
11	Hlth	1.322225	1.007291	1.278138	1.767156	1.084149	1.029696		-0.27085	-0.75987	-0.07686	-0.0224
12	MedEq	1.026958	0.79123	0.98997	1.385035	0.867987	0.861212		-0.19874	-0.59381	-0.07676	-0.06998
13	Drugs	1.205275	0.926374	1.15923	1.640123	1.000863	0.981791		-0.23286	-0.71375	-0.07449	-0.05542
14	Chems	1.137755	0.890478	1.13661	1.574174	0.968584	0.947858		-0.24613	-0.6837	-0.07811	-0.05738
15	Rubbr	1.086841	0.85794	1.116136	1.515194	0.909923	0.953281		-0.2582	-0.65725	-0.05198	-0.09534
16	Txtls	1.367236	1.03972	1.308619	1.856725	1.143525	1.105732		-0.2689	-0.81701	-0.1038	-0.06601
17	BldMt	1.134971	0.881697	1.103299	1.545867	0.932087	0.943689		-0.2216	-0.66417	-0.05039	-0.06199
18	Cnstr	1.407299	1.085175	1.311427	1.898117	1.152928	1.156959		-0.22625	-0.81294	-0.06775	-0.07178
19	Steel	1.833539	1.451247	1.819282	2.534913	1.605049	1.506401		-0.36803	-1.08367	-0.1538	-0.05515
20	FabPr	2.13939	1.50365	2.154994	2.877809	1.641854	1.578193		-0.65134	-1.37416	-0.1382	-0.07454
21	Mach	1.199239	0.924848	1.189916	1.640211	1.017342	1.028902		-0.26507	-0.71536	-0.09249	-0.10405
22	ElcEq	1.169512	0.908043	1.12489	1.642201	0.985766	0.961191		-0.21685	-0.73416	-0.07772	-0.05315
23	Autos	1.267115	0.993053	1.218601	1.753438	1.091089	1.077749		-0.22555	-0.76039	-0.09804	-0.0847
24	Aero	1.021728	0.768588	1.062799	1.420766	0.847234	0.833177		-0.29421	-0.65218	-0.07865	-0.06459
25	Ships	1.488053	1.139012	1.444609	1.975257	1.238975	1.198532		-0.3056	-0.83624	-0.09996	-0.05952
26	Guns	0.995821	0.716697	0.978106	1.383004	0.756898	0.809228		-0.26141	-0.66631	-0.0402	-0.09253
27	Gold	2.977735	2.28754	2.88888	3.883746	2.477847	2.334465		-0.60134	-1.59621	-0.19031	-0.04693
28	Mines	1.959432	1.498735	1.759065	2.535789	1.625056	1.571602		-0.26033	-1.03705	-0.12632	-0.07287
29	Coal	4.098543	3.210201	3.933835	5.74877	3.600658	3.258079		-0.72363	-2.53857	-0.39046	-0.04788
30	Oil	1.592979	1.223311	1.505084	2.13997	1.337769	1.277197		-0.28177	-0.91666	-0.11446	-0.05389
31	Util	0.989923	0.736415	0.888471	1.255452	0.773927	0.834533		-0.15206	-0.51904	-0.03751	-0.09812
32	Telcm	0.894211	0.668196	0.900868	1.205218	0.747131	0.764753		-0.23267	-0.53702	-0.07894	-0.09656
33	PerSv	1.215438	0.966456	1.17475	1.662627	1.050051	1.016672		-0.20829	-0.69617	-0.0836	-0.05022
34	**BusSv**	**1.004064**	**0.765655**	**1.005711**	**1.390553**	**0.843894**	**0.851718**	**0.773338**	**-0.24006**	**-0.6249**	**-0.07824**	**-0.08606**
35	Hardw	1.180443	0.89711	1.163436	1.647018	0.946957	0.957695		-0.26633	-0.74991	-0.04985	-0.06059
36	Softw	1.103118	0.813315	1.072354	1.482534	0.894307	0.886372		-0.25904	-0.66922	-0.08099	-0.07306
37	Chips	1.189346	0.895403	1.15822	1.639076	0.966905	0.957992		-0.26282	-0.74367	-0.0715	-0.06259
38	LabEq	1.062702	0.805264	1.05315	1.464991	0.866554	0.877859		-0.24789	-0.65973	-0.06129	-0.07259
39	Paper	0.939574	0.688791	0.921353	1.231835	0.74693	0.814674		-0.23256	-0.54304	-0.05814	-0.12588
40	Boxes	1.209443	0.934363	1.197897	1.64568	0.995747	1.007494		-0.26353	-0.71132	-0.06138	-0.07313
41	Trans	1.16832	0.897071	1.134345	1.608627	0.970819	0.954322		-0.23727	-0.71156	-0.07375	-0.05725
42	Whlsl	0.957341	0.754995	0.897222	1.288179	0.812168	0.824554		-0.14223	-0.53318	-0.05717	-0.06956
43	Rtail	0.918302	0.69726	0.89691	1.248268	0.75479	0.807571		-0.19965	-0.55101	-0.05753	-0.11031
44	Meals	0.956561	0.695118	0.92543	1.312723	0.770551	0.781026		-0.23031	-0.6176	-0.07543	-0.08591
45	Banks	1.273093	0.959419	1.212897	1.730132	1.028014	1.011346		-0.25348	-0.77071	-0.06859	-0.05193
46	Insur	0.96245	0.73746	0.924703	1.28772	0.786128	0.819748		-0.18724	-0.55026	-0.04867	-0.08229
47	RlEst	1.303525	1.007962	1.241796	1.790034	1.08322	1.078203		-0.23383	-0.78207	-0.07526	-0.07024
48	Fin	1.184566	0.886949	1.140983	1.60941	0.934919	0.921013		-0.25403	-0.72246	-0.04797	-0.03406
49	Other	0.926088	0.689	0.904187	1.269136	0.747592	0.771392		-0.21519	-0.58014	-0.05859	-0.08239

8.2 LASSO's commonly used parameters

Rank	Industry	Count
1	MedEq	126
2	Hshld	118
3	Hlth	92
4	Smoke	88
5	Toys	85
6	Clths	80
7	BldMt	76
8	Food	72
9	Cnstr	71
10	Rubbr	70
11	Soda	65
12	Autos	63
13	Beer	62
14	Rtail	61
15	Softw	56
16	Telcm	56
17	Guns	53
18	Coal	53
19	Oil	52
20	Hardw	51
21	Aero	48
22	Mach	48
23	BusSv	48
24	Ships	47
25	Meals	45

Rank	Industry	Count
26	Chips	45
27	Txtls	43
28	Util	43
29	Boxes	41
30	Agric	40
31	Steel	40
32	Paper	40
33	Banks	39
34	Whlsl	38
35	Fin	37
36	Mines	37
37	Fun	36
38	Books	36
39	RlEst	36
40	Drugs	35
41	Trans	35
42	ElcEq	34
43	Insur	34
44	Gold	33
45	Other	33
46	PerSv	32
47	LabEq	32
48	FabPr	32
49	Chems	32