LEARNING TEMPORAL RELATIONSHIPS BETWEEN FINANCIAL SIGNALS

Dawei Cheng[†], *Yi Tu*[†], *Zhibin Niu*[‡] *and Liqing Zhang*[†]

[†]Key Laboratory of Shanghai Education Commission for Intelligent Interaction and Cognitive Engineering, Dep. of Computer Science and Engineering, Shanghai Jiao Tong University, China

[‡] School of Computer Software, Tianjin University, China

ABSTRACT

Portfolio risk control is vital to financial institutions: investors seek to build equities with the highest return but with minimum risk. However, a general phenomenon is significant comovement among many financial signals, such as stocks and futures. One investment strategy is to choose less correlated assets. Classic approaches quantifying such relationships in real financial markets make it difficult to exclude factors such as market trends and autocorrelation. In this paper, we propose a signal process perspective for quantitative measurement. A machine learning based algorithm is designed to model returns, taking account of market sensitivity, autocorrelation, and relationships with other stocks. We then extend the model training algorithm using regularized least square and gradient descent to estimate parameters. A penalty factor is designed in the optimization function to address extreme large negative returns. After denoising common factors, the learned *pure* relationship parameters are applied to construct a relationship matrix. Finally, we use this matrix to build portfolios by constrained optimization. Empirical experiments on two stock datasets show that the proposed method outperforms several state-of-the-art methods in terms of mean average precision and cumulative returns.

Index Terms— financial signal, temporal relationship, factor model, portfolio risk.

1. INTRODUCTION

The development of financial markets means that quantitative measurement is essential for modern investment activities. Any investor who holds a portfolio of financial assets wants to find better ways to control risks. Commonly, higher returns are in accordance with larger risks [1]. Based on different situations, investors tend to confront two main issues: 1). Given a tolerant risk ratio, maximize the return of a portfolio. 2). Given an expected return, minimize the investment risk. Both situations require effectively curbing the comovement among portfolios [2, 3]. Therefore, selecting less interrelated equities is becoming essential, as is finding a new way to diagnose the temporal relationship among financial signals.

The first step requires modeling the return of financial series [4, 5]. Machine learning techniques are usually applied to infer the distribution of variations by training with historical data [6–8]. Finding the relationship among financial signals is a key process of the above methods, either by building a correlation or similarity matrix [9], [10] or by training a model [11–13] from historical distribution.

These methods are hampered by two drawbacks: 1). The relationship is built by historical data, which fails to exclude *noisy* signals, such as market trends and autocorrelations. 2). In order to curb the risk, finding the *pure* relationship in large losses is the key problem. But above methods give equal weight to both large and small losses.

Therefore, we propose a similarity measurement from downside in financial signals to avoid large losses in portfolios. First, we describe daily returns by a regression model. Then we utilize the relationships as a regression factor, while improving it's purity by adding other two factors, market sensitivity and autocorrelation. The large losses penalty parameter φ is designed to learn the relationship during downside. In addition, as market sensitivity is volatile and the relationship among stocks tends to remain smooth, we propose a regularized cost function that responds quickly to market trends while simultaneously stabilizing the relationship among stocks.

In brief, the main contributions in this paper can be summarized as follows:

- Proposing a novel regression model to fit daily returns by introducing the market sensitivity factor α, autocorrelation parameter γ, and relationships of equities $\omega_{x,y}$.
- Providing a signal process perspective for financial engineering. To the best of our knowledge, this is the first study of exploring *pure* relationship to curb large losses among financial signals by denoising market sensitivity and autocorrelation.

The work was supported by the National Basic Research Program of China (2015CB856004), the Key Basic Research Program of Shanghai Science and Technology Commission, China (15JC1400103, 16JC1402800)

- Extending the model training algorithm using regularized least square to estimate the parameters. A vector of λ is applied to balance the frequently changing market quotations and relatively stable connections among stocks.

2. PROPOSED METHODS

Based on the above intuition, we propose a TRS(**T**emporal **R**elationships between Financial **S**ignals) algorithm for modeling daily returns by necessary factors, autocorrelation and relational parameters. We then train the model using the regularized least square method. Moreover, gradient descent algorithm is used to minimize regularized square errors. The learned relational model is then applied to build a constrained portfolio.

2.1. Problem Formulation and Modeling

Let $P_{t,i}$ denote the price of stock i on day t. In general, financial models focus on returns rather than on prices [14], because returns are a complete summary of both profit and loss and they have both theoretical and empirical properties that make them more attractive than prices [15].

The *net return* $x_{t,i}$ of stock i on day t is defined as:

$$x_{t,i} = \frac{P_{t,i}}{P_{t-1,i}} - 1 \tag{1}$$

Then, the *cumulative return* of recent T days is simply defined by Eq. 2.

$$x_{t,i}(k) = \prod_{k=1}^{T-1} (x_{t-k} + 1) - 1 \tag{2}$$

Given the historical time series data of n equities on day t, the *net return* matrix is stated: $X = \{x_{t,i}\} \in \mathbb{R}^{T \times n}$, where $1 \leq t \leq T, 1 \leq i \leq n$. We then designed a machine learning based approach, named TRS, to model the statistical properties of future returns in financial assets. This is defined as:

$$\widehat{x_{t,i}} = \sum_{k=1}^{3} \alpha_{i,k} \cdot d_{\hat{t},k} + \sum_{m=1}^{p-1} \gamma_{i,m} \cdot x_{t-m,i} + \sum_{j=1;j\neq i}^{n} \omega_{i,j} \cdot u_{t,j} \tag{3}$$

where $d_{\hat{t}} = \{1, x_{\hat{t}}, \epsilon_t\}$, $x_{\hat{t}}$ and ϵ_t are the market index factor and the markets' large event adjustment parameter.

$$u_{t,j} = x_{t,j} - \sum_{k=1}^{3} \alpha_{j,k} \cdot d_{\hat{t},k} - \sum_{m=1}^{p-1} \gamma_{j,m} \cdot x_{t-m,j} \tag{4}$$

In this model, we learn the parameters of stock i on a sequence of historical data; these include the equities comprehensive return parameter $\alpha_{i,k}$, autocorrelation factor $\gamma_{i,m}$ and relationships with others $\omega_{i,j}$.

2.2. Customized Estimation by Least Squares

Least squares [16, 17] is a standard approach for regression and classification problems, and is used to find the minimized sum of the squares of errors. In the proposed TRS model, using historical training data (return matrix $X \in \mathbb{R}^{T \times n}$), we improve the standard least square cost function by adding the customized factor φ:

$$\min_{\alpha_*, \gamma_*, \omega_*} \sum_{t=1, i=1}^{T,n} \left(e^{-(x_{t,i}-\varphi)^2} \left\| x_{t,i} - \hat{x_{t,i}} \right\|_2^2 \right) \tag{5}$$

Factor φ is designed to maximize the weight of large losses in equities by highlighting the cost boundary, and is estimated by cross-validation. We use a minus number in the exponential function because TRS is designed to give greater weight to negative returns.

For standard regression problems as $X \cdot \omega = Y$, X is the feature matrix and Y is the label vector. Then, the learned parameter ω is given by:

$$\hat{\omega} = (X^T X)^{-1} X^T Y \tag{6}$$

In order to tackle over-fitting, the regularized parameter λ is introduced in least square solutions. Then we have:

$$\hat{\omega} = (X^T X + \lambda I)^{-1} X^T Y \tag{7}$$

As regularized least squares give equal penalty coefficients to all parameters, we propose another regularized method; this can add a penalty to large changes in relational coefficients while dynamically adjusting to market changes speedily. Therefore, by adding a regularization term, the optimization problem becomes:

$$\min_{\alpha_*, \gamma_*, \omega_*} \sum_{i=1}^{n} \left(\sum_{t=1}^{T} e^{-(x_{t,i}-\varphi)^2} \left\| x_{t,i} - \hat{x_{t,i}} \right\|_2^2 + \sum_{k=1}^{|\Theta_{i,T}|} \lambda_{c1,k} \Theta_{i,T,k}^2 + \sum_{l=1}^{|\Theta_{i,T}|} \lambda_{c2,l} (\Theta_{i,T,l} - \Theta_{i,T-1,l})^2 \right) \tag{8}$$

Where $\Theta_{i,T} = \{\alpha_{i,*}, \gamma_{i,*}, \omega_{i,*}\}$ and $\Theta_{i,T-1}$ are the parameters learned for stock i on (T-1) day. Vector λ_{c1} and λ_{c2} are the customized penalty parameters. They are estimated by cross-validation. φ is the large losses penalty parameter, because the maximum ambiguity is at the boundary. We then use a gradient descent algorithm to minimize this cost function. For each $x_{t,i} \in \mathbb{R}$, the parameters are updated by:

$$\alpha_{i,k} \leftarrow \alpha_{i,k} + \eta \left(F_{t,i} \cdot d_{\widehat{t},k} - \lambda_{c1}^{k} \cdot \alpha_{i,k}^{T} - \lambda_{c2}^{k} \cdot (\alpha_{i,k} - \alpha_{i,k}^{(T-1)}) \right), \quad k = 1,2,3$$
$$\gamma_{i,m} \leftarrow \gamma_{i,m} + \eta \left(F_{t,i} \cdot x_{t-m,i} - \lambda_{c1}^{3+m} \cdot \gamma_{i,m}^{T} - \lambda_{c2}^{3+m} \cdot (\gamma_{i,m} - \gamma_{i,m}^{(T-1)}) \right), \quad m = 1,2,...,p$$
$$\omega_{i,j} \leftarrow \omega_{i,j} + \eta \left(F_{t,i} \cdot u_{t,j} + F_{t,j} \cdot u_{t,i} - 2 \cdot \lambda_{c1}^{3+p+j} \cdot \omega_{i,j} - 2 \cdot \lambda_{c2}^{3+p+j} \cdot (\omega_{i,j} - \omega_{i,j}^{(T-1)}) \right), \quad j = 1,2,...,n; j \neq i \tag{9}$$

In this procedure, $F_{t,i} = e^{-(x_{t,i}-\varphi)^2} \cdot (x_{t,i} - \widehat{x_{t,i}})$. Besides, η is the learning rate that is dynamically adjusted by line search.

Table 1. Relational Matrix C of TRS.

	1	⋯	i	⋯	j	⋯	n
1	σ_1^2	⋯	$\omega_{1,i}$	⋯	$\omega_{1,j}$	⋯	$\omega_{1,n}$
⋯	⋯	⋯	⋯	⋯	⋯	⋯	⋯
i	$\omega_{i,1}$	⋯	σ_i^2	⋯	$\omega_{i,j}$	⋯	$\omega_{i,n}$
⋯	⋯	⋯	⋯	⋯	⋯	⋯	⋯
j	$\omega_{j,1}$	⋯	$\omega_{j,i}$	⋯	σ_j^2	⋯	$\omega_{j,n}$
⋯	⋯	⋯	⋯	⋯	⋯	⋯	⋯
n	$\omega_{n,1}$	⋯	$\omega_{n,i}$	⋯	$\omega_{n,j}$	⋯	σ_n^2

Once we obtain the trained parameters $\omega_{i,j}$, a relationship matrix can be built as shown in Table. 1. We then utilize this matrix C to build the optimal portfolio. For a given expected return x_e, investors want to find the minimum variance of the portfolio. Therefore, the weights w of the equities in a portfolio are formulated by optimizing problems as Eq. 10.

$$\begin{aligned} &\min_{w} \frac{1}{2} w^T C C^T w \\ &\text{subject to: } \sum_{i=1}^{n} \overline{x_i} w_i \geq x_e \\ &\sum_{i=1}^{n} w_i = 1; w_i \in [0,1], i \in \{1,2,\cdots,n\} \end{aligned} \tag{10}$$

Where, $\overline{x_i}$ is the expected return of stock i, and C is the relationship matrix built by the parameters learned above. As CC^T is positive semi-definite, the target function is convex; consequently, quadratic programming could be applied to find a global optimal solution in this problem.

2.3. Convergence and Complexity Analysis

We use at least T days of historical data to train our model in order to update the autocorrelation coefficient effectively. We initialize the parameters randomly for the first training process; we then use the previous day's parameters as the original state of current day. Since the time windows series data slide is only one day per time, convergence can always be achieved within a few iterations.

For the storage complexity, during the iterations, we need to hold $\mathbb{R}$, $x_{\widehat{T}}$ and ϵ_T in the memory, which costs $O(Tn)$ space. Θ_T and $\{u_{t,i}\}$ are also needed when updated, bringing an additional $O(Tn+n^2)$ space. Here T is the number of days and n is the number of equities. Therefore, the total storage complexity is $O(2Tn + n^2)$.

For the computational complexity, in every iteration, we need $O(Tn)$ time to compute $\{u_{t,i}\}$, $O(Tn^2)$ time to update Θ_T. Thus, the computational complexity of one iteration is $O(Tn^2)$, while the total for the TRS algorithm over m iterations is $O(Tmn^2)$.

3. EXPERIMENTS

3.1. Datasets

We run our experiments on all companies currently listed in CSI 300 and $S\&P$ 500 that have traded between January 1, 2005 and January 1, 2015. These companies are derived from ten sectors: Industrials, Health Care (H.C.), Information Technology (I.T.), Financial, Utilities, Materials, Consumer Staples, Consumer Discretionary, Energy and Telecommunication Services.

Based on financial domain knowledge and statistical information from our datasets [18], we define that there is an *event* if one stock's return in a day is less than minus 9 percent, so φ is set to 0.09. If more than two equities experienced *events* in a day, we call this is an *event day*. In total, there are 238 *event days* among CSI 300 companies and 179 *event days* for $S\&P$ 500 companies. It should be noted that the average returns on *event days* differ significantly: C4.322% for CSI 300 and C1.150% for $S\&P$ 500.

3.2. Top K Negative Return Experiments

In this test, our proposed method is applied to $S\&P$ 500 and CSI 300 datasets. Using at least 50 days of historical data, we train and update the parameters daily. On the next day, given a set of equities returns A, we predict the rest set of equities returns B (which is disjoint with A), comparing the predicted results with the true values. This test is designed to evaluate the accuracy of relationship matrix C, which is constructed by $\omega_{a,b}, \{a \in A, b \in B\}$. As mentioned in Section 2, this papers main topics are modeling and learning relationship parameters. Therefore, after training the proposed model with historical data on all equities, we could obtain $\omega_{a,b,t-1}$ and matrix C_{t-1} in day t-1. Then, given returns of A on the next day, the returns of equities in the rest set B could be predicted by the model in Eq. 3. For example, pick b in day t from test set B, its return is calculated by Eq. 11.

$$\widehat{x_{t,b}} = \sum_{k=1}^{3} \alpha_{b,k} \cdot d_{\widehat{t},k} + \sum_{m=1}^{p-1} \gamma_{b,m} \cdot x_{t-m,b} + \sum_{a=1}^{|A|} \omega_{b,a,t-1} \cdot u_{t,a} \tag{11}$$

Then, we compare $\widehat{x_{t,b}}$ with real values to measure the models precision; more specifically, the precision of the relationship parameters $\omega_{a,b}$.

The model is trained on event days in each sector. We compare our method with AR, $ARMA$, $ARIMA$, $GARCH$, CR, PCR and FAC [19]. In TRS and FAC, we use market factor $d_{\widehat{t}}$ and returns of set A on the current day, $\omega_{a,b,t-1}$ on the previous day and lagged return x_{t-m} to predict $x_{t,b}$, the return of b on day t. For CR and PCR, the relationship on the previous day and returns of set A on the current day t are applied. Accordingly, lagged return x_{t-m} is used in $AR(1)$, $ARMA(1,1)$, $ARIMA(1,1,1)$ and $GARCH(1,1)$ models. We use ω from the previous day and use the current day market factors and returns of the portfolio set to evaluate models.

The main purpose of this paper is to learn relationships among financial signals; therefore, the key is to apply $\omega_{a,b,t-1}$ from the previous day rather than the current day. For market sensitivity factors and returns of set A, as known and given values, these are updated by the current day values.

We evaluate our methods on the *event days*, then exclude sectors that have fewer than 10 such days. We randomly split data to 80 percent for training and the rest for test. We run the model 100 times, recording the mean and variance of MAP scores [20]. The results are shown in Fig. 1. TRS outperforms seven other methods in all sectors of both markets. This advantage is more significant in Chinese stock market CSI 300 companies, as presented in Fig. 1(a).

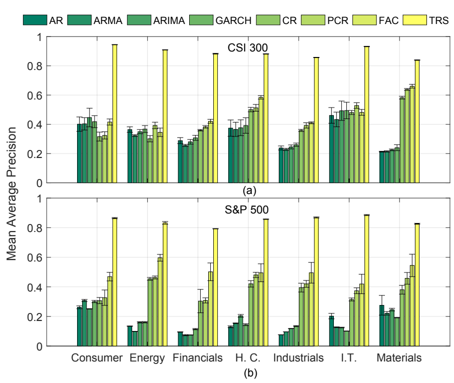

Fig. 1. MAP scores of 100 test runs in seven sectors of CSI 300 and $S\&P$ 500.

3.3. Build Portfolio with Relationship Matrix

Curbing large losses in stock portfolios is the major application of this method, which is known as a tradeoff between desired return and risk. For a portfolio construction problem, given expected return x_e, the minimum variance portfolio is the optimal portfolio, which is optimized by Eq. (10).

In conventional methods, C is the covariance of returns and $\overline{x_i}$ is the desired return of stock i. Normally, these are calculated from historical data. In order to build a portfolio as formulated in Eq. 10, correlations among stocks are needed. Thus, we compare the performance with the covariance matrix, referred to as COV, and FAC.

The expected return is negative correlated with portfolio diversity in a limited period. Thus, we utilize the desired expected return x_e to control the amount of diversification. Given historical data, maximum expected returns and minimum risk causal relative are major targets in this experiment. We apply the model to two sectors with the most events. We compare TRS with FAC and COV, and the cumulative returns in the financial sector of $S\&P$ 500 are shown in Fig. 2. Between 2005 and 2008, our model is comparable with two other approaches, FAC and COV. However, when the market situation is tough in 2008, TRS and FAC models outperform COV soon afterwards. Since mid-2009, the cumulative return of TRS is consistently larger than FAC. This demonstrates the effectiveness of our model in building optimal portfolios.

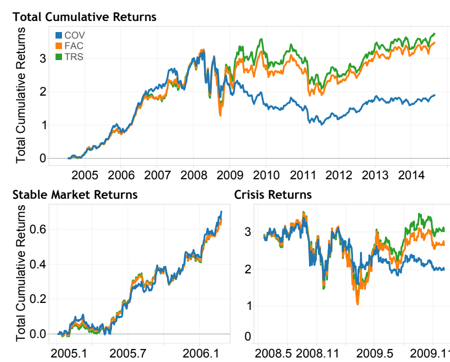

Fig. 2. Cumulative returns of portfolios in the financial sector, as built by our methods, COV and FAC.

4. CONCLUSION

In this paper, we have proposed a temporal relationship learning approach on financial signals by denoising market sensitivity and autocorrelation. The learned relationship parameters among financial signals could effectively control large losses of a built portfolio. Specifically, we model daily returns by autocorrelation and factor methods. The order of autocorrelation is optimized by cross-validation; we then train the model by regularized least squares and gradient descent methods. Two other factors are designed for quick response to market fluctuations and its relationship with others. Then we construct a matrix by relationship parameters that are later utilized to build portfolios. Finally, two kinds of experiments are applied to demonstrate the effectiveness of this model. The results show that our method consistently outperforms other state-of-the-art methods.

5. REFERENCES

[1] Kee-Hong Bae, G Andrew Karolyi, and Reneé M Stulz, "A new approach to measuring financial contagion," *Review of Financial studies*, vol. 16, no. 3, pp. 717–763, 2003.

[2] Ke Zhou, Xin Li, and Hongyuan Zha, "Collaborative ranking: improving the relevance for tail queries," in *Proceedings of the 21st ACM international conference on Information and knowledge management*. ACM, 2012, pp. 1900–1904.

[3] Kristin J Forbes and Roberto Rigobon, "No contagion, only interdependence: measuring stock market comovements," *The journal of Finance*, vol. 57, no. 5, pp. 2223–2261, 2002.

[4] René M Stulz, "Rethinking risk management," *Journal of applied corporate finance*, vol. 9, no. 3, pp. 8–25, 1996.

[5] Kevin Dowd, Andrew JG Cairns, and David Blake, "Mortality-dependent financial risk measures," *Insurance: Mathematics and Economics*, vol. 38, no. 3, pp. 427–440, 2006.

[6] Christopher M Bishop, "Pattern recognition," *Machine Learning*, vol. 128, 2006.

[7] Goce Ristanoski, Wei Liu, and James Bailey, "A time-dependent enhanced support vector machine for time series regression," in *Proceedings of the 19th ACM SIGKDD international conference on Knowledge discovery and data mining*. ACM, 2013, pp. 946–954.

[8] Yiyong Feng, Daniel P Palomar, et al., "A signal processing perspective on financial engineering," *Foundations and Trends® in Signal Processing*, vol. 9, no. 1–2, pp. 1–231, 2016.

[9] Hsin-Yu Ha, Yimin Yang, Fausto C Fleites, and Shu-Ching Chen, "Correlation-based feature analysis and multi-modality fusion framework for multimedia semantic retrieval," in *2013 IEEE International Conference on Multimedia and Expo (ICME)*. IEEE, 2013, pp. 1–6.

[10] Jonathan Masci, Michael M Bronstein, Alexander M Bronstein, and Jürgen Schmidhuber, "Multimodal similarity-preserving hashing," *IEEE transactions on pattern analysis and machine intelligence*, vol. 36, no. 4, pp. 824–830, 2014.

[11] Graham W Taylor, Geoffrey E Hinton, and Sam T Roweis, "Two distributed-state models for generating high-dimensional time series," *Journal of Machine Learning Research*, vol. 12, no. Mar, pp. 1025–1068, 2011.

[12] Philémon Brakel, Dirk Stroobandt, and Benjamin Schrauwen, "Training energy-based models for time-series imputation.," *Journal of Machine Learning Research*, vol. 14, no. 1, pp. 2771–2797, 2013.

[13] Shuai Xiao, Mehrdad Farajtabar, Xiaojing Ye, Junchi Yan, Xiaokang Yang, Le Song, and Hongyuan Zha, "Wasserstein learning of deep generative point process models," in *Advances in Neural Information Processing Systems*, 2017, pp. 3250–3259.

[14] John Y Campbell, Andrew Wen-Chuan Lo, Archie Craig MacKinlay, et al., *The econometrics of financial markets*, vol. 2, princeton University press Princeton, NJ, 1997.

[15] Jeroen Kerkhof and Bertrand Melenberg, "Backtesting for risk-based regulatory capital," *Journal of Banking & Finance*, vol. 28, no. 8, pp. 1845–1865, 2004.

[16] Fong-Ching Yuan and Chao-Hui Lee, "Using least square support vector regression with genetic algorithm to forecast beta systematic risk," *Journal of Computational Science*, vol. 11, pp. 26–33, 2015.

[17] Martin Einemo and Hing Cheung So, "Weighted least squares algorithm for target localization in distributed mimo radar," *Signal Processing*, vol. 115, pp. 144–150, 2015.

[18] C Ronald Sprecher and Mars A Pertl, "Large losses, risk management and stock prices," *Journal of Risk and Insurance*, pp. 107–117, 1983.

[19] Gartheeban Ganeshapillai, John V Guttag, and Andrew Lo, "Learning connections in financial time series.," in *ICML (2)*, 2013, pp. 109–117.

[20] Emine Yilmaz and Javed A Aslam, "Estimating average precision with incomplete and imperfect judgments," in *Proceedings of the 15th ACM international conference on Information and knowledge management*. ACM, 2006, pp. 102–111.