On the use of graphical models for valuation of financial assets

Daniel Martin (dlmartin) [1] **Siddharth Satpathy (ssatpat1)** [1] **Jueheng Zhu (juehengz)** [1]

Abstract

The bond market is very important to the economy. Compared to equities, bonds have lower liquidity and transparency; hence, there is less public data available. Adding information from correlated assets with greater liquidity would help increase forecasting accuracy for bond prices. In this work, we construct a pipeline for bond forecasting with factors. We choose factors that are considered to affect bond prices in finance literature, recover the structure of a graphical model, and use the resulting factors for forecasting.

1. Overview

Our project is on using graphical models for valuing financial assets, specifically corporate bonds.

Graphical models are commonly used in the field of econometrics for predicting macroeconomic risk, as they can model a firm's risk of bankruptcy (default) conditioned on external factors. Hidden Markov Models are referred to as "regime-switching models" for their use in predicting the onset of bull or bear markets. Besides risk modeling, most applications of graphical models have been used for stock price and risk forecasting.

The markets for less-liquid assets such as bonds are even larger than stocks: the global bond market is valued over \$100 trillion, as compared to S&P's estimate of \$64 trillion for the global equity market (Federated Investors, Inc., 2017). However, they are less active. This leads to less transparency. As we have access to prices for bonds and other assets, we intend to apply extant stock models to these datasets.

Traditionally, when economists do asset value prediction, they first come up with a theory, fix the regression structure, and then figure out the factor loads. Here we use an alternative approach. We investigate the variable correlations, feed them into a graphical model and let the data tell us the prediction structure. This is a novel application.

2. Literature Survey

Semi supervised learning (SSL) based algorithms have been used in Park & Shin (2012) to investigate interrelations and complexities between factors through a network. The suggested method connects individual networks based on similarities between factors and extracts influence of the final similarities in the connected input factors and response factors as its prediction value. The proposed stock prediction model model considers causal complexities and interrelations in various economic indices and factors like exchange rates, oil prices, stock prices in other countries, money interest rates and economic situations by joining time series data to SSL. The structure of this SSL prediction model is in the form of a graph where nodes represent time series variables that influence stock prices. And, edges between nodes (adjacency matrix) denote connection strengths between two sets of time series. Connection strengths in this adjacency matrix can take binary values or Gaussian values depending on Euclidean distance between nodes. Predictions on unlabeled nodes are made after learning based on similarity matrices calculated at previous time steps. As such this SSL method relies on not only input and target variables but also interrelations between them. Tests of this method on stocks listed in KOSPI returned an area under curve (AUC) value of 0.72. The authors extended this work by proposing a hierarchical graphical model (Park & Shin, 2013). The proposed model has two layers. The top layer contains global indicators such as the directions of market indices and important commodities, while the bottom layer contains the individual stocks in the Korean Stock Exchange (KOSPI). A latent layer is included in between.

In Cerchiello et al. (2017), the authors introduce a novel framework, based on graphical Gaussian models, which can estimate systemic risks with stochastic network models based on two different sources, *viz.*, financial markets and financial tweets, and propose a way to combine them, using a Bayesian approach. Such a model has been shown to be able to derive the network that best describes interrelationships between financial institutions on the basis of

[1]Carnegie Mellon University, Pittsburgh, PA 15213, USA.

10708 Class Project, Spring, 2019.

their market price data, and hence, the risk contagion. Twitter data gives us a source of data that can supplement market prices. The data considered in this paper comprises of financial tweets on a number of banks which have been collected on a daily basis. Daily variation of this 'bank sentiment' from Twitter is used to build a undirected Gaussian graphical model which is combined with graphical models based on market data, by means of a Bayesian approach. An advantage of this model lies in the fact that it can include systemic risk models for institutions that are not publicly listed, using just the tweet data component.

In Zhu & Barber (2015), the authors use hierarchical models to estimate multiple related Gaussian graphical models (GGM) on the same sets of variables. The said hierarchical model encapsulates our prior belief about the shared structure across multiple graphs. This hierarchical model is optimized to find the maximum a posteriori (MAP) estimate of precision matrix Ω. The ensuing optimization combines a likelihood term with non-convex log-shift penalty functions. The use of non-convexity in penalty helps in threshold weak signals to zero, while leading to reduced shrinkage on edges with strong signals. This leads to good selection and estimation performances. One thing worth noting here is that, the mentioned optimization problem is convex under some mild conditions even with the use of non-convex penalty. Experiments of this method are done with stock price and bikeshare data. Comparisons of the results of this method with methods that use convex penalty functions shows that the use of non-convexity in penalty functions leads to less bias on strong signals, thereby making it possible to obtain good selection and estimation performance at the same time.

In Cerchiello & Giudici (2016), the authors propose a new Gaussian graphical model to estimate systematic risks. This is the first work where risk estimation has been done through the use of financial market and balance sheet data in a combined perspective. In this model, the conditional dependencies between financial institutions are reduced to correlations between countries, and correlations between institutions within countries. These correlations are then used to find systemic country effects and idiosyncratic bank-specific effects. The model proposed in this work focuses on structural learning to infer the network model that best describes interrelationships between financial institutions based on the given data. Experiments of this model are done to estimate systemic risks of large banks in the European Union. Such an analysis has been shown to accurately identify central institutions whose failure could result in breakdown of the entire banking system.

Denev (2015) focuses on modeling financial networks as PGMs. With proper calibration, Bayesian Nets, Markov Random Fields, Chain Graphs and Directed Cyclic Graphs are used to capture the conditional independence and further predict both probabilities of default and asset returns.

As most financial data are time series, special technique is required. Reeson (2009) employees Dynamic Linear Models to 30 funds from Vanguard and does portfolio analysis using Gaussian graphical models.

Filiz et al. (2012) uses toric and Ising models for correlated defaults prediction. It explains default dependence, fat-tail of the loss distribution and implied correlation smile, while providing a calibration algorithm based on maximum likelihood estimation.

Belloni et al. (2016) propose a way to do financial risk management using prediction from quantile graphical models. These models capture non-Gaussian settings in econometric applications. They also show that QGM can represent tail interdependence, which is extremely useful to model extreme events.

Giudici & Spelta (2013) compares marginal correlation networks, static Bayes nets, dynamic Bayes nets (also contains a vector autoregressive component), and Granger causality networks for modeling time series. The application is the estimation of systematic risk for different countries using total liabilities of international banks.

Balcilar et al. (2015) uses economic policy uncertainty (EPU) measures for South Africa and twenty other markets combined with twenty other factors, including index returns and macroeconomic indicators. From a Gaussian graphical model, the authors estimated the rolling-window posterior probabilities of excess returns conditioned over each of these factors.

Hidden Markov models are also commonly used in economic forecasting. Salhi et al. (2016) show HMM's performing well against GARCH, while Alvi (2018) demonstrates crude oil price forecasting using HMM's and Belief networks.

Finally, we can investigate graphical models that use non-classical probabilities. Moreira et al. (2018) compares classical Bayes nets with quantum (non-classical) Bayes nets by using them to model the process of an online loan application. While quantum networks can also become intractable for large state spaces, they are able to represent more-complex interference effects between variables.

3. Methodology

In our work, we intend to look at the effects of existing factors on future bond prices. This is inspired by common approaches used by financial analysts: companies are affected not only by marketwide forces but also by sector-specific forces. For instance, investors always note a company's level of debt when measuring its credit risk. Companies with a higher debt-to-asset ratio can be more risky; hence, investors may demand a higher return when purchasing the company bonds.

As a company may be affected by many sector-specific factors, we have decided to focus on companies that operate primarily within a single sector. We chose the petroleum industry as it is centered around crude oil, a widely-traded commodity.

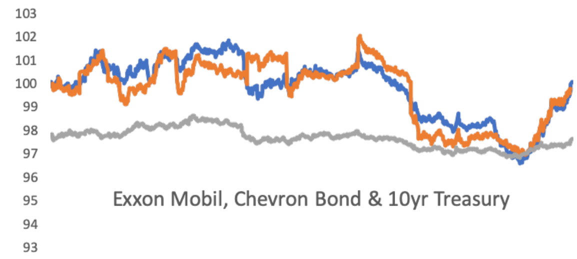
From a modeling perspective, one can begin by examining whether the available factors are correlated in order to estimate an undirected graphical model. For this purpose, we chose to apply the graphical Lasso (Hastie et al., 2015). We will first describe how we obtained and preprocessed our data, then follow with a description of the various estimation methods we applied.

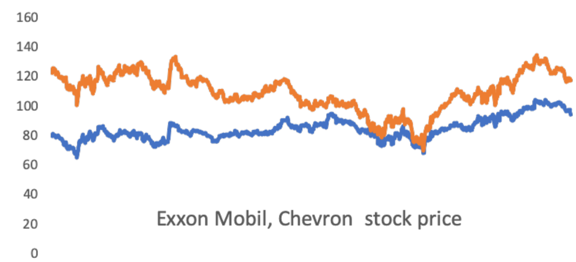
4. Data collection & processing

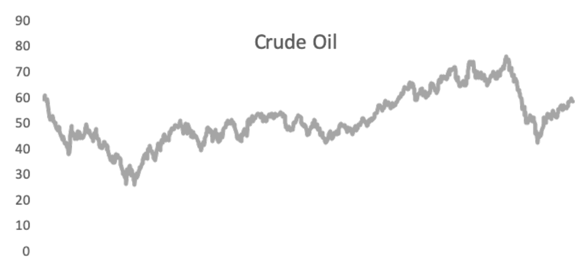
All data are drawn from Bloomberg terminal. We extract the following nine time series:

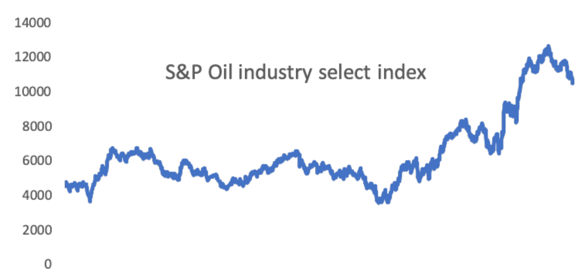
- Oil and gas company, Chevron Corp. US 5-year benchmark bond daily last price, 3/26/2014-3/25/2019
- Chevron US stock daily last price, 3/26/2014-3/25/2019
- Oil and gas company, ExxonMobil Corp. US 5-year benchmark bond daily last price, 06/19/2015-03/25/2019
- ExxonMobil US stock daily last price, 3/26/2014-3/25/2019
- S&P Oil industry select index last price, 3/26/2014-3/25/2019
- US government 10-year treasury quotes, 3/26/2014-3/25/2019
- West Texas Intermediate (WTI) crude oil daily spot commodity price, 03/26/2014-03/25/2019
- Chevron Corp. total debt/total assets ratio, quarterly, 2014Q1-2018Q4
- ExxonMobil Corp. total debt/total assets ratio, quarterly, 2014Q1-2018Q4

We draw them in the following figures.









Clearly they are all closely related, which is the motivation for linking them using graphical models.

These are three different types of time series: trading quotes, asset prices, and financial statements statistics. Before we train our model on these data we need to process them with noise-filter, auto-correlation analysis, and seasonality analysis, respectively.

Corporate bond price (or equivalently, bond yield) is considered to reflect the market's belief of the the default risk of the bond issuer. Similarly, treasury yield represents investors' confidence in the US financial system. However, last price quotes incorporate trading noises that may mask the true underlying information. We use a plain version of Kalman filter algorithm (see Welch & Bishop (1995)) to unravel this time series, and feed our model with he processed series.

Commodity price of crude oil and the stock prices of ExxonMobil and Chevron are all asset prices. They share several significant characteristics: prices everyday is highly correlated and tend to be persistent than in standard random walk model; the distribution of price return is fat-tailed compared to standard normal distribution; the volatility of prices will be clustered around certain values. All factors tend to interact with each other, which makes it impossible to separately address these issues. We employ an Autoregressive integrated moving average (ARIMA) model for the conditional mean, and a Generalized AutoRegressive Conditional Heteroskedasticity (GARCH) model for the conditional variance. For hyper-parameter setup we follow Mohammadi & Su (2010). Since the S&P Oil industry select index is just a weighted average of the stock prices in oil industry, the processing method is the same as with individual asset prices.

Financial statements statistics is highly affected by the seasonality of market supply and demand. A simple way to cope with this issue is a combination of AR and MA models. First, we regress debt/asset ratio on the moving average of the ratio of the same quarters in the last few years. We then obtain the seasonality effect. Finally, we adjust for it whenever oscillation around the underlying trend arises.

5. Code

It is essential to keep track of conditional independences between nodes in graphs in machine learning and statistics. Gaussian distributions are completely defined by two variables *viz.* their mean and variance. Zero correlations between these variables will imply statistical independence. One can use a similar analogy between between conditional independence and the inverse covariance for Gaussian distributions. Such an equivalence can be used in probability distributions like multivariate Gaussians to estimate inverse covariance matrices. Indices of zero and nonzero values in an inverse covariance matrix (for a network where variables are modelled by Gaussian distributions) can give us information about conditional independence or lack thereof. Such estimation is widely used to uncover conditional dependencies and independences in gene regulatory networks in cellular biology and neural interactions. In our project, we seek to analyze conditional dependencies and independences in networks which contain information about bond prices from two companies (ExxonMobil and Chevron) and oil prices. We use a software package called '`skggm`'(Laska & Narayan, 2017) which is geared toward Gaussian graphical models for this exercise.

`skggm` has the following submodules for estimation of inverse covariance matrix:

1. **QuicGraphicalLasso:** *QuicGraphicalLasso* is a variant of QUIC (QUadratic approximation of Inverse Covariance matrices) (Hsieh et al., 2014) which is a second-order algorithm that solves the ℓ_1-regularized Gaussian maximum likelihood estimation (MLE). For n independently drawn, p-dimensional Gaussian random samples $X \in \mathbb{R}^{n\times p}$ with sample covariance $\hat{\Sigma}$, it is possible to calculate the maximum likelihood estimate of the inverse covariance matrix Θ by using a graphical lasso method. Given a regularization penality $\lambda > 0$, the ℓ_1-regularized Gaussian MLE for Θ can be written as the following:

$$\hat{\Theta}(\Lambda) = \underset{\Theta>0}{\arg\min}\ \log\det\Theta + \mathrm{tr}(\hat{\Sigma}\Theta) + \sum_{i,j=1}^{p} \lambda_{ij}|\Theta_{ij}| \tag{1}$$

In equation 1 , $\lambda_{ij} \in \Lambda^{p\times p}$ which is a symmetric matrix with non negative entries. To ensure that Θ stays positive definite, we do not penalize the diagonal entries λ_{jj}. For off-diagonal elements ($i \neq j$), we have $\lambda_{ij} = \lambda ji \equiv \lambda\ \forall i \neq j$. Here, Σ is the sample covariance matrix.

In QuicGraphicalLasso, we modify the objective given in equation 1 to the following.

$$\hat{\Theta}(\Lambda) = \underset{\Theta>0}{\arg\min}\ \log\det\Theta + \mathrm{tr}\left[R(\hat{\Sigma})\Theta\right] + \sum_{i,j=1}^{p} \lambda_{ij}|\Theta_{ij}| \tag{2}$$

$R(\hat{\Sigma})$ is given in equation 3.

$$R(\hat{\Sigma}) = \left[\mathrm{diag}(\hat{\Sigma})\right]^{-1/2} \hat{\Sigma} \left[\mathrm{diag}(\hat{\Sigma})\right]^{-1/2} \tag{3}$$

Among the advantages of QuicGraphicalLasso over GraphicalLasso are support for a matrix penalization term and speed.

2. **QuicGraphicalLassoCV:** *QuicGraphicalLassoCV* is an optimized cross-validation model selection implementation of QuicGraphicalLasso. It finds a sparse inverse covariance with cross-validated choice of the $L1$ penalty. As like QuicGraphicalLasso, QuicGraphicalLassoCV implements the Graphical Lasso algorithm with matrix penalization.

3. **QuicGraphicalLassoEBIC:** *QuicGraphicalLassoEBIC* computes a sparse inverse covariance matrix estimation using quadratic approximation andExtended Bayesian Information Criteria (EBIC) for model selection (convenience Class). More information on EBIC can be found in Foygel & Drton (2010).

4. **AdaptiveGraphicalLasso:** *AdaptiveGraphicalLasso* uses a two step estimation procedure.
   - Get an initial sparse estimate.
   - Obtain a new penalization matrix from the original estimate. In this step, the resulting coefficients are used to generate adaptive weights and QuicGraphicalLasso is performed for a refit with these weights.

5. **ModelAverage:** *ModelAverage* works by subsampling the training data and computing a graphical lasso estimate where the penalty of a random subset of coefficients has been scaled. By performing this double randomization several times, ModelAverage assigns high scores to features that are repeatedly selected across randomizations. In essence, ModelAverage is an ensemble meta-estimator which computes different fits with a user-specified estimator and averages the support of the resulting precision estimates. Here, a variant of graphical lasso is implemented in two steps:
   - Get bootstrap samples by randomly subsampling **X**.
   - Draw a random matrix penalty.

6. **Ledoit-Wolf estimator:** Ledoit-Wolf is a particular form of shrinkage, where the shrinkage coefficient is computed using techniques outlined in Ledoit & Wolf (2004).

Figures 1, 2, 3 and 4 show results of covariance and inverse covariance estimators from different model selection method when nine features *i.e. bond prices* for Chevron and Exxon-Mobil, *bond yield percentages* for Chevron and Exxon-Mobil, *stock prices* for Chevron and Exxon Mobil, oil prices, S & P *oil indices* and 10 year *Treasury prices* are used.

Figures 10, 11, 12 and 13 in section A show results of covariance and inverse covariance estimators from different model selection method when five features *i.e.*, *bond prices* from Chevron and Exxon-Mobil, *bond yield percentages* from Chevron and Exxon-Mobil and *oil prices* are used.

In our estimates, we see that the S&P Oil Index has a lower covariance with the other factors besides the Chevron and Exxon stock prices. While we will retain this factor for our current forecasts, it does not add as much information as the other factors.

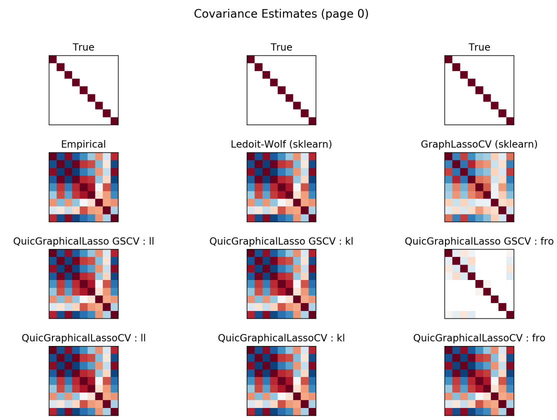


Figure 1. Inverse covariance estimates from standard (empirical), Ledoit-Wolf, GraphLassoCV, QuicGraphicalLasso, QuicGraphicalLasso + GridSearchCV (GSCV) and QuicGraphicalLassoCV methods. The symbols “ll”, “kl” and “fro” depict score metric: log-likelihood, kl and frobenius respectively. Nine features *i.e. bond prices* for Chevron and Exxon-Mobil, *bond yield percentages* for Chevron and Exxon-Mobil, *stock prices* for Chevron and Exxon Mobil, oil prices, S & P *oil indices* and 10 year *Treasury prices* are used during the computation of these estimators.

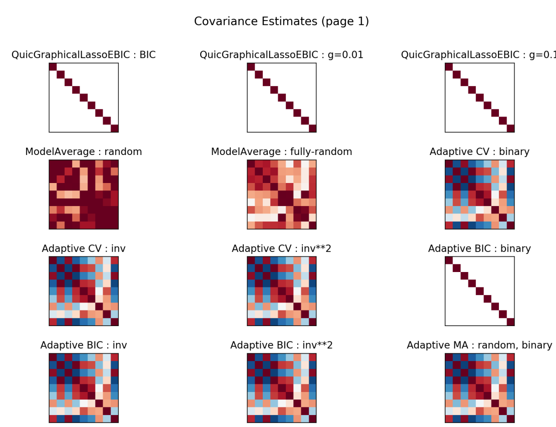


Figure 2. Inverse covariance estimates from standard (empirical), QuicGraphicalLasso-EBIC, ModelAverage, Adaptive CV and Adaptive BIC for different parameters. Nine features *i.e. bond prices* for Chevron and Exxon-Mobil, *bond yield percentages* for Chevron and Exxon-Mobil, *stock prices* for Chevron and Exxon Mobil, oil prices, S & P *oil indices* and 10 year *Treasury prices* are used during the computation of these estimators.

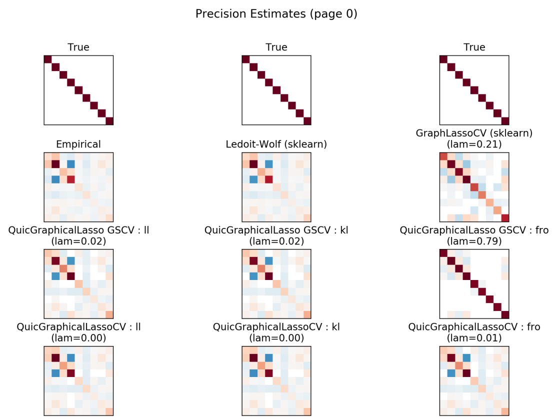


Figure 3. Precision estimates from standard (empirical), Ledoit-Wolf, GraphLassoCV, QuicGraphicalLasso, QuicGraphicalLasso + GridSearchCV (GSCV) and QuicGraphicalLassoCV methods. The symbols "ll", "kl" and "fro" depict score metric: log-likelihood, kl and frobenius respectively. Nine features *i.e. bond prices* for Chevron and Exxon-Mobil, *bond yield percentages* for Chevron and Exxon-Mobil, *stock prices* for Chevron and Exxon Mobil, oil prices, S & P *oil indices* and 10 year *Treasury prices* are used during the computation of these estimators.

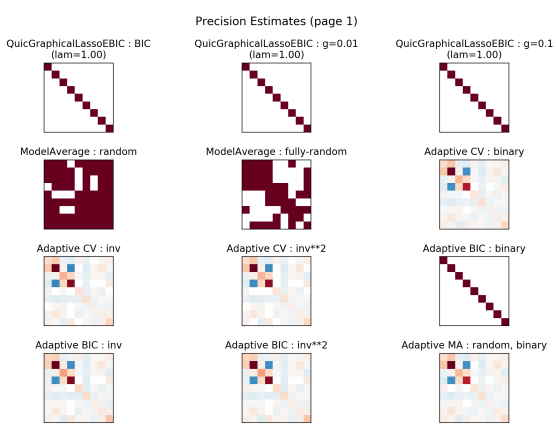


Figure 4. Precision estimates from standard (empirical), QuicGraphicalLasso-EBIC, ModelAverage, Adaptive CV and Adaptive BIC for different parameters. Nine features *i.e. bond prices* for Chevron and Exxon-Mobil, *bond yield percentages* for Chevron and Exxon-Mobil, *stock prices* for Chevron and Exxon Mobil, oil prices, S & P *oil indices* and 10 year *Treasury prices* are used during the computation of these estimators.

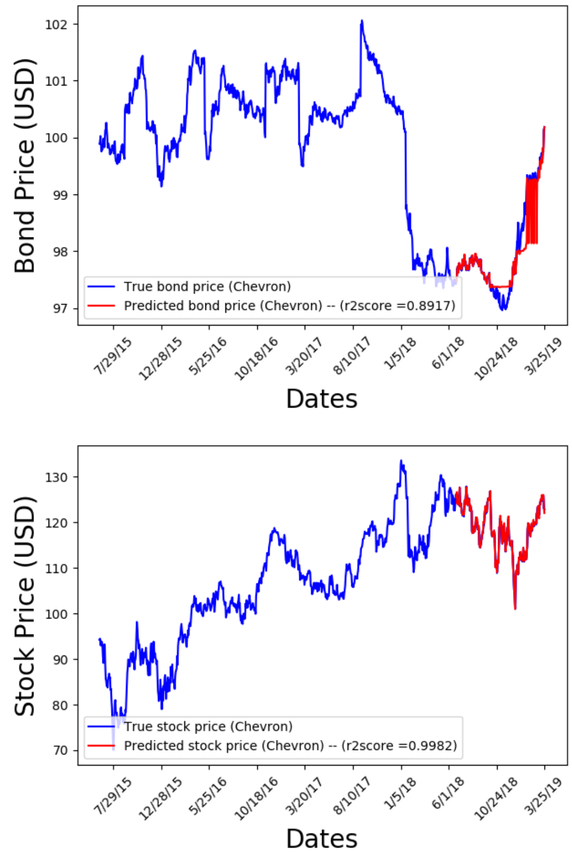


Figure 5. Top plot shows bond prices for Chevron and bottom plot shows stock prices for Chevron. In both plots, the blue lines show actual values of prices while red line show forecasted values of prices. Nine time series based features, viz. *bond prices* for Chevron and Exxon-Mobil, *bond yield percentages* for Chevron and Exxon-Mobil, *stock prices* for Chevron and Exxon Mobil, oil prices, S & P oil indices and 10 year Treasury prices are used for forecasting. The forecasting results shown in these plots are obtained through the use of Random Forest regressor.

6. Forecasts

After we find conditional (in)dependencies using graphical lasso methods, we move to forecast values of various random vectors using information gotten from the use of graphical lasso methods. Two techniques are popular when it comes to forecasting of values of from time series data - 1.⟩ Random Forests and, 2.⟩ Recurrent Neural Networks.

6.1. Random Forests

Random Forest is a popular supervised learning model which can accept a vector $x = (x_1, \ldots, x_k)$ for each observation and predict output values y. They can be used for both regression and classification tasks with the use of multiple decision trees. In our work, we aim to use Ran-

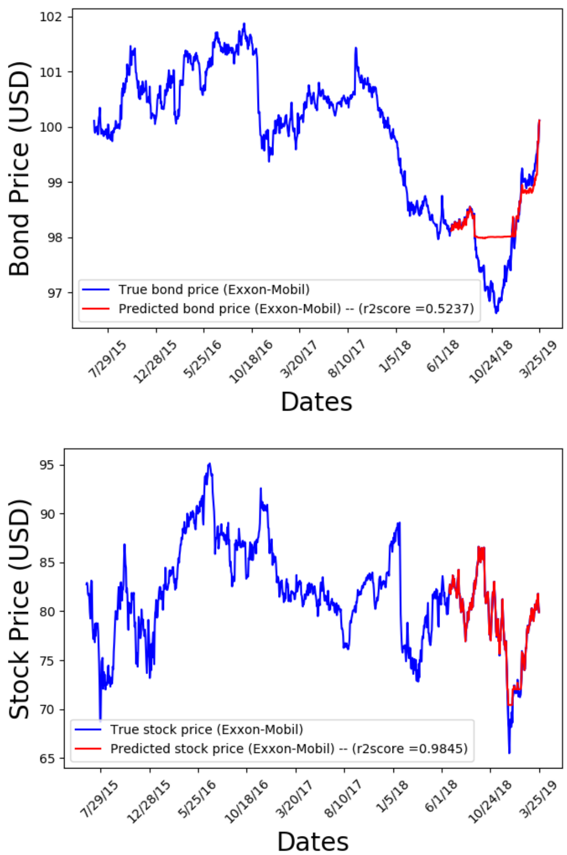


Figure 6. Top plot shows bond prices for Exxon-Mobil and bottom plot shows stock prices for Exxon-Mobil. In both plots, the blue lines show actual values of prices while red line show forecasted values of prices. Nine time series based features, viz. *bond prices* for Chevron and Exxon-Mobil, *bond yield percentages* for Chevron and Exxon-Mobil, *stock prices* for Chevron and Exxon Mobil, oil prices, S & P oil indices and 10 year Treasury prices are used for forecasting. The forecasting results shown in these plots are obtained through the use of Random Forest regressor.

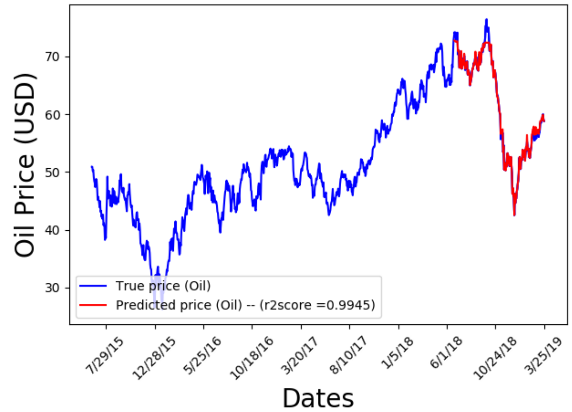


Figure 7. This plot shows oil prices. The blue line shows true prices of oil, while the red line shows predicted prices of oil. Nine time series based features, viz. *bond prices* for Chevron and Exxon-Mobil, *bond yield percentages* for Chevron and Exxon-Mobil, *stock prices* for Chevron and Exxon Mobil, oil prices, S & P oil indices and 10 year Treasury prices are used for forecasting. The forecasting results shown in these plots are obtained through the use of Random Forest regressor.

dom Forest regressors to forecast bond prices, stock prices for Chevron and Exxon-Mobil and prices of crude oil.

To achieve this, we consider nine time series based features, viz. *bond prices* for Chevron and Exxon-Mobil, *bond yield percentages* for Chevron and Exxon-Mobil, *stock prices* for Chevron and Exxon Mobil, oil prices, S & P oil indices and 10 year Treasury prices. Each of these features can be considered as a random vector. For each of these random vectors, we have use feature standardization. This makes the values of each feature in the data have zero-mean and unit-variance.

Once we have feature scaled our data, we can use it to train a Random Forest regressor. We divide the time-series data into sections of 80 and 20 percent splits. The 80 percent split is the training data and it corresponds to the older time snapshots of the data. The 20 percent split is used for forecasting and it corresponds to the most recent section of the time series data. After dividing our data in this 80-20 split regions we use a Random Forest regressor with 20 number of trees in the forest. We forecast data for bond and stock prices of Chevron and Exxon-Mobil and oil prices. Figures 5, 6 and 6 show results that we get from Random Forest regression. We use coefficient of determination (R^2 regression score function) to determine the effectiveness of our predicted results. For forecasting of stock prices, we obtain R^2 score functions of 0.9982 and 0.9845 for Chevron and Exxon-Mobil. We get R^2 score functions of 0.8917 and 0.5237 for forecasts of bond prices for Chevron and Exxon-Mobil respectively. For predictions of oil price,

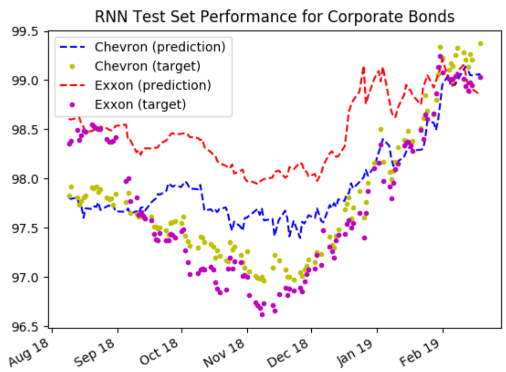


Figure 8. Company forecasts using the RNN

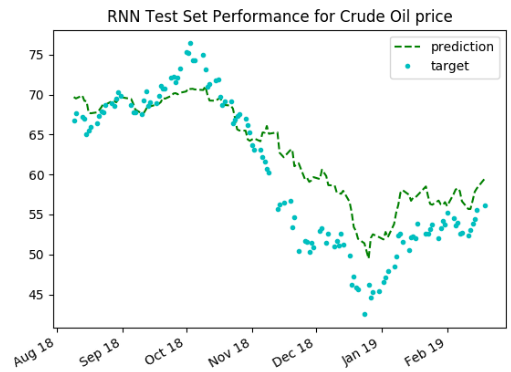


Figure 9. Crude oil forecasts using the RNN

we get a R^2 score function of 0.9945.

We conduct many experiments with different numbers of trees in our random forests and find that 20 is the optimal number of trees (using R^2 score function as metric to determine effectiveness of forecasts). Further increase of number of trees does not significantly increase the goodness of our forecasts. We also conduct experiments by choosing different combinations of features (from the aforementioned nine time series based features). Figures 14, 15 and 16 in section B show results of forecasts when five features are used. Our experiments show that best results are obtained when all nine features are used for forecasting.

6.2. Recurrent Neural Network

The Recurrent Neural Network (RNN) is a type of deep network used for predicting sequential data, since the recurrent network units can "remember" past data. We use the same features as before (except bond yields), which will be standardized before use. We will train our RNN to forecast all seven features using a moving window of 32 data points. Our RNN has an LSTM layer with 32 recurrent units. We used the Mean Squared Error (MSE) to evaluate our results, shown in Figures 8 and 9. The model achieved an MSE of 0.0569 on Chevron, 0.3843 on Exxon, and 10.39 on the crude oil (note the differences in data scale). On a visual basis, the Random Forest achieves a better fit on crude, while the RNN has a better fit to the bond price.

Discussion and Future Work

In this work, we have demonstrated a pipeline for explainable bond forecasting. Bond forecasting is often hampered by a lack of liquidity and transparency. While the finance literature has postulated relationships between bond prices and related factors, their usage has been mostly to explain past performance. By contrast, we actively discover correlated factors from available data, and then use them for future forecasting.

Our results on bonds from two major oil companies have shown the benefits of this approach. Not only can our model forecast the bond prices, but it can also forecast the major underlying commodity (oil) as well.

Future work can include several directions. First, we can see whether this model holds for bonds of different term lengths from the same issuer. Investor demands are different for longer and shorter bonds, so there may differences in factor correlations. Also, we can construct models for other sectors, which can vary based on the economic structure of the sector. Finally, we can use the explanatory relationships of our model for related purposes such as financial risk modeling.

A. QuicGraphicalLasso results with 5 features

.

In this section in figures 10, 11, 12 and 13, we show results of covariance and inverse covariance estimators from different model selection method when five features *i.e.*, *bond prices* from Chevron and Exxon-Mobil, *bond yield percentages* from Chervron and Exxon-Mobil and *oil prices* are used.

B. Random Forest results with 5 features

In this section we show results of forecasts obtained from Random Forest when only five features, *viz.*, bond prices from Chevron and Exxon-Mobil, bond yield percentages from Chevron and Exxon-Mobil and oil prices are used. The results shown here are from a Random Forest regressor

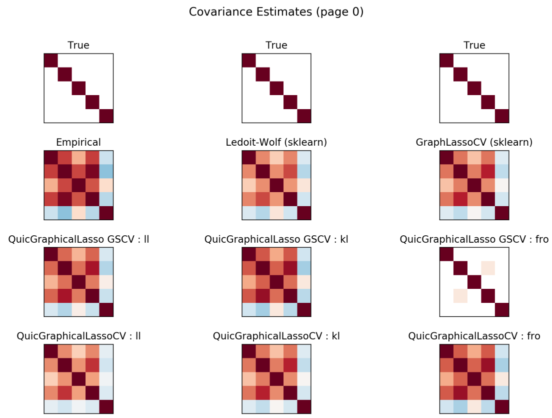


Figure 10. Inverse covariance estimates from standard (empirical), Ledoit-Wolf, GraphLassoCV, QuicGraphicalLasso, QuicGraphicalLasso + GridSearchCV (GSCV) and QuicGraphicalLassoCV methods. The symbols "ll", "kl" and "fro" depict score metric: log-likelihood, kl and frobenius respectively. Five features *i.e.*, bond prices from Chevron and Exxon-Mobil, bond yield percentages from Chervron and Exxon-Mobil and oil prices are used during the computation of these estimators.

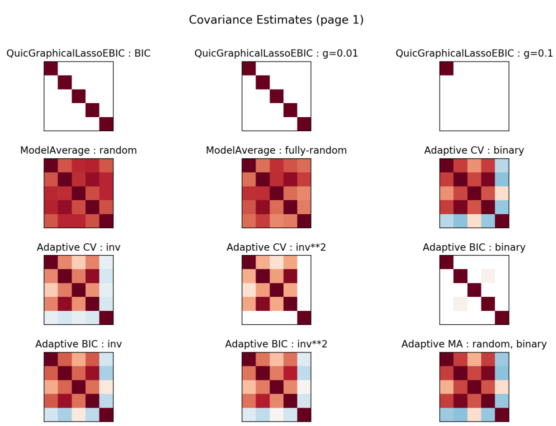


Figure 11. Inverse covariance estimates from standard (empirical), QuicGraphicalLasso-EBIC, ModelAverage, Adaptive CV and Adaptive BIC for different parameters. Five features *i.e.*, bond prices from Chevron and Exxon-Mobil, bond yield percentages from Chervron and Exxon-Mobil and oil prices are used during the computation of these estimators.

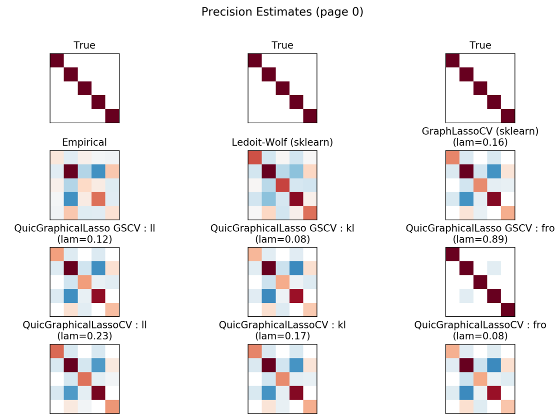


Figure 12. Precision estimates from standard (empirical), Ledoit-Wolf, GraphLassoCV, QuicGraphicalLasso, QuicGraphicalLasso + GridSearchCV (GSCV) and QuicGraphicalLassoCV methods. The symbols "ll", "kl" and "fro" depict score metric: log-likelihood, kl and frobenius respectively. Five features *i.e.*, bond prices from Chevron and Exxon-Mobil, bond yield percentages from Chervron and Exxon-Mobil and oil prices are used during the computation of these estimators.

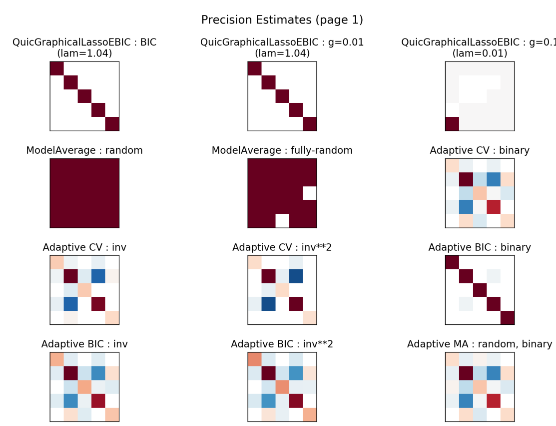


Figure 13. Precision estimates from standard (empirical), QuicGraphicalLasso-EBIC, ModelAverage, Adaptive CV and Adaptive BIC for different parameters. Five features *i.e.*, bond prices from Chevron and Exxon-Mobil, bond yield percentages from Chervron and Exxon-Mobil and oil prices are used during the computation of these estimators.

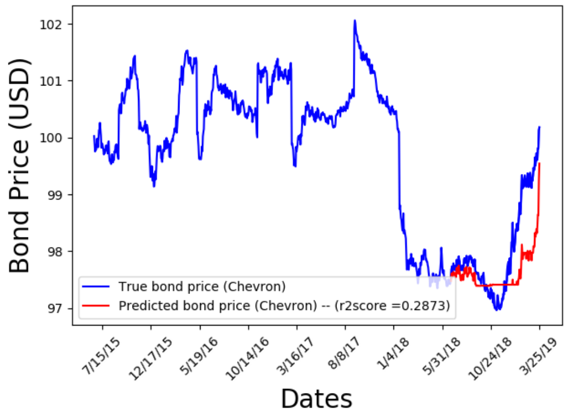


Figure 14. This plot shows bond prices for Chevron. The blue line shows actual values of prices while red line shows forecasted values of prices. Five time series based features, viz. *bond prices* for Chevron and Exxon-Mobil, *bond yield percentages* for Chevron and Exxon-Mobil, and oil prices were used for forecasting. The forecasting results shown in these plots are obtained through the use of Random Forest regressor.

which uses 20 trees. Figures 14, 15 and 16 show results of forecasts when five features are used. In each of these plots we find that the coefficient of determination is smaller than the corresponding cases which were discussed in 6.1.

C. RNN results with 3 features

In Figures 17 and 18, we provide the RNN results using the bond prices for Chevron and Exxon as well as the crude oil price. The model achieved an MSE of 0.1424 on Chevron, 0.2264 on Exxon, and 1.927 on the crude oil (note the differences in data scale).

References

Alvi, D. A. Application of probabilistic graphical models in forecasting crude oil price. Bachelor thesis, Department of Computer Science, University College London, 2018.

Balcilar, M., Gupta, R., Modise, M. P., and Mwamba, J. M. W. Predicting South African equity premium using domestic and global economic policy uncertainty indices: Evidence from a Bayesian graphical model. Technical Report 2015-96, Department of Economics, University of Pretoria, Pretoria, RSA, 2015.

Belloni, A., Chen, M., and Chernozhukov, V. Quantile graphical models: Prediction and conditional independence with applications to systemic risk. *arXiv preprint arXiv:1607.00286*, 2016.

Cerchiello, P. and Giudici, P. Conditional graphical models for systemic risk estimation. *Expert Syst. Appl.*, 43(C):

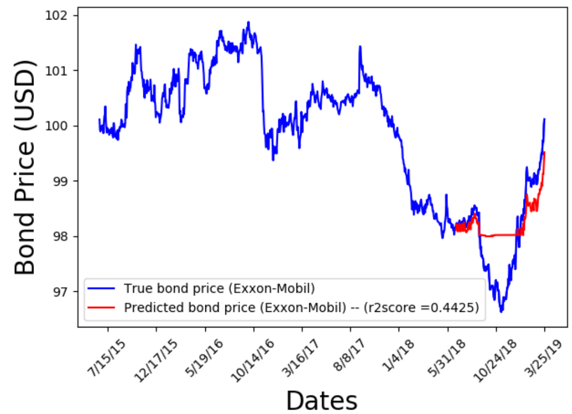


Figure 15. This plot shows bond prices for Exxon-Mobil. The blue line shows actual values of prices while red line shows forecasted values of prices. Five time series based features, viz. *bond prices* for Chevron and Exxon-Mobil, *bond yield percentages* for Chevron and Exxon-Mobil, and oil prices were used for forecasting. The forecasting results shown in these plots are obtained through the use of Random Forest regressor.

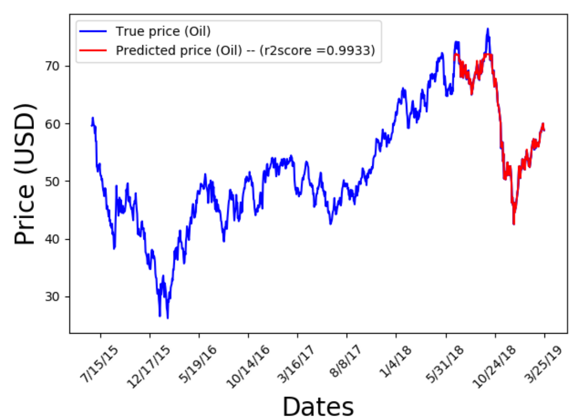


Figure 16. This plot shows oil prices. The blue line shows true prices of oil, while the red line shows predicted prices of oil. Five time series based features, viz. *bond prices* for Chevron and Exxon-Mobil, *bond yield percentages* for Chevron and Exxon-Mobil, and oil prices were used for forecasting. The forecasting results shown in these plots are obtained through the use of Random Forest regressor.

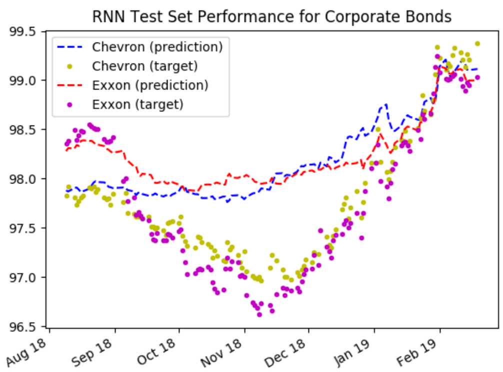


Figure 17. Company forecasts using the RNN with 3 factors

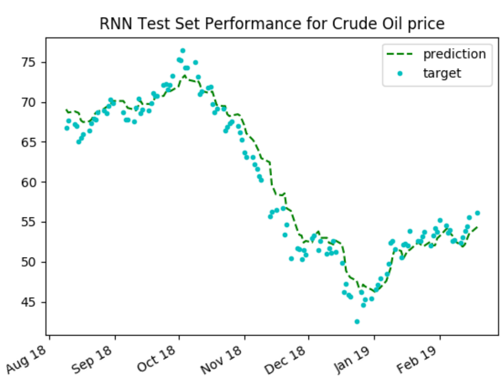


Figure 18. Crude oil forecasts using the RNN with 3 factors

165–174, January 2016. ISSN 0957-4174. doi: 10.1016/j.eswa.2015.08.047. URL https://doi.org/10.1016/j.eswa.2015.08.047.

Cerchiello, P., Giudici, P., and Nicola, G. Twitter data models for bank risk contagion. *Neurocomputing*, 264:50 – 56, 2017. ISSN 0925-2312. doi: https://doi.org/10.1016/j.neucom.2016.10.101. URL http://www.sciencedirect.com/science/article/pii/S0925231217311049. Machine learning in finance.

Denev, A. *Probabilistic Graphical Models in Finance*. Risk Books, 2015.

Federated Investors, Inc. Infographic: The bond market. Online, June 2017. URL http://www.federatedinvestors.com/FII/daf/pdf/G45692-06_2017.pdf.

Filiz, I. O., Guo, X., Morton, J., and Sturmfels, B. Graphical models for correlated defaults. *Mathematical Finance: An International Journal of Mathematics, Statistics and Financial Economics*, 22(4):621–644, 2012.

Foygel, R. and Drton, M. Extended bayesian information criteria for gaussian graphical models. *arXiv e-prints*, art. arXiv:1011.6640, Nov 2010.

Giudici, P. and Spelta, A. Graphical network models for international financial flows. Technical Report 52 (10-13), Department of Economics and Management, University of Pavia, Pavia, Italy, 2013.

Hastie, T., Tibshirani, R., and Wainwright, M. *Statistical Learning with Sparsity: The Lasso and Generalizations*. Chapman & Hall/CRC, 2015. ISBN 1498712169, 9781498712163.

Hsieh, C., Sustik, M., Dhillon, I., and Ravikumar, P. Quic: Quadratic approximation for sparse inverse covariance estimation. *Journal of Machine Learning Research*, 15: 2911–2947, 10 2014. ISSN 1532-4435.

Laska, J. and Narayan, M. skggm 0.2.7: A scikit-learn compatible package for Gaussian and related Graphical Models, July 2017. URL https://doi.org/10.5281/zenodo.830033.

Ledoit, O. and Wolf, M. A well-conditioned estimator for large-dimensional covariance matrices. *Journal of Multivariate Analysis*, 88 (2):365 – 411, 2004. ISSN 0047-259X. doi: https://doi.org/10.1016/S0047-259X(03)00096-4. URL http://www.sciencedirect.com/science/article/pii/S0047259X03000964.

Mohammadi, H. and Su, L. International evidence on crude oil price dynamics: Applications of arima-garch models. *Energy Economics*, 32(5):1001–1008, 2010.

Moreira, C., Haven, E., Sozzo, S., and Wichert, A. Process mining with real world financial loan applications: Improving inference on incomplete event logs. *PLoS ONE*, 13(12):e0207806, 2018.

Park, K. and Shin, H. Forecasting stock price movement with semi-supervised learning. In *Proceedings of the International Conference on Data Mining (DMIN)*, 2012.

Park, K. and Shin, H. Stock price prediction based on hierarchical structure of financial networks. In Lee, M., Hirose, A., Hou, Z.-G., and Kil, R. M. (eds.), *Proceedings of the 20th International Conference on Neural Information Processing (ICONIP 2013)*, pp. 456–464, Daegu, Korea, 2013. Springer-Verlag.

Reeson, C. Financial time series graphical modeling and portfolio analysis. Bachelor thesis, Duke University, 2009.

Salhi, K., Deaconu, M., Lejay, A., Champagnat, N., and Navet, N. Regime switching model for financial data: Empirical risk analysis. *Physica A*, 461:148–157, 2016.

Welch, G. and Bishop, G. An introduction to the kalman filter. Technical Report 95-041, Department of Computer Science, University of North Carolina at Chapel Hill, 1995.

Zhu, Y. and Barber, R. F. The Log-Shift Penalty for Adaptive Estimation of Multiple Gaussian Graphical Models. In Lebanon, G. and Vishwanathan, S. V. N. (eds.), *Proceedings of the Eighteenth International Conference on Artificial Intelligence and Statistics*, volume 38 of *Proceedings of Machine Learning Research*, pp. 1153–1161, San Diego, California, USA, 09–12 May 2015. PMLR. URL http://proceedings.mlr.press/v38/zhu15.html.