Global Market Correlation Strategy - SVM

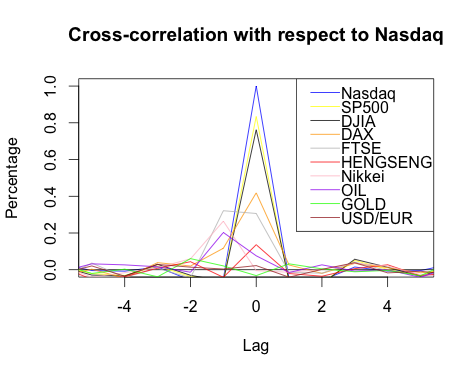
Support vector machine (SVM) explanation:

A support vector machine (SVM) is a classifier, a tool to achieve machine learning. Its core algorithm is to seek the optimal hyper-plain that best separates training sets. In other words, a support vector machine builds a model that assigns new data into its category.

Strategy idea:

The idea is from a thought that Asian markets lead the US markets. The purpose of this research is to analyze this phenomenon quantitatively.

Details:

The first part of my code is data processing and correlations analysis. I grabbed historical open and close prices of the required markets from Yahoo Finance and Quandl (https://www.quandl.com). The data range I chose was from 2006-04-15 to 2015-04-15. I used the NASDAQ index as the base and replaced missing data by linear interpolation. Since the goal was to analyze the correlations of those markets, the daily return of each market was the primary data I used for all operations. The analysis of correlations of those markets shows in the next. 

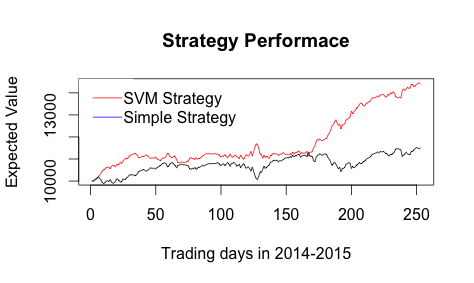
From the above figure, you can clearly see the relations between Nasdaq and other markets.

From the above figure, you can see the relations between Nasdaq and other markets.

The second part of my code built the SVM model. Based on this cross-correlations analysis, I chose the markets with the highest correlations to the Nasdaq as the features to train my SVM model. Firstly, I chose the daily returns of Nasdaq as the prediction target and labeled them as "1"(positive) and "-1"(negative). Besides, I also normalized the features data by dividing them into -1 and 1. Last but not least, I included the relative long-term returns (varied from 1 to 30 days) into my features. Used the daily returns data from 2006 to2014 as the training set, I developed my first SVM model. However, the prediction accuracy rate was not satisfying. So I tried to improve the model next.

In the third part, I tried to minimize trading risk by picking out risky points. To do this, I divided daily return of Nasdaq into 3 classes, "-1"(negative), "0"(neutral) and "1"(positive). I used F1 measure to find out the best neutral range. In this way, I can hold my money when the model predicts "0" and wait for the better chance. The simulation shows that this multi-class SVM model works very well.

The last part of my code is the simulation of the SVM model. The simulation date range is 2014-04-15 to 2015-04-15, and my trading strategy is straightforward: buy when the model predicts "1", sell when it predicts "-1" and hold money when it predicts "0".



The simple strategy in the above figure means that I simply buy in Nasdaq every day no matter in what conditions.

Conclusion: My SVM strategy model gains around 50% annual return, which is almost 40% higher than that of the simple strategy. Besides, the max loss in one day is around 2%, only 72 /252 days I would lose money. Trading days also decrease from 252 to around 200 days, which reduces tax and trading fees. Furthermore, the Sharpe ratio of my strategy is around 4.78(based on 0.19% annual risk-free interest rate).

I have to point out that this strategy has a flaw that the DAX and FTSE index I use does not close before the time U.S. market opens. These markets' open time have some interaction with each other, so we can't apply this trading strategy in the real world precisely as the simulation. However, we can overcome this imperfection with more accurate history prices.

**Two more thoughts on SVMs stock market application:**

First, instead of the index prediction, SVMs may be able to help us to identify which stock is worth holding. There are many standards about selecting stocks, and SVMs can help us apply these standards to stock markets to find the best portfolio. I have read a paper about applying SVM to identify bankrupt companies. They used profit margin, debt ratio, cash flow ratio and other accounting attributes as the training features to classify bankrupt companies. We may also identify the "good" companies similarly. We may also rank those "good" companies, and buy the top ones.

On the other hand, traditionally we use SVMs in handwriting and image recognition. Thus, it makes sense to migrate that application to the financial industry by using SVMs to identify what is "good" news for one stock and buy that particular stock at the "good" time.These are some untested thoughts about SVM application on stock markets.