Predicting Abnormal Returns From News Using Text Classification

Ronny Luss

Joint work with Alexandre d'Aspremont

Department of Operations Research and Financial Engineering
Princeton University

July 20, 2009

Thanks to Jonathan Lange and Kevin Fan for research assistance.

A Market Classification Problem

Microsoft issues the following press release at 10:30 am on Wednesday:

*LONDON - Dec. 12, 2007 - Microsoft Corp. has **acquired** Multimap, one of the United Kingdom's top 100 technology companies and one of the **leading** online mapping services in the world. The **acquisition** gives Microsoft a **powerful** new location and mapping technology to complement existing offerings such as Virtual Earth, Live Search, Windows Live services, MSN and the aQuantive advertising platform, with future **integration potential** for a range of other Microsoft products and platforms. Terms of the deal were not disclosed.*

Goal: Given the last hour of Microsoft prices and the press release, we want a model that produces a binary output at 10:30 am (when the news comes out):

$$\begin{cases} +1 & \text{if the absolute return on Microsoft from 10:30-11:30} \geq \rho \\ -1 & \text{if the absolute return on Microsoft from 10:30-11:30} < \rho \end{cases}$$

Bag-of-words:

increas	decreas	acqui	lead	up	down	bankrupt	powerful	potential	integrat
0	0	2	1	0	0	0	1	1	1

Time series of returns:

r_1	r_2	r_3	r_4	r_5
.02	.01	.005	-.005	0

Experimental Setup

News changes over time! We want to train a model on recent news and only test on news that is published in the *short term*.

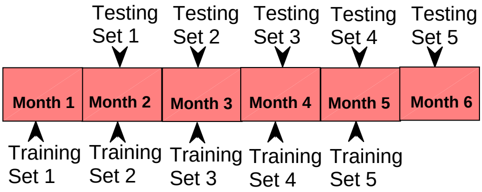


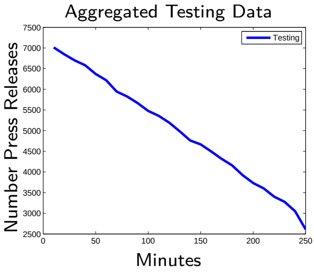
Figure: Chronological training and testing with a moving window

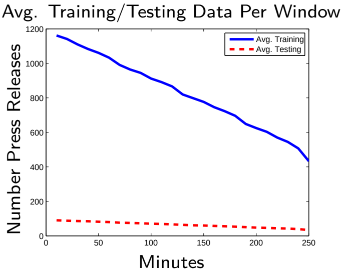
All results shown will use training on **one** year of news and testing on the following **one** month of news. Test results are aggregated.

Data

Data Set: PR Newswire press releases

- Time Period: January 2000 to December 2007.
- Results are based on 128 companies chosen based on quantity of releases.
- We only consider news published during the business day.
- Intraday price data obtained from Wharton Research Data Services.



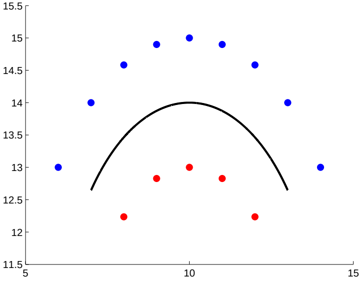


Text classification in finance

- *Text Mining Systems for Market Response to News: A Survey* (2006) by Mittermayer and Knolmayer
- Lavrenko et.al. (2000) uses **Naive Bayes** to choose from 5 categories obtained by slope of regression with 10-minute stock price data.
- Thomas (2003) uses **Decision Rules** to categorize by headlines with daily data for trading strategies.
- Mittermayer and Knolmayer (2006) uses **SVM** with various kernels to predict 15 minutes into the future. Uses 4 classes. Uses PR Newswire from April-December 2002.
- Kogan et.al. (2009) use Support Vector Regression to forecast stock return volatility based on text in SEC mandated 10-K reports.

Support Vector Machines

$\Phi : x \rightarrow \Phi(x)$ is a mapping to a linearly separable space:



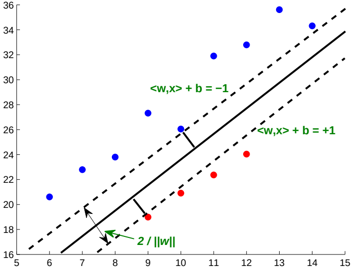


Figure: Input Space vs. Feature Space

Mercer's Condition: $K \succeq 0 \Rightarrow K$ is a kernel $\Rightarrow \exists \Phi$ s.t. $K_{ij} = \langle \Phi(x_i), \Phi(x_j) \rangle$

Performance Measures

We *optimize* the **Annualized Sharpe Ratio** of the following game:

For every press release published, make a bet on whether or not an abnormal return will occur and receive a payoff of $\pm\$1$.

Annualized Sharpe Ratio $= \frac{\text{Expected Return}*(\text{Periods Per Year})^{1/2}}{\text{Risk}}$

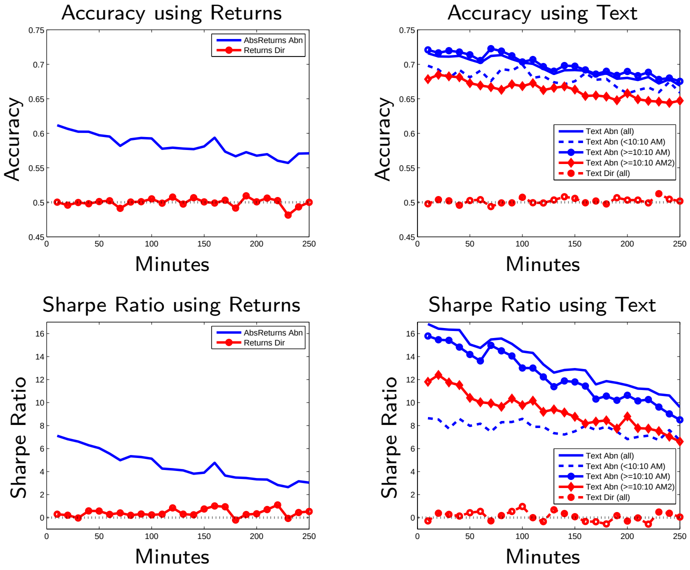
Expected return is calculated as the average return of playing the above game for each press release on each day of the data horizon.

Another measure we use is **Accuracy** $= \frac{\text{Number of correct predictions}}{\text{Total number of predictions}}$

Abnormal Returns Definition

- Sort the absolute returns following all news in the training set.
- Define $T = 75^{th}$ percentile of absolute returns as threshold.
- For the i^{th} article, label $\begin{cases} y_i = 1, & |r_i| >= T \\ y_i = -1, & |r_i| < T \end{cases}$

Predicting Abnormal Returns *(75% with only SVM)*

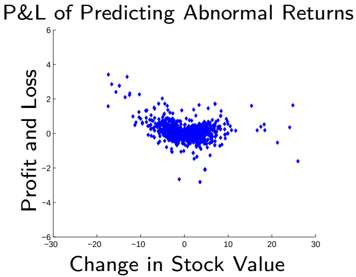


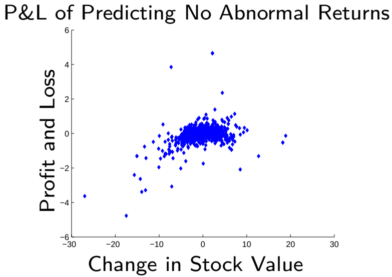
Strategy: Δ Hedged Covered Call Options

IF predicting an abnormal return: Buy 1 call options and sell Δ shares of stock.
Tomorrow, exit positions.

IF predicting NO abnormal return: Sell 1 call options and buy Δ shares of stock.
Tomorrow, exit positions.

where Δ is defined as the change in call option price resulting from a $1 increase in stock price.





NOTE: Options data taken from *OptionMetrics* through *WRDS*.

Predicting Daily Abnormal Returns

Features	Strategy	Accuracy	Sharpe Ratio	# Trades
Text	TRADE ALL	.63	.75	3752
Abs Returns	TRADE ALL	.54	-1.01	3752
Text	LONG ONLY	.63	2.02	1953
Abs Returns	LONG ONLY	.54	1.15	597
Text	SHORT ONLY	.62	-1.28	1670
Abs Returns	SHORT ONLY	.54	-1.95	3155

Kernel Optimization

Suppose K_1 and K_2 are good text and absolute returns kernels.

How can we combine the kernels?

From Lanckriet et al. 2004, we can *learn* kernels using the framework:

$$\min_{K \in \mathcal{K}} \omega_C(K) \tag{1}$$

where

$$\omega_C(K) = \max_{\{0 \leq \alpha \leq C, \alpha^T y = 0\}} \alpha^T e - \frac{1}{2} \alpha^T \mathrm{diag}(y) K \mathrm{diag}(y) \alpha \tag{2}$$

is an upper bound on the probability of misclassification.

One way to combine the kernels is with positive linear combinations.

$$\mathcal{K} = \{K : K = d_1 K_1 + d_2 K_2, d_i \geq 0\} \tag{3}$$

Kernel Optimization

The most recent formulation in Rakotomamonjy et al. (2008) uses:

$$\min \; J(d) \text{ s.t. } \sum_i d_i = 1, d_i \geq 0 \tag{4}$$

where

$$J(d) = \max_{\{0 \leq \alpha \leq C, \alpha^T y = 0\}} \alpha^T e - \frac{1}{2}\alpha^T \mathrm{diag}(y)(\sum_i d_i K_i)\mathrm{diag}(y)\alpha \tag{5}$$

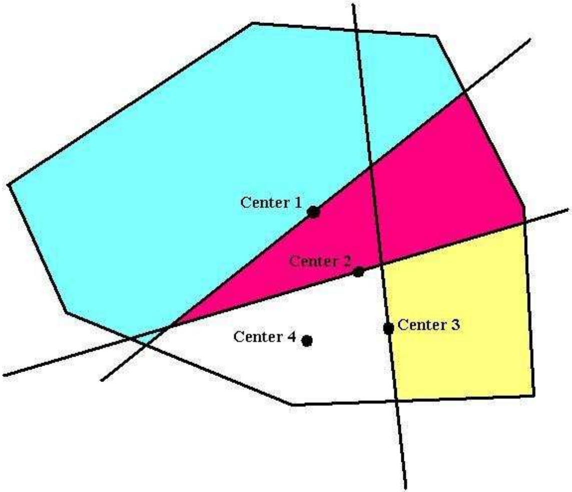
The gradient of J can be calculated by:

$$\frac{\partial J}{\partial d_i} = -\frac{1}{2}\alpha^{*T}\mathrm{diag}(y)K_i\mathrm{diag}(y)\alpha^* \tag{6}$$

where α^* is the optimal solution to SVM using the kernel $\sum_i d_i K_i$

- **Every computation of $J(d)$ or $\nabla J(d)$ requires an SVM computation.**
- **Multiple SVM computations per iteration for gradient methods!**

Analytic Center Cutting Plane Method



Find the center, make a cut, shrink the feasible region, and repeat.

Algorithm 1 Analytic center cutting plane method

1: Compute d_i as the analytic center of $\mathcal{L}_i = \{d \in \mathbf{R}^n | A_i d \leq b_i\}$ by solving:

$$d_{i+1} = \underset{x \in \mathbf{R}^n}{\operatorname{argmin}} \; -\sum_{i=1}^{m} \log(b_i - a_i^T x)$$

where a_i^T represents the i^{th} row of coefficients from A_i in $\mathcal{L}_i$, m is the number of rows in A_i, and n is the dimension of d (the number of kernels).

2: Compute $\nabla J(d)$ from (6) at the center d_{i+1} and update the (polyhedral) localization set:

$$\mathcal{L}_{i+1} = \mathcal{L}_i \cap \{d \in \mathbf{R}^n | \nabla J(d_{i+1})(d - d_{i+1}) \geq 0\}$$

3: If $m \geq 3n$, reduce the number of constraints to $3n$.

4: If gap $\leq \epsilon$ stop, otherwise go back to step 1.

One SVM computation per iteration!

How do these algorithms compare against each other?

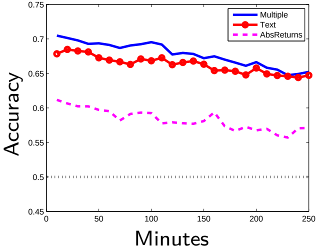
	Max	simpleMKL				accpmMKL			
Dim	# Kern	# Kern	# Iters	# SVMs	Time	# Kern	# SVMs	Time	LIBSVM Time
500	3	2.0	3.4	27.2	48.6	3.0	7.1	**13.7**	**0.6**
	7	2.6	3.4	39.5	47.9	7.0	12.0	**15.5**	**1.8**
	11	3.6	3.2	41.0	37.3	10.9	15.3	**17.4**	**3.3**
1000	3	2.0	2.0	29.3	164.5	3.0	6.3	**36.7**	**2.4**
	7	2.4	3.6	53.3	240.3	6.8	11.7	**40.0**	**6.8**
	11	3.9	3.6	57.8	214.6	10.6	14.9	**48.1**	**12.7**
2000	3	2.0	1.0	24.0	265.8	3.0	5.0	**79.4**	**7.2**
	7	3.3	1.5	30.4	209.6	7.0	10.5	**110.5**	**25.2**
	11	6.0	2.3	40.5	253.2	11.0	14.4	**141.4**	**46.5**
3000	3	2.0	1.0	24.0	435.5	3.0	6.0	**248.9**	**17.9**
	7	4.0	2.0	38.0	591.4	7.0	6.8	**221.7**	**39.0**
	11	6.0	2.0	39.8	648.9	11.0	8.0	**244.8**	**66.8**

Table: Numerical performance of simpleMKL versus accpmMKL for classification on **Text Classification Data**.

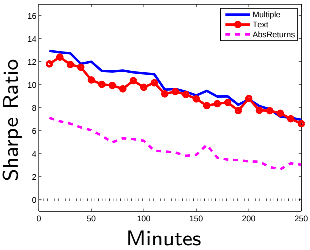
Kernel Optimization - Improvements? (75% Threshold)

13 possible kernels: 1 linear text, 1 linear absolute returns, 4 gaussian text, 4 gaussian absolute returns, 1 linear timestamp and day of week, 1 identity matrix.

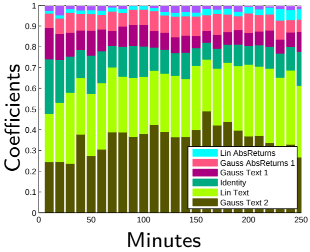
Accuracy using Multiple Kernels



Sharpe Ratio using Multiple Kernels



Coefficients with Multiple Kernels 75^{th} %



Further Directions

- Δ hedged covered call options for intraday predictions.
- Predict directions of price movements - can kernel optimization help? So far unfortunately no.
- Topic tracking.
- Kernel optimization with unrestricted d (need to solve a large SDP).
- Feature selection, aggregating features.
- Multi-class SVM.
- Support Vector Regression.