SENIOR THESIS

# GPU Optimized Machine Learning Algorithms for Low-Volatility Stock Portfolio Options

Julian Gilyard
April 17, 2016

## CONTENTS

# THANKS

Keeping with the ever present traditional of the senior honors these, I would like to take this time to thank some of the people that have been individually responsible for my success throughout my undergraduate career. These individuals have played a significant role in providing me with clarity, opportunities and understanding. Without there support, I would not have been able to thrive during my time at Wake Forest University. As much as this thesis is for me, it is for all of the individuals that have helped me along the way.

Matthias Gobbert is a professor at UMBC in Maryland. He was my primary supervisor during my NSA/NSF sponsored REU. Dr. Gobbert provided me with an invaluable understanding of research tools, high performance computing and and UNIX systems. His teaching style is second to none and created a life long friendship between myself and my teammates In 2015, he helped me present and co-author a paper that was published in SIAM's journal SIURO.

Professor Jiang has been half of the integral part to my mathematical economics major. As much as professor Jiang is an adviser, he is a friend. He has provided me with a plethora of life advice regarding decisions for my future and how to apply myself more fully within the discipline of mathematical. I have always appreciate his candidness and ability to teach. Because of him, I believe I am a better Mathematician and human.

Dr. Phillips, while neither in a computation field of finance or computer science has been an integral part of my maturation. Dr. Phillips is the scholarship adviser and a personal friend. He was the adviser of the Vienna Flow house in 2014 when I studied abroad. While I lived there, I matured as a person, wrote models for monitoring Bitcoin traffic and explored the depths of my

own humanity. Dr Phillips' influence in my life has allowed me to thoroughly experience my Wake Forest time while learning to understand the meaning behind life. He may be the most complete and passionate individual that I have ever met. He pedagogical vantage point and methodological approach to life has forever changed my understanding of what is means to live life. I vicariously live through his stories and picture life better every time I have a conversation with him.

Professor Chen is the epitome of a learned individual. He is capable of understanding concepts and always willing to let me explore the boundaries of how economics applies. He is the second half of my mathematical economics major. Within the economics department, I have relished the opportunities to work with him and further my understanding of numerous concepts. During my senior year, when I heard that he was offering game theory, I immediately took the class and was refreshingly challenged. He allowed me to look at the importance of applying deep concepts and invited me to apply game theory in non-traditional roles. Thanks for letting me write my game theory paper on bowling.

Professor Gambill, while out of the people mentioned here I have known the least, has nonetheless been a truly inspirational mentor and teacher. Professor Gambill teaches the library science class in which students create individual research projects and gain an understanding of research and its implications. She embodies the spirit of what is right at Wake Forest University. She has inspired me to acquire higher levels of knowledge while expressing myself within scholarly perspective. Without her help this paper would not have been possible. Because of her, I gained a reinvigorated desire for research and scholarship.

Dr. Cotrell is a revered professor of the economics department at Wake Forest University. I took his econometric class during my junior year and the concepts that he taught me have provided a lifelong appreciation for research and applied economics. Dr. Cotrell taught me the value of collecting data and applying it to a larger concept and context. Without his help, I would have no concept of econometric theory or learning how to identify the importance of everyday items to a larger idea. Dr. Cotrell while being an economic professor is also a computational genius. He created, wrote and now maintains the economic regression software GRETL. I hope that in the future he will serve as a friend, resource and adjunct professor in the computer science department.

Professor Pauca, is a professor of the Computer Science department. He was my first computer science professor at Wake Forest University and inspired me to expand my scope at Wake Forest University. He was taught me to enjoy computer science as a discipline and to fully embrace the culture. Through his class, I learned how to program android applications while making a life long connection to the discipline as a whole. Individually he has made tremendous strides in making technology useful to disabled people and his been an inspiration to me.

Dylan Stamer is a strategist at UBS and inspired me to apply my computer science major to finance. His mentoring has been invaluable throughout my maturation. Dylan provided me a significant amount of real world knowledge about finance while providing me with opportunities to expand my horizons. In 2016, I will return to UBS to work with him. I consider him to be the primary reason that I was able to work at UBS while being a large friend and mentor.

Joe Stewart is one of the most inspirational individuals that I have met in my life. Joe inspired

me to take finance and the world by the palm of my hands and to run with it. He is the head of US hedge fund sales at UBS and will be my current employer. Joe has taken a unique role in my life my providing me with numerous opportunities to apply my skills in looking an unique hybrids for markets domestic and international. Without his support, I would not be employed or have a significant understanding of the lateral correlations and movements of markets. While Joe is adept at finance, he is impact extends past that to a friendship for me. He has mentored me an a way that few could by placing me in unique situations for research and influence. In the future, Joe Stewart and I will be working towards looking at tertiary market movements and seeing the impact of largely leveraged markets.

Dr. Samuel Cho is pretty much everything to me. He has been a mentor, teacher, friend, confidant, resource,and understanding mentor that I am honored to have worked under during my tenure at Wake Forest University. I first met Dr. Cho when I presented my first app from my first computer science class in 2013. After viewing that application, Dr. Cho offered me a position to research in his lab. Coming into his lab, I knew nothing of research, MD simulations, GPUs (Graphics Processing Units), computational complexity or life; however, my interactions and constant briefings from him provided me with context for all of those experiences and more. He gave my the strength to pursue my passions and lead me to unimaginable places. When I look back at what I have done at Wake Forest University, I see that almost all of it has to do with Dr. Cho. Dr. Cho has provided me to learning opportunities, presentation scenarios and a life long friendship. He taught me the value of quality work, the true meaning of teamwork and the knowledge of the impact of failure. I don't know where I would be without him. He is possibly

the greatest person that I have had the pleasure of interacting with in my life. Outside of being able to research with Dr. Cho, I have had the pleasure of calling him my friend. Whenever, I have needed to talk with someone about life, research, understanding or context, he has been available. Dr. Cho can tell you able times that I have cried in his office, rambled for hours about unique concepts that he already was an expert in, and about my many mistakes. In the fact the the majority of my work has dealt with research Dr. Cho taught me how to expand my horizons and to truly pursue ideas that I thought were worthy. He gave me the courage to take nontraditional approaches to numerous ideas while expanding my mind. In the time that I have known Dr. Cho, he has not once inhibited my ability to learn or experiment. In fact, he has been always willing to give me a shot to go after the most obscure topics with significant importance. In the amount that Dr. Cho has given me, I doubt that I will be able to ever repay him. I suspect that he is rarest form of person that exist, one that challenges the ideas of the status quo while appreciating the current reality and encouraging others to do the same. Sam, I am sure that forever we will be friends; you will always be my mentor; and, I will never forget you.

Mom and Dad: If I wrote about how much you mean to me this paper would just be a history of our lives together. In order to preserve brevity, I will just state the obvious. Thanks for everything, for without you, I wouldn't be here today. ;)

# 1 Introduction

This project is a hybrid construction between economics, finance and computer science. We seek to identify characteristics of low volatility equities while attempt to forecast if the equities stay

within the realm of profit for specific options strategies. While volatility and value and positively correlated, given the black scholaes formula, we seek to look only for low volatility strategies as this gives us a cheaper and more reliable approach for looking at applications to finance as a whole. All of the packages used are open source and the source code provided is available online through my personal website and github repository[1] [2].

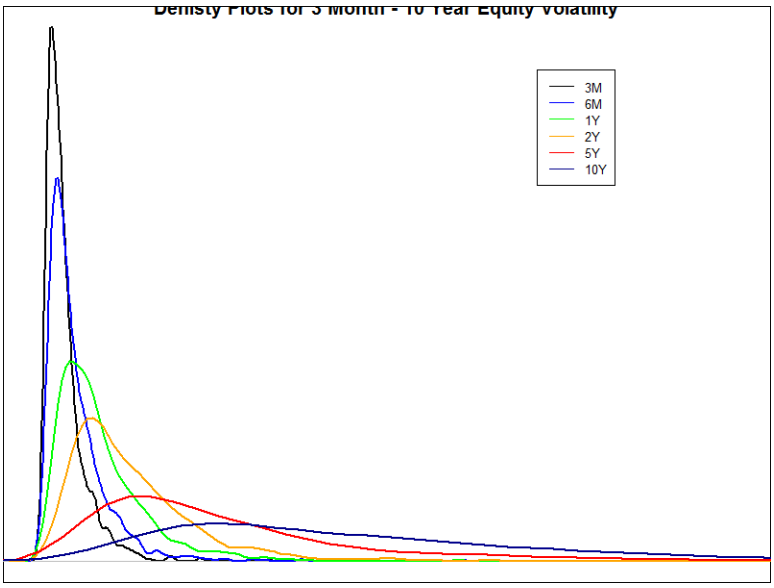


Figure 1.1: Standard Deviations of the Market

After we identify these low volatility equities in one of five time horizons (3 months, 6 months, 1 year, 2 year and 5 year) we look at numerous machine learning algorithms and see which ones are most adept at identifying the desired outcome for low volatility. We will be using a combination of algorithms then present the most compelling algorithms comparing the results to one another. This will allow us to gauge performance from a run time perspective and an accuracy perspective. Traditional metrics for machine learning such as confusion matrices will serve as a litmus test to determine how our algorithms will perform in real world scenarios.

In order to prepare for this project, we enlisted the help of Coursera's machine learning class taught in R. This provided us with industry experience and a wide variety of additional techniques that have been used to perform machine learning algorithms. This class is cross taught at the University of Pennsylvania and is provided for free online. The class covered R's implementation of the Caret package for machine learning, current testing methods, forecasting ideologies, co-variance matrices, data mining techniques and a complete understanding of data cleaning.

Our goal is to see how well these algorithms perform, select the best algorithm that performs the most accurate under our testing set then to see how we can increase the performance of our algorithms by porting them to GPUs. We will use a combination of leverageable packages through R and CUDA. In the end we seek to find performance gains and accurate predictors for a equities while displaying useful real world performance. Future implications for this work can be to limit negative market exposure or to dynamically craft baskets for clients that want particular exposure to companies in a specific sector but at a quantifiable risk profile. In order to provide validity to our testing methods we divide our data into two separate categories.

# 2 Methodology and Approach

## 2.1 Machine Learning Class

In order to have a starting point we take the time to enroll in the machine learning class.[3] This not only gives us exposure to industry standards but provides validity behind the reasoning for our actions. The verification that our methods are solid have allowed us to expand our scope for research while ensuring that our testing methods are true. The class was taught online and was

part of an online 4 week forum.

The class provided real world examples regarding the success of machine learning and is applications to multiple fields. The class begins with historical introduction then later increases in complexity as more concrete models come into play. The class while free and only four weeks dives in to topics including naive Bayes, random forests, regression modeling and classification trees.

Given the individual strengths and weakness of each modeling technique that is proposed there is a significant amount of information that can be glean from the class. The first concept is that determine the question is vital to the success of any machine learning algorithm. Regardless of the desired technique creating a valid question with viable outputs and quantifiable inputs is not only valid but useful. Additionally, crafting questions over scenarios where data is plentiful is an encouraging situation. This provides those looking to do machine learning algorithms with breadth and history to apply more methods while fine tuning already available skills. This class is offered as an online extension and package of the Data Science specialization offered by Coursera and John's Hopkins.

Separate from the creation of questions and basic algorithms the class submerses students in the ever important topic of data cleaning and mining. This technique was useful when gathering data for our project in the fact that all data is not created equal. The class provides examples where data is missing and holes need to be filled. Filling the holes is addressed by using a combination of zeroing out, estimating, and removing variables. The concept of data cleaning provides accurate rationale for why it is an important step in the process of creating algorithms.

Following the usefulness of cleaning data, we are approached with the concept of data visualization. Numerous skills within the class provide reasoning for the usefulness of data visualization. Data visualization, provides readers, skimmers and those equally familiar with the field and understanding of what authors are attempting to portray. The majority of the visualizations in this paper originate from R's Caret package and ggplot. These are the exact packages that are directly referenced in the class. These direct examples will provide additional explanations for our choosing of certain variables and correlations for outputs.

## 2.2 CARET PACKAGE

The caret package which stands for Classification and Regression training is "a set of functions that attempt to streamline the process for creating predictive models" [4]. The benefit of using caret is that we have a simple package to automate reading in data that originates in CSV file format, data can be cleaned quickly, multiple machine learning algorithms can be combined and data can be easily separated into testing/training sets. The caret package provides direct splicing of data, k-means separation and automated separation of data. This gives clarity to the procedure and allows others to reproduces the results with a limited amount of knowledge regarding machine learning techniques and provisions[5]. The package operates within R's framework and leverages 25 separate packages to produces the desired output of regression and classification models.

## 2.3 Data Mining

In order to gather data for the project, initially we were perplexed. We wanted to capture a substantial portion of the market while looking to work with historical data for back-testing and results. Instead of directly looking at every single domestic stock and attempting to glean specific data from multiple databases, we Bloomberg's financial terminal which has been graciously supplied by Wake Forest University's Business school. In order to access this database, we created custom Bloomberg scripts to interface with Bloomberg's Historical Data API. These scripts were written in Visual Basic for applications in order to run natively on the terminals and to provide malleability.

Bloomberg's api can be directly accessed through Microsoft Excel's Bloomberg plugin. This provided direct access and ease when looking to convert the files to csv files to improve readability. This lead us to look for multiple directions for determining how we would find equities. While Bloomberg provides live and historical data for public and private bonds, equities, portfolios and ETFs. We weren't sure what was the best approach when considering a market-wide approach. At one point in time, we considering making a Rather than to select stocks individually, we look to capture a market-wide phenomenon by looking at the industry standard approach.

Through the process of mining this data we make approximately 1500 stock with 2500 historical price data points and 11 unique characteristics. This lead to approximately 3.7 million api requests. So in order to get the data, data was mined over the duration of 3 days as Bloomberg seemingly limits the number of api requests that come from academic Bloomberg accounts[6].

In order to scrap our data we chose the S& P Composite 1500 Index. It is followed under

domestic markets under the ticket of SPR. This was chosen because it provides the financial industry standard for a total market benchmark of the US market. The official definition is "The S& P Composite 1500Âő combines three leading indices, the S& P 500Âő, the S& P MidCap 400Âő, and the S& P SmallCap 600Âő to cover approximately 90 percent of the U.S. market capitalization. It is designed for investors seeking to replicate the performance of the U.S. equity market or benchmark against a representative universe of tradable stocks."

This level and representative of the US equity market provides us with a a good overviews for testing and training models because it the index reaches across numerous sectors and looks to provide the most liquid examples of these stocks. The metrics used by Standard and Poors (S& P) provide a significant amount of context regarding the quality and product that they decided to present. The S& P 1500 is also known as the TMIX (Total Market Index). All shares listed in the TMIX must pass a strict liquidity, market capitalization and float criteria to be maintained in the index. Liquidity is the concept that an equity should be able to be sold and bought within a reasonable amount of time. Meaning that shares and being actively purchased and sold. The liquidity concerns for the S& P 1500 state that the stock should trade a minimum of 250,000 shares each of the six months leading up to its evaluation date and for companies with multiple stock classes, like Berkshire Hathaway and Google, their classes will be independently evaluated. This ensures that investors won't be stuck with stocks mirroring "inactive" stocks. The public float criteria states that atleast 50 percent of the stock must be public. This ensures that one particular entity can't control a majority of the price fluctuations or majority control of the public company. The market capitalization rules are as follows, "Unadjusted company market capitalization of US

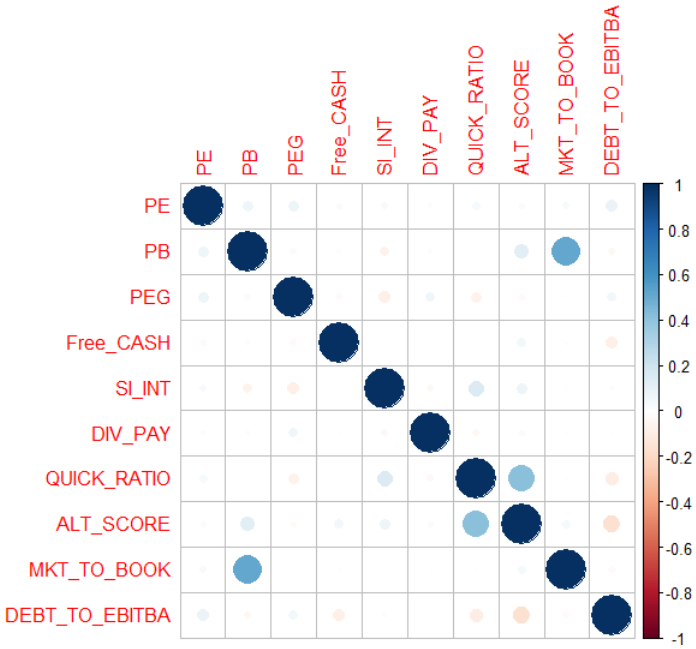


Figure 2.1: Correlation Matrix for Covariates

5.3 billion or more for the S& P 500, US 1.4 billion to US 5.9 billion for the S& P MidCap 400, and US 400 million to US 1.8 billion for the S& P SmallCap 600." [7]

In order to clean the data. We looked at the entire S & P 1500. From there we truncated our data down to only equities with atleast 10 years worth of history. This provided context and allowed us to extend our time line for modeling and forecasting. After this data truncation, we found ourselves with a little more than 1200 equities. As a result, we were able to have a representative of approximately 80 percent of the USA's market cap. Afterwards we were able to run a correlation matrix over all of the different covariates in order to prevent multicollinearity. We found that there we no significant overlaps throughout the data thus allowing us to conclude that there will be minimal overlap throughout the regressions. After checking for the appearance of multicollinearity. We looked to provided correlations between three major covariates in

Dividend Pay, Price to Book Ratio and PEG ratio. This allowed us to look at the data is see how there is apparent overlap. Thankfully we see minimal concentrations of the data and observe few outliers. Following that we normalized the data to be bound between 0 and zero. This was done primarily to help our machine learning algorithms with their consistency and to limit the variance in our linear regressions.

Talk about regression Models for Correlation

# 3 Literature Review

## 3.1 First Paper

One of the initial papers that I looked at to provide inspiration for my research was written by Dave McKenney and Troy White. In 2011 they published a paper entitled Stock Trading Strategy Creation using GP on GPU. They notice that previously their had been no intersection between GP, Equities and GPUs. The idea behind the paper is two fold. It investigates the involvement of genetic programming (GP) and its ability to predict equity movements while looking at the speed improvements that occur when running these algorithms on GPU vs traditional CPU metrics. McKenney and White utilize NVIDIA enabled CUDA devices much like those that are avalible at Wake Forest University. These devices allow for the paralization of multiple algorithms and look to reduce the amount of computational time required for algorithms that can be used in a predictive fashion. While the speedup improvements were interesting, the most critical point of the paper comes in the form of understanding the implications of genetic programming alogrithms towards stock trading. While we do not directly use genetric programming alogrithms, this opens the

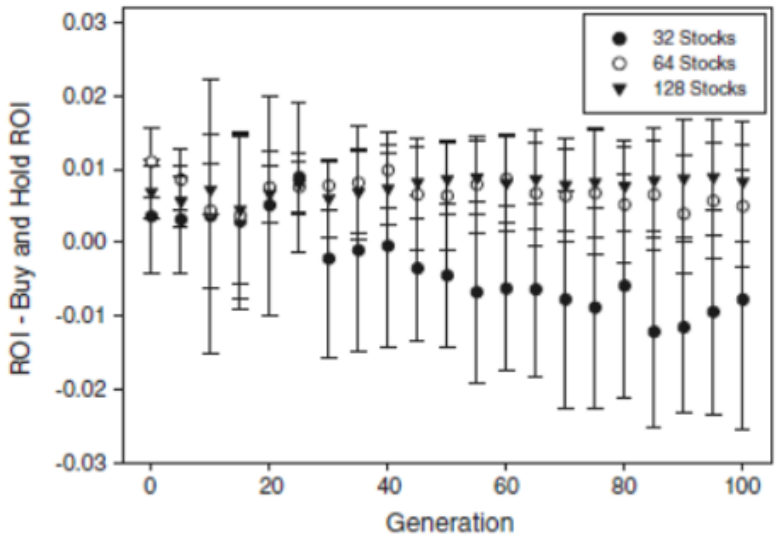


**Fig. 10** Profitability of different GP runs on testing data with 1 SD *error bars* (using 256 training days)

Figure 3.1: Limited Returns with Algo

flood gates for the utilziation of alogrihtms as a whole for the procress of backtesting and figuring out improvements for algorithms. From a GPU oriented perspective, the speedup is predictable. At the most parallizable points in the paper, the GPU speedup is 600x faster than the CPU oriented code. This given the fact that GPUs most have an order of 400-500x more cores than the average CPU and the math required for GP is embarssingly parallel. [8]

The takeaways from the paper are interesting to say the least. The algorithm that they choose determines different buy and sell points based off of signals from traditional trading metrics. At each point, they purchase all available shares with the 10,000 USD allocated for the account. The technicals that are involved use Money Flow Index, Ease of Movement, Commodity Channel Index, MACD and Volume Indices. When back testing the algorithm, researchers found that

smaller portfolios provided the highest return on their investment. Their 32 stock portfolio outperformed their 128 stock portfolio in every case; however, in real world scenarios the reverse was true. The stock portfolio of 128 outperformed the portfolio of 32 leading researchers to conclude that in real life the reduction in variance and larger amount of stocks lead to limited loss but also limited gain. Among their real world profits, they at the end of 100 simulations were left profiting between 0 percent at the 32 stock portfolio and 2 percent at the 128 stock portfolio leading them to think that the algorithm won't outperform buy and hold strategies; however, it open the door for further development, such as expansions for different algorithms, attempts to use different strategies and possible responses for economic indicators (Interest Rates, Sock Index Values, Federal Reserve Minutes, etc) [8].

## 3.2 Second Paper

The Second major paper that I used for inspiration was entitled Using GPU-CPU architecture to speed up a GA-based real-time system for trading the stock market. Their idea was to streamline the process of analyzing large quantities of data while looking to avoid the large psychological reaction of traders that they encounter when investing in financial markets. They cite that comparitively they are able to report a profit of 870 percent for the S & P 500 over a 10 year period of (1996 to 2006) versus the consensus average profit of 273 percent over the same period of time. [9].

This paper provided a list of GA based alogirithms that have been implmeneted on a large scale. They cite the software GeneHunter which was created by a company called MBA Ware

located in Virginia. The company has 2,500 customers from more than 40 diferent companies. GeneHunter's concept has been used to create rules that look at the NYSE index. The focus of his paper is to take the concept of using genetic algorithms that require a substantial amount of time to run and apply them to intraday trading instead of simply daily trading. As a result, it requires significantly more computation power as "it is not possible to wait for hours to obtain the investment decision result provided by the mechanical trading system when the investments have to be done continuously during the day." While the desired outcome was to use this algorithm on intra-day trading data, it is used on daily data because its is more accessible.

Prior to addressing this paper, determining indicators for our alogirhtm development was difficult, outlined in Nunez[9], there is a long list of flucuating indcators that can be used. In the paper, the drive their genetic algorithm by using the metrics of price-earnings ratio, price-book value raio, price-cash flow raio, debt over book euiity, sales growthm, net income growth, cash flow growth, return on assets, turnover growth, and profit margin growth. Nunez et al, cites 7 other authors that have found these metrics to be effective in their research. This provided us with a subset of useful measures. In the paper, they are required to compute these ratios; however, because they are freely avaliable through bloomberg, we do not have to compute these values. A genetic algorithm is used to determine short indicators and long position indicators. They select, Price to Cash Flow ratio, Leverage ratio, Growth in Sales, and Increase in turnover for short positions and Market to book, growth in sales and Return on asset for long positions. They then use this to provide metrics to position for short or long moves. An example of a rule provided is that if the PER of a company is above 20 then move to a short position if it is below 10

invest in a long position. Each indicator then quantifies a parameter.

Instead of directly crafting their machine learning algorithm, They choose to use a software tool known as Jacket for parallelsization and implemntation details. The software package is written in Matlab and natively allows for GPU implementation. This prvoided substantial reasons why software packages should be utilized over the creation of individuals tools. Throughout their study, they find that their for loop structures provided the most acute areas for parallelziation and functionality. A major issue noted with this sitution is that the memory capacity of the GPU limits the number of generations that can be executed on a GPU.

Short selling is the selling of a security that the seller does not own, or any sale that is completed by the delivery of a security borrowed by the seller. Short selling is a legitimate trading strategy. Typically, a short sale involves the sale of a security at the current price which is settled with shares lent to the short seller by a third party. The seller makes the short sale on the assumption that the price of the security will go down. If this occurs, the short seller will purchase shares to lock in a profit, extinguish the short position and replace the shares previously borrowed. Of course, if the stock rises in price, the short seller may elect to close out the position through a purchase, and absorb the resulting loss. Firms are required to report their short positions as of settlement on the 15th of each month. A compilation is published eight business days after. (NasDaq Publications) COmpare differences between the Sqrt and regular distributions Correlations $SUMMARY(NO_PROCESS_DATA)$

## 3.3 BLACK SCHOALES

# 4 GPU Rationale

# 5 Keywords

# 6 Model Creation and Functions

# 7 Results

# References


[1] J. Gilyard, "Personal website," 2016.

[2] J. Gilyard, "Project repository," 2016.

[3] D. J. Leek, D. R. Peng, and D. B. Caffo, "Practical machine learning," 2016.

[4] M. Kuhn, "The caret package," 2016.

[5] M. Kuhn, "Building predictive models in r using the caret package," 2008.

[6] H. B. School, "Bloomberg limits," 2015.

[7] *S & P US Indices Methodology*.

[8] D. Mckenney and T. White, "Stock trading strategy using gp on gpu," *FOCUS*, 2011.

[9] I. Contreras, Y. Jiand, J. I. Hidalgo, and L. Nunez-Letamendia, "Using a gpu-cpu arhitecture to speed up a ga-based real-time system for trading the stock market," *Soft Comput*, 2012.