In August 2012 Knight Capital, a leading high-frequency-trading firm, collapsed in just hours after a new algorithm loaded into its system went rogue, causing losses of $440m at frightening speed[1]. Instead of its dismissal, computer-trading moved beyond share trading and seize dominance in some of the biggest capital markets in the world: the corporate- and government-bond markets as well as those for credit derivatives, interest-rate swaps and currency swaps. Man AHL, a well-established $18.8bn quant fund provider, has been conducting research into machine-learning for trading purposes since 2009, and using it as one of the techniques to manage client money since 2014[2]. A machine-learning algorithm however has yet to beat conventional trading strategies based on rules set by humans.

Recurrent reinforcement learning can provide immediate feedback to optimize the strategy, ability to produce real valued actions or weights naturally without resorting to the discretization necessary for value function approach.

Scope, Input-Output behavior of the System

Our project is based on the system described in the paper by M. Dempster and V. Leemans[3] that utilized a recurrent reinforcement learning system to create an automated trading system. We will build on their system by adding a machine learning component that will take market and trade data as inputs to generate trading signals. The trading signals will be passed to the recurrent reinforcement learning algorithm to determine the optimal way to trade those signals.

The recurrent reinforcement learning algorithm will optimize using a metric that maximizes pnl while minimizing the volatility of losing days. A commonly used metric is the Sortino Ratio which is the average return divided by the standard deviation of negative returns.

The recurrent reinforcement learning algorithm will be allowed 3 basic actions: Buy, Sell, Do nothing. Additional risk management constraints of a stop loss, maximum daily drawdown, and maximum position limit will be included as well.

1. Trade Signal Prediction Algo: Supervised Machine Learning, Kalman Filter, Non-Supervised Learning
2. Trade Optimization Algo: Recurrent reinforcement learning
3. Objective Function: Sortino Ratio
4. Actions: Buy (long), Sell (short), Neutral (do nothing)
5. Constraints: Maximum position limit, trade stop loss, maximum daily drawdown
6. Optimizer: Gradient Ascent

Preliminary data, and Examples of Inputs and Outputs

The preliminary data set consists of 1 minute open, high, low, and close data for the front month Hang Seng Index future from November 1, 2016 to August 31, 2017 compiled from historical raw tick data from the Hong Kong Futures Exchange. Included in the data are:

- Average Trade Price
- Traded Volume
- Bid Quantity and Bid Price for 3 levels

---

[1] “Rise of the robotraders”, The Economist, Nov 18th 2013

[2] “Machine-learning promises to shake up large swathes of finance”, The Economist, May 25th 2017

[3] “An Automated FX Trading System Using Adaptive Reinforcement Learning”, M. Dempster and V. Leemans, 2004.

- Ask Quantity and Ask Price for 3 levels
- Bid Ask Spread
- LnReturn for 1, 5, 10, 30 minute intervals

Also included is a classification prediction with probabilities generated from a Machine Learning model using proprietary book, trade, and technical indicators. The targets are:

- +1 for a 1 minute forward return greater than + 10 points in the market that maintains that level for an additional 30 seconds
- -1 for a 1 minute forward return of -10 points in the market that maintains that level for an additional 30 seconds
- 0 for everything else

The outputs from the recurrent reinforcement learning algorithm should include the Sortino Ratio, daily pnl, and daily trade count.

Baseline and Oracle

The baseline model is the supervised learning model with expert designed features that will be fed into the recurrent reinforcement learning algorithm. This current model has been backtested against a simplistic trading algorithm and can generate a modest positive Sortino Ratio.

The oracle is the actual target predictions that are fed into a recurrent reinforcement learning algorithm. The oracle model will represent the best trading strategy possible to maximize the Sortino ratio.

We will also attempt to bridge the gap between the baseline and oracle with other prediction models such as using a deep learning neural network (non-supervised) and a Kalman filter to predict forward returns.

The major challenge that we will most likely encounter is the limited amount of data that we have to train and test the algorithm. We currently have intraday tick data from November 1, 2016 to August 31, 2017.

Search and value iteration might be able to address the challenge of having a limited amount of data.

Similar Projects

The value function approach had dominated the field throughout the last thirty years. There are few public known applications of recurrent reinforcement learning in the financial world.

- J. Moody, et al. first proposed adaptive algorithm called recurrent reinforcement learning (RRL) for discovering investment policies[4]. The direct reinforcement approach differs from dynamic programming and reinforcement algorithms such as TD-learning and Q-learning, which attempt to estimate a value function for the control problem. The RRL direct reinforcement framework enables a simpler problem representation, avoids Bellman's curse of dimensionality and offers compelling advantages in efficiency.
- David Lu aimed to have AI to trade itself with downside protection and as few tweaks on the parameters as possible, based on deep learning and LSTM recurrent neural networks. His controlled experiments demonstrated using FX data with skewed returns distributions as a successful USDGBP trading agent[5].

[4] "Learning to Trade via Direct Reinforcement", John Moody and Matthew Saffell, IEEE Transactions on Neural Networks, Jul 2001

[5] "Agent Inspired Trading Using Recurrent Reinforcement Learning and LSTM Neural Networks", David W. Wu, Jul 23rd 2017