

# 利用深度学习动态调整交易策略

2017.06.13

王尉同/罗欣/吴凡/郭晓睿/宋飞龙/王摧/曾文星/周龙梅/张子萃/黄薇/罗思蜀

Fintech2017 训练营第二期第 2 小组

## 一、背景

### 1.1 程序化交易

随着市场的交易数据越来越多，对大量数据进行快速分析的需求越来越迫切。另外在一些高频交易中，依靠人发出交易指令很可能滞后导致获利机会的丢失。

为了解决这些问题，计算机技术被应用在交易中。依靠程序自动分析已有的数据，发掘套利的机会并发出交易的指令，从而获利。目前程序化交易的应用主要有如下两个方面：

1. 对基本面信息的分析，可采用程序自动分析公司财务报表之类的文件，得到该公司的资金，财务，发展状况等有价值的基本面信息帮助交易人员决策。
2. 对交易信息的分析，采用程序分析证券之间的相关性，在整个市场中寻找相关性较强的证券对，做配对交易或者分析证券的自相关性预测证券价格走势。除此之外，现在运用比较多的是根据量化人员的交易经验分析证券价格、波动的模式，得出一些均线策略，反转策略，突破策略等。

程序化交易（算法交易）优势主要体现在历史数据评估、高效执行、无主观情绪输入、可度量评价、交易频率等方面。但其开发成本也较高、技巧复杂。

一方面，经过大规模的对数据的分析得出的判断具有更高的准确性和可信度；另一方面，程序可分析出比较抽象的模式，这些模式凭人的经验可能很难从历史数据发现。不仅如此，快速的交易指令的发出使得对高置信概率的套机机会把握的更加及时。然而，现有的程序化交易算法还有很多的不足。大多数都需要设计者有丰富的交易经验，对交易经验进行提炼，得出交易思想，然后用程序实现。这样的方式受限于交易人员的交易经验，不利于充分分析市场数据来发掘更多套利机会。

### 1.2 程序化交易流程

程序化交易通常基于某种平台，经过以下流程：

1. 提出假设
2. 建立模型
3. 回测验证
4. 执行交易

其交易策略通常基于：市场微观结构研究（for HFT 高频交易）；基金结构套利(fund structure arbitrage)；机器学习、人工智能算法。

其中机器学习的作用通常在限价订单簿建模(Limit Order Book Modeling), 历史价格学习, 添加文本情感等辅助特征, 基于反馈的强化学习等方面。

### 1.3 深度学习

近 5 年来, 深度学习在计算机视觉、自然语言处理等领域遍地开花, 大幅度提高了各大领域关键问题的准确率。其端到端训练的框架、自动抽取特征的能力、强大地通用型和可扩展性, 使其成为人工智能领域最火热的话题。目前最流行的深度学习结构, 有空间上权值共享的卷积神经网络和时间上权值共享的循环神经网络 (convolutional neural network, CNN), 后者主要代表是因克服了梯度弥散而流行的长短时(Long-short term, LSTM) 神经网络。CNN 在图像数据上效果很好, 而 LSTM 则主要用于序列数据上。本课题的股票数据可以视作时间序列输入, 可以直接用 LSTM 建模。

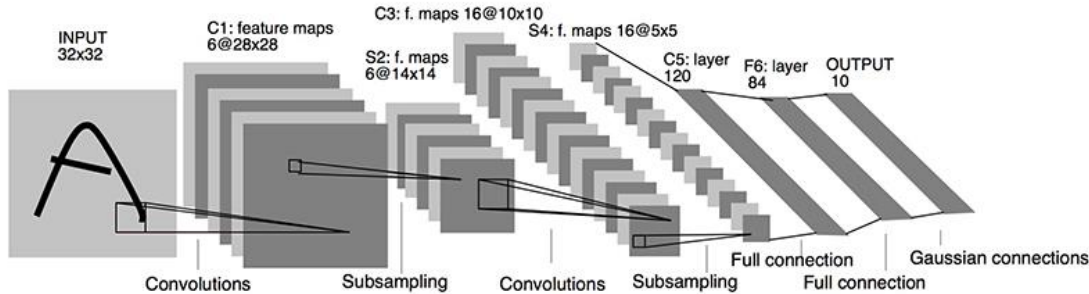


Figure 1 LeNet5 手写体识别

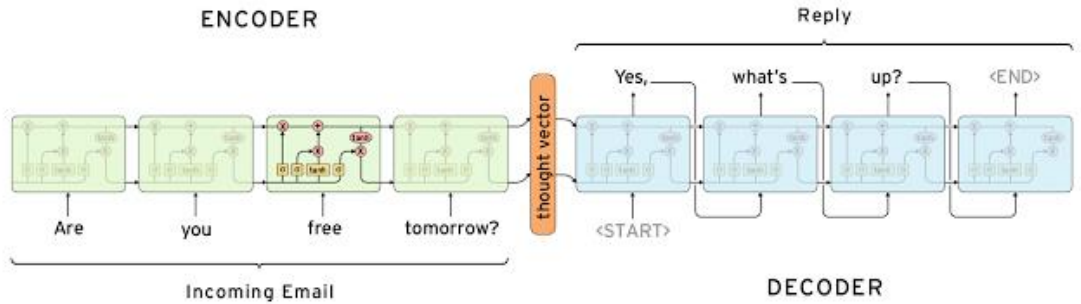


Figure 2 基于 LSTM 编解码框架的邮件自动回复

### 1.4 递归神经网络 RNN

在递归神经网络中, 我们加入了另一组神经元之间的连接。这些连接允许把同一个隐藏层的神经元的激活值在下一个序列步骤反馈回自身。简单的说, 在序列的每一个步骤中, 隐藏层除了从下层接受信号外, 还从自身接受信号。此机制给了递归神经网络一个重要的性质: 记忆。递归神经网络中具有循环结构, 可以使得信息持续的保存。

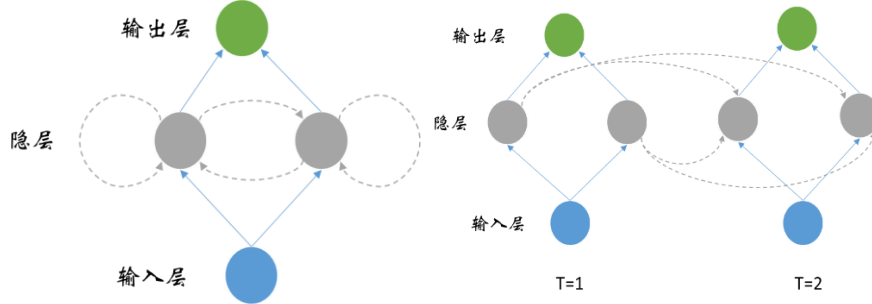


Figure 3 递归神经网络 (RNN) 的循环结构

将上面  $t$  时刻隐层的循环体结构拆开就会得到如下的递归神经网络信息处理的形式，

$$h_t = \phi(W_{x_{t-1} \rightarrow h_{t-1}} \cdot x_t + W_{h_{t-2} \rightarrow h_{t-1}} \cdot h_{t-1} + b) \quad (1)$$

其中， $W_{x_{t-1} \rightarrow h_{t-1}}$  为  $t-1$  时刻输入到输出连接的权重， $W_{h_{t-2} \rightarrow h_{t-1}}$  是隐层自身上  $t-2$  时刻到  $t-1$  时刻的连接权重， $b$  为网络连接权值的偏置（一个小随机数）。如果只考虑隐层自身上的传递过程，不考虑非线性的部分，那么我们会得到一个简化的递归形式，

$$h_t = W_{h_{t-2} \rightarrow h_{t-1}} \cdot h_{t-1} \quad (2)$$

如果将起始时刻的隐藏状态信息  $h_0$  向第  $t$  时刻传递，可以得到

$$h_t = (W_{h_{t-2} \rightarrow h_{t-1}} \cdot W_{h_{t-3} \rightarrow h_{t-2}} \cdots W_{h_0 \rightarrow h_1}) \cdot h_0 \quad (3)$$

当  $W_{h_{t-1} \rightarrow h_t}$  的特征值小于 1 时， $W_{h_{t-2} \rightarrow h_{t-1}} \cdot W_{h_{t-3} \rightarrow h_{t-2}} \cdots W_{h_0 \rightarrow h_1}$  向 0 衰减；当  $W_{h_{t-1} \rightarrow h_t}$  的特征值大于 1 时， $W_{h_{t-2} \rightarrow h_{t-1}} \cdot W_{h_{t-3} \rightarrow h_{t-2}} \cdots W_{h_0 \rightarrow h_1}$  向  $\infty$  扩增。这种情况下想要传递的  $h_0$  中的信息就会被掩盖，并不能传递到  $h_t$ 。

因此，这种循环递归神经网络可以实现短时的记忆功能，对于长时间的时间关系，学习效果就会非常差。基于此，我们的解决方案采用的是克服短时记忆限制的长短期记忆网络算法。

## 1.5 长短期记忆网络模型 (LSTM)

### 1) 概述：

长短期记忆网络 (LSTM) 算法是递归神经网络的一种，主要用于序列数据的建模，其核心结构是在标准的递归神经 ( $h_t = \phi(W_{x_{t-1} \rightarrow h_{t-1}} \cdot x_t + W_{h_{t-2} \rightarrow h_{t-1}} \cdot h_{t-1} + b)$ ) 的基础上装备了若干个控制数级的“门”。即可以理解成在 RNN 整体中加入其它神经网络（“门”），而这些“门”只是控制数级，控制信息的流动量的大小。

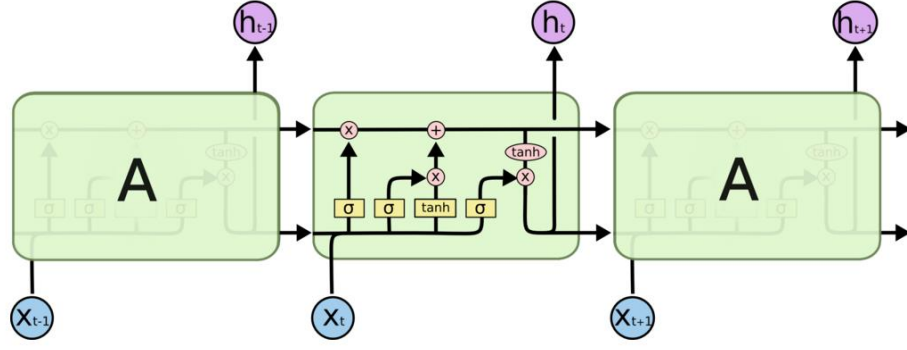


图 3. LSTM 中递归模块的展开图

## 2) LSTM 核心理念

关于“门”的概念，我们可以把它看成是激活函数，它的输入量是控制依据，输出量是值为 (0,1) 的数值，表示该如何调节其他数据的数级的控制方式。因此“门”所产生的输出会用于控制其他数据的数级，相当于过滤器的作用，即有选择的让信息通过。一般情况，“门”是由一个 sigmoid 神经网络层和逐点乘法运算组成。其中 sigmoid 层输出的“0”表示“禁止任何信息成分通过”，“1”表示“让所有成分通过”。

在 LSTM 中，网络首先构建了三个“门”来进行信息流通量的控制，分别为输入门  $i_t$ 、遗忘门  $f_t$ 、输出门  $o_t$ 。

$$\begin{aligned} i_t &= \text{sigmoid}(W_{xi} \cdot x_t + W_{hi} \cdot x_{t-1} + b_i) \\ f_t &= \text{sigmoid}(W_{xf} \cdot x_t + W_{hf} \cdot x_{t-1} + b_f) \\ o_t &= \text{sigmoid}(W_{xo} \cdot x_t + W_{ho} \cdot x_{t-1} + b_o) \end{aligned} \quad (4)$$

其中，

- 输入门  $i_t$ ：控制有多少信息可以流入隐层的记忆细胞
- 遗忘门  $f_t$ ：控制有多少上一时刻的记忆细胞中的信息可以累积到当前时刻的记忆细胞中。
- 输出门  $o_t$ ：控制有多少当前时刻的记忆细胞中的信息可以流入当前隐藏状态  $h_t$  中。

其次，关于信息传递的信息流，其来源只有三处，当前的输入  $x_t$ ，上一时刻的隐藏状态  $h_{t-1}$ ，上一时刻的细胞状态  $c_{t-1}$ ，其中  $c_{t-1}$  是额外制造出来、可线性自连接的单元。真正的信息流来源可以说只有当前的输入  $x_t$ ，上一时刻的隐藏状态  $h_{t-1}$  两处。三个“门”的控制依据，以及数据的更新都是来源于这两处。

再次，关于历史信息的积累，采用式 (5) 进行计算，

$$c_t = f_t \odot c_{t-1} + i_t \odot \tanh(W_{xc} \cdot x_t + W_{hc} \cdot h_{t-1} + b_c) \quad (5)$$

其中， $\tanh(W_{xc} \cdot x_t + W_{hc} \cdot h_{t-1} + b_c)$  是本次要积累的信息来源。所以历史信息的积累并不是靠隐藏状态  $h$  自身，而是靠记忆细胞这个自连接来积累。在积累时，靠遗忘门来限制上一时刻的记忆细胞的信息，并靠输入门来限制新信息。

最后，我们要对当前隐藏状态进行计算，采用下式 (6)。

$$h_t = o_t \odot \tanh(c_t) \quad (5)$$

当前隐藏状态  $h_t$  是从  $c_t$  计算得来的，由于  $c_t$  是以线性的方式自我更新的，所以现将其加入带有非线性功能的  $\tanh(c_t)$ ，之后在依靠输出门  $o_t$  的过滤得到当前的隐藏状态  $h_t$ 。

## 二、问题整体分析

问题：  
根据某金融标的（伦敦黄金价格）价格，通过应用深度学习技术，让交易员实现自动更新市场认知，动态调整交易策略。

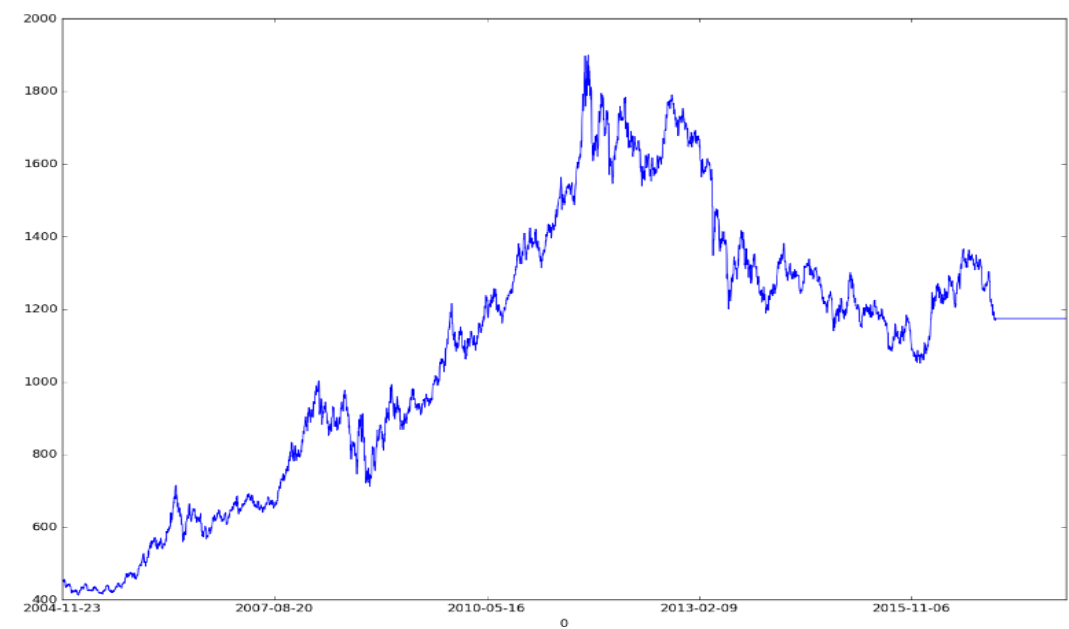


Figure 3 伦敦金价随时间变化图 2004/11/23- 2016/12/8

分析:  
利用深度学习技术在严格管控风险的前提下动态调整交易动作使得收益最大化。  
已知,  
为了在可解释的框架下分析此问题, 我们先尝试取一定时间窗口 T (30 天) 内的金价和辅助特征作为输入, 用 LSTM 预测金价单日涨幅, 在此基础上, 设计风险期望可调的交易策略。  
从这种思路出发, 我们设计了如下解决方案：

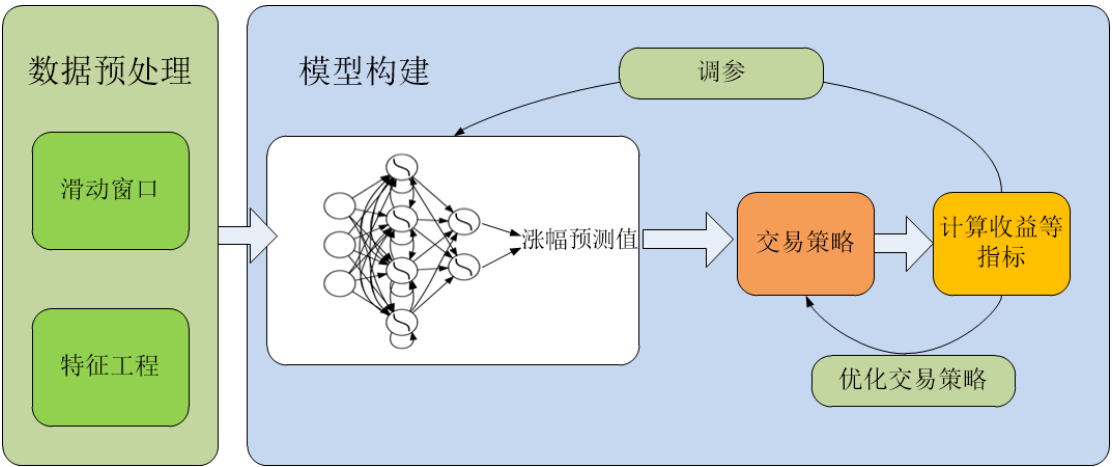


Figure 4 整体解决方案

将问题分解为两步，预测次日涨幅，制定交易策略。两步建模主要考虑以下优点：

1. 用 LSTM 学习输入的序列特征并作预测，在深度学习领域已经是一个研究很深入的方向——序列学习，这一步既可以专一考虑深度学习算法的调参和优化，又有明确的优化目标。因而单独出来，也可以给出很好的解释。
2. 交易策略一方面涉及到金融领域的先验知识，另一方面也需加入不同的风险控制项，因而可以先假定之前的预测在某种阈值概率下是可信的，在不同风险需求下去优化交易策略的参数。
3. 市场整体行情的变化已经考虑在第一步预测中，可以根据实时行情的变化对模型进行重新训练或进一步 finetune 微调。行情的变化带来风险需求的变化，可以实时改变第二步的策略。因而在 2 步都可以根据行情变化对最终策略进行动态调整。

## 三、基于 LSTM 的金价次日涨幅预测

### 3.1 输入特征分析

除了直接的金价序列，表 2 中还给了相关的一些金融数据的时间序列变化。这里我们先用机器学习的方法筛选出关联较大的几个特征。

给定有效数据从 2004/11/23 到 2016/12/8，由于预测过程在周末等非交易日是不能做买卖交易的，虽然周末新闻事件很可能对非交易日后的金价造成影响，但考虑到预测和行为的一致性，我们先去掉了非交易日的数据。

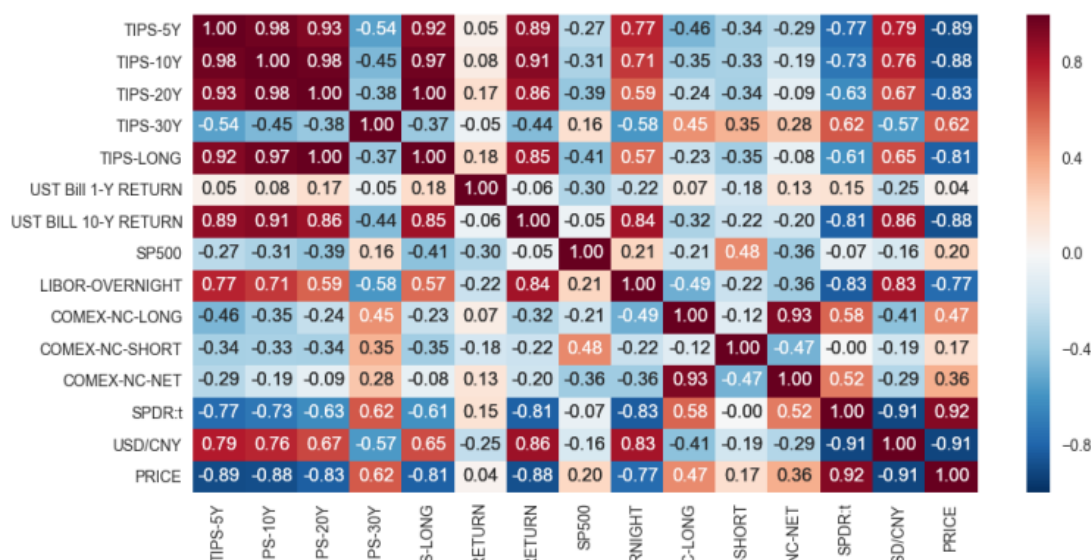


Figure 5 所有特征（包括金价）相关性热图

所有特征相关性矩阵如上图所示，与金价 price 较为相关的特征是：TIPS-5Y 和 TIPS-10Y (Treasury Inflation-Protected Securities)5 年和 10 年的通货膨胀保值债券，SPDR:t (SPDR 黄金 ETF 持仓量)，USD/CNY (美元兑人民币汇率)，UST BILL 10-Y RETURN (美国国债收益率十年期) 等。

特征名称为['price', 'TIPS-5Y', 'TIPS-10Y', 'TIPS-20Y', 'TIPS-LONG', 'UST BILL 10-Y RETURN', 'LIBOR-OVERNIGHT', 'SPDR:t', 'USD/CNY'], 共 9 个。



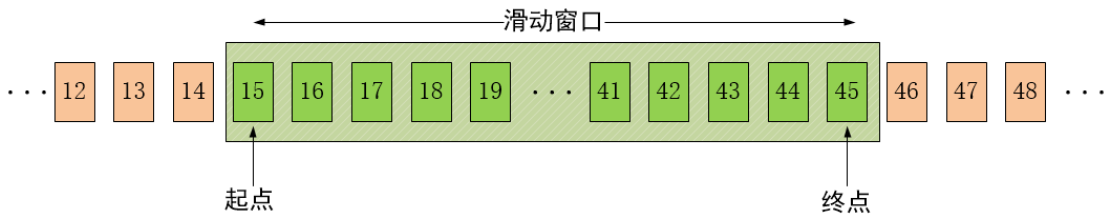
其中，金融常识可知金价走势通常与美元负相关，这与我们的分析符合。  
由于这 9 个输入特征，先各自归一化，再用滑动时间窗采样作为 LSTM 的输入序列。

### 3.2 滑动时间窗

采用滑动时间窗方式进行对所选特征及价格进行序列内归一化处理。经过多次试验，我们选取时间窗大小为 30。即所有标准化后的 10 个特征均做以下时间窗处理：

$$\bar{x}_t = \frac{x_t}{x_{t-30}} - 1$$

如下图所示：



滑动时间窗使得 LSTM 的学习有了明确的目标，即学习时间窗内价格序列的相对变化趋势，捕捉时间维度上的关联。

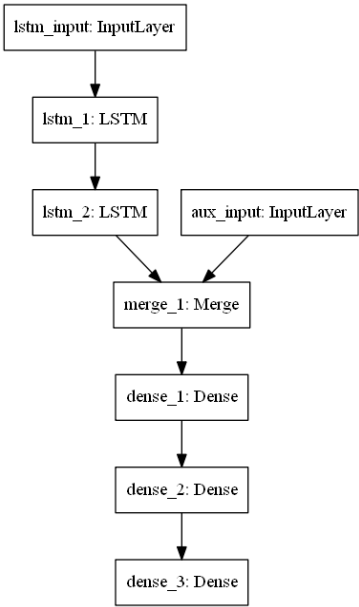
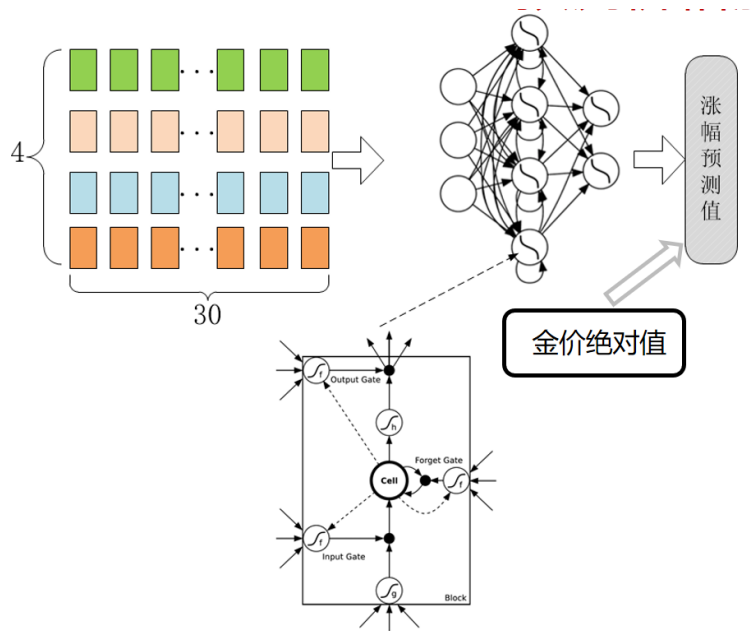
### 3.3 LSTM 训练

我们将整个数据后 10%作为测试集，前 90%中的 80%和 20%分别作为训练集和验证集。

时间窗序列化后输入到 2 层的 LSTM。

LSTM 有如下优点：

1. 能有效地捕捉滑动时间窗内的序列前后的关联。
2. 有效避免梯度弥散，长短程依赖通过自动学习权值。



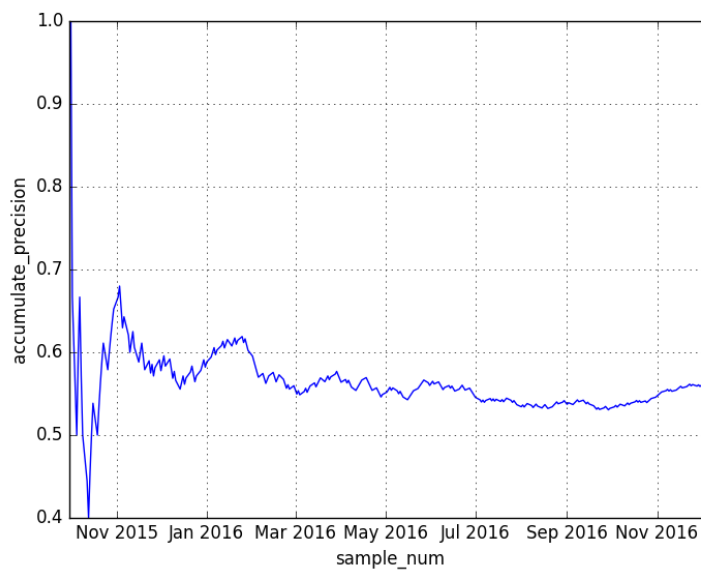
时间窗归一化只考虑了相对值，其实金价绝对值也会对其预测造成影响，因为金价本身应该在某一个范围内波动，如超过 1600 美元/盎司再上涨的空间就不大了。因而将金价绝对值作为全连接层的输入特征。

由于监督训练用的是 30 日窗口，序列预测输出是相对于 30 天之前的涨幅： $\hat{a}_t = \frac{\hat{d}_t}{d_{t-30}} - 1$

应该做一步换算，转化为单日相对涨幅： $\hat{z}_t$ 。

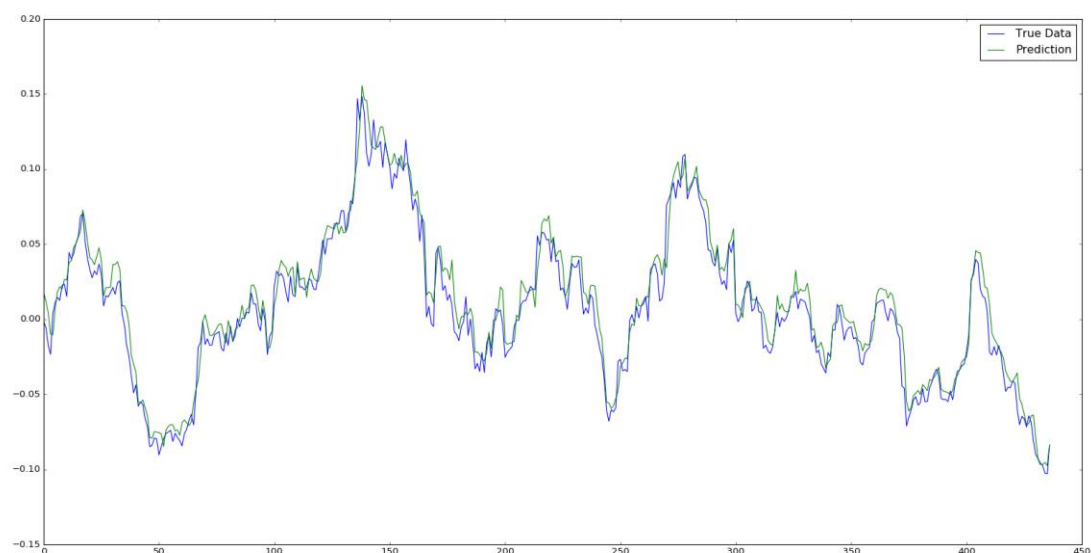
此步可以得到测试集的涨幅预测。

训练集部分，涨跌预测准确率变化如下图：



整体预测准确率为 56%。这一准确率表明，预测的涨跌方向准确率都不是很高，因而交易策略应该更多地考虑到一些 LSTM 预测错误所带来的风险变化和策略鲁棒性。

预测涨幅与真实涨幅画图如下：





## 四、基于预测与风险控制的交易策略

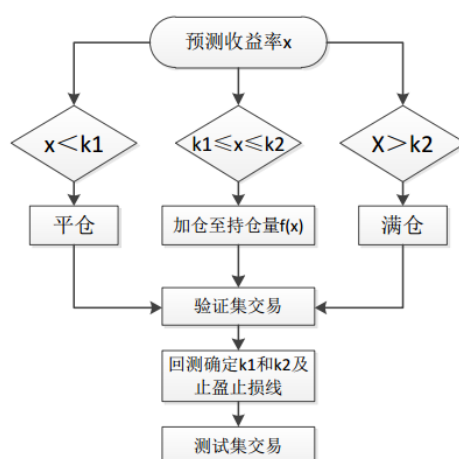
### 4.1 策略描述与动机

投资风险是指对未来投资收益的不确定性,在投资中可能会遭受收益损失甚至本金损失的风险。在实际交易中,在一个月中得到的收益极有可能在一周内亏完,因此降低一个策略的风险是至关重要的。由此可见一个完整的交易策略必须包含对风险的评估。

在实际中,我们所构建的模型有一定误差,并且,根据历史数据建立的模型并不一定能准确的预测未来行情的走势。因此通过一些手段降低风险是必要的。

我们基于止损止盈的思想构建了一个能够动态调整仓位的函数,并结合模型所给的预测信号制定出一个能够满足不同风险偏好的交易策略。

流程图如下：



### 4.2 策略理解

#### 1) 因势而变：根据模型实时效果动态调整交易策略

模型通过历史数据的信息来预测未来的走势,但行情总是在变换的,我们就需要对模型进行更新。在当前行情下模型失效的时候,我们会选择重新训练模型以适应新的行情。

#### 2) 因您而变：根据投资者的风险偏好动态调整交易策略

根据不同投资者的不同风险偏好,设置止损线,当亏损超过止损线时及时平仓止损,可以帮助投资者保护本金,降低风险;设置止盈线,当盈利超过止盈线时及时平仓锁定收益,可以帮助投资者把钱赚稳,降低风险。

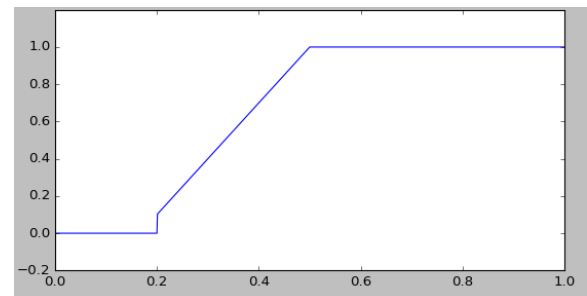
### 4.3 仓位选择

仓位选择函数基于以下两点构建：

1. 利用平仓分隔点  $k_1$ ，以及满仓分隔点  $k_2$  来调整风险。
2. 假设仓位与基于模型预测出的收益率线性相关。

函数(1)即为仓位选择函数，其中  $x$  为根据 LSTM 算法预测的涨幅， $k_1$  表示平仓分隔点的可调参数， $k_2$  为表示满仓分隔点的可调参数。图(1)为该函数取 0.2 时的图像，其中  $x$  轴为预测涨幅， $y$  轴为持仓量，从图上可以看出：

- 1) 该策略根据算法所给的信号决定持仓量，当预测涨幅越大，。
- 2)  $K_1$  为平仓分隔点，一旦预测的涨幅低于了  $k_1$ ，就平仓。 $k_1$  越小，策略风险越大。
- 3)  $K_2$  为满仓分隔点，一旦预测的涨幅高于  $k_1$ ，就满仓。 $k_2$  越小，策略风险越大。



$$f(x, k_1, k_2) = \begin{cases} 0 & x < k_1 \\ \frac{0.9(x - k_1)}{k_2 - k_1} + 0.1 & k_1 \leq x \leq k_2 \\ 1 & k_2 < x \end{cases} \quad (1)$$

### 4.4 风险描述

在实际操作时，将基于回测结果的好坏来调整参数  $k_1$ ， $k_2$ （我们将通过最大回撤率、盈亏比、胜率来描述策略的风险，用年化收益率来描述策略的好坏。）再根据不同的风险需求选择不同大小的  $k_1$ ， $k_2$ 。以下为实验结果：

**策略(1)： $k_1=-0.003$ ,  $k_2=0$ ，不设止盈止损线**

高风险高回报

年收益率	最大回撤率	盈亏比	胜率
5.17%	3.63%	1.35	54.39%

**策略(2)： $k_1=0$ ,  $k_2=0.0002$ ，止盈止损线 30%**

适中风险适中汇报

年收益率	最大回撤率	盈亏比	胜率
3.03%	3.07%	1.015	61.9%

**策略(3)： $k_1=0.0001$ ,  $k_2=0.00018$ ，止盈止损线 10%**

低风险低回报

年收益率	最大回撤率	盈亏比	胜率
2.21%	2.93%	0.876	63.16%

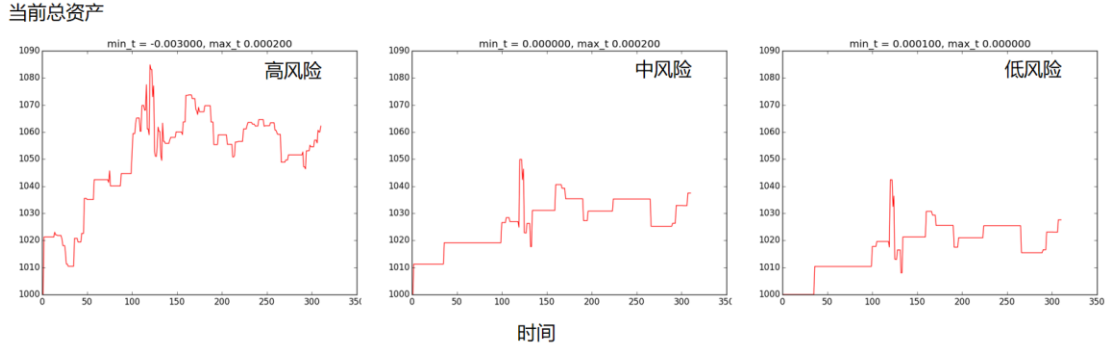


Figure 6 不同风险下资产随时间变化情况

## 4.5 策略实现过程

依据预测的每日收益率，按照上述的交易策略完成交易操作，最后计算每日的年化收益率、最大回撤率、胜率和盈亏比来评价交易策略的实际执行结果。下面将首先介绍这四个策略评价指标的计算方法和意义。

### (1) 年化收益率

设第  $i$  天的总资产为  $A_i$ ，初始总资产为  $A_0$ ，一年的交易日数量为 252 天，则可以计算

年化收益率： $\left(\frac{A_i - A_0}{A_0}\right)^{\frac{252}{i}}$ 。年化收益率每日更新一次，是交易策略执行结果好坏最直接的评价指标。

### (2) 最大回撤率

最大回撤率表示最坏情况（即最高点买入，最低点卖出）时的亏损率。设第  $i$  天的总资产

为  $A_i$ ，设第  $j$  天的总资产为  $A_j$ ，且有  $i < j$ ，则最大回撤率的用公式表示为： $\max(\frac{A_i - A_j}{A_i})$ 。

最大回撤率每日更新一次，是评估交易策略风险最重要的指标之一。在计算第  $k$  天的最大回撤时，我们先找到第  $k$  天之前总资产的最大值，即为  $A_i$ ；再找出第  $i$  天到第  $k$  天之间总资产的最小值，即为  $A_j$ ，代入公式进行计算。

### (3) 胜率

从零仓位开始买入、调整再到平仓记为一次交易。设交易的总次数为  $N_{sell}$ ，这其中赚钱

的交易次数为  $N_{win}$ ，亏本的交易次数为  $N_{lose}$ ，则可以计算胜率  $\frac{N_{win}}{N_{sell}}$ 。胜率在每次平仓时更

新一次，是交易策略执行结果好坏的评价指标之一。

### (4) 盈亏比

胜率高并不意味着赚的钱多，如果赚钱的交易中收益率低，而亏本的交易中亏损率高，最后的总结果可能反过来是亏损。因此我们需要计算加权的胜率即盈亏比，赋予的权值是平

均获利幅度  $\overline{R_{win}}$  和平均亏损幅度  $\overline{R_{lose}}$ 。盈亏比在每次平仓时更新一次，用公式表示为

$$\frac{N_{win} \cdot \overline{R_{win}}}{N_{lose} \cdot \overline{R_{lose}}}。$$

策略评价指标有两个应用，直观的应用是在测试集中评价策略执行结果的好坏，作为最终的输出结果。除此之外，我们还划分了验证集，在验证集中计算这一套指标，利用输出结果来确定不同风险等级下  $k_1$  和  $k_2$  的值。

## 4.6 三种不同风险等级下阈值的确定

在验证集中，我们使用了随机搜索（Random Search）和网格搜索（Grid Search）两种方法搜索  $k_1$  和  $k_2$  的值。搜索的意思是：代入不同的  $k_1$  和  $k_2$  运行策略并计算评价指标，根据评价指标确定合适的  $k_1$  和  $k_2$ 。

随机搜索是初步尝试，在一个大范围的区间里随机选择参数，通过模型在验证集上的表现得到最佳参数的大致区间。在这个基础上使用网格搜索，即在区间内通过设定步长搜索得到模型在验证集上的最佳参数。

## 五、总结

1. 用 LSTM 学习伦敦金价及相关历史数据来预测次日涨幅，涨跌正确率 56%。
2. 按照不同的风险偏好选择交易策略超参数，在测试集上达到了较好的回测效果。
3. 2 步的框架经过超参数的经验初始化后，可以在线地自动完成整套交易过程。
4. 交易策略的动态调整体现在：
  - 1) 根据投资者的不同风险偏好，动态调整持仓量；
  - 2) 当涨幅预测准确率下降时，重新训练模型。
5. 多模型融合可能会带来性能提升和动态调整的灵活性。
6. 若将交易动作为学习目标进行建模，可加入强化学习的训练框架。但目前相关做法的性能并不理想。可以参考 multi-agent, q-learning 等内容。

## 六、参考

1. <https://yq.aliyun.com/articles/68463>
2. <http://machinelearningmastery.com/sequence-classification-lstm-recurrent-neural-networks-python-keras/>

3. <http://blog.csdn.net/u011414416/article/details/46724699>

## 七、致谢

非常感谢招商银行如此负责地组织此次训练营!特别要感谢全程陪伴和指导我们的所有招行工作人员。从接机时兴奋的招手到送机时一直等在安检口确保我们按期登机,是他们的热情投入,让我们在紧凑的日程中仍能感受到温暖和关怀。他们每一个细节的默默付出,才成就了我们此次美好而难忘的训练营。也非常荣幸能够结识到这么多优秀的志同道合的小伙伴。从一次次的讨论、争论到分工、合作,我们组也从一个各抒己见的零散状态,进化到为按同一个主线方案各自出力的密切合作状态。也正是这股招行传递给我们的认真与务实,让我们第2小组的同学,决定在训练营结束后,将此次的所有代码和文档整理起来,也就有了本文这篇报告。也期待我们与招行,与导师们,与小伙伴们的缘分能有延续!

2017.6.13  
起草于招银大学