基于神经网络的股票预测分析

胡笛 2620170029　　顾炎极 2620170030　王辉 2620170027

源码及数据所在 github 仓库：https://github.com/hudibbi/gupiaoproject

摘要

股票预测是指以准确的调查统计资料和股市信息为依据，从股票市场的历史、现状和规律性出发，运用科学的方法，对个股或大盘的未来走势做出预测。本文将介绍以股票的日交易数据为基础，利用数据挖掘技术对股票进行预测，主要包括数据探索，指标建立，数据标准化，指标筛选，模型建立，模型测评几个步骤。

关键词：股票；预测；数据；挖掘；

1. 导论

1.1 问题描述

在金融系统的预测研究中，股票预测是一个非常热门的课题。这是因为股票市场具有高收益与高风险并存的特性，随着股市的发展，人们不断在探索其内在规律，对于股市规律的认识逐步加深，产生各种各样的股市预测方法。但是，股票市场作为一种影响因素众多、各种不确定性共同作用的复杂的巨系统，其价格波动往往表现出较强的非线性的特征。利用其历史数据可以对股票进行一定的预测。

1.2 问题分析及文章结构

为了能实现股票的预测，最基本的也是不可或缺的条件是有每一支股票每日交易的日线数据，即包括日期、开盘价、最高价、最低价、收盘价、成交量这六个基本变量。在得到原始数据后，为了能够较准确地刻画每支股票，我们需要一些指标去描述这些股票的特征。然后将数据标准化来消除变量间的量纲（单位）影响和变异大小因子的影响，使变量具有可比性。再根据相关性来筛选指标，使用神经网络来进行数据分类，最后进行了模型的测评。

2. 数据的处理及探索

2.1 股票日交易数据的获取

为了能实现股票的预测，最基本的也是不可或缺的条件是有每一支股票每日交易的日线 数据，即包括日期、开盘价、最高价、最低价、收盘价、成交量这六个基本变量。我们利用 choice 金融终端的 excel 插件将股票交易历史数据导入 excel 中，最终选取的是深市编号在 0 到 1000 的股票的交易数据，数据存放在 szl000_data 文件夹中，为随后的进一步挖掘提供了数据支持。

2.2 计算指标

为了能够较准确地刻画每支股票，我们需要一些指标去描述这些股票的特征。对股票来说，历史上很多技术分析大师已经发明或设计了无数的股票技术指标，经过对股票常用技术指标的筛选，最后筛选并确定的指标如下表所示：

指标标记	指标名称	计算方式
s_x1	当日涨幅	(当日收盘价一前第 n 日收盘价）/前第 n 日收盘价 x 100%
s_x2	2 曰涨幅	
s_x3	5 日涨幅	
s_x4	10 日涨幅	
s_x5	30 日涨幅	
s_x6	10 日涨跌比率 ADR	10 日内股票上涨天数之和/N 日内股票下跌天数之和
s_x7	10 日相对强弱指标 RSI	RSI= 100xRS/(l+RS). RS = n 日的平均上涨点数/n 日的平均下跌点数
s_x8	当日 K 线值	(收盘价-开盘价）/(最高价-最低价）
s_x9	3 日 K 线值	(收盘价一 3 日前开盘价）/ (3 日内最高价-3 日内最低价）
s_x10	6 日 K 线值	(收盘价一 6 日前开盘价）/(6 日内最高价-6 日内最低价）
s_x11	6 日乖离率（BIAS)	乖离率=[(当日收盘价一 6 日平均价)/6 日平均价]x 100%
s_x12	10 日乖离率（BIAS)	乖离率=[(当日收盘价-10 日平均价)/1〇日平均价]x 100%
s_x13	9 曰 RSV	(n 日收盘价-n 日最低价)/(n 日最高价一 n 日最低价)x 100%
s_x14	30 日 RSV	
s_x15	90 日 RSV	
s_x16	当日 OBV 量比	

s_x17	5 日 OBV 量比	n 日 OBV / 5 日 OBV
s_x18	10 日 OBV 量比	
s_x19	30 日 OBV 量比	
s_x20	60 日 OBV 量比	
s_y	分类指标	根据未来 1 日与 3 日涨幅来确定 s_y 为 1 或 -1

具体指标的计算代码为 github 仓库中的 step2，在 matlab 执行的结果是产生两个数据文件：

1)历上好、坏及一般股票的样本数据文件为 train orginal sample.xlsx,该文件数据的主要是用于训练模型。

2)当日所有股票的指标文件，该文件的数据主要用于预测未来股票的涨跌潜力。

2.3 数据标准化

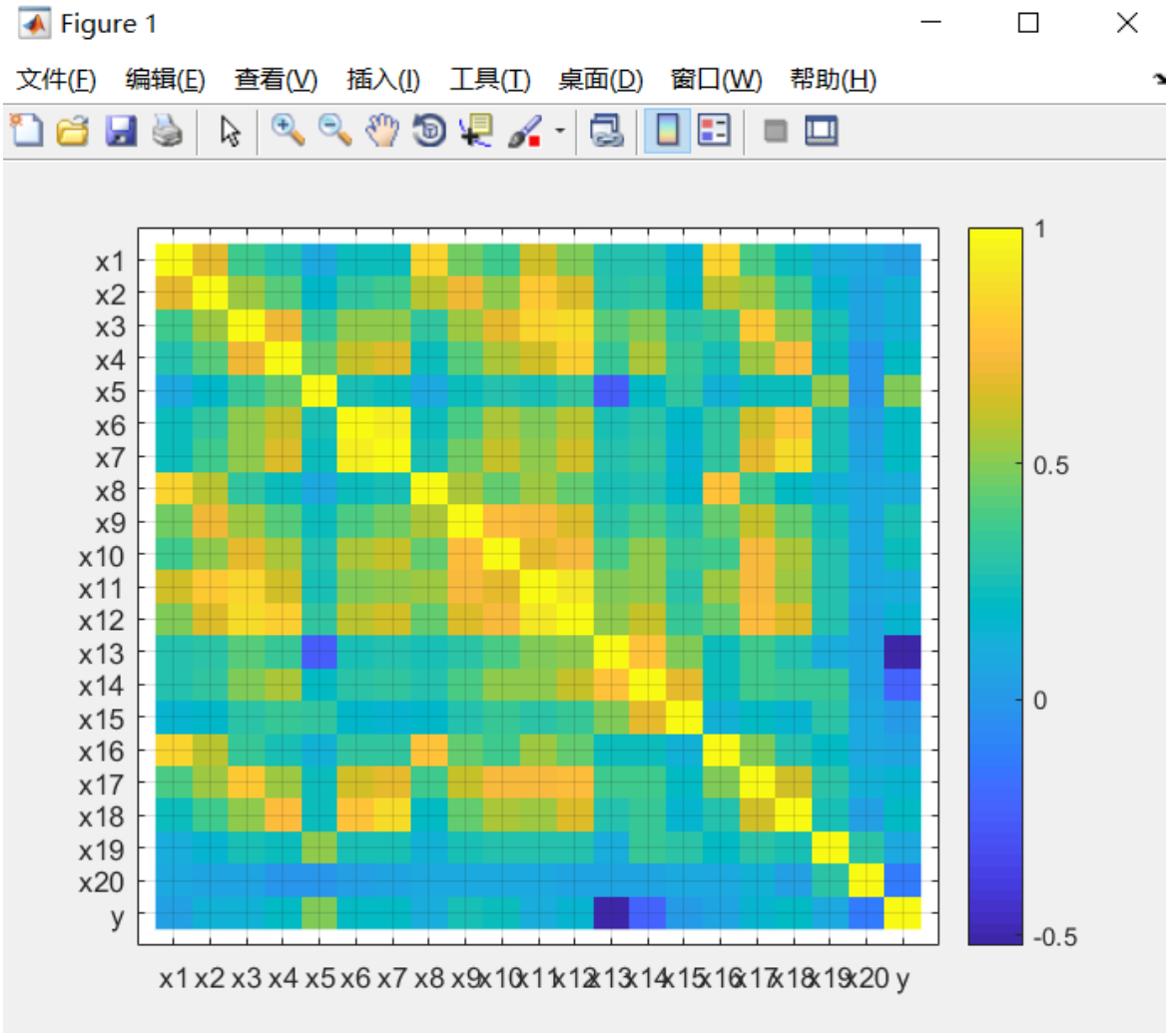
数据标准化的目的是消除变量间的量纲（单位）影响和变异大小因子的影响，使变量具有 可比性。这里将用均值方差归一化法来对数据进行标准化，所得数据在[0，1]之间，代码为 step3，在 matlab 执行此段代码后，训练样本和预测样本都被进行了标准化，且分别被保存在 train_sample.xlsx 和 forecast_sample.xlsx 两个文件中。

2.4 变量筛选

数据归一化后，其实可以直接用于训练模型，但并不确定这些变量是否都有效，如果有效性差，不仅使程序需要处理的数据量增多，而且还会影响模型的准确程度，因为相关性差或数据 质量差的变量有可能稀释模型的作用，所以一般都会对变量进行进一步的筛选。

我们采用数据相关性分析方法来确定变量之间的相关性，并定义一个相关系数阈值，来最终筛选出效果显著的变量，具体代码为 step4，在 matlab 执行此代码， 首先会得到变量间的相关系数矩阵及相关系数图(如下图所示)，从该

图可以看出，x1∽X20 与 y 的相关性有显著差异。



再设定一个相关系数阈值，可由这个阈值来确定选哪些变量，这里取 0.2。程序执行完成后，可以从两个数据文件 selected_tdata.xlsx 和 selected_fdata.xlsx 中发现，有 8 个变量被选中，部分结果如下表所示，第一列是股票编号，随后的 8 列是被选中的变量值，最后一列是分类变量，用于描述是否是好股票。

10	0	0.069164	0.302117	0.173961	0.285799	0.19314	0.128049	0.063319	1
16	0.737536	0.612279	0.361291	0.341493	0.460639	0.706977	0.453238	0.62247	1
17	1	1	1	1	0.336484	0.417322	0.611387	0.754004	1
18	0.642994	0.573002	0.77472	0.844089	0.78096	0.84974	0.426179	0.380197	1
25	1	1	0.77472	0.844089	0.673976	0.843547	0.539033	0.662783	1
25	1	0.923603	0.77472	0.844089	0.459012	0.589257	0.40598	0.529239	1
25	0.824973	0.930236	0.568177	0.676557	0.854067	0.709632	0.311905	0.40163	1
28	0.776759	1	0.444087	0.509025	0.636303	0.807272	0.44011	0.627117	1
30	0.544017	0.507041	0.568177	0.676557	0.615123	0.704927	0.499873	0.512607	1
34	0.538355	0.340719	0.77472	0.844089	0.657035	0.726068	0.380084	0.367861	1
39	0.249701	0.879735	0.361291	0.341493	0.310216	0.280028	0.140917	0.635209	1
45	0.751344	0.465547	0.568177	0.676557	0.820673	1	0.477708	0.485824	1
48	0.748947	0.820479	0.77472	0.844089	0.369029	0.685691	0.333438	0.646545	1
66	0.553435	0.499977	0.77472	0.844089	0.392919	0.493113	0.128049	0.226869	1
68	0.201307	0.39149	0.444087	0.509025	0.285934	0.365659	0.19924	0.144097	1
70	0.454509	0.415319	0.444087	0.509025	0.721773	0.781299	0.4766	0.484484	1

3. 模型的建立及评估

3.1 股票预测的方法

股票的预测方法有多种，我们最终利用分类的思想对股票进行预测。分类是在已有数据的基础上，根据各个对象的共同特性，构造或通过学习生成一个分类函数或一个分类模型，利用这个分类模型把其他数据映射到给定类别中的某一个的过程，我们使用神经网络来训练分类，最后利用神经网络训练该组数据，使其数据达到最大的区分度，此时得到的模型就可以用来预测。比如计算了某股票当日所有指标变量后代人该模型中，就可以预测该股票未来的走势分别属于这两类的可能性。

3.2 模型的训练及评估

利用分类算法对训练 样本进行训练并实现预测的效果。这里我们运用 MATLAB 自带的神经网络工具箱中的函数进行训练，代码为 step5，模型的执行结果是得到一个所有股票的排序表格（Forecastjesult.xlsx ），如下表所示，排序的依据是最后 1 列模型预测出的数据，这个数据可以理解为股票未来增长的概率，如果对这个概率按四舍五入取整，得到的则是分类数据。这个结果的作用是，在实际股票买卖过程中，可以选择排名靠前的股票买入，反之卖出。

	A	B	C	D	E	F	G	H	I	J
1	65	1	1	1	1	0.217464	0.689387	0.615622	0.933314	1.076462
2	802	0.649562	0.714952	0.590378	0.669138	0.533305	0.493489	0.119175	0.450005	0.995385
3	985	0.489474	0.388007	0.219643	0.032438	0.289402	0.922103	0.458649	0.370715	0.985637
4	582	0.350914	0.507703	0.590378	0.669138	0.58377	0.410922	0.118595	0.226798	0.940392
5	66	0.846695	0.593295	0.590378	0.669138	0.551252	0.670699	0.293865	0.605941	0.885136
6	751	1	1	0.87818	0.881371	0.332703	0.595813	0.626997	0.948292	0.88133
7	707	0	0.650724	0.302097	0.244671	0.699561	0.544556	0.236814	0.403214	0.830667
8	819	1	0.888569	0.87818	0.881371	0.613117	0.822664	0.666953	0.978776	0.826818
9	522	0.343439	0.942634	0.417467	0.456905	0.029334	0.000374	0.035146	0.607885	0.778539
10	521	0.710836	1	0.302097	0.244671	0.396943	0.315258	0.393372	0.913728	0.75364
11	885	0.285331	0.703407	0.417467	0.456905	0.358856	0.272492	0.060355	0.374782	0.73032
12	592	0.661045	1	0.87818	0.881371	0.739005	0.935041	0.7914	1	0.713166
13	29	0.085371	0.415442	0.417467	0.456905	0.703882	0.434806	0.247843	0.275041	0.701449
14	597	0	0.840481	0.302097	0.244671	0.435429	0.293521	0.205304	0.566584	0.696442
15	88	0.698538	0.985261	0.219643	0.032438	0	0.348754	0.447649	0.572891	0.694298
16	56	0.039639	0.407894	0.219643	0.032438	0.343148	0.187949	0.035146	0.158252	0.677229
17	883	0	0.790437	0.417467	0.456905	0.290722	0.235434	0.132728	0.558075	0.671945
18	830	0.448327	0.749364	0.417467	0.456905	0.431399	0.284155	0.119175	0.490821	0.666662

最后模型进行评估，评估用的是历史数据，所用的验证方法是全集验证。最终结果如下图所示，可以得到模型分类的正确率和错误率，从图中可以看出，正确率还是明显高于错误率。只要交易次数较多，从概率的角度盈利的能力已经是非常可观的。