

NATIONAL TAIWAN UNIVERSITY

FINANCIAL TECHNOLOGY

# **Project 2: Stock Price Prediction**

Lecturer:

Che Lin (林澤)

Author: Tadeo Hepperle, Student ID: A11922105

October 25, 2022

<i>CONTENTS</i>	1
-----------------	---

## Contents

<b>1 Introduction</b>	<b>2</b>
1.1 Description of the data . . . . .	2
1.2 Description of methods . . . . .	3
1.2.1 Last Value Model . . . . .	3
1.2.2 Moving Average Model . . . . .	3
1.2.3 Recurrent Neural Network . . . . .	3
1.2.4 LSTM . . . . .	4
1.2.5 GRU . . . . .	5
<b>2 Results</b>	<b>5</b>
2.1 Descriptive Statistics . . . . .	5
2.2 Last Value Model . . . . .	7
2.3 Moving Average Model . . . . .	7
<b>3 Discussion</b>	<b>8</b>

# 1 Introduction

In this project historical data on the Apple Inc. stock is used to predict the stocks price on the next day. This is a regression task for which we are going to compare multiple methods:

- A simple model always using the current days Close price to predict the next day's Close price.
- A moving average model, that predicts the next day's Close price by a learned linear combination of the past 30 Close prices.
- A Recurrent Neural Network with one layer
- An LSTM model
- A GRU model

## 1.1 Description of the data

The dataset contains various technical features concerning the Apple Inc. stock price in the time interval between 2011/01/03 and 2012/12/31 a total of 754 data points. For each data point the following 6 features are available:

- Open - price of the stock when markets opened
- High - highest price of the stock on that day
- Low - lowest price of the stock on that day
- Close - price of the stock when markets closed
- Volume - the total volume of the stock traded
- Adjusted Close - stock's value after accounting for any corporate actions

In addition to these inherent features, 5 other features were computed:

- MA10 - 10 day moving average

- MA30 - 30 day moving average
- K - from the KD-indicator line chart
- D - from the KD-indicator line chart
- Weekday - one hot encoded into 7 different binary features

The data points are continuous without gaps except for weekends and other days where the stock market is closed. For method selection and model evaluation the data was split into train, validation and test sets:

- Train set - stock price from 2011/01/01 - 2012/12/31
- Validation set - stock price from 2013/01/01 - 2013/06/30
- Test set - stock price from 2013/07/01 - 2013/12/30

## 1.2 Description of methods

### 1.2.1 Last Value Model

The simplest model to predict the next day's Close price is to just use the current day's Close price as the prediction. This model does not need any training and can be understood as a baseline model other models can be compared to in terms of their train and test loss.

### 1.2.2 Moving Average Model

A moving average in its more general definition is a linear combination of the values a subsequence of the sequence. In this model we want to form a linear combination of the past 30 Close values to predict the next Close price. For this we need to learn 30 weights.

### 1.2.3 Recurrent Neural Network

The recurrent neural network (RNN) is specified by 2 parameters: *input\_size* and *hidden\_size*. In this project we always just use an RNN with 1 layer. The

RNN learns over a period of  $time\_steps = 30$  iterations. In each iteration, an input vector of size  $input\_size$  and a hidden state vector of size  $hidden\_size$  are concatenated into a combined vector which is fed through a ReLU-activated linear layer to produce the hidden state for the next iteration. In the initial iteration the hidden state is a zero-vector. After the  $time\_steps$  iterations of this process, the hidden state is mapped by a linear layer to a single scalar value (the Close price we want to predict). Mini-batch gradient descent is used to train such an RNN.

#### 1.2.4 LSTM

LSTM stands for long short-term memory. The LSTM tries to improve on the RNN by using a cell state in addition to the hidden state, which is also a vector of  $hidden\_size$ . While the hidden state represents a short term memory because the gradients deteriorate over time, the cell state is connected via constant gradient flow and can store information for much longer.

$$f_t = \sigma_g(W_f x_t + U_f h_{t-1} + b_f)$$

$$i_t = \sigma_g(W_i x_t + U_i h_{t-1} + b_i)$$

$$o_t = \sigma_g(W_o x_t + U_o h_{t-1} + b_o)$$

$$\tilde{c}_t = \sigma_c(W_c x_t + U_c h_{t-1} + b_c)$$

$$c_t = f_t \odot c_{t-1} + i_t \odot \tilde{c}_t$$

$$h_t = o_t \odot \sigma_h(c_t)$$

In the equation above  $x_t$  stands for the input feature vector at a given time  $t$  starting from  $t = 1$  up to  $t = 30$  in our case.  $h_t$  represents the hidden state, while  $c_t$  is the cell state. In each iteration of the forward pass the formulas above are applied, to generate a new  $h_t$  and  $c_t$  until finally  $h_{30}$  is mapped by a linear layer to a single scalar: the predicted Close value of the next day.

In theory, one could have multiple layers in each iteration, we will just use 1-layer LSTMs and GRUs in this project.

### 1.2.5 GRU

GRU stands for gated recurrent unit. In comparison to the LSTM it lacks the input/update gate and the cell state but has been found to produce similar results at a lower computational cost in many applications. In each forward pass the following formulas are applied to get from the input  $x_t$  and the hidden state  $h_t$  to the next hidden state:

$$\begin{aligned} z_t &= \sigma_g(W_z x_t + U_z h_{t-1} + b_z) \\ r_t &= \sigma_g(W_r x_t + U_r h_{t-1} + b_r) \\ \hat{h}_t &= \phi_h(W_h x_t + U_h(r_t \odot h_{t-1}) + b_h) \\ h_t &= z_t \odot \hat{h}_t + (1 - z_t) \odot h_{t-1} \end{aligned}$$

## 2 Results

First we will provide descriptive statistics and graphics of the dataset before moving on to compare the regression methods with each other.

### 2.1 Descriptive Statistics

The price of the Apple Inc. stock fluctuated between \$67.9 and \$134.1 in the given time frame. A volume between \$800.000 and \$15.000.000 was traded every active day. The 3 charts in Figure 2.1 display the price development, traded volume and KD-line in the entire timeframe of almost 2 years.

In the topmost candle-chart the blue line represents the 10-Day moving average, while the orange line is the 30-Day moving average.

In the bottom chart shows the K (blue line) and D statistics (orange line), both common technical indicators.

Before applying any prediction model the features and labels (Close price of the next day) were min-max normalized to a range between 0 and 1. The weekdays were computed from the date and one-hot-encoded.



Figure 1: Sequence diagram

## 2.2 Last Value Model

As a baseline for all further learning we need to know how well we can predict the next Close price with a very simple model: What if we just take the last day's Close price as the prediction? Such a model yields results on the 3 data set slices like displayed in Table 1 and gives price predictions like shown in Figure 2.



Figure 2: Predictions of the Last Value Model

Table 1: MSE of the Last Value Model on train, valaidation and test set

	Train set	Validation set	Test set
MSE	0.00102	0.00035	0.00028

It will be compared to other models below.

## 2.3 Moving Average Model

Pytorch was used to train a neural network with 1 lineaer layer that models the next Close Price as a Moving Average of the last 30 days with variable weights. The weights were initialized to resemble the Last Value model at first, s.t.  $w = [0, 0, 0, \dots, 1]$  and  $Card(w) = 30$ .



### **3 Discussion**