

人民币汇率波动的研究与建模

姜博恩^{*} 纪传宇[†] 凌源飞[‡]

2023 年 6 月

摘要

本研究旨在对 2005 年 10 月 1 日至 2020 年 9 月 30 日期间以英镑为单位的人民币外汇汇率进行时间序列分析. 通过对人民币汇率的变动进行研究, 我们可以深入了解该汇率的波动性和相关特征.

首先, 我们对人民币汇率时间序列进行了描述性统计分析, 以了解其基本统计特征和变动情况. 然后通过对是否有季节因素进行 ANOVA 检验后, 应用 Holt 两指数平滑模型, 对汇率的趋势和季节性进行了分解, 以获取更准确的时间序列信息.

接下来, 我们进行了平稳性检验和相关性检验, 以验证数据的平稳性和寻找可能存在的相关关系. 然后, 通过对取对数差分后的汇率序列进行分析, 建立 ARMA 模型, 对其之后残差序列进行 ARCH 效应检验, 并通过建立 GARCH 模型消除其条件异方差性, 最后通过 AIC 等准则对不同非白噪声建模模型的选择.

通过比较不同模型的拟合效果, 我们发现 ARMA(2,2)-GARCH(1,1) 模型在拟合人民币汇率时间序列方面表现出色. 该模型能够更准确地捕捉到汇率变动中的长期趋势和短期波动, 同时有效消除了残差序列中的异方差现象.

最后, 利用建立的 ARMA(2,2)-GARCH(1,1) 模型, 我们对未来一定期间内人民币汇率进行了预测. 通过计算置信区间和条件标准差的变化趋势, 我们能够对未来人民币汇率走势的预测, 虽然效果有限, 但也能金融机构和投资者提供有关风险管理和决策制定的一定的参考.

本研究的结果为我们对以英镑为单位的人民币外汇汇率的波动性和未来走势有了更深入的了解, 并为金融市场参与者提供了重要的信息和决策支持. 我们的研究方法和模型对于其他类似时间序列数据的分析也具有一定的借鉴意义.

关键词 汇率数据, Holt 两指数, ARMA, ARMA-GARCH

1 绪论

1.1 问题背景

汇率作为一种关键的金融时间序列, 受到国内外学者广泛的关注. 汇率受到多种因素的影响, 包括国际收支, 利率, 通货膨胀, 市场预期, 不同国家的宏观经济政策以及政府干预等. 某币的汇率的波动程度反映了某国国际金融关系的紧密程度. 因此, 研究汇率序列对于引导经济主体制定金融政策和投资决策具有重要意义. 在研究汇率序列时, 必须考虑各种因素的综合影响, 并采用科学合理的方法和模型进行分析和预测. 只有这样, 我们才能更好地了解 and 掌握汇率波动的规律和趋势, 做出更准确的决策, 实现经济的增长和稳定.

^{*}学号: 20338109. 主要完成了 ARMA-GARCH 模型部分的撰写, 构建论文结构等工作.

[†]学号: 19331053. 主要完成了确定性分析、描述性统计的撰写, 摘要和问题背景构建等工作

[‡]学号: 19331088. 主要完成了原始序列分析、差分序列的 ARMA 模型构建与分析等工作

我们利用以英镑为单位的外汇汇率的数据为所研究的数据集, 包括 2005 年 10 月 1 日至 2020 年 9 月 30 日期间的人民币汇率的日常观察结果, 共计 $n = 3790$ 个观察值, 希望对该汇率进行正确的模型选择以及合理的预测.

首先绘制未经过预处理的数据的时序图: 从图中可以看出不仅该序列不平稳, 而且不平稳的幅度很大, 这是因

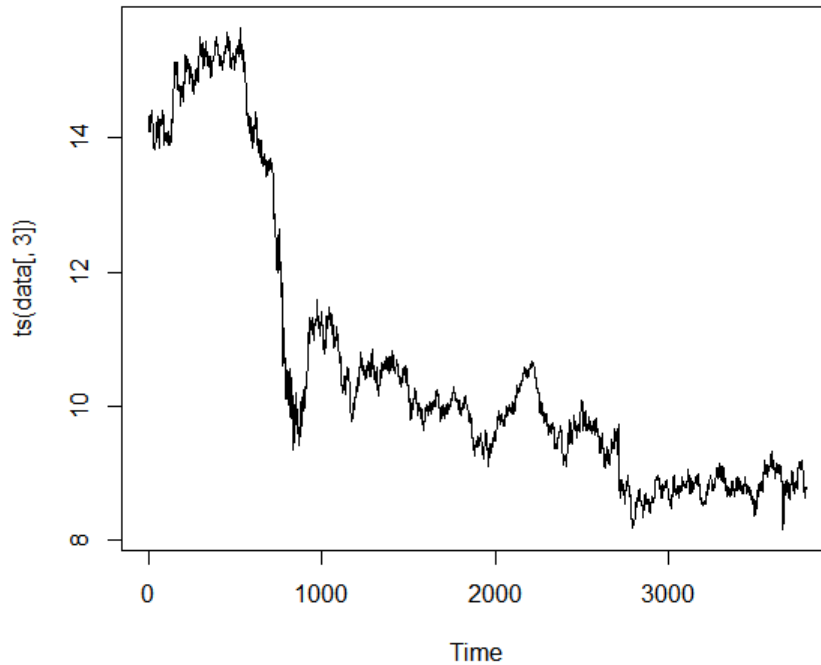


图 1: 未经过预处理的时序图

为经济数据的取值范围通常非常广, 跨度较大, 而且变化幅度也可能很大. 这样的数据分布会导致经济时间序列的方差和均值随时间变化而发生显著变化, 因此对于不同时间点的数据来说, 它们之间的差异可能被过度强调.

为了解决这个问题, 取对数可以将数据转换成一个更可比较的尺度. 因为对数函数是一个单调递增函数, 所以取对数可以压缩原始数据的幅度范围, 使得变化幅度较小的数据也可以得到足够的重视. 此外, 对数转换还可以使得数据更加接近正态分布, 这样可以更容易地进行统计分析和建模. 因此, 在经济时间序列中, 取对数是一种常见的数据预处理方法, 可以提高数据的可比性和可解释性, 有助于更准确地分析和预测经济现象.

由于汇率序列通常呈现出比较大的波动范围和不稳定的方差, 且随着汇率的变动幅度增大, 其对实际经济变动的影响程度并不是线性的.

因此, 我们希望通过取对数从而将大幅波动的汇率序列转化为相对平稳的序列, 降低序列的波动幅度和方差, 同时也能够保持序列的趋势和变化方向不变. 此外, 还希望通过取对数使得序列的变化更加对数正态分布, 更符合许多统计模型的假设条件.

对取对数后的时间序列数据绘制时序图以及进行平稳性检验:

Listing 1: 读取数据并进行平稳性检验

```
data <- BoE_exchange_rates
dat<-log(data[,3])
CHN<-ts(dat)
plot(CHN)
adf.test(dat)
```

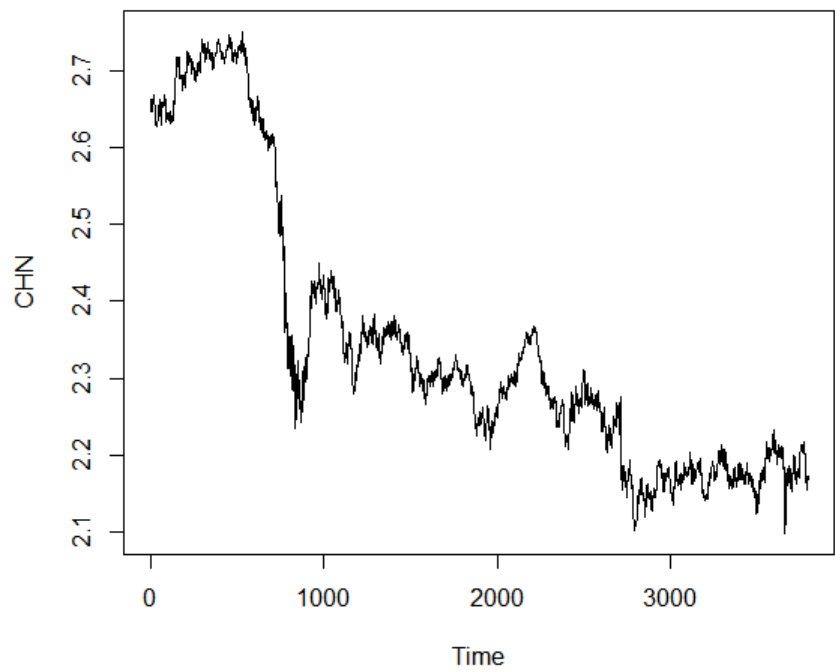


图 2: 经过取对数处理的时序图

Listing 2: 输出平稳性检验的结果

```
Augmented Dickey-Fuller Test
alternative: stationary

Type 1: no drift no trend
      lag  ADF p.value
[1,]  0 -1.37  0.188
[2,]  1 -1.36  0.193
[3,]  2 -1.38  0.187
[4,]  3 -1.41  0.176
[5,]  4 -1.42  0.173
[6,]  5 -1.46  0.159
[7,]  6 -1.39  0.182
[8,]  7 -1.39  0.182
[9,]  8 -1.43  0.169

Type 2: with drift no trend
      lag  ADF p.value
[1,]  0 -1.51  0.523
[2,]  1 -1.54  0.512
[3,]  2 -1.53  0.515
[4,]  3 -1.56  0.505
[5,]  4 -1.51  0.523
[6,]  5 -1.46  0.537
[7,]  6 -1.47  0.534
[8,]  7 -1.49  0.527
```

```

[9,] 8 -1.47 0.536
Type 3: with drift and trend
      lag  ADF p.value
[1,] 0 -2.02 0.566
[2,] 1 -2.08 0.544
[3,] 2 -2.04 0.558
[4,] 3 -2.03 0.562
[5,] 4 -1.97 0.589
[6,] 5 -1.87 0.630
[7,] 6 -1.96 0.593
[8,] 7 -1.98 0.584
[9,] 8 -1.91 0.616
----
Note: in fact, p.value = 0.01 means p.value <= 0.01

```

经过检验发现无论在哪种模型下都不能通过平稳性检验,所以我们尝试对取对数后的数据进行一阶差分,一阶差分后我们进行平稳性检验:

Listing 3: 对数据进行差分,并生成时间序列

```

dat<-diff(dat)
CHN<-ts(dat)
plot(CHN)
adf.test(dat)

```

Listing 4: 对差分后的数据进行平稳性检验

```

Augmented Dickey-Fuller Test
alternative: stationary

Type 1: no drift no trend
      lag  ADF p.value
[1,] 0 -60.3 0.01
[2,] 1 -43.7 0.01
[3,] 2 -35.9 0.01
[4,] 3 -31.7 0.01
[5,] 4 -29.4 0.01
[6,] 5 -25.4 0.01
[7,] 6 -23.3 0.01
[8,] 7 -22.5 0.01
[9,] 8 -21.7 0.01
Type 2: with drift no trend
      lag  ADF p.value
[1,] 0 -60.3 0.01
[2,] 1 -43.7 0.01
[3,] 2 -35.9 0.01
[4,] 3 -31.8 0.01
[5,] 4 -29.4 0.01
[6,] 5 -25.5 0.01
[7,] 6 -23.3 0.01
[8,] 7 -22.5 0.01
[9,] 8 -21.7 0.01
Type 3: with drift and trend
      lag  ADF p.value
[1,] 0 -60.3 0.01
[2,] 1 -43.7 0.01

```

```
[3,]  2 -35.9    0.01
[4,]  3 -31.8    0.01
[5,]  4 -29.4    0.01
[6,]  5 -25.5    0.01
[7,]  6 -23.3    0.01
[8,]  7 -22.5    0.01
[9,]  8 -21.7    0.01
----
Note: in fact, p.value = 0.01 means p.value <= 0.01
```

在检验中我们可以看出不管在哪个模型下,数据都通过了 ADF 检验,即进行了取对数后差分的数据满足了平稳性的要求。

满足了平稳性要求后我们进行数据是否是白噪声的检验:

H_0 : 延迟期数小于或等于 m 期的序列值之间不相关

$\rho_1 = \rho_2 = \dots = \rho_m = 0, \forall m \geq 1$

H_1 : 延迟期数小于或等于 m 期的序列值之间有相关性

至少存在某个 $\rho_k \neq 0, \forall m \geq 1, k \leq m$

我们利用 LB 统计量进行检验:

Listing 5: 进行纯随机检验

```
#利用LB统计量进行检验
for(i in c(6,12,18)){
  print(Box.test(dat,lag=i))
}
```

Listing 6: 纯随机性检验的结果

```
Box-Pierce test

data:  dat
X-squared = 18.072, df = 6, p-value = 0.006054

Box-Pierce test

data:  dat
X-squared = 37.963, df = 12, p-value = 0.0001557

Box-Pierce test

data:  dat
X-squared = 52.81, df = 18, p-value = 2.807e-05
```

从结果可以看出,所有的 p 值都大于 0.05,所以无法在 95% 的置信度下拒绝原假设,即数据是非白噪声序列,所以我们可以对该数据进行时间序列建模。

1.2 课题介绍

原始数据是人民币对英镑的汇率的时间序列数据,本文灵感主要来源于 [EV22] 文章中的应用,数据来源可以参考该论文的支撑材料¹。对于上述预处理后的数据,仅仅差分并不是我们要研究的问题,由于我们考察的是收益率的

¹https://github.com/sebastian-engelke/extremal_tree_learning/tree/master/application/data

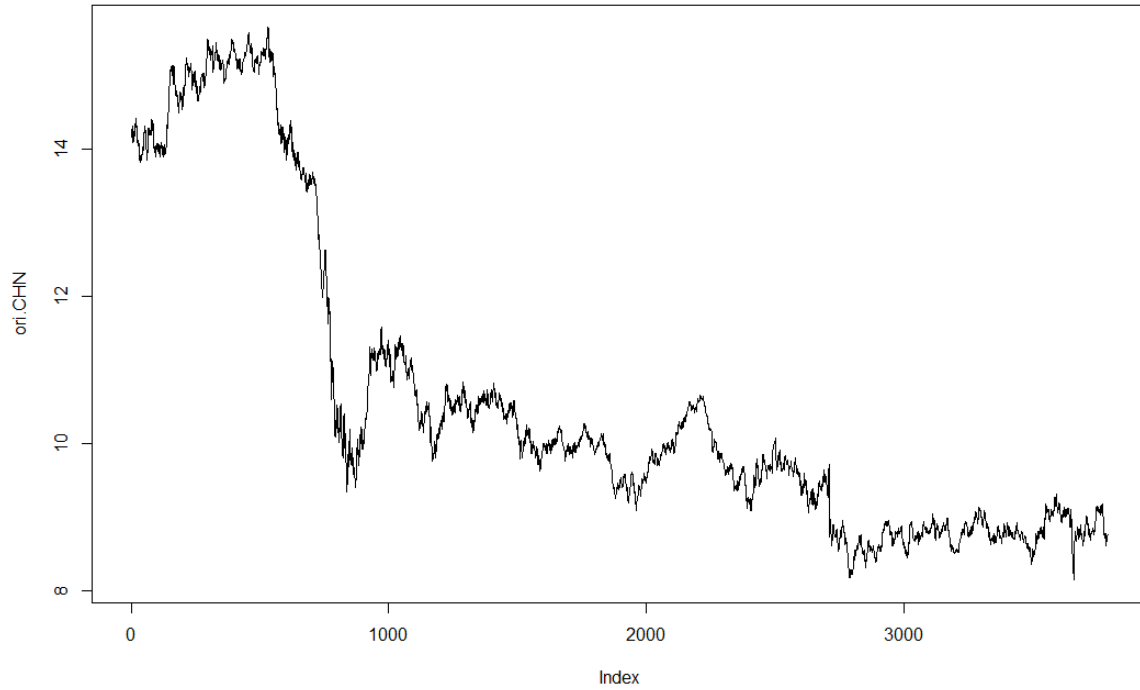


图 3: 人民币对英镑汇率 (2005 年 10 月 1 日至 2020 年 9 月 30 日)

问题, 更加合理的数据预处理方法是计算每天的对数收益率 (daily log-returns). 这是有实际理论依据的, 事实上, 设 x_t 是数据的实际值, 则 $r_t = \frac{x_t - x_{t-1}}{x_{t-1}}$ 就是收益率, 则我们有 $1 + r_t = \frac{x_t}{x_{t-1}}$, 并且根据 Taylor 展示,

$$\log(1 + r_t) = \log\left(\frac{x_t}{x_{t-1}}\right) = \log(x_t) - \log(x_{t-1}) \approx r_t.$$

因此为研究该收益率数据, 对数据进行对数差分运算是更合理的选择, 即

Listing 7: 转换为对数收益率数据

```
nlr <- apply(data, 2, FUN = function(x) diff(log(x))) #ndifflog
```

从图4中可以明显看出该序列似乎是稳定的, 平均收益率接近零, 但是波动 (变化) 很大的周期往往聚集在一起. 这些类型的金融数据分析的一个问题就是要预测未来收益率的波动性. 在这种情况下, ARCH 和 GARCH 模型 [Eng82; Bol86] 就是用来处理这些问题的.

1.3 章节安排

首先我们对原始时间序列进行非平稳时间序列的分析, 最后拟合了一个不带有季节项的因素分解模型, 并使用 Holt 两指数平滑模型, 对汇率的趋势进行了提取, 获得了更准确的时间序列信息. 在已经对原始序列进行了平稳性和纯随机性的检验后, 在本文的剩余部分, 我们首先对原始非平稳序列进行了确定性分析, 以确定该时间序列的一些特征. 然后, 我们尝试建立了普通的 ARMA 模型进行进一步的分析. 然而, 我们发现对于该数据集而言, ARMA 模型无法很好地总结出该时间序列的特征, 因此我们转而探索其他模型.

由于该时间序列表现出一定时期内波动性或可变性的变化, 我们决定采用条件异方差时间序列建模的方法. 最后, 我们确定了该数据的时间序列建模为 $ARMA(2, 2) - GARCH(1, 1)$ 模型.

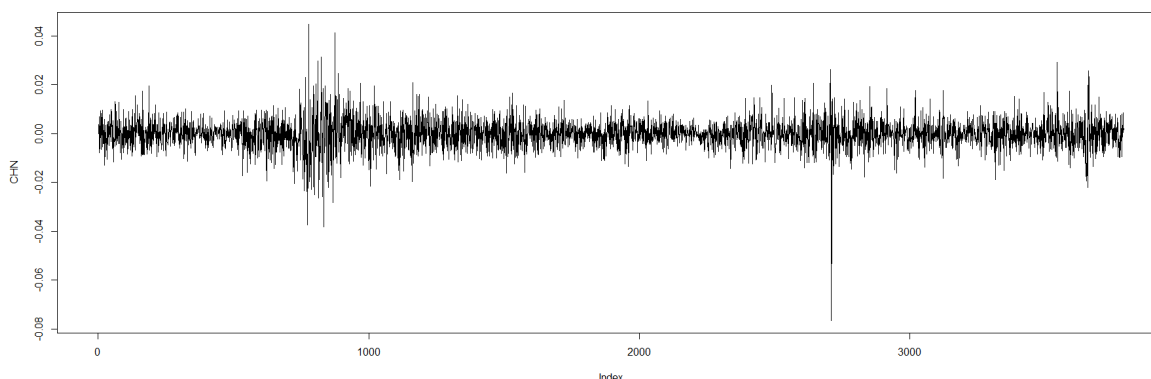


图 4: 人民币汇率的日收益率 (2005 年 10 月 1 日至 2020 年 9 月 30 日)

2 原始序列分析

2.1 数据变换

为了降低数据的噪声以及波动性, 同时更好地捕捉汇率地长期趋势和周期性, 我们将汇率的日数据平均成月数据, 时间序列图如下:

Listing 8: 将原始时间序列进行数据变换

```
library(lubridate)
library(dplyr)

CHNO <- data[, "CHN"]
CHNO_df <- data.frame(date = names(CHNO), value = CHNO)
CHNO_df$year <- year(as.Date(CHNO_df$date))
CHNO_df$month <- month(as.Date(CHNO_df$date))
CHNO_df$day <- day(as.Date(CHNO_df$date))
CHNO_df$year_month <- floor_date(as.Date(CHNO_df$date), "month")

monthly_mean <- CHNO_df %>%
  group_by(month) %>%
  summarise(mean_value = mean(value))
plot(monthly_mean, type="l")

year_monthly_mean <- CHNO_df %>%
  group_by(year_month) %>%
  summarise(mean_value = mean(value))

plot(year_monthly_mean, type="l")
```

2.2 非平稳时间序列分析

通过观察时序图, 我们可以注意到在第三年的某个时间段内, 汇率出现了突然下降的情况. 这种突发的变化可能是由于国家或政策的影响所引起的. 基于这个观察, 我们决定将重点放在原始时间序列的后半部分, 即对该时间段进行数据特征的提取与描述.

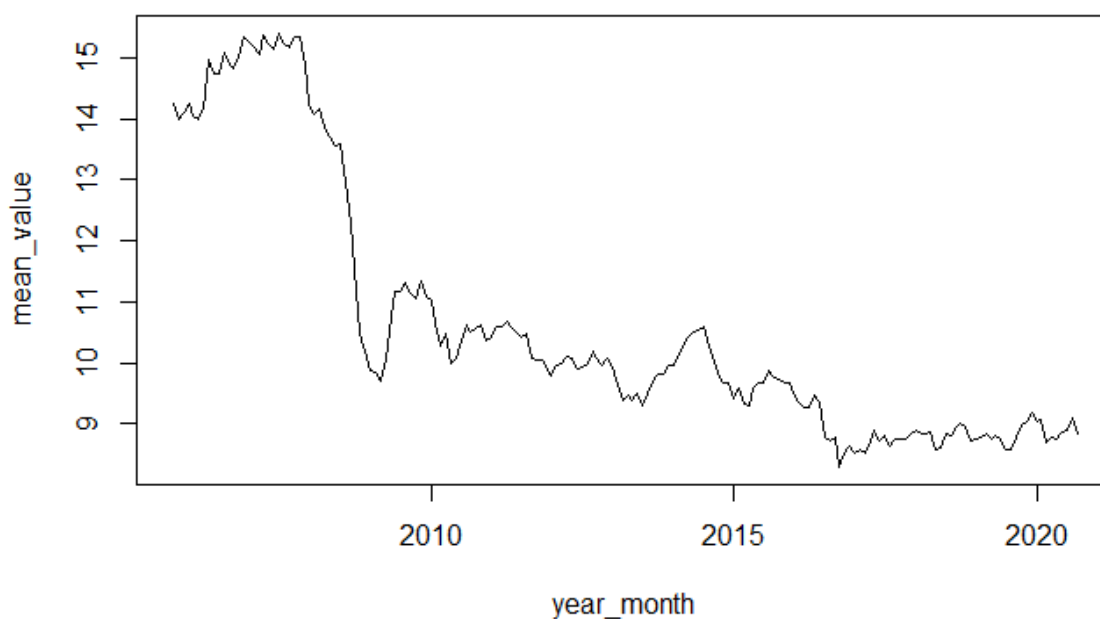


图 5: 原始人民币汇率的月收益率时间序列图

Listing 9: 提取目标数据

```
sea1<-ts(year_monthly_mean[40:171,2],frequency = 12,start = (2009))
plot(sea1, ylab='exchange rate')
```

从图中可以看出汇率随月度时间的变化有一定的线性趋势,但是是否有季节趋势无法通过时序图看出.

做关于十二个月的 ANOVA 检验:

$$H_0: \beta_1 = \beta_2 = \dots \beta_{12} \text{ v.s. } H_1: \beta_1 \dots \beta_{12} \text{ 不全相等.}$$

```
CHNO_df$month <- factor(CHNO_df$month)
summary(aov(value ~ month, data = CHNO_df))
```

Df	Sum Sq	Mean Sq	F value	Pr(>F)
month	11	36	3.271	0.732
Residuals	3779	16891	4.470	0.709

从检验结果看出 P 值 = 0.709, 即在置信度为 95% 的情况下不能拒绝原假设, 所以该时间序列没有季节效应.

由于 Holt 两参数指数平滑适用于对含有线性趋势的序列进行拟合与预测, 所以我们利用该方法对汇率的时间序列进行拟合.

Holt 两参数指数平滑的模型结构为:

$$x_t = a_0 + bt + \varepsilon_t$$

式中, a_0 为截距; b 为斜率; ε_t 为随机波动, $\varepsilon_t \sim N(0, \sigma^2)$.



图 6: 人民币汇率的月收益率时间序列图

该模型结构还可以等价表达为如下递推公式:

$$\begin{aligned} x_t &= a_0 + b(t-1) + b + \varepsilon_t \\ &= (x_{t-1} - \varepsilon_{t-1}) + b + \varepsilon_t \end{aligned}$$

不妨记

$$\begin{aligned} a(t-1) &= x_{t-1} - \varepsilon_{t-1} \\ b(t) &= b + \varepsilon_t \end{aligned}$$

显然, $a(t-1)$ 是序列在 $t-1$ 时刻截距的无偏估计值, $b(t)$ 是序列在 t 时刻斜率的无偏估计值. 于是模型可表示为:

$$x_t = a(t-1) + b(t)$$

Holt 两参数指数平滑就是分别使用简单指数平滑的方法, 结合序列的最新观察值, 不断修匀截距项 $\hat{a}(t)$ 和斜率项 $\hat{b}(t)$, 递推公式如下:

$$\begin{aligned} \hat{a}(t) &= \alpha x_t + (1 - \alpha)[\hat{a}(t-1) + \hat{b}(t-1)] \\ \hat{b}(t) &= \beta[\hat{a}(t) - \hat{a}(t-1)] + (1 - \beta)\hat{b}(t-1) \end{aligned}$$

式中, x_t 为序列在 t 时刻得到的最新观察值; α, β 均为平滑系数, 满足 $0 < \alpha, \beta < 1$.

Listing 10: 拟合两指数 Holt 模型

```
fit<-HoltWinters(sea1,gamma=F)
fit
```

Listing 11: 两指数 Holt 模型的输出结果

```
Holt-Winters exponential smoothing with trend and without seasonal component.
```

```
Call:
```

```
HoltWinters(x = sea1, gamma = F)
```

```
Smoothing parameters:
```

```
alpha: 1
```

```
beta : 0.003853306
```

```
gamma: FALSE
```

```
Coefficients:
```

```
[,1]
```

```
a 9.19312000
```

```
b -0.01917329
```

得出的模型为:

$$x_t = 9.19312 - 0.01917329t + \varepsilon_t$$

拟合与预测的效果分别为:

Listing 12: 两指数 Holt 模型的预测结果

Point Forecast	Lo 80	Hi 80	Lo 95	Hi 95
Jan 2020	9.173947	8.933614	9.414279	8.806390
Feb 2020	9.154773	8.814236	9.495311	8.633967
Mar 2020	9.135600	8.717726	9.553475	8.496516
Apr 2020	9.116427	8.632979	9.599875	8.377057
May 2020	9.097254	8.555704	9.638803	8.269025
Jun 2020	9.078080	8.483704	9.672456	8.169060
Jul 2020	9.058907	8.415677	9.702137	8.075172
Aug 2020	9.039734	8.350776	9.728691	7.986064
Sep 2020	9.020560	8.288414	9.752706	7.900840
Oct 2020	9.001387	8.228164	9.774610	7.818845
Nov 2020	8.982214	8.169705	9.794722	7.739589
Dec 2020	8.963041	8.112787	9.813294	7.662691

Listing 13: 数据整理

```
true_dat<-year_monthly_mean[172:177,2]
com<-cbind(fore$mean[1:6],true_dat,(fore$mean[1:6]-true_dat)^2)
colnames(com)<-c("预测数据","真实数据","每个月份的残差平方")
com
```

由于汇率是受多种因素影响的复杂变量, 其未来走势受到市场需求和供应的影响. 随着时间的推移, 市场需求和供应的因素可能发生变化, 导致长期预测的准确性下降, 所以我们取前六期预测的值与真实值进行对比.

Listing 14: 输出结果

	预测数据	真实数据	每个月份的残差平方
1	9.173947	9.047168	0.016072795
2	9.154773	9.064815	0.008092517
3	9.135600	8.677100	0.210222365
4	9.116427	8.776930	0.115258101
5	9.097254	8.735595	0.130797092
6	9.078080	8.870623	0.043038624

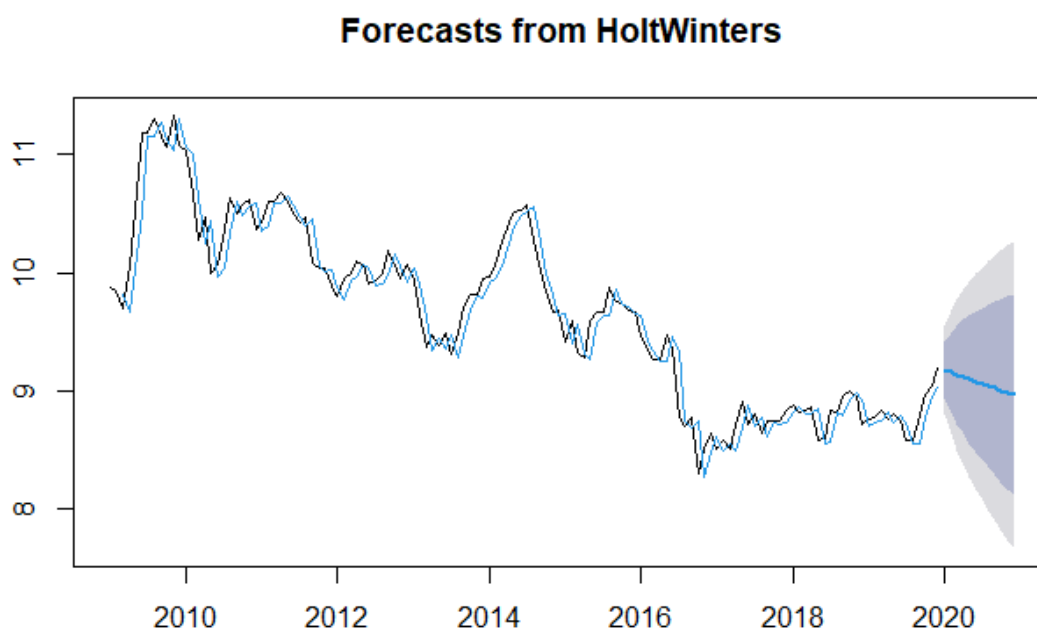


图 7: 人民币汇率 Holt 两参数指数平滑预测效果图

从结果可以看出前两期的预测较为准确, 但是后面期数的预测的偏移越来越大, 由于汇率数据是实时更新的, 所以可以在新的一天数据更新后再进行一次建模去预测下一天的汇率, 从而提高预测的准确度.

对经过 Holt 两参数指数平滑预测的残差进行平稳性和随机性检验:

Listing 15: 对残差进行平稳性和随机性检验

```
plot(fore$residuals)
re<-ts(fore$residuals)
adf.test(re)
for(i in c(6,12,18,24)){
  print(Box.test(re,lag=i))
}
```

Listing 16: 平稳性和随机性检验结果

```
Augmented Dickey-Fuller Test
alternative: stationary

Type 1: no drift no trend
lag    ADF p.value
[1,]   0 -10.01    0.01
[2,]   1  -7.19    0.01
[3,]   2  -5.60    0.01
[4,]   3  -6.48    0.01
[5,]   4  -5.73    0.01
Type 2: with drift no trend
```

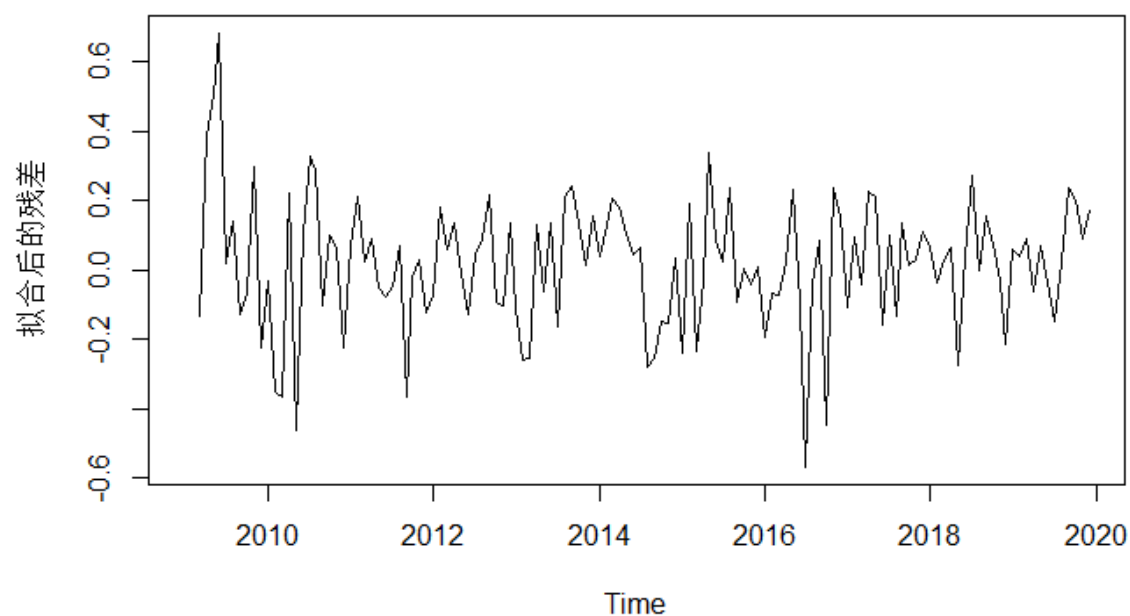


图 8: 人民币汇率 Holt 两参数指数平滑拟合后的残差图

```
lag      ADF p.value
[1,]    0 -10.06    0.01
[2,]    1  -7.22    0.01
[3,]    2  -5.61    0.01
[4,]    3  -6.46    0.01
[5,]    4  -5.73    0.01
Type 3: with drift and trend
lag      ADF p.value
[1,]    0 -10.03    0.01
[2,]    1  -7.18    0.01
[3,]    2  -5.55    0.01
[4,]    3  -6.38    0.01
[5,]    4  -5.69    0.01
----
Note: in fact, p.value = 0.01 means p.value <= 0.01

Box-Pierce test

data:  re
X-squared = 5.6962, df = 6, p-value = 0.4581

Box-Pierce test

data:  re
X-squared = 16.815, df = 12, p-value = 0.1567
```

```
Box-Pierce test

data: re
X-squared = 20.387, df = 18, p-value = 0.3115

Box-Pierce test

data: re
X-squared = 26.826, df = 24, p-value = 0.3126
```

根据检验结果和残差时序图的观察,我们可以得出结论:拟合后的残差呈现出平稳的白噪声特征,这意味着利用 Holt 两参数指数平滑进行拟合已经成功地提取了数据中的信息. 因此,该时序图展示了拟合模型的良好符合度. 即 Holt 两参数指数平滑这一模型已经捕捉到了数据中的大部分可解释变动,可以很好地适应了数据的特征,并且可以用于分析和预测.

3 ARMA 模型的拟合

3.1 模型识别

我们首先尝试建立普通的 ARMA 模型. 为了完成定阶,我们先绘制并分析了序列的自相关、偏自相关系数图,然后使 ARMA(p,q) 模型中的 p,q 遍历 0-5 并根据 AIC 准则选取了最优的模型. 我们首先绘制自相关系数、偏自相关系数图:

Listing 17: 绘制自相关系数, 偏自相关系数图

```
par(mfrow = c(1, 2))
acf(CHN)
pacf(CHN)
```

模型	AR(p)	MA(q)	ARMA(p, q)
PACF 图	第p 个条后截尾	拖尾	前p 个条无规律, 其后拖尾
ACF 图	拖尾	第q 个条后截尾	前q 个条无规律, 其后拖尾

从本例的自相关图可以看出,延迟大于等于 1 阶自相关系数几乎全部在 2 倍标准差范围内波动,且自相关系数衰减没有显著的规律性. 偏自相关系数在 2 倍标准差范围内外不断波动,没有显著的规律性. 随后,我们在 0-5 范围内遍历 ARMA(p,q) 模型中的 p,q 并根据 AIC 准则选取最佳模型:

Listing 18: 遍历所有可能的情况

```
aic_matirx<-matrix(0,nrow = 6,ncol = 6)
for(i in 1:6){
  for(j in 1:6){
    xfit=arima(CHN,order=c((i-1),0,(j-1)))
    aic_matirx[i,j]<-AIC(xfit)
  }
}
aic_matirx
```

Listing 19: 输出遍历的结果

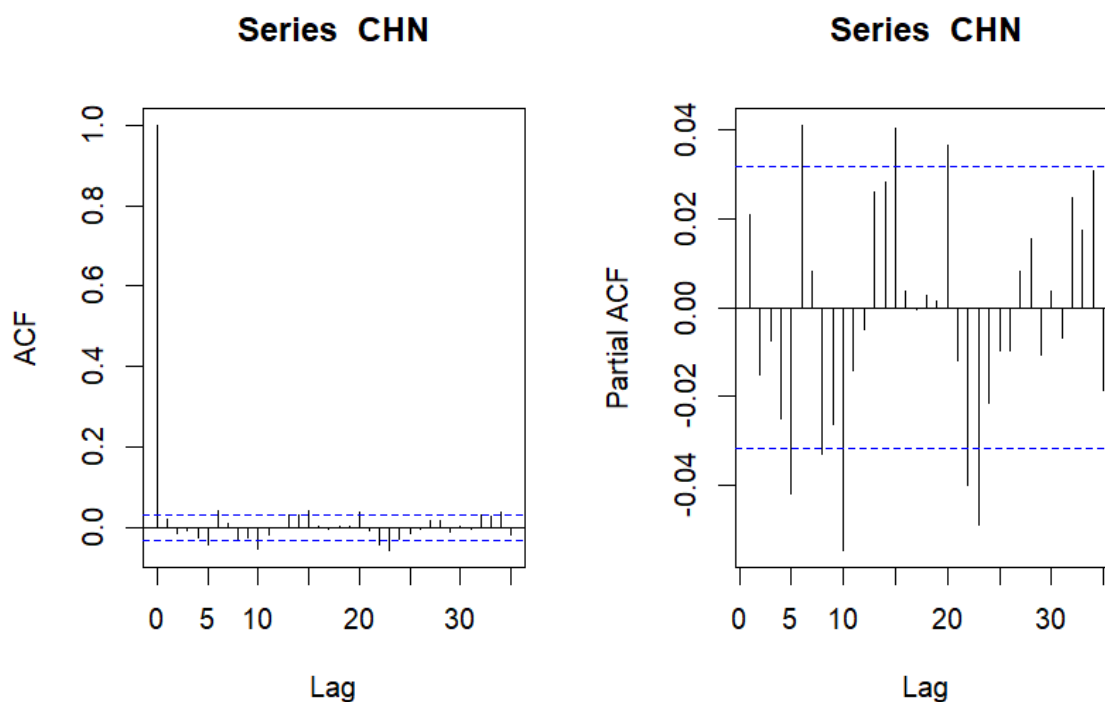


图 9: 人民币汇率的日收益率的自相关和偏自相关图

	[,1]	[,2]	[,3]	[,4]	[,5]	[,6]
[1,]	-27799.32	-27799.00	-27797.81	-27796.06	-27796.07	-27802.89
[2,]	-27798.96	-27796.98	-27795.82	-27794.08	-27796.70	-27803.32
[3,]	-27797.80	-27795.79	-27797.36	-27795.99	-27823.70	-27804.53
[4,]	-27796.02	-27794.00	-27795.99	-27823.41	-27800.30	-27802.63
[5,]	-27796.42	-27796.22	-27801.59	-27821.46	-27827.33	-27819.16
[6,]	-27801.06	-27802.08	-27803.91	-27802.23	-27803.81	-27825.31

Listing 20: 手动选出最好的结果

```
best<-which(aic_matirx==aic_matirx[which.min(aic_matirx)],arr.ind=T)
print(best-1)
```

Listing 21: AIC 准则下手动的模型定阶

```
row col
[1,] 4 4
```

根据 AIC 准则, 我们最终将模型定阶为 ARMA(4,4).

3.2 参数估计

在完成模型定阶以后, 我们利用条件最小二乘法与极大似然估计混合方法进行参数估计, 相关命令和输出结果如下:

Listing 22: 拟合 ARMA(4,4) 模型

```
fit <- arima(CHN, order = c(4, 0, 4))
fit
```

Listing 23: 拟合结果

```
Call:
arima(x = CHN, order = c(4, 0, 4))

Coefficients:
      ar1      ar2      ar3      ar4      ma1      ma2      ma3      ma4 intercept
    0.1834  -0.6749  -0.0073  -0.7790  -0.1628   0.6454   0.0257   0.7378    -1e-04
s.e.  0.2494   0.2257   0.1800   0.1117   0.2626   0.2263   0.1763   0.1217     1e-04

sigma^2 estimated as 3.771e-05:  log likelihood = 13923.67,  aic = -27827.33
```

根据输出结果, 我们确定该模型口径为:

$$x_t = -2.2778 \times 10^{-4} + 0.1834x_{t-1} - 0.6749x_{t-2} - 0.0073x_{t-3} - 0.7790x_{t-4} + \epsilon_t - 0.1628\epsilon_{t-1} + 0.6454\epsilon_{t-2} + 0.0257\epsilon_{t-3} + 0.7378\epsilon_{t-4}, \text{Var}(\epsilon_t) = 3.771 \times 10^{-5}.$$

3.3 模型检验

在确定了模型口径之后, 我们首先进行模型的显著性检验, 即残差序列的白噪声检验, 原假设和备择假设分别为:

$$H_0: \rho_1 = \rho_2 = \dots = \rho_m = 0, \forall m \geq 1$$

$$H_1: \text{至少存在某个 } \rho_k \neq 0, \forall m \geq 1, k \leq m$$

拟合模型显著性检验的相关命令和输出结果如下:

Listing 24: 模型显著性检验

```
ts.diag(fit)
```

考察残差序列的白噪声检验结果, 可以看出在 25 阶延迟下白噪声检验统计量的 P 值小于 0.05, 我们认为这个拟合模型的残差序列属于非白噪声序列, 即该拟合模型不成立. 综上, 我们认为 ARMA 模型不能很好地拟合该序列, 于是我们推测可能是由于该序列存在异方差性, 所以我们利用 Q 检验和 LM 检验对残差进行异方差检验.

Q 检验的构造思想是: 如果残差序列方差非齐, 且具有集群效应, 那么残差平方序列通常具有自相关性. 所以方差非齐检验可以转化为残差平方序列的自相关性检验.

Q 检验统计量实际上就是 $\{\epsilon_t^2\}$ 的 LB 统计量

$$Q(q) = n(n+2) \sum_{i=1}^q \frac{\rho_i^2}{n-i}$$

原假设成立时, Q 检验统计量近似服从自由度为 $q-1$ 的 χ^2 分布:

$$Q(q) \sim \chi^2(q-1)$$

LM 检验的构造思想是: 如果残差序列方差非齐, 且具有集群效应, 那么残差平方序列通常具有自相关性. 我们就可以尝试使用自回归模型拟合残差平方序列:

$$\epsilon_t^2 = w + \sum_{j=1}^q \lambda_j \epsilon_{t-j}^2 + e_t$$

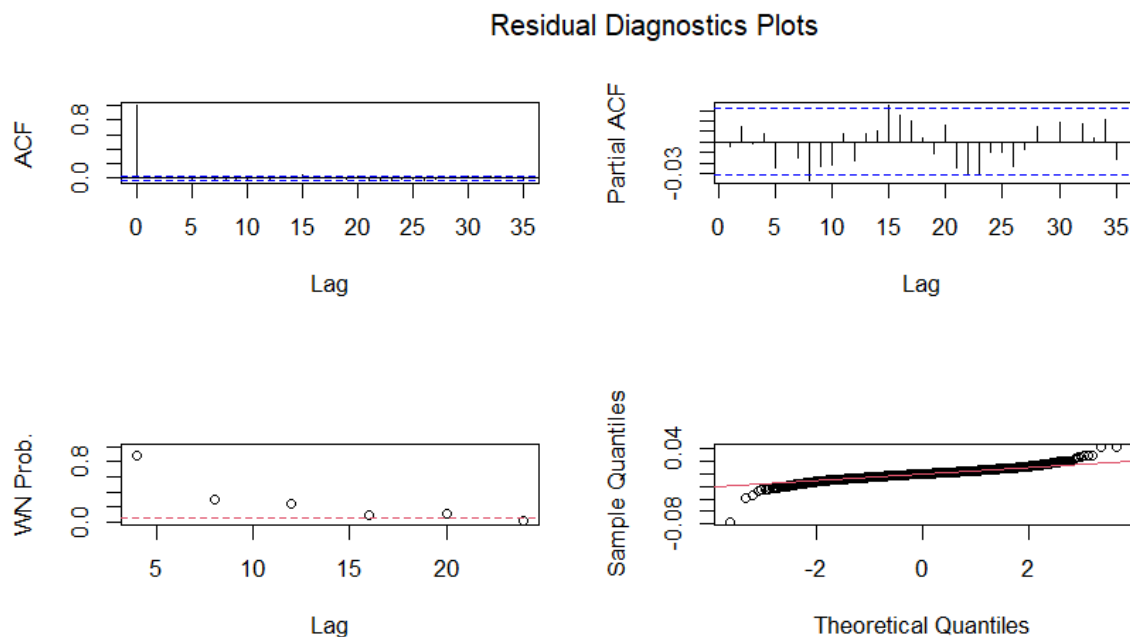


图 10: 人民币汇率的日收益率拟合模型显著性检验图

于是方差齐性检验就可以转化为这个方程是否显著成立的检验. 如果该方程显著成立 (至少存在一个参数 λ_j 显著非零), 那就意味着残差平方序列具有自相关性, 可以用该回归方程提取自相关信息. 反之, 则意味着残差平方序列不存在显著的自相关性, 不能拒绝方差齐性假定. 所以 LM 检验实际上就是残差平方序列 $\{\varepsilon_t^2\}$ 自回归方程的显著性检验.

考虑假设检验问题:

$$H_0: \text{残差平方序列纯随机} \quad \text{v.s.} \quad H_1: \text{残差平方序列具有自相关性}$$

该假设条件等价于:

$$H_0: \lambda_1 = \cdots = \lambda_q \quad \text{v.s.} \quad H_1: \lambda_1, \cdots, \lambda_q \text{ 不全为零}$$

Listing 25: 进行 LM 检验

```
for (i in 1:5) print(ArchTest(dat,lag=i))
```

Listing 26: 输出结果

```
ARCH LM-test; Null hypothesis: no ARCH effects

data:  dat
Chi-squared = 78.453, df = 1, p-value < 2.2e-16

ARCH LM-test; Null hypothesis: no ARCH effects

data:  dat
```



```

Chi-squared = 86.456, df = 2, p-value < 2.2e-16

      ARCH LM-test; Null hypothesis: no ARCH effects

data: dat
Chi-squared = 110.74, df = 3, p-value < 2.2e-16

      ARCH LM-test; Null hypothesis: no ARCH effects

data: dat
Chi-squared = 154.2, df = 4, p-value < 2.2e-16

      ARCH LM-test; Null hypothesis: no ARCH effects

data: dat
Chi-squared = 196.1, df = 5, p-value < 2.2e-16

> for (i in 1:6) {
+   print(Box.test(CHN^2, type = "L", lag = i))
+ }

      Box-Ljung test

data: CHN^2
X-squared = 78.534, df = 1, p-value < 2.2e-16

      Box-Ljung test

data: CHN^2
X-squared = 95.202, df = 2, p-value < 2.2e-16

      Box-Ljung test

data: CHN^2
X-squared = 129.42, df = 3, p-value < 2.2e-16

      Box-Ljung test

data: CHN^2
X-squared = 195.83, df = 4, p-value < 2.2e-16

      Box-Ljung test

data: CHN^2
X-squared = 273.08, df = 5, p-value < 2.2e-16

      Box-Ljung test

```

```
data: CHN^2
X-squared = 293.39, df = 6, p-value < 2.2e-16
```

LM 检验和 Q 检验都显示该序列显著方差非齐, 所以存在异方差, 于是我们转而使用 ARMA-GARCH 模型去拟合以及预测.

4 ARMA-GARCH 模型的拟合与预测

从图4中我们可以看出, 残差序列 (如果单纯将模型看成一个 ARMA(0, 0) 模型) 的波动在大部分时间段是平稳的, 但在某些时刻波动持续偏大, 在默写时段波动持续偏小, 呈现出集群效应 (volatility cluster).

对时间序列 x_t , 在 (中心化)ARMA(p,q) 模型 $\Phi(B)x_t = \Theta(B)\varepsilon_t$ 中, 当误差是白噪声 $WN(0, \sigma_\varepsilon^2)$ 时, 则由过去的信息都条件方差 σ_ε^2 是不依赖于过去的常数, 但是由于该数据集的波动性依赖于过去 (根据纯随机性检验), 因此使用条件异方差的模型.

接下来对选取的人民币对英镑汇率在 2005 年 10 月 1 日到 2020 年 9 月 30 日期间对数收益率的历史数据, 逐步建立 ARMA-GARCH 族模型, 并逐步优化, 求出最优拟合模型.

4.1 探索性数据分析

为了对感兴趣的数据进行初步的了解, 我们首先给出对数收益率的数字特征

Listing 27: 探索性数据分析

```
> mean(CHN)
[1] -0.0001266175
> sd(CHN)
[1] 0.006177818
> kurtosis(CHN)
[1] 12.53141
> mean(((CHN - mean(CHN)) / sd(CHN))^3)
[1] -0.431552
> shapiro.test(CHN)

      Shapiro-Wilk normality test

data: CHN
W = 0.94652, p-value < 2.2e-16

> Box.test(CHN, lag = 20, type = "L")

      Box-Ljung test

data: CHN
X-squared = 58.91, df = 20, p-value = 1.049e-05
```

即得到以下的结果 从表中的数据可以看出人民币对英镑的对数收益率分布有如下的特征: 均值为负, 但是绝对值接

统计量	均值	标准差	峰度	偏度	Shapiro-Wilk	Ljung-Box's Q(20)
实现	-0.000127	0.00618	12.5314	-0.431552	0.9465 ($p < 2.2 \times 10^{-6}$)	58.91 (1.049×10^{-5})

近零, 说明在长期持有人民币或英镑未必会带来收益, 峰度值 (12.5314) 大于 3, 说明分布是典型的“尖峰肥尾”分布. 偏度值为负说明数据是一个左偏分布, 为检验该序列是否服从正态分布, 我们采用 Shapiro-Wilk 检验, 发现 p 值很小, 即认为该序列不服从正态分布. 之前的 LB 统计量也表明该数据不是一个白噪声序列.

4.2 模型理论

自回归条件异方差模型 (ARCH 模型) 假设在历史数据已知的情况下, 零均值, 纯随机残差序列具有异方差性. 给波动率建模提供一个系统框架的洗一个模型是 [Eng82] 中提出的自回归条件异方差 (ARCH) 模型. 这个模型允许在一个 ARMA 模型中, 条件误差方差依赖于过去的平方和的形式, 而不是像现在有独立误差的 ARMA 模型中的常数的形式.

例. 最简单的 ARCH 模型, $ARCH(1)$, 可以进行如下的建模,

$$\begin{aligned}x_t &= \sigma_t \varepsilon_t \\ \sigma_t^2 &= \alpha_0 + \alpha_1 x_{t-1}^2\end{aligned}$$

其中 ε_t 是标准 Gauss 白噪声,

例. 一个具有 $ARCH(s)$ 模型误差的 $AR(p)$ 过程可以表示为

$$x_t = \sum_{i=1}^p \phi_i x_{t-i} + \phi_0 + v_t$$

其中 $\mathbb{E}[v_t \mid v_{t-1}, v_{t-2}, \dots] = 0$, 且 $h_t = \text{Var}[v_t \mid v_{t-1}, v_{t-2}, \dots] = \mathbb{E}[v_t^2 \mid v_{t-1}, v_{t-2}, \dots] = w_0 + \sum_{i=1}^s w_i v_{t-i}^2$ 其中 $w_0 > 0, w_i \geq 0, i = 1, \dots, s$.

在由 [Bol86] 介绍的广义 ARCH 模型 (GARCH) 中, 则使用了如下的模型形式,

$$\begin{cases} x_t = f(t, x_{t-1}, x_{t-2}, \dots) + \varepsilon_t \\ \varepsilon_t = \sqrt{h_t} e_t \\ h_t = \omega + \sum_{i=1}^p \eta_i h_{t-i} + \sum_{j=1}^q \lambda_j \varepsilon_{t-j}^2 \end{cases}$$

其中, $f(t, x_{t-1}, x_{t-2}, \dots)$ 为 $\{x_t\}$ 的确定性信息的拟合模型, $e_t \stackrel{i.i.d.}{\sim} N(0, \sigma^2)$. 这个模型被记为 GARCH(p, q). 于是

$$\text{Var}[\varepsilon_t \mid \varepsilon_{t-1}, \varepsilon_{t-2}, \dots] = \mathbb{E}[h_t e_t^2 \mid \varepsilon_{t-1}, \varepsilon_{t-2}, \dots] = h_t \mathbb{E}[e_t^2] = h_t \sigma^2$$

4.3 模型识别

由于先前检验中 LM 检验统计量充分的大, 因此我们在这里选用 GARCH(1, 1) 模型, 事实上, 一般场合下 GARCH(1, 1) 可以很好的拟合异方差的问题 [殷金 10]. 因此这里重点考虑 ARMA 部分的参数估计, 我们分别考虑 ARMA(p, q) 中参数均从 1 到 3 变化, 有如下的结果

Listing 28: 遍历可能的结果进行模型识别

```
pb <- progress_bar$new(total = 9)
ans <- matrix(0, nrow = 3, ncol = 3)
AIC <- matrix(0, nrow = 3, ncol = 3)
BIC <- matrix(0, nrow = 3, ncol = 3)
lik <- matrix(0, nrow = 3, ncol = 3)
for (p in 1:3) {
  for (q in 1:3) {
    spec <- ugarchspec(variance.model = list(model = "sGARCH", garchOrder = c(1, 1)), mean.model =
      list(armaOrder = c(p, q)))
```

```

# "sGARCH" for the standard GARCH model
# "eGARCH" for the exponential GARCH model
# "gjrGARCH" for the GJR-GARCH model (also known as the TGARCH model)
# "iGARCH" for the integrated GARCH model
# "csGARCH" for the component-wise GARCH model
# "avGARCH" for the asymmetric volatility GARCH model
# "ngARCH" for the non-linear GARCH model
# "nGARCH" for the non-Gaussian GARCH model

fit <- ugarchfit(spec, CHN[train])
forecast <- ugarchforecast(fit, n.ahead = 790)
ans[p, q] <- mean((forecast@forecast$seriesFor - CHN[test])^2)
log_likelihood <- sum(fit@fit$log.likelihoods)
lik[p, q] <- log_likelihood
# Get the number of estimated parameters in the model
num_parameters <- length(coef(fit))
# Calculate the number of observations
num_observations <- length(data)
# Calculate AIC and BIC
AIC[p, q] <- -2 * log_likelihood + 2 * num_parameters
BIC[p, q] <- -2 * log_likelihood + num_parameters * log(num_observations)
pb$tick()
}
}

```

我们对序列进行划分, 前 3000 个数据当成是训练数据, 后 790 个数据当成是验证数据, 以检验我们模型的精度, 计算后的结果为

	q = 1	q = 2	q = 3
p = 1	22605.82	22607.91	22609.80
p = 2	22607.94	22609.93	22611.87
p = 3	22609.58	22611.23	22627.48

表 1: 计算出 AIC 的取值

	q = 1	q = 2	q = 3
p = 1	22643.26	22651.59	22659.72
p = 2	22651.62	22659.85	22668.03
p = 3	22659.50	22667.40	22689.88

表 2: 计算出 BIC 的取值

表1, 2分别展示了不同模型 AIC, BIC 的取值,

Listing 29: 根据遍历结果进行模型定阶

```

> AIC_loc <- which(AIC == min(AIC), arr.ind = TRUE)
> AIC_loc
      row col
[1,]    1   1
> BIC_loc <- which(BIC == min(BIC), arr.ind = TRUE)
> BIC_loc
      row col

```

```
[1,] 1 1
```

从拟合精度而言, AIC 显示选择模型 ARMA(3, 3)-GARCH(1, 1) 更好, BIC 显示选择模型 ARMA(1, 1)-GARCH(1, 1). 然而从预测的角度而言,

Listing 30: 根据遍历结果进行模型定阶

```
> min_loc <- which(ans == min(ans), arr.ind = TRUE)
> min_loc
      row col
[1,]  2   2
```

ARMA(2, 2)-GARCH(1, 1) 模型有更小的均方误差, 综合比较后认为 ARMA(2, 2)-GARCH(1, 1) 模型的表现最好, 该模型的表达式为

$$r_t - \mu = \phi_1 x_{t-1} + \phi_2 x_{t-2} + e_t + \theta_1 e_{t-1} + \theta_2 e_{t-2}$$

其中 $\{e_t\}$ 是白噪声, 且 $e_t \stackrel{i.i.d.}{\sim} N(0, \sigma_t^2)$, 且满足

$$\sigma_t^2 = \omega + \alpha_1 e_t^2 + \beta_1 \sigma_{t-1}^2$$

其中 $\omega \geq 0, \alpha_1 \geq 0, \beta_1 \geq 0, \alpha_1 + \beta_1 < 1$.

模型输出的结果为

Listing 31: 输出的结果

```
Optimal Parameters
-----
      Estimate Std. Error  t value Pr(>|t|)
mu      -0.000053   0.000087  -0.612984  0.53989
ar1       0.444650   0.674203   0.659520  0.50956
ar2      -0.028154   0.449491  -0.062635  0.95006
ma1      -0.465836   0.671303  -0.693928  0.48773
ma2       0.017767   0.462687   0.038400  0.96937
omega     0.000000   0.000000   0.669363  0.50326
alpha1    0.054083   0.006306   8.577001  0.00000
beta1     0.943147   0.006097  154.688486  0.00000
```

注意到两个截距项都有较大的 p 值, 因此我们限制截距项为 0, 即可得到,

Listing 32: 输出的结果

```
> spec <- ugarchspec(variance.model = list(model = "sGARCH", garchOrder = c(1, 1)),
+                    mean.model = list(armaOrder = c(2, 2)), fixed.pars = list(mu = 0, omega = 0))
> fit <- ugarchfit(spec, data[train])
> show(fit)
Optimal Parameters
-----
      Estimate Std. Error  t value Pr(>|t|)
mu           0.00000      NA      NA      NA
ar1          1.13252   0.035393  31.998    0
ar2         -0.94908   0.021694 -43.749    0
ma1         -1.12715   0.044293 -25.448    0
ma2          0.92833   0.025175  36.876    0
omega        0.00000      NA      NA      NA
alpha1       0.04623   0.003778  12.236    0
beta1        0.95277   0.003802 250.606    0
```

Robust Standard Errors:

	Estimate	Std. Error	t value	Pr(> t)
mu	0.00000	NA	NA	NA
ar1	1.13252	0.060653	18.6722	0
ar2	-0.94908	0.023525	-40.3430	0
ma1	-1.12715	0.074731	-15.0827	0
ma2	0.92833	0.030175	30.7649	0
omega	0.00000	NA	NA	NA
alpha1	0.04623	0.006862	6.7371	0
beta1	0.95277	0.006441	147.9300	0

从而各个参数估计结果均显著, 因此得到的估计方程为

$$r_t = -0.000053 + 0.444650r_{t-1} - 0.028154r_{t-2} + e_t - 0.465836e_{t-1} + 0.017767e_{t-2}$$

$$\sigma_t^2 = 0.00000 + 0.054083e_t^2 + 0.943147\sigma_{t-1}^2$$

若限制两个截距项为 0, 则可以写出

$$r_t = 1.13252r_{t-1} - 0.94908r_{t-2} + e_t - 1.12715e_{t-1} + 0.92833e_{t-2}$$

$$\sigma_t^2 = 0.04623e_t^2 + 0.95277\sigma_{t-1}^2$$

上述模型既是我们建立好的 ARMA(2, 2)- GARCH(1, 1) 模型。

可以看出上述模型对 ARMA(2,2) 模型进行了一定程度的修正, $\alpha_1 + \beta_1 < 1$ 满足 GARCH 模型的约束条件, 从方差非常数项的系数均显著大于 0 可知, 未来的波动确实收到过去波动的正向影响, 表现出波动率聚集地现象。

4.4 基于不同分布假定下的 ARMA-GARCH 模型

受 [张东旭 16] 启发, 由于我们之前的假定是残差服从正态分布, 我们拟合后的模型给出的残差的 QQ 图如图 11 所示,

Listing 33: 残差的 QQ 图

```
spec <- ugarchspec(variance.model = list(model = "sGARCH", garchOrder = c(1, 1)),
                  mean.model = list(armaOrder = c(2, 2)), fixed.pars = list(mu = 0, omega = 0))
fit <- ugarchfit(spec, data)
qqnorm(fit@fit$residuals, col = "grey", pch = 16, cex = 1)
qqline(fit@fit$residuals, datax = FALSE, distribution = qnorm,
       probs = c(0.25, 0.75), qtype = 7, col = "steelblue", lwd = 2)
```

从图中可以看出, 残差并不满足正态分布, 我们考虑误差分布为 t 分布或者是广义误差分布 (GED), 我们比较模型的拟合程度

Listing 34: 拟合不同分布假定下的模型

```
spec <- ugarchspec(variance.model = list(model = "sGARCH", garchOrder = c(1, 1)),
                  mean.model = list(armaOrder = c(2, 2)), fixed.pars = list(mu = 0, omega = 0))
fit0 <- ugarchfit(spec, data)
show(fit0)

spec <- ugarchspec(variance.model = list(model = "sGARCH", garchOrder = c(1, 1)),
                  mean.model = list(armaOrder = c(2, 2)), fixed.pars = list(mu = 0, omega = 0),
                  distribution.model = "ged")
fit1 <- ugarchfit(spec, data)
show(fit1)
```

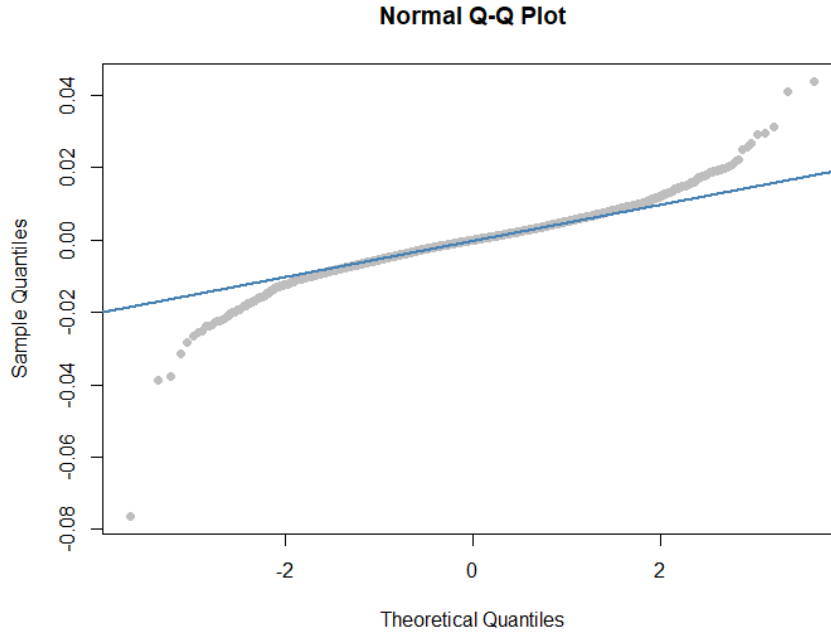


图 11: ARMA(2, 2)- GARCH(1, 1) 模型拟合后残差的 QQ 图

```
spec <- ugarchspec(variance.model = list(model = "sGARCH", garchOrder = c(1, 1)),
  mean.model = list(armaOrder = c(2, 2)), fixed.pars = list(mu = 0, omega = 0),
  distribution.model = "std")
fit2 <- ugarchfit(spec, data)
show(fit2)
```

得到结果如表3所示.

ARMA(2,2)-GARCH(1,1)	Norm	T	GED
log likelihood	14266.59	14332.5	14331.91
AIC	-7.5254	-7.5596	-7.5588
BIC	-7.5155	-7.5481	-7.5456

表 3: ARMA-GARCH 模型在不同分布假定下的比较结果

但是由于输出的结果 t 分布情况下对应的参数不显著, 因此我们选用 GED 作为最终的模型, 于是此时模型的形式为

$$r_t = \phi_1 x_{t-1} + \phi_2 x_{t-2} + e_t + \theta_1 e_{t-1} + \theta_2 e_{t-2}$$

其中 $\{e_t\}$ 是白噪声, 且服从 GED, GED 的密度函数为

$$f(x | \sigma_p, p) = \frac{1}{2\sigma_p p^{1/p} \Gamma(1 + \frac{1}{p})} \exp\left(-\frac{|x|^p}{p\sigma_p^p}\right)$$

且满足

$$\sigma_t^2 = \alpha_1 e_t^2 + \beta_1 \sigma_{t-1}^2$$

其中 $\alpha_1 \geq 0, \beta_1 \geq 0, \alpha_1 + \beta_1 < 1$.

Listing 35: 输出的结果

```
Robust Standard Errors:
      Estimate Std. Error  t value Pr(>|t|)
mu      0.000000      NA      NA      NA
ar1     1.186809    0.019946   59.501     0
ar2    -0.936889    0.013267  -70.621     0
ma1    -1.199133    0.022168  -54.093     0
ma2     0.930346    0.012793   72.724     0
omega   0.000000      NA      NA      NA
alpha1  0.042854    0.001620   26.459     0
beta1   0.956146    0.001269  753.599     0
skew    0.985210    0.020492   48.077     0
shape   1.458929    0.064321   22.682     0
> fit2@fit$coef[7]+fit2@fit$coef[8] <1
alpha1
TRUE
```

因此最终拟合结果为

$$r_t = 1.186809r_{t-1} - 0.936889r_{t-2} + e_t - 1.199133e_{t-1} + 0.930346e_{t-2}$$

$$\sigma_t^2 = 0.042854e_t^2 + 0.956146\sigma_{t-1}^2$$

从拟合结果可以看出, 上述模型依然满足 GARCH 模型限定条件. $\alpha_1 + \beta_1 < 1$ 满足 GARCH 模型的约束条件, 从方差非常数项的系数均显著大于 0 可知, 未来的波动确实收到过去波动的正向影响, 表现出波动率聚集地现象.

4.5 模型预测

首先, 我们对上述提出的模型进行模型预测.

Listing 36: 模型预测

```
plot(forecast@forecast$seriesFor, type = "l", main = "ARMA-GARCH Forecast", xlab = "Period", ylab =
     "Forecasted Values")
plot(forecast@forecast$sigmaFor, type = "l", main = "ARMA-GARCH Forecast", xlab = "Period", ylab =
     "Forecasted Values")
> head(forecast@forecast$seriesFor)
      1978-03-20 08:00:00
T+1      0.0002466585
T+2      0.0005200704
T+3      0.0002233451
T+4     -0.0002102637
T+5     -0.0005744882
T+6     -0.0004242966
```

画出预测结果如图12所示,

根据输出的结果我们可以看出对波动项的预测也仅仅能给出一些短期的预测, 预测方差也随着时间的增大而增大.

Listing 37: 预测效果的评价

```
> # MSE
> mean((forecast@forecast$seriesFor - CHN[test])^2)
[1] 3.253168e-05
> # MAE
> mean(abs(forecast@forecast$seriesFor - CHN[test]))
[1] 0.004277216
```

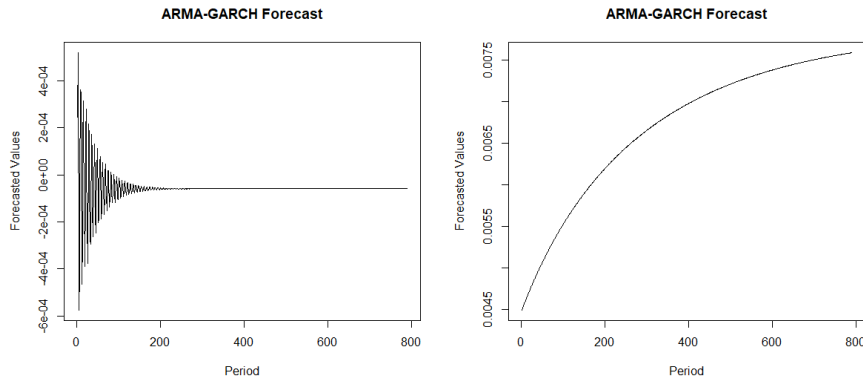



图 12: 预测结果

分别计算 MSE 和 MAE, 可以得到 3.253168×10^{-5} 与 0.004277216.

5 总结

本研究对以英镑为单位的人民币汇率进行了时间序列分析. 基本统计分析表明该序列是非平稳的非白噪声序列. 我们首先利用确定性模型对原始数据进行分析, 为了降低数据的噪声和波动性将日数据平均成月数据, 通过 ANOVA 检验发现序列中不存在季节效应. 接着, 我们采用 Holt 两指数平滑模型对原始时间序列进行趋势项建模, 通过残差分析验证了趋势项可以充分提取该序列的信息.

尽管人民币汇率是非平稳的时间序列, 但经过对数差分处理后, 序列变为平稳非白噪声序列. 因此, 我们采用 ARMA 模型进行建模, 并得到了最优模型 ARMA(4,4). 然而, 对模型拟合后的残差序列进行分析发现异方差现象的存在, 因此需要进一步建立 GARCH 模型. 通过 AIC 与 BIC 准则的综合考虑, 我们确定了 ARMA(2,2)-GARCH(1,1) 模型具有最佳的拟合效果.

由于拟合后的残差不满足正态分布假设, 我们尝试改变残差的假定分布, 进一步对人民币对数收益率进行拟合. 结果表明, 在广义误差分布 (GED) 假定下, GARCH 模型的拟合效果最好. 基于这一结果, 我们在广义误差分布 (GED) 假定下建立了 ARMA(2,2)-GARCH(1,1) 模型. 预测效果展现出只能对短期的波动效果进行预测.

综上所述, 本研究通过多个步骤的时间序列分析, 对以英镑为单位的人民币汇率进行了全面的建模和预测. 研究表明, ARMA(2,2)-GARCH(1,1) 模型能够较好地捕捉到汇率变动的长期趋势和短期波动, 为金融机构和投资者提供了重要的决策依据. 本研究的方法和结果对类似时间序列数据的分析具有借鉴意义.

References

- [Bol86] Tim Bollerslev. “Generalized autoregressive conditional heteroskedasticity”. In: *Journal of econometrics* 31.3 (1986), pp. 307–327.
- [Eng82] Robert F Engle. “Autoregressive conditional heteroscedasticity with estimates of the variance of United Kingdom inflation”. In: *Econometrica: Journal of the econometric society* (1982), pp. 987–1007.
- [EV22] Sebastian Engelke and Stanislav Volgushev. “Structure learning for extremal tree models”. In: *Journal of the Royal Statistical Society Series B: Statistical Methodology* 84.5 (2022), pp. 2055–2087.
- [张东旭 16] 张东旭. “基于 ARMA-GARCH 模型族的上证指数收益率波动的实证分析”. MA thesis. 清华大学, 2016.
- [殷金 10] 殷红 and 金永红. 计量经济学理论与方法. 清华大学出版社, 2010.