问题：

在doubs河流鱼群研究中，请筛选VOLPla站点有关CHE鱼类生物量和密度的记录，构建表3。其中，第1列为序号，第2列为时间戳（stamp），第3列为生物量（Biomass）。请根据下面问题作答。

|  |  |  |
| --- | --- | --- |
| no | Date | Biomass |
| 1854 | 1994-06-21 | 0.9500713 |
| 1855 | 1995-06-13 | 0.9519654 |
| 1856 | 1996-06-18 | 8.3632565 |
| ︙ | ︙ | ︙ |

1. library(tidyverse) 表3 doubs河1994-2020年CHE鱼类个体生物量
2. library(randomForest)
3. data <- read.table("fishBiomassData.txt ")
4. mydata <- data |>
5. subset(STATION=="VOLPla" & SP == "CHE") |>
6. select(DATE, BIOMASS)
7. ts <- ts(mydata$BIOMASS, start=1994, frequency=1)
8. ts\_org <- window(ts, end = 2018)
9. ts\_trf <- ts.org |> log() |> diff(1)
10. lag\_order <- 2
11. horizon <- 2
12. ts\_mbd <- embed(ts.trd, lag.order + 1)
13. Y\_train <- ts\_mbd[, 1]
14. X\_train <- ts\_mbd[, -1]
15. y\_test <- window(ts, start = 2019, end = 2020)
16. x\_test <- ts\_mbd[nrow(ts\_mbd), c(1:lag\_order)]
17. pred\_rf <- numeric(horizon)
18. for (i in 1:horizon){set.seed(1)
19. fit\_rf <- randomForest(X\_train, y\_train)
20. pred\_rf[i] <- predict(fit\_rf, X\_test)
21. y\_train <- y\_train[-1]
22. X\_train <- X\_train[-nrow(X\_train), ] }
23. pred\_rf
24. exp\_term <- exp(cumsum(pred\_rf))
25. last\_obs <- as.vector(tail(ts\_org, 1))
26. backtrans\_fc <- last\_obs \* exp\_term
27. y\_pred <- ts(backtrans\_fc, start = 2019, frequency = 1)
28. library(fpp2)
29. forecast::accuracy(as.numeric(y\_pred), as.numeric(y\_test))
30. ts\_fc <- cbind(ts,pred = c(rep(NA, length(ts\_org)), y\_pred))
31. plot\_fc <- ts\_fc |> autoplot() + theme\_minimal()
32. plot\_fc

1) 这是构建机器学习时间序列预测模型，根据代码，请简述时间序列预测建模基本原理以及基本过程。

①时间序列预测建模基本原理：利用回归模型，基于时间序列的历史数据，识别出数据的模式，如趋势、季节性和非平稳性，从而进行更准确的未来值预测。在此过程中，数据可能需要经过一系列的预处理和转化，以便适合模型的输入需求。

②时间序列预测建模的基本过程：

**第一步**：构建训练矩阵，将数据从文件中读取，进行筛选和数据变换，最后构建滞后特征矩阵。

library(tidyverse) # 加载 tidyverse 包

library(randomForest) # 加载 randomForest 包

data <- read.table("fishBiomassData.txt ") # 读取数据

mydata <- data |>

subset(STATION == "VOLPla" & SP == "CHE") |> # 筛选特定站点和物种

select(DATE, BIOMASS) # 选择日期和生物量

ts <- ts(mydata$BIOMASS, start = 1994, frequency = 1) # 构建时间序列对象

ts\_org <- window(ts, end = 2018) # 提取到2018年的数据

ts\_trf <- ts\_org |> log() |> diff(1) # 对数变换 + 一阶差分

lag\_order <- 2 # 设置滞后阶数

ts\_mbd <- embed(ts\_trf, lag\_order + 1) # 构建滞后特征矩阵

**第二步**：分割训练集和测试集，将构建好的训练矩阵分为训练集和测试集

Y\_train <- ts\_mbd[, 1] # 分割出目标变量 (y)

X\_train <- ts\_mbd[, -1] # 分割出特征变量 (X)

y\_test <- window(ts, start = 2019, end = 2020) # 定义测试集（2019-2020）

x\_test <- ts\_mbd[nrow(ts\_mbd), c(1:lag\_order)] # 定义测试特征

**第三步**：训练和评价模型，使用训练集训练随机森林模型

horizon <- 2 # 设定预测步数

pred\_rf <- numeric(horizon) # 初始化预测结果向量

for (i in 1:horizon) { # 循环进行逐步预测

set.seed(1) # 设置随机种子

fit\_rf <- randomForest(X\_train, Y\_train) # 训练随机森林模型

pred\_rf[i] <- predict(fit\_rf, x\_test) # 生成预测

Y\_train <- Y\_train[-1] # 更新目标变量

X\_train <- X\_train[-nrow(X\_train), ] # 更新特征矩阵

}

**第四步**：评估模型，对模型的预测结果进行反变换并评价其准确性，同时可视化结果

exp\_term <- exp(cumsum(pred\_rf)) # 计算反变换因素

last\_obs <- as.vector(tail(ts\_org, 1)) # 获取最后观察值

backtrans\_fc <- last\_obs \* exp\_term # 反变换预测值

y\_pred <- ts(backtrans\_fc, start = 2019, frequency = 1) # 构建最终预测时间序列

library(fpp2) # 加载 fpp2 包

accuracy\_metrics <- forecast::accuracy(as.numeric(y\_pred), as.numeric(y\_test)) # 评估模型准确性

ts\_fc <- cbind(ts, pred = c(rep(NA, length(ts\_org)), y\_pred)) # 合并时间序列和预测结果

plot\_fc <- ts\_fc |> autoplot() + theme\_minimal() # 可视化预测结果

2）除了基于原始值（生物量）构造滞后特征外，还可从时间戳提取并构造建模特征，请简述从时间戳构造建模特征的依据是什么？

①日期时间特征(date time features)：即观察值与具体日期或季节有关，如日低温与雨季等有重要关系，可以选择把日期或季节作为特征。

②滞后特征(lag features)：如今年的11月份数据与以往年份的11月份数据更相关，即要关注历史上的今天。

③窗口特征(window features)：建模精度不仅与采用或选择的滑动窗口大小密切相关，而且与窗口内的均值、最大值等具体数据也有关。