BỘ GIÁO DỤC VÀ ĐÀO TẠO
TRƯỜNG ĐẠI HỌC DÂN LẬP HẢI PHÒNG
----------o0o----------

KHAI PHÁ DỮ LIỆU VỚI R

ĐỒ ÁN TỐT NGHIỆP ĐẠI HỌC HỆ CHÍNH QUY

Ngành: Công Nghệ Thông Tin

BỘ GIÁO DỤC VÀ ĐÀO TẠO
TRƯỜNG ĐẠI HỌC DÂN LẬP HẢI PHÒNG
----------o0o----------

KHAI PHÁ DỮ LIỆU VỚI R

ĐỒ ÁN TỐT NGHIỆP ĐẠI HỌC HỆ CHÍNH QUY
Ngành: Công Nghệ Thông Tin

Sinh viên thực hiện: Trần Văn Ngọc.
Giáo viên hướng dẫn: Thạc sĩ Nguyễn Thị Thanh Thoan.
Mã số sinh viên: 121223.

BỘ GIÁO DỤC VÀ ĐÀO TẠO
TRƯỜNG ĐẠI HỌC DÂN LẬP HẢI PHÒNG

CỘNG HÒA XÃ HỘI CHỦ NGHĨA VIỆT NAM
Độc lập – Tự do – Hạnh phúc
----------o0o----------

NHIỆM VỤ THIẾT KẾ TỐT NGHIỆP

Sinh viên: TRẦN VĂN NGỌC Mã số sinh viên: 121223

Lớp: CT1201 Ngành: Công nghệ thông tin

Tên đề tài: KHAI PHÁ DỮ LIỆU VỚI R

NHIỆM VỤ ĐỀ TÀI

1. Nội dung và các yêu cầu cần giải quyết trong nhiệm vụ đề tài tốt nghiệp

+ Tìm hiểu Ngôn Ngữ R
+ Tìm hiểu Khai Phá Dữ Liệu
+ Tìm hiểu bài toán áp dụng và demo chương trình

2. Các số liệu cần thiết để thiết kế, tính toán.

-Dữ liệu từ thị trường New York Stock Exchange từ tháng 4/1970 đến tháng 5/2002

3. Địa điểm thực tập

CÁN BỘ HƯỚNG DẪN ĐỀ TÀI TỐT NGHIỆP

Người hướng dẫn thứ nhất:
Họ và tên: ……Nguyễn Thị Thanh Thoan…….. .
Họchàm, học vị: ……Thạc Sỹ…….. . .
Cơ quan công tác: Khoa Công Nghệ Thông Tin – Đại Học Dân Lập Hải Phòng
Nội dung hướng dẫn: ……
……+Tìm hiểu Ngôn Ngữ R…….
……+Tìm hiểu Khai Phá Dữ Liệu Với R……. .
……+Tìm hiểu bài toán áp dụng và Demo chương trình….
Người hướng dẫn thứ hai:
Họ và tên: …….
Học hàm, học vị: …….. .
Cơ quan công tác: …….
Nội dung hướng dẫn: ……. . . . . .
. . . . . .
. . . . . .
. . . . . .
……
……
Đề tài tốt nghiệp được giao ngày tháng năm 2012
Yêu cầu phải hoàn thành trước ngày tháng năm 2012

Đã nhận nhiệm vụ: Đ. T. T. N
Sinh viên

Đã nhận nhiệm vụ: Đ. T. T. N
Cán bộ hướng dẫn Đ. T. T. N

Hải phòng, ngày tháng năm 2012
HIỆU TRƯỞNG

GS. TS. NGƯTTrần Hữu Nghị

PHẦN NHẬN XÉT TÓM TẮT CỦA CÁN BỘ HƯỚNG DẪN

1. Tinh thần thái độ của sinh viên trong quá trình làm đề tài tốt nghiệp:
   ……
   ……
   ……
   ……
   ……
   ……
2. Đánh giá chất lượng của đề tài tốt nghiệp (so với nội dung yêu cầu đã đề ra trong nhiệm vụ đề tài tốt nghiệp)
   ……
   ……
   ……
   ……
   ……. . .
3. Cho điểm của cán bộ hướng dẫn:
   (Điểm ghi bằng số và chữ)
   ……
   ……
   ……. . . ……

Ngày tháng năm 2012

Cán bộ hướng dẫn chính

(Ký, ghi rõ họ tên)

PHẦN NHẬN XÉT ĐÁNH GIÁ CỦA CÁN BỘ CHẤM PHẢN BIỆN ĐỀ TÀI TỐT NGHIỆP

1. Đánh giá chất lượng đề tài tốt nghiệp (về các mặt như cơ sở lý luận, thuyết minh chương trình, giá trị thực tế, …)

2. Cho điểm của cán bộ phản biện:

(Điểm ghi bằng số và chữ)

……

……

……. . .

Ngày tháng năm 2012

Cán bộ chấm phản biện

(Ký, ghi rõ họ tên)

Mục Lục

LỜI CẢM ƠN

Trong lời đầu tiên của báo cáo đồ án tốt nghiệp "Khai Phá Dữ Liệu Với R" này, em muốn gửi những lời cám ơn và biết ơn chân thành nhất của mình tới tất cả những người đã hỗ trợ, giúp đỡ em về kiến thức và tinh thần trong quá trình thực hiện đồ án.

Trước hết, em xin chân thành cám ơn Cô Giáo - Ths. Nguyễn Thị Thanh Thoan - Giảng viên Khoa Công Nghệ Thông Tin, Trường ĐHDL Hải Phòng, người đã trực tiếp hướng dẫn, nhận xét, giúp đỡ em trong suốt quá trình thực hiện đồ án.

Xin chân thành cảm ơn các thầy cô trong Khoa Công Nghệ Thông Tin và các phòng ban nhà trường đã tạo điều kiện tốt nhất cho em cũng như các bạn khác trong suốt thời gian học tập và làm tốt nghiệp.

Cuối cùng em xin gửi lời cảm ơn đến gia đình, bạn bè, người thân đã giúp đỡ động viên em rất nhiều trong quá trình học tập và làm Đồ án Tốt Nghiệp.

Do thời gian thực hiện có hạn, kiến thức còn nhiều hạn chế nên Đồ án thực hiện chắc chắn không tránh khỏi những thiếu sót nhất định. Em rất mong nhận được ý kiến đóng góp của thầy cô giáo và các bạn để em có thêm kinh nghiệm và tiếp tục hoàn thiện đồ án của mình.

Em xin chân thành cảm ơn!

Hải Phòng, ngày 25 tháng 12 năm 2012

Sinh viên

Trần Văn Ngọc

Chương 1: Giới Thiệu Ngôn Ngữ R

I. Khái quát chung

1. Giới thiệu R

Năm 1996, trong một bài báo quan trọng về tính toán thống kê, hai nhà thống kê học Ross Ihaka và Robert Gentlan thuộc Trường đại học Auckland, New Zealand phác hoạ một ngôn ngữ mới cho phân tích thống kê mà họ đặt tên là R. Sáng kiến này được rất nhiều nhà thống kê học trên thế giới tán thành và tham gia vào việc phát triển R.

Vậy R là gì? Nói một cách ngắn gọn, R là một phần mềm sử dụng cho phân tích thống kê và vẽ biểu đồ. Thật ra, về bản chất, R là ngôn ngữ máy tính đa năng, có thể sử dụng cho nhiều mục tiêu khác nhau, từ tính toán đơn giản, toán học giải trí(recreational mathatics) , tính toán ma trận(matrix) , đến các phân tích thống kê phức tạp. Vì là một ngôn ngữ, cho nên người ta có thể sử dụng R để phát triển thành các phần mềm chuyên môn cho một vấn đề tính toán cá biệt.

2. Ưu điểm của R

- R có chứa nhiều loại kỹ thuật thống kê: mô hình hóa tuyến tính và phi tuyến, kiểm thử thống kê cổ điển, phân tích chuỗi thời gian, phân loại, phân nhóm, v. v. và đồ họa. R
- R cũng có tính mở rộng cao bằng cách sử dụng các gói cho người dùng đưa lên cho một số chức năng và lĩnh vực nghiên cứu cụ thể.
- Một điểm mạnh khác của R là nền tảng đồ họa có thể tạo ra những đồ thị chất lượng cao cùng các biểu tượng toán học.
- Dù R được dùng chủ yếu bởi những nhà thống kê và cũng có thể dùng làm một công cụ tính toán ma trận tổng quát với các kết quả đo đạc cạnh tranh so với GNU Octave và đối thủ thương mại của nó, MATLAB. Giao diện RWeka đã được thêm vào phần mềm khai phá dữ liệu phổ biến Weka, cho phép đọc/ghi

định dạng arff vì vậy cho phép sử dụng tính năng khai phá dữ liệu trong Weka và thống kê trong R

- Ngôn ngữ R có rất nhiều ưu điểm so với các ngôn ngữ lập trình bậc cao như C , C++ , Java….
- R có khả năng điều khiển dữ liệu và lưu trữ số liệu, R còn có tính nguyên bản.
- R cho phép sử dụng ma trận đại số.
- Có thể sử dụng bảng băm và các biểu thức chính quy
- R cũng hỗ trợ lập trình hướng đối tượng.
- Khả năng biểu diễn đồ họa phong phú.
- Ngôn ngữ R cũng cung cấp các cấu trúc điều khiển cơ bản như các ngôn ngữ lập trình bậc cao khác. Ví dụ như :If…else…;while…. ;for……vv.

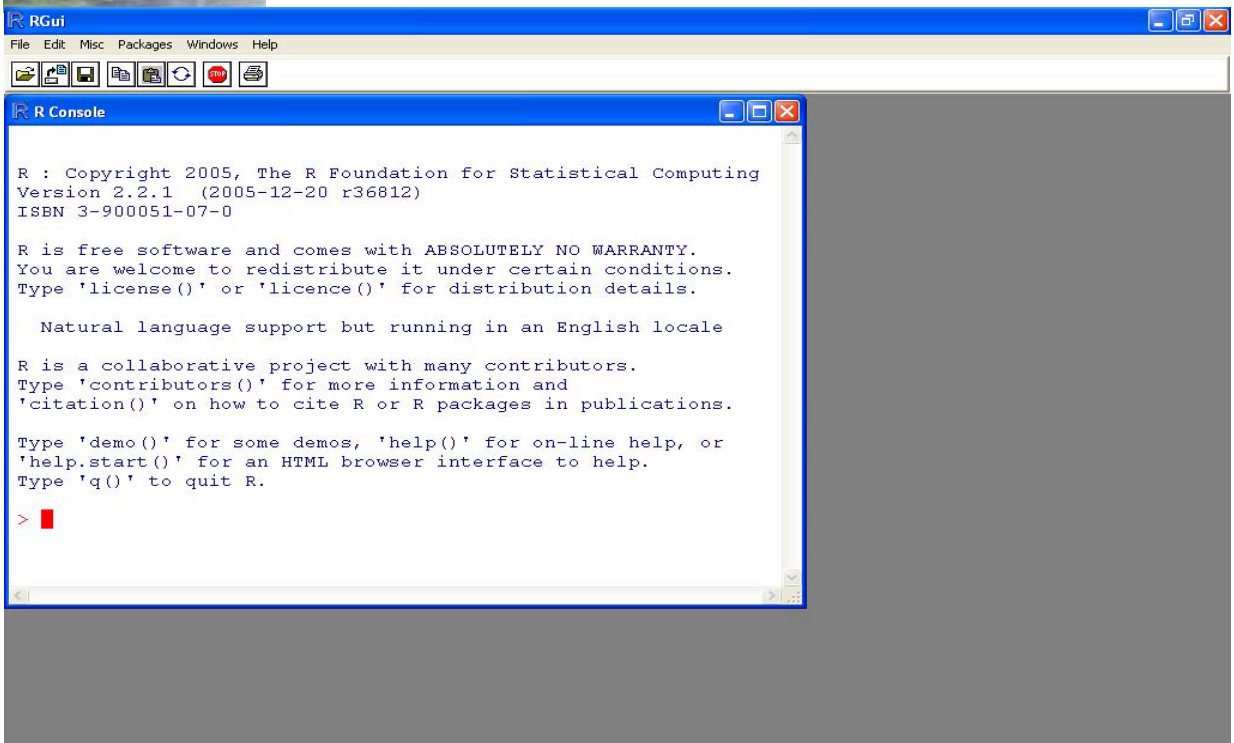
II. Hướng dẫn sử dụng R

1. Cài đặt và giao diện

Để sử dụng R, việc đầu tiên là phải cài đặt R trong máy.

Khi đã tải R xuống máy tính, bước kế tiếp là cài đặt vào máy tính. Để làm việc này, chỉ đơn giản nhấn chuột vào tài liệu trên và làm theo hướng dẫn cách cài đặt trên màn hình. Đây là một bước rất đơn giản, chỉ cần 1 phút là việc cài đặt R có thể hoàn tất.

Sau khi hoàn tất việc cài đặt, một *icon* sẽ xuất hiện trên *desktop* của máy tính(Hình bên). Đến đây thì đã sẵn sàng sử dụng R. Cóthể nhấp chuột vào icon này và sẽ có một cửa sổ như sau:



Hình 1. 1 Giao diện ngôn ngữ R

2. Nhập dữ liệu trong R

Dữ liệu mà R hiểu được phải là dữ liệu trong một data.frame.

1) **Nhập dữ liệu trực tiếp từ dòng lệnh theo cấu trúc từ hàm c():**

Tên_biến_lưu_dữ_liệu <- c(pt1, pt2, …, ptn)

- VD: a <- c(4,67,87,4,5,3)

  b <- c(9,8,7,5,6,5,22)

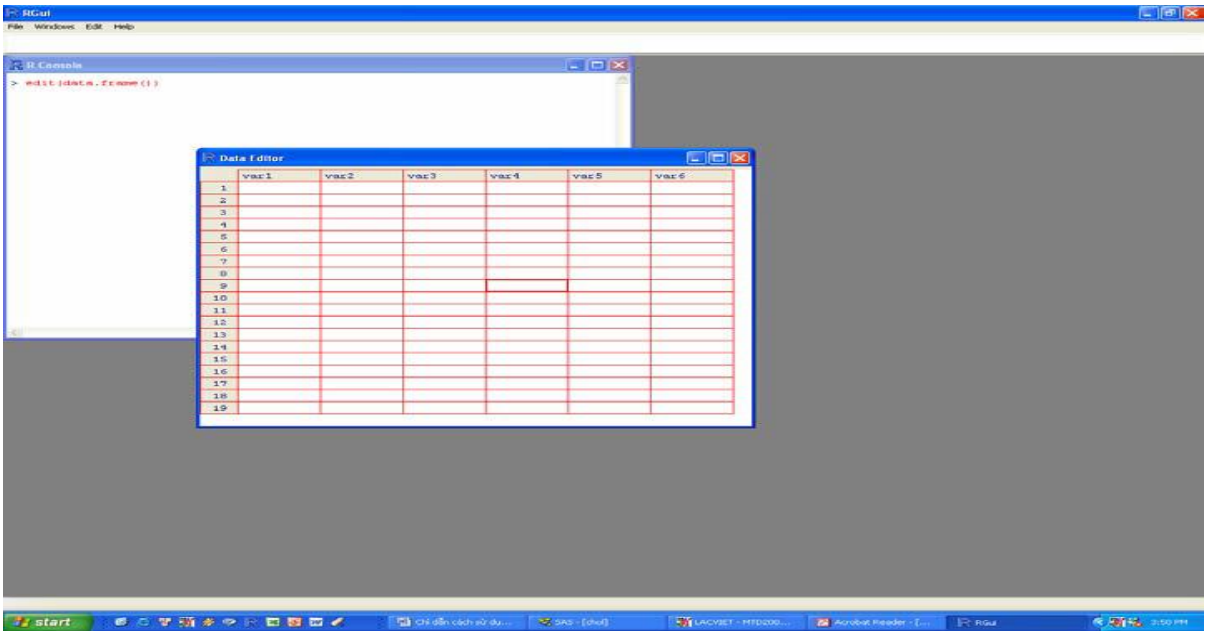
Ghép các biến riêng lẻ nhập bằng hàm c thành một khung dữ liệu để sử dụng sau này:

Tên_biến <- data.frame(tham_số_1,tham_số_2, ……. , tham_số_n)

Lưu tên tệp: *save(tên_biến ,file="tênfile.rdata")*

2) Nhập dữ liệu từ cửa sổ nhập Data Editor:

> **e***dit(data.frame())*



3) Nhập dữ liệu từ File text

- Tênbiến <- read.table("path file", header=TRUE)

Xem lại nội dung tệp vừa nhập:

- names(Tênbiến)

Lưu lại dưới dạng tệp R để xử lý sau này

- *save(tên_biến ,file="tênfile.rdata")*

4) Nhập dữ liệu từ File excel

- Lưu tệp excel dưới đuôi *.csv
- Đọc tệp csv với cú pháp sau:

Tênbiến <- read.csv("đường_dẫn_đến_file_csv",HEADER=true)

- Tham số *HEADER = true* cho R biết dòng đầu tiên của file xls được chọn làm tên của các cột.
- Lưu lại tệp để sử dụng lần sau bằng lệnh save().

2. 5 Nhập từ tệp SPSS: read. spss

Phần mềm thống kê SPSS lưu dữ liệu dưới dạng "sav". Chẳng hạn như nếu đã có một dữ liệu có tên là testo. sav trong thư mục c:\works\insulin, và muốn chuyển dữ liệu này sang dạng R có thể hiểu được, cần sử dụng lệnh read. spss trong package có tên là foreign. Các lệnh sau đây sẽ hoàn tất dễ dàng việc này:

Việc đầu tiên cho truy nhập foreign bằng lệnh library:

> library(foreign)

Việc thứ hai là lệnh read. spss:

> setwd("c:/works/insulin")

> testo <- read. spss("testo. sav", to. khung dữ liệu=TRUE)

Lệnh thứ hai read. spss yêu cầu R đọc số liệu từ "testo. sav", và cho vào một

data. frame có tên là testo.

Bây giờ có thể lưu testo dưới dạng R để xử lí sau này bằng lệnh sau đây:

> save(testo, file="testo. rda")

3. Văn Phạm R

- R là một ngôn ngữ "đối tượng ". Do đó, các dữ liệu trong R được chứa trong object.
- R phân biệt chữ hoa và chữ thường. VD: R khác với r
- Việc đặt tên một đối tượng hay một biến số trong R khá linh hoạt, tên một đối tượng phải được viết liền nhau và không đặt trùng với những đối tượng đã có.

- Khi có 2 chữ rời nhau R dùng dấu chấm để thay vào giữa khoảng trống. VD: read.table, data.frames.

4. Các lệnh hệ thống

4. 1 Lệnh về môi trường vận hành của R

getwd()	Cho biết thư mục hiện hành là gì
setwd(c:/works)	Chuyển thư mục vận hành về c:\works(chú ý R dùng "/")
options(prompt="R>")	Đổi prompt thành R>
options(width=100)	Đổi chiều rộng cửa sổ R thành 100 characters
options(scipen=3)	Đổi số thành 3 số thập phần(thay vì kiểu 1. 2E-04)
options()	Cho biết các thông số về môi trường của R

4. 2 Lệnh cơ bản

ls()	Liệt kê các đối tượng trong bộ nhớ
rm(object)	Xóa bỏ đối tượng
seach()	Tìm hướng

4. 3 Trợ giúp trong R

Ngoài lệnh ***args()*** R còn cung cấp lệnh ***help()*** để người sử dụng có thể hiểu "Văn phạm" của từng hàm. Chẳng hạn như muốn biết hàm ***lm*** có những tham số gì chỉ cần gõ lệnh: *>help()*

hay

>?lm

một cửa sổ sẽ hiện ra bên ngoài của màn hình chỉ rõ cách sử dụng ra sao và thậm chí có cả ví du.

Sử dụng lệnh *help. start()* một cửa sổ sẽ xuất hiện chỉ dẫn toàn bộ hệ thống R.

Hàm ***apropos*** cũng rất có ích vì nó cung cấp cho tất cả các hàm trong R bắt đầu bằng kí tự mà muốn tìm. Chẳng hạn như muốn biết hàm nào trong R có kí tự "lm" thì chỉ gõ lệnh:

> apropos(lm)

5. Tổ chức dữ liệu trong R

Sử dụng R cho các phép tính ma trận

- *Nhập dữ liệu vào ma trận theo cú pháp:*

  >tenbien <- Matrix(biendl, nrow)↵

  VD: ta có ma trận A có dạng A =

  Khi nhập vào R sẽ nhập như sau:

  > x <- c(4,5,6,7,8,9,10,11,12)

  > A <- matrix(x, nrow=3)

  Cho 2 ma trận A và B:

  - Cộng (trừ) 2 ma trận: > A+ (-)B
  - Nhân 2 ma trận: > A %*%B
  - Ma trận nghịch đảo: > solve(A)
  - Ngoài ra R có một gói Matrix chuyên thiết kế cho tính toán ma trận.

6. Các lệnh lập trình trong R

Sẽ quay lại với dữ liệu chol trong ví dụ 1. Để tiện việc theo dõi xin nhắc lại rằng đã nhập số liệu vào trong một dữ liệu R có tên là chol từ một text file có tên là chol. txt:

> setwd("c:/works/stats")

> chol <- read. table("chol. txt", header=TRUE)

> attach(chol)

6. 1 Kiểm tra số liệu trống không(missing value)

Trong nghiên cứu, vì nhiều lí do số liệu không thể thu thập được cho tất cả đối tượng, hay không thể đo lường tất cả biến số cho một đối tượng. Trong trường hợp đó, số liệu trống được x là "missing value". R x các số liệu trống không là NA. Có một số kiểm định thống kê đòi hỏi các số liệu trống không phải được loại ra trước khi phân tích. R có một lệnh rất có ích cho việc này: na. omit, và cách sử dụng như sau:

> chol. new <- na. omit(chol)

Trong lệnh trên, yêu cầu R loại bỏ các số liệu trống không trong khung dữ liệu chol và đưa các số liệu không trống vào khung dữ liệu mới tên là chol. new. Chú ý lệnh trên chỉ là ví dụ, vì trong dữ liệu chol không có số liệu trống không.

6. 2 Tách rời dữ liệu: subset

Nếu vì một lí do nào đó, chỉ muốn phân tích riêng cho nam giới, có thể tách chol ra thành hai khung dữ liệu, tạm gọi là nam và nu. Để làm chuyện này, dùng lệnh subset(data, cond) , trong đó dữ liệu là khung dữ liệu mà muốn tách rời, và cond là điều kiện. Ví dụ:

> nam <- subset(chol, sex==”Nam”)

> nu <- subset(chol, sex==”Nu”)

Sau khi ra hai lệnh này, đã có 2 dữ liệu mới tên là nam và nu. Chú ý điều kiện sex == “Nam” và sex == “Nu” dùng ==thay vì = để chỉ điều kiện chính xác.

Tất nhiên, cũng có thể tách dữ liệu thành nhiều khung dữ liệu khác nhau với những điều kiện dựa vào các biến số khác. Chẳng hạn như lệnh sau đây tạo ra một khung dữ liệu mới tên là old với những bệnh nhân trên 60 tuổi:

> old <- subset(chol, age>=60)

> dim(old)

[1] 25 8

Hay một khung dữ liệu mới với những bệnh nhân trên 60 tuổi và nam giới:

> n60 <- subset(chol, age>=60 & sex==”Nam”)

> dim(n60)

[1] 9 8

6. 3 Chiết số liệu từ một data . frame

Trong chol có 8 biến số. Có thể chiết dữ liệu chol và chỉ giữ lại những biến số cần thiết như mã số(id) , độ tuổi(age) và total cholestrol(tc) . Để ý từ lệnh names(chol) rằng biến số id là cột số 1, age là cột số 3, và biến số tc là cột số 7. có thể dùng lệnh sau đây:

>data2 <- chol[, c(1, 3, 7)]

6. 4 Nhập hai khung dữ liệu thành một:merge

Giả dụ như có dữ liệu chứa trong hai khung dữ liệu. Dữ liệu thứ nhất tên là d1 gồm3cột: id, sex, tc nhưsau:

id sex tc

1 Nam 4. 0

2 Nu 3. 5

3 Nu 4. 7

4 Nam 7. 7

5 Nam 5. 0

6 Nu 4. 2

7 Nam 5. 9

8 Nam 6. 1

9 Nam 5. 9

10 Nu 4. 0

Dữ liệu thứ hai tên là d2 gồm 3 cột: id, sex, tg như sau:

id sex tg

1 Nam 1. 1

2 Nu 2. 1

3 Nu 0. 8

4 Nam 1. 1

5 Nam 2. 1

6 Nu 1. 5

7 Nam 2. 6

8 Nam 1. 5

9 Nam 5. 4

10 Nu 1. 9

11 Nu 1. 7

Hai dữ liệu này có chung hai biến số id và sex. Nhưng dữ liệu d1 có 10 dòng, còn dữ liệu d2 có 11 dòng. có thể nhập hai dữ liệu thành một khung dữ liệu bằng cách dùng lệnh merge như sau:

```
> d <- merge(d1, d2, by="id", all=TRUE)
> d
```

id sex. x tc sex. y tg

1 1 Nam 4. 0 Nam 1. 1

2 2 Nu 3. 5 Nu 2. 1

3 3 Nu 4. 7 Nu 0. 8

4 4 Nam 7. 7 Nam 1. 1

5 5 Nam 5. 0 Nam 2. 1

6 6 Nu 4. 2 Nu 1. 5

7 7 Nam 5. 9 Nam 2. 6

8 8 Nam 6. 1 Nam 1. 5

9 9 Nam 5. 9 Nam 5. 4

10 10 Nu 4. 0 Nu 1. 9

11 11 <NA> NA Nu 1. 7

Trong lệnh merge, yêu cầu R nhập 2 dữ liệu d1 và d2 thành một và đưa vào khung dữ liệu mới tên là d, và dùng biến số id làm chuẩn. Để ý thấy bệnh nhân số 11 không có số liệu cho tc, cho nên R cho là NA(một dạng "not available") .

6. 5 Mã hóa số liệu(data coding)

Trong việc xử lí số liệu dịch tễ học, nhiều khi cần phải biến đổi số liệu từ biến liên tục sang biến mang tính cách phân loại. Chẳng hạn như trong chẩn đoán loãng xương, những phụ nữ có chỉ số T của mật độ chất khoáng trong xương(bonineral density hay BMD) bằng hay thấp hơn -2. 5 được x là"loãng xương", những ai có BMD giữa -2. 5 và -1. 0 là "xốp xương"(osteopenia) , và trên-1. 0 là "bình thường". Ví dụ, có số liệu BMD từ 10 bệnh nhân như sau:

-0. 92, 0. 21, 0. 17, -3. 21, -1. 80, -2. 60, -2. 00, 1. 71, 2. 12, -2. 11

Để nhập các số liệu này vào R có thể sử dụng *hàm* c như sau:

bmd <- c(-0. 92, 0. 21, 0. 17, -3. 21, -1. 80, -2. 60,

-2. 00, 1. 71, 2. 12, -2. 11)

Để phân loại 3 nhóm loãng xương, xốp xương, và bình thường, có thể dùng mã số 1, 2 và 3. Nói cách khác, muốn tạo nên một biến số khác(hãy gọi là diagnosis) gồm 3 giá trị trên dựa vào giá trị của bmd. Để làm việc này, sử dụng lệnh:

tạm thời cho biến số diagnosis bằng bmd

>diagnosis<-bmd

biến đổi bmd thành diagnosis

> diagnosis[bmd <= -2. 5] <- 1

> diagnosis[bmd > -2. 5 & bmd <= 1. 0] <- 2

>diagnosis[bmd>-1. 0]<-3

tạo thành một data frame

>data<-khung dữ liệu(bmd, diagnosis)

liệt kê để kiểm tra x lệnh có hiệu quả không

> data

bmd diagnosis

1 -0. 92 3

2 0. 21 3

3 0. 17 3

4 -3. 21 1

5 -1. 80 2

6 -2. 60 1

7 -2. 00 2

8 1. 71 3

9 2. 12 3

10 -2. 11 2

6. 6 Biến đổi số liệu bằng cách dùng *replace*

Một cách biến đổi số liệu khác là dùng replace, nhưng cách này tương đối phức tạp hơn. Tiếp tục ví dụ trên, biến đổi từ bmd sang diagnosis như sau:

> diagnosis <- bmd

> diagnosis <- replace(diagnosis, bmd <= -2. 5, 1)

> diagnosis <- replace(diagnosis, bmd > -2. 5 & bmd <= 1. 0, 2)

> diagnosis <- replace(diagnosis, bmd > -1. 0, 3)

6. 7 Biến đổi thành yếu tố(*factor*)

Trong phân tích thống kê, phân biệt một biến số mang tính *yếu tố* và biến số liên tục bình thường. Biến số yếu tố không thể dùng để tính toán như cộng trừ nhân chia, nhưng biến số số học có thể sử dụng để tính toán. Chẳng hạn như trong ví dụ bmd và diagnosis trên, diagnosis là yếu tố vì giá trị trung bình giữa 1 và 2 chẳng có ý nghĩa thực tế gì cả; còn bmd là biến số số học.

Nhưng hiện nay, diagnosis được x là một biến số số học. Để biến thành biến số yếu tố, cần sử dụng *hàm* factor như sau:

```
> diag <- factor(diagnosis)
> diag
[1] 3 3 3 1 2 1 2 3 3 2
Levels:1 2 3
```

Chú ý R bây giờ thông báo cho biết diag có 3 bậc:1, 2 và 3. Nếu yêu cầu R tính số trung bình của diag, R sẽ không làm theo yêu cầu này, vì đó không phải là một biến số số học:

```
> mean(diag)
[1] NA
Warning message:
argument is not numeric or logical: returning NA in: mean. default(diag)
```

Dĩ nhiên, có thể tính giá trị trung bình của diagnosis:

```
> mean(diagnosis)
[1]
2. 3
```

Nhưng kết quả 2. 3 này không có ý nghĩa gì trong thực tế cả.

6. 8 Chia nhóm bằng *cut*

Với một biến liên tục, có thể chia thành nhiều nhóm bằng hàm cut. Ví dụ, có biến age như sau:

```
> age <- c(17, 19, 22, 43, 14, 8, 12, 19, 20, 51, 8, 12, 27, 31, 44)
```

Độ tuổi thấp nhất là 8 và cao nhất là 51. Nếu muốn chia thành 2 nhóm tuổi:

> cut(age, 2)

[1](7. 96, 29. 5](7. 96, 29. 5](7. 96, 29. 5](29. 5, 51](7. 96, 29. 5](7. 96, 29. 5](7. 96, 29. 5](7. 96, 29. 5] [9](7. 96, 29. 5](29. 5, 51](7. 96, 29. 5](7. 96, 29. 5](7. 96, 29. 5](29. 5, 51](29. 5, 51]

Levels:(7. 96, 29. 5](29. 5, 51]

cut chia biến age thành 2 nhóm: nhóm 1 tuổi từ 7. 96 đến 29. 5; nhóm 2 từ 29. 5 đến 51. có thể đếm số đối tượng trong từng nhóm tuổi bằng hàmtable như sau:

>table(cut(age, 2))

(7. 96, 29. 5](29. 5, 51]

11 4

Trong lệnh sau đây, chia biến độ tuổi thành 3 nhóm và đặt tên ba nhóm là “low”, “medium” và “high”:

> ageg <- cut(age, 3, labels=c("low", "medium", "high"))

[1] low low low high low low low low low high low low medium medium

[15] high

Levels: low medium high

> ageg <- cut(age, 3, labels=c("low", "medium", "high"))

> table(ageg)

Ageg

low medium high

10 2 3

Tất nhiên, cũng có thể chia age thành 4 nhóm(quartiles) bằng cách cho những thông số 0, 0. 25, 0. 50 và 0. 75 như sau:

cut(age,

breaks=quantiles(age, c(0, 0. 25, 0. 50, 0. 75, 1)) ,

labels=c(“q1”, “q2”, “q3”, “q4”) ,

include. lowest=TRUE)

6. 9 Tập hợp số liệu bằng *cut2(*Hmisc)

Hàm cut trên chia biến số theo giá trị của biến, chứ không dựa vào số mẫu, cho nên số lượng mẫu trong từng nhóm không bằng nhau. Tuy nhiên, trong phân tích thống kê, có khi cần phải phân chia một biến số liên tục thành nhiều nhóm dựa vào phân phối của biến số nhưng số mẫu bằng hay tương đương nhau. Chẳng hạn như đối với biến số bmd có thể "cắt" dãy số thành 3 nhóm với số mẫu tương đương nhau bằng cách dùng function cut2(trong package Hmisc) như sau:

```
> # nhập package Hmisc để có thể dùng function cut2
> library(Hmisc)
> bmd <- c(-0. 92, 0. 21, 0. 17, -3. 21, -1. 80, -2. 60, -2. 00, 1. 71, 2. 12, -2. 11)
> # chia biến số bmd thành 2 nhóm và để trong đối tượng group
> group <- cut2(bmd, g=2)
> table(group)
group
[-3. 21, -0. 92) [-0. 92, 2. 12]
5        5
```

Như thấy qua ví dụ trên, g = 2 có nghĩa là chia thành 2 nhóm(g=group) . R tự động chia thành nhóm 1 gồm giá trị bmd từ -3. 21 đến -0. 92, và nhóm 2 từ -0. 92 đến 2. 12. Mỗi nhóm gồm có 5 số.

Tất nhiên, cũng có thể chia thành 3 nhóm bằng lệnh:

```
> group <- cut2(bmd, g=3)
```

Và với lệnh table sẽ biết có 3 nhóm, nhóm 1 gồm 4 số, nhóm 2 và 3 mỗi nhóm có 3 số:

```
> table(group)
group
[-3. 21, -1. 80) [-1. 80, 0. 21) [ 0. 21, 2. 12]
4        3        3
```

7. Các hàm thống kê và đồ thị

7. 1 Các hàm thống kê

7. 1 1 Hàm số thống kê

min(x)	Số nhỏ nhất của biến số x
max(x)	Số lớn nhất của biến số x
which. max(x)	Tìm dòng nào có giá trị lớn nhất của biến số x
which. min(x)	Tìm dòng nào có giá trị nhỏ nhất của biến số x
sum(x)	Số tổng của biến số x
range(x)	Khác biệt giữa max(x) va min(x)
mean(x)	Số trung bình của biến số x
median(x)	Số trung vị(median) của biến số x
sd(x)	Độ lệch chuẩn(standard deviation) của biến số x
var(x)	Phương sai(variance) của biến số x

7. 1. 2 Phân phối thống kê

pnorm(x, mean, sd)	Phân phối chuẩn
plnorm(x, mean, sd)	Phân phối chuẩn logarit
pt(x, df)	Phân phối t
pf(x, n1, n2)	Phân phối F

7. 1. 3 Phân tích thống kê

t. test	Kiểm định t
pairwise. t. test	Kiểm định t cho paired design
var. test	Kiểm định phương sai
bartlett. test	Kiểm định nhiều phương sai
wilcoxon. test	Kiểm định Wilcoxon
kruskal. test	Kiểm định Kruskal
friedman. test	Kiểm định Friedman
lm(y ~ x)	Phân tich hồi qui tuyến tính(linear regression)

7. 2 Đồ thị

7. 2. 1 Một số hàm vẽ đồ thị

plot(y~x)	Vẽ đồ thị y va x(scatter plot)
hist(x)	Vẽ đồ thị y va x(scatter plot)
plot(y ~ x | z)	Vẽ hai biểu đồ x va y theo từng nhóm của z
pie(x)	Vẽ đồ thị tròn
boxplot(x)	Vẽ đồ thị theo dạng hình hộp
qqnorm(x)	Vẽ phân phối quantile của biến số x
qqplot(x, y)	Vẽ phân phối quantile của biến số y theo x
barplot(x)	Vẽ biểu đồ hình khối cho biến số x
hist(x)	Vẽ histogram cho biến số x
stars(x)	Vẽ biểu đồ sao cho biến số x
abline(a, b)	Vẽ đường thẳng với intercept=a va slope=b
abline(h=y)	Vẽ đường thẳng ngang
abline(v=x)	Vẽ đường thẳng đứng
abline(lm. object)	Vẽ đồ thị theo mô hinh tuyến tính

7. 2. 2 Một số thông số cho đồ thị

pch	Kí hiệu để vẽ đồ thị(pch = *p*lotting *ch*aracters)
mfrow, mfcol	Tạo ra nhiều cửa sổ để vẽ nhiều đồ thị cùng một lúc(*m*ulti*fr*ame)
xlim, ylim	Cho giới hạn của trục hòanh và trục tung
xlab, ylab	Viết tên trục hòanh và trục tung
lty, lwd	Dạng và kích thước của đường biểu diễn
cex, mex	Kích thước và khoảng cách giữa các kí tự.
col	Màu sắc

Chương 2: Khai Phá Dữ Liệu

Hiện nay trên sách báo, trong các cuộc hội thảo, tiếp thị sản phẩm ứng dụng công nghệ thông tin, người ta nói rất nhiều về *khai phá dữ liệu* hay có người còn gọi là đào mỏ dữ liệu(data mining) . Và chắc chắn trong không ai là không từng một lần được nghe thấy từ này. Vậy *Khai phá dữ liệu* là gì? Và tại sao lại có nhiều người lại nói đến vấn đề này trong cả công nghiệp máy tính lẫn trong hoạt động kinh doanh đến như vậy?

2. 1 Khai phá dữ liệu là gì

2. 1. 1Khái niệm

Khai phá dữ liệu là một khái niệm ra đời vào những năn cuối của thập kỷ 80. Nó bao hàm một loạt các kỹ thuật nhằm phát hiện ra các thông tin có giá trị tiềm ẩn trong các tập dữ liệu lớn(các kho dữ liệu) . Về bản chất, khai phá dữ liệu liên quan đến việc phân tích các dữ liệu và sử dụng các kỹ thuật để tìm ra các mẫu hình có tính chính quy(regularities) trong tập dữ liệu.

Năm 1989, Fayyad, Piatestsky-Shapiro và Smyth đã dùng khái niệm *Phát hiện tri thức trong cơ sở dữ liệu*(Kownledge Discovery in Database – KDD) để chỉ toàn bộ quá trình phát hiện các tri thức có ích từ các tập dữ liệu lớn. Trong đó, *khai phá dữ liệu* là một bước đặc biệt trong toàn bộ quá trình, sử dụng các giải thuật đặc biệt để chiết xuất ra các mẫu(pattern)(hay các mô hình) từ dữ liệu.

2. 1. 2Các bước của quá trình khai phá dữ liệu

Các giải thuật khai phá dữ liệu thường được mô tả như những chương trình hoạt động trực tiếp trên tệp dữ liệu. Với các phương pháp học máy và thống kê trước đây, thường thì bước đầu tiên là các giải thuật nạp toàn bộ tệp dữ liệu vào trong bộ nhớ. Khi chuyển sang các ứng dụng công nghiệp liên quan đến việc khai phá các kho dữ liệu lớn, mô hình này không thể đáp ứng được. Không chỉ bởi vì nó không thể nạp hết dữ liệu vào trong bộ nhớ mà còn vì khó có thể chiết xuất dữ liệu ra các tệp đơn giản để phân tích được.

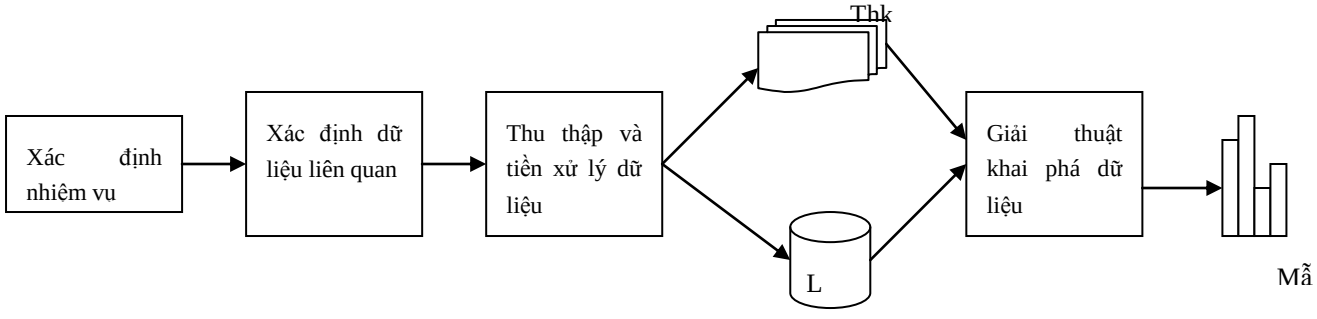
Quá trình xử lý khai phá dữ liệu bắt đầu bằng cách xác định chính xác vấn đề cần giải quyết. Sau đó sẽ xác định các dữ liệu liên quan dùng để xây dựng giải pháp. Bước tiếp theo là thu thập các dữ liệu có liên quan và xử lý chúng thành dạng sao cho giải thuật khai phá dữ liệu có thể hiểu được. Về lý thuyết thì có vẻ rất đơn giản nhưng khi

thực hiện thì đây thực sự là một quá trình rất khó khăn, gặp phải rất nhiều vướng mắc như: các dữ liệu phải được sao ra nhiều bản(nếu được chiết xuất vào các tệp) , quản lý tập các tệp dữ liệu, phải lặp đi lặp lại nhiều lần toàn bộ quá trình(nếu mô hình dữ liệu thay đổi) , v. v…

Sẽ là quá cồng kềnh với một giải thuật khai phá dữ liệu nếu phải truy nhập vào toàn bộ nội dung của CSDL và làm những việc như trên. Vả lại, điều này cũng không cần thiết. Có rất nhiều giải thuật khai phá dữ liệu thực hiện dựa trên những thống kê tóm tắt khá đơn giản của CSDL, khi mà toàn bộ thông tin trong CSDL là quá dư thừa đối với mục đích của việc khai phá dữ liệu.

Bước tiếp theo là chọn thuật toán khai phá dữ liệu thích hợp và thực hiện việc khai phá dữ liệu để tìm được các mẫu(pattern) có ý nghĩa dưới dạng biểu diễn tương ứng với các ý nghĩa đó(thường được biểu diễn dưới dạng các luật xếp loại, cây quyết định, luật sản xuất, biểu thức hồi quy, …) .

Đặc điểm của mẫu phải là mới(ít nhất là đối với hệ thống đó) . Độ mới có thể được đo tương ứng với độ thay đổi trong dữ liệu(bằng cách so sánh các giá trị hiện tại với các giá trị trước đó hoặc các giá trị mong muốn) , hoặc bằng tri thức(mối liên hệ giữa phương pháp tìm mới và phương pháp cũ như thế nào) . Thường thì độ mới của mẫu được đánh giá bằng một hàm logic hoặc một hàm đo độ mới, độ bất ngờ của mẫu. Ngoài ra, mẫu còn phải có khả năng sử dụng tiềm tàng. Các mẫu này sau khi được xử lý và diễn giải phải dẫn đến những hành động có ích nào đó được đánh giá bằng một hàm lợi ích. Ví dụ như trong dữ liệu các khoản vay, hàm lợi ích đánh giá khả năng tăng lợi nhuận từ các khoản vay. Mẫu khai thác được phải có giá trị đối với các dữ liệu mới với độ chính xác nào đó.



Hình 2. 1. Quá trình khai phá dữ liệu.

Với các giải thuật và các nhiệm vụ của khai phá dữ liệu rất khác nhau, dạng của các mẫu chiết xuất được cũng rất đa dạng. Theo cách đơn giản nhất, sự phân tích cho ra

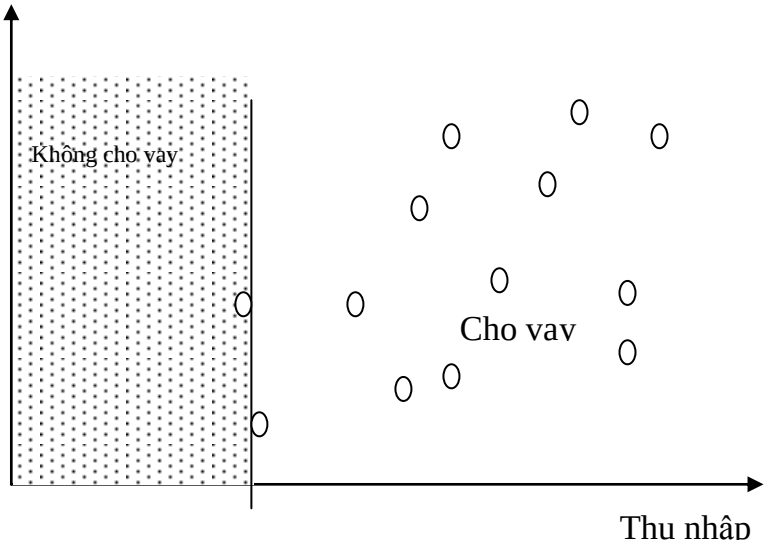
kết quả chiết xuất là một báo cáo về một số loại(có thể bao gồm các phép đo mang tính thống kê về độ phù hợp của mô hình, các dữ liệu lạ, v. v…) . Trong thực tế đầu ra phức tạp hơn nhiều, mẫu chiết xuất được có thể là một mô tả xu hướng, có thể là dưới dạng văn bản, một đồ thị mô tả các mối quan hệ trong mô hình, cũng có thể là một hành động, ví dụ như yêu cầu người dùng làm gì với những gì khai thác được trong dữ liệu. Một mẫu chiết xuất được từ một công cụ khai phá tri thức khác lại có thể là một dự đoán x số lượng bánh kẹo bán ra vào dịp Tết sẽ tăng lên bao nhiêu phần trăm, v. v… Hình 2. 2 là một ví dụ minh họa kết quả của việc khai phá dữ liệu khách hàng xin vay vốn, với một lựa chọn t, mẫu chiết xuất được là một luật "*Nếu thu nhập < t đồng thì khách hàng vay bị vỡ nợ*".

Dạng của mẫu chiết xuất được có thể được phân loại bởi kiểu mẫu dữ liệu mà nó mô tả. Các mẫu liên vùng(interfield pattern) liên quan đến các giá trị của các trường trong cùng một bản ghi(ví dụ: Nếu thủ tục=phẫu thuật thì ngày nằm viện>5) . Các mẫu liên bản ghi liên quan đến các giá trị được tổng hợp từ một nhóm các bản ghi ví dụ như bệnh nhân mắc bệnh đau dạ dày khó ăn gấp hai lần những người bình thường khác; hoặc xác định những phần có ích ví dụ như nhóm các công ty có lợi nhuận. Việc khai thác các mẫu liên bản ghi là dạng tổng kết dữ liệu. Đối với dữ liệu phụ thuộc thời gian, mối quan hệ liên bản ghi có thể cũng xác định các xu hướng quan tâm(ví dụ như sản lượng bán hàng tăng 20% so với năm ngoái) .

Ta cũng có thể phân loại dạng mẫu chiết xuất được theo khả năng mô tả của chúng. Ví dụ như mẫu chiết xuất được của quá trình khai phá dữ liệu theo số lượng liên quan đến các giá trị trường số sử dụng các công thức toán học. Mẫu của quá trình khai phá dữ liệu theo chất lượng tìm ra một mối quan hệ logic giữa các trường. Ta phân biệt hai dạng này vì các kỹ thuật khai phá khác nhau thường được sử dụng trong các trường hợp khác nhau. Ví dụ như các mối quan hệ số lượng tuyến tính tìm thấy rất dễ dàng bằng các phương pháp hồi quy tuyến tính trong khi khai phá theo định tính lại không thể dùng được các phương pháp này.

Kỹ thuật khai phá dữ liệu thực chất không có gì mới. Nó là sự kế thừa, kết hợp và mở rộng của các kỹ thuật cơ bản đã được nghiên cứu từ trước như học máy, nhận dạng, thống kê(hồi quy, xếp loại, phân nhóm) , các mô hình đồ thị, các mạng Bayes, trí tuệ nhân tạo, thu thập tri thức hệ chuyên gia, v. v… Tuy nhiên, với sự kết hợp tài tình của khai phá dữ liệu, kỹ thuật này có ưu thế hơn hẳn các phương pháp trước đó, đ lại nhiều triển vọng trong việc ứng dụng phát triển nghiên cứu khoa học cũng như làm tăng mức lợi nhuận trong các hoạt động kinh doanh.

2. 1. 3Ví dụ minh họa



Để minh họa hoạt động cũng như mẫu chiết xuất được của quá trình khai phá dữ liệu, sẽ dùng chủ yếu một ví dụ đơn giản như đã cho trên Hình 2. 2. Hình 2. 2 mô tả một tập dữ liệu hai chiều gồm có 23 điểm mẫu. Mỗi điểm biểu thị cho một khách hàng đã vay ngân hàng. Trục hoành biểu thị cho thu nhập, trục tung biểu thị cho tổng dư nợ của khách hàng. Dữ liệu khách hàng được chia thành hai lớp: dấu x biểu thị cho khách hàng bị vỡ nợ, dấu o biểu thị cho khách hàng có khả năng trả nợ. Tập dữ liệu này có thể chứa những thông tin có ích đối với các tổ chức tín dụng trong việc ra quyết định có cho khách hàng vay nữa không. Ví dụ như ta có mẫu "*Nếu thu nhập < t đồng thì khách hàng vay sẽ bị vỡ nợ*" như mô tả trên Hình 2. 2.

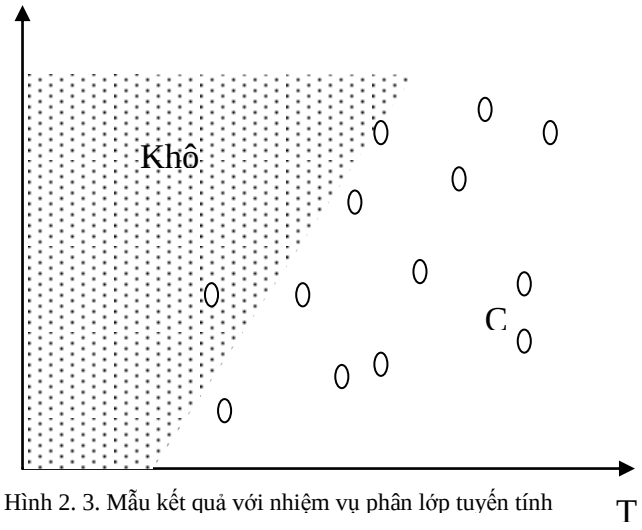
2. 2 Nhiệm vụ chính của Khai phá dữ liệu

Rõ ràng rằng mục đích của khai phá dữ liệu là các tri thức chiết xuất được sẽ được sử dụng cho lợi ích cạnh tranh trên thương trường và các lợi ích trong nghiên cứu khoa học.

Do đó, ta có thể coi mục đích chính của khai thác dữ liệu sẽ là *mô tả*(description) và dự đoán(prediction) . Các mẫu mà khai phá dữ liệu phát hiện được nhằm vào mục đích này. *Dự đoán* liên quan đến việc sử dụng các biến hoặc các trường trong cơ sở dữ liệu để chiết xuất ra các mẫu là các dự đoán những giá trị chưa biết hoặc những giá trị trong tương lai của các biến đáng quan tâm. *Mô tả* tập trung vào việc tìm kiếm các mẫu mô tả dữ liệu mà con người có thể hiểu được.

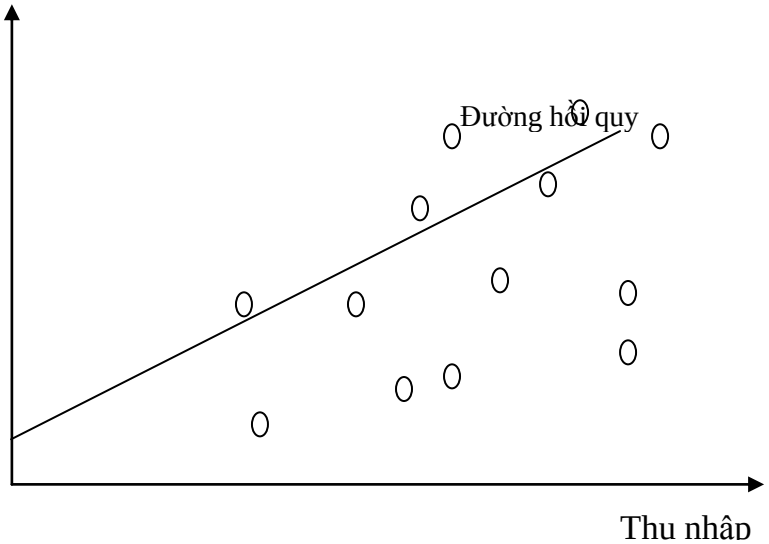
Để đạt được hai mục đích này, nhiệm vụ chính của khai phá dữ liệu bao gồm như sau:

- *Phân lớp(Classification) :* Phân lớp là việc học một hàm ánh xạ(hay phân loại) một mẫu dữ liệu vào một trong số các lớp đã xác định(Hand 1981; Weiss & Kulikowski 1991; McLachlan 1992) . Ví dụ về việc sử dụng phương pháp phân lớp trong khai phá dữ liệu là ứng dụng phân lớp các xu hướng trong thị trường tài chính(Apte. & Hong) và ứng dụng tự động xác định các đối tượng đáng quan tâm trong các cơ sở dữ liệu ảnh lớn(Fayyad, Djorgovski, & Weir) . Hình 2. 3 mô tả đầu ra của nhiệm vụ khai phá dữ liệu phân lớp đối với tập dữ liệu khách hàng đã nêu trên. Đó là một mẫu chia tập dữ liệu khách hàng thành hai miền tuyến tính. Mẫu này có thể sẽ cho phép tổ chức tín dụng quyết định có cho các khách hàng vay hay không.



Hình 2. 3. Mẫu kết quả với nhiệm vụ phân lớp tuyến tính

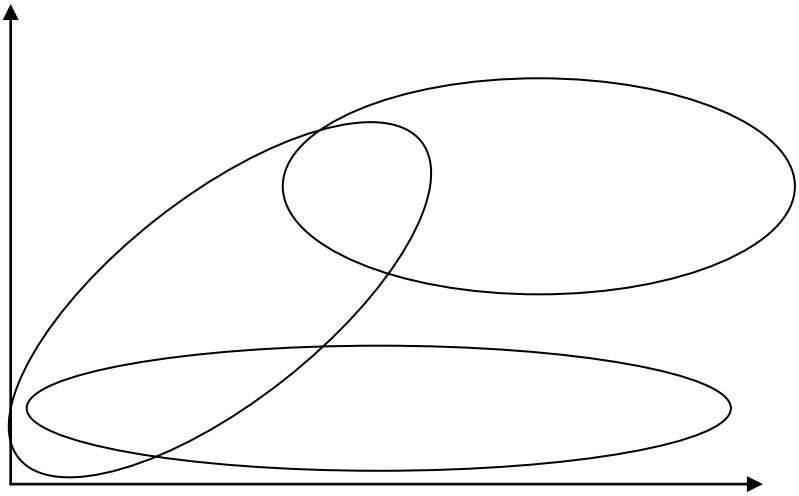
- *Hồi quy(Regression) :* Hồi quy là việc học một hàm ánh xạ từ một mẫu dữ liệu thành một biến dự đoán có giá trị thực. Có rất nhiều ứng dụng khai phá dữ liệu với nhiệm vụ hồi quy, ví dụ như dự đoán số lượng biomass xuất hiện trong rừng biết các phép đo vi sóng từ xa, đánh giá khả năng tử vong của bệnh nhân biết các kết quả xét nghiệm chuẩn đoán, dự đoán nhu cầu tiêu thụ một sản phẩm mới bằng một hàm chỉ tiêu quảng cáo, dự đoán theo thời gian với các biến đầu vào là các giá trị của mẫu dự đoán trong quá khứ, v. v… Hình 2. 4 mô tả mẫu kết quả dự đoán tổng dư nợ của khách hàng với nhiệm vụ khai phá dữ liệu là hồi quy. Đường hồi quy tuyến tính cho thấy những



Hình 2. 3. Mẫu kết quả với nhiệm vụ hồi quy

khách hàng có thu nhập càng cao thì tổng dư nợ càng lớn. Mẫu kết quả này không phù hợp với quy luật và điều đó là dễ hiểu vì ta thấy đường hồi quy tuyến tính ở đây không vét cạn được hết các trường hợp xảy ra mà chỉ mô tả được mối liên hệ của một số rất ít khách hàng.

- *Phân nhóm(Clustering) :* Là việc mô tả chung để tìm ra các tập xác định các nhóm hay các loại để mô tả dữ liệu(Titterington, Smith & Makov 1985; Jain & Dubes 1988) . Các nhóm có thể tách riêng nhau hoặc phân cấp hoặc gối lên nhau. Có nghĩa là một dữ liệu có thể vừa thuộc nhóm này, vừa thuộc nhóm kia. Các ứng dụng khai phá dữ liệu có nhiệm vụ phân nhóm như: phát hiện tập các khách hàng có phản ứng giống nhau trong cơ sở dữ liệu tiếp thị, xác định các loại quang phổ từ các phương pháp đo tia hồng ngoại(Cheesan & Stutz) . Hình 2. 5 mô tả các mẫu của quá trình khai phá dữ liệu với nhiệm vụ phân nhóm. Ở đây, các mẫu là các nhóm khách hàng được xếp thành ba nhóm gối lên nhau. Các điểm nằm trong cả hai nhóm chứng tỏ khách hàng có thể thuộc cả hai loại trạng thái. Chú ý rằng với nhiệm vụ này, khách hàng không được phân biệt như cũ nữa(không dùng các dấu x và o) mà được phân biệt theo nhóm(thay bằng dấu +) . Liên quan chặt chẽ đến việc phân nhóm là nhiệm vụ đánh giá mật độ xác suất, bao gồm các kỹ thuật đánh giá dữ liệu, hàm mật độ xác suất đa biến liên kết của tất cả các biến/các trường trong cơ sở dữ liệu(Silverman 1986) .



Hình 2. 3. Mẫu kết quả với nhiệm vụ phân nhóm

- *Tóm tắt(summarization) :* Liên quan đến các phướng pháp tìm kiếm một mô tả tóm tắt cho một tập con dữ liệu. Ví dụ như việc lập bảng các độ lệch chuẩn và trung bình cho tất cả các trường. Các phương pháp phức tạp hơn liên quan đến nguồn gốc của các luật tóm tắt(Agrawal et al.) , khai thác

mối liên hệ hàm giữa các biên(Zbowicz & Zytkow) . Các kỹ thuật tóm tắt thường được áp dụng cho các phân tích dữ liệu tương tác có tính thăm dò và tạo báo cáo tự động.

- *Mô hình hóa phụ thuộc(Dependency Modeling) :* Bao gồm việc tìm kiếm một mô hình mô tả sự phụ thuộc đáng kể giữa các biến. Các mô hình phụ thuộc tồn tại dưới hai mức: mức cấu trúc của mô hình xác định(thường ở dạng đồ họa) các biến nào là phụ thuộc cục bộ với nhau, mức định lượng của một mô hình xác định độ mạnh của sự phụ thuộc theo một thước đo nào đó. Ví dụ như các mạng phụ thuộc xác suất sử dụng độc lập có điều kiện để xác định khía cạnh có cấu trúc của một mô hình và các xác suất hoặc tương quan để xác định độ mạnh của sự phụ thuộc(Heckerman; Glymour et al. , 1987) . Các mạng phụ thuộc xác suất đang ngày càng tìm thấy nhiều ứng dụng trong các lĩnh vực khác nhau như phát triển các hệ chuyên gia y tế áp dụng tính xác suất từ các cơ sở dữ liệu, thu thập thông tin, mô hình hóa gen di truyền của người.
- *Phát hiện sự thay đổi và lạc hướng(Change and Deviation Detection)) :* Tập trung vào khai thác những thay đổi đáng kể nhất trong dữ liệu từ các giá trị chuẩn hoặc được đo trước đó(Berndt & Cliffort; Guyon et al. ; Klosegen; Matheus et al. ; Basseville & Nikiforov 1993) .

Vì các nhiệm vụ khác nhau này yêu cầu số lượng và các dạng thông tin rất khác nhau nên chúng thường ảnh hưởng đến việc thiết kế và chọn giải thuật khai phá dữ liệu khác nhau. Ví dụ như giải thuật tạo cây quyết định tạo ra được một mô tả phân biệt được các mẫu giữa các lớp nhưng không có các tính chất và đặc điểm của lớp.

2. 3 Các phương pháp Khai phá dữ liệu

Quá trình khai phá dữ liệu là quá trình phát hiện mẫu, trong đó, giải thuật khai phá dữ liệu tìm kiếm các mẫu đáng quan tâm theo dạng xác định như các luật, cây phân lớp, quy hồi, phân nhóm, v. v…

2. 3. 1 Các thành phần của giải thuật khai phá dữ liệu

Giải thuật khai phá dữ liệu bao gồm 3 thành phần chính như sau: biểu diễn mô hình, đánh giá mô hình, tìm kiếm mô hình.

- *Biểu diễn mô hình:* Mô hình được biểu diễn bằng một ngôn ngữ L để mô tả các mẫu có thể khai thác được. Nếu sự mô tả quá bị hạn chế thì sẽ không thể học được hoặc sẽ không thể có các mẫu tạo ra được một mô

hình chính xác cho dữ liệu. Ví dụ một mô tả cây quyết định sử dụng phân chia các nút theo trường đơn, chia không gian đầu vào thành các mặt siêu phẳng song song với các trục thuộc tính. Phương pháp cây quyết định như vậy không thể khai thác được từ dữ liệu dạng công thức x=y dù cho tập học có to đến đâu đi nữa. Vì vậy, việc quan trọng là người phân tích dữ liệu cần phải hiểu đầy đủ các giả thiết mô tả. Một điều cũng khá quan trọng là người thiết kế giải thuật cần phải diễn tả được các giả thiết mô tả nào được tạo ra bởi giải thuật nào. Khả năng mô tả mô hình càng lớn thì càng làm tăng mức độ nguy hiểm do bị học quá và làm giảm đi khả năng dự đoán các dữ liệu chưa biết. Hơn nữa, việc tìm kiếm sẽ càng trở nên phức tạp hơn và việc giải thích mô hình cũng khó khăn hơn.

Mô hình ban đầu được xác định bằng cách kết hợp biến đầu ra(phụ thuộc) với các biến độc lập mà biến đầu ra phụ thuộc vào. Sau đó phải tìm những tham số mà bài toán cần tập trung giải quyết. Việc tìm kiếm mô hình sẽ đưa ra được một mô hình phù hợp với các tham số được xác định dựa trên dữ liệu(trong một số trường hợp, mô hình được xây dựng độc lập với dữ liệu trong khi đối với một số trường hợp khác thì mô hình và các tham số lại thay đổi để phù hợp với dữ liệu) . Trong một số trường hợp, tập dữ liệu được chia thành tập dữ liệu học và tập dữ liệu thử. Tập dữ liệu học được sử dụng để làm cho các tham số của mô hình phù hợp với dữ liệu. Mô hình sau đó sẽ được đánh giá bằng cách đưa các dữ liệu thử vào mô hình và thay đổi lại các tham số cho phù hợp nếu cần. Mô hình lựa chọn có thể là phương pháp thống kê như SASS, v. v…, một số giải thuật học máy(ví dụ như suy diễn cây quyết định và các kỹ thuật học có thầy khác) , mạng neuron, suy diễn hướng tình huống(case-based reasoning) , các kỹ thuật phân lớp.

- *Đánh giá mô hình:* Đánh giá x một mẫu có đáp ứng được các tiêu chuẩn của quá trình phát hiện tri thức hay không. Việc đánh giá độ chính xac dự đoán dựa trên đánh giá chéo(cross validation) . Đánh giá chất lượng mô tả liên quan đến độ chính xác dự đoán, độ mới, khả năng sử dụng, khả năng hiểu được của mô hình. Cả hai chuẩn thống kê và chuẩn logic đều có thể được sử dụng để đánh giá mô hình. Ví dụ như luật xác suất lớn nhất có thể dùng để lựa chọn các tham số cho mô hình sao cho xử lý phù hợp nhất với tập dữ liệu học. Việc đánh giá mô hình được thực hiện qua kiểm tra dữ liệu(trong một số trường hợp kiểm tra với tất cả các dữ liệu, trong một số

trường hợp khác chỉ kiểm tra với dữ liệu thử) . Ví dụ như đối với mạng neuron, việc đánh giá mô hình được thực hiện dựa trên việc kiểm tra dữ liệu(bao gồm cả dữ liệu học và dữ liệu thử) , đối với nhiệm vụ dự đoán thì việc đánh giá mô hình ngoài kiểm tra dữ liệu còn dựa trên độ chính xác dự đoán.

- *Phương pháp tìm kiếm:* phương pháp tìm kiếm bao gồm hai thành phần: tìm kiếm tham số và tìm kiếm mô hình. Trong tìm kiếm tham số, giải thuật cần tìm kiếm các tham số để tối ưu hóa các tiêu chuẩn đánh giá mô hình với các dữ liệu quan sát được và với một mô tả mô hình đã định. Việc tìm kiếm không cần thiết đối với một số bài toán khá đơn giản: các đánh giá tham số tối ưu có thể đạt được bằng các cách đơn giản hơn. Đối với các mô hình chung thì không có các cách này, khi đó giải thuật "tham lam" thường được sử dụng lặp đi lặp lại. Ví dụ như phương pháp giảm gradient trong giải thuật lan truyền ngược(backpropagation) cho các mạng neuron. Tìm kiếm mô hình xảy ra giống như một vòng lặp qua phương pháp tìm kiếm tham số: mô tả mô hình bị thay đổi tạo nên một họ các mô hình. Với mỗi một mô tả mô hình, phương pháp tìm kiếm tham số được áp dụng để đánh giá chất lượng mô hình. Các phương pháp tìm kiếm mô hình thường sử dụng các kỹ thuật tìm kiếm heuristic vì kích thước của không gian các mô hình có thể thường ngăn cản các tìm kiếm tổng thể, hơn nữa các giải pháp đơn giản(closed form) không dễ đạt được.

2. 3. 2 Một số phương pháp khai thác dữ liệu phổ biến

2. 3. 2. 1 Phương pháp quy nạp(induction)

Một cơ sở dữ liệu là một kho thông tin nhưng các thông tin quan trọng hơn cũng có thể được suy diễn từ kho thông tin đó. Có hai kỹ thuật chính để thực hiện việc này là suy diễn và quy nạp.

- *Phương pháp suy diễn:* Nhằm rút ra thông tin là kết quả logic của các thông tin trong cơ sở dữ liệu. Ví dụ như toán tử liên kết áp dụng cho bảng quan hệ, bảng đầu chứa thông tin về các nhân viên và phòng ban, bảng thứ hai chứa các thông tin về các phòng ban và các trưởng phòng. Như vậy sẽ suy ra được mối quan hệ giữa các nhân viên và các trưởng phòng. Phương pháp suy diễn dựa trên các sự kiện chính xác để suy ra các tri thức mới từ các thông tin cũ. Mẫu chiết xuất được bằng cách sử dụng phương pháp này thường là các luật suy diễn. Với tập dữ liệu khách hàng vay vốn ở

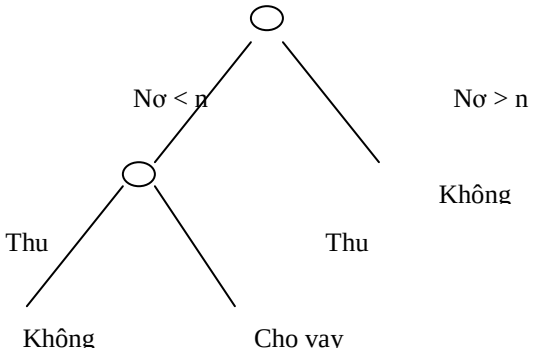
trên, ta có mẫu chiết xuất được với ngưỡng thu nhập t là một luật như sau: "*Nếu thu nhập của khách hàng lớn hơn t đồng thì khách hàng có khả năng trả nợ*".

- *Phương pháp quy nạp:* phương pháp quy nạp suy ra các thông tin được sinh ra từ cơ sở dữ liệu. Có nghĩa là nó tự tìm kiếm, tạo mẫu và sinh ra tri thức chứ không phải bắt đầu với các tri thức đã biết trước. Các thông tin mà phương pháp này đ lại là các thông tin hay các tri thức cấp cao diễn tả về các đối tượng trong cơ sở dữ liệu. Phương pháp này liên quan đến việc tìm kiếm các mẫu trong CSDL.

Trong khai phá dữ liệu, quy nạp được sử dụng trong cây quyết định và tạo luật.

2. 3. 2. 2 Cây quyết định và luật

- *Cây quyết định:* Cây quyết định là một mô tả tri thức dạng đơn giản nhằm phân các đối tượng dữ liệu thành một số lớp nhất định. Các nút của cây được gán nhãn là tên các thuộc tính, các cạnh được gán các giá trị có thể của các thuộc tính, các lá mô tả các lớp khác nhau. Các đối tượng được phân lớp theo các đường đi trên cây, qua các cạnh tương ứng với các giá trị của thuộc tính của đối tượng tới lá. Hình 2. 6 mô tả một mẫu đầu ra có thể của quá trình khai phá dữ liệu dùng phương pháp cây quyết định với tập dữ liệu khách hàng xin vay vốn.



Hình 2. 6. Mẫu kết quả với phương pháp cây

- *Tạo luật:* Các luật được tạo ra nhằm suy diễn một số mẫu dữ liệu có ý nghĩa về mặt thống kê. Các luật có dạng NẾU P THÌ Q, với P là mệnh đề đúng với một phần trong CSDL, Q là mệnh đề dự đoán. Ví dụ ta có một mẫu phát hiện được bằng phương pháp tạo luật: nếu giá 1 cân táo thấp hơn 5000 đồng thì số lượng táo bán ra sẽ tăng 5%. Những luật như thế này

được sử dụng rất rộng rãi trong việc mô tả tri thức trong hệ chuyên gia. Chúng có thuận lợi là dễ hiểu đối với người sử dụng.

Cây quyết định và luật có ưu điểm là hình thức mô tả đơn giản, mô hình suy diễn khá dễ hiểu đối với người sử dụng. Tuy nhiên, giới hạn của nó là mô tả cây và luật chỉ có thể biểu diễn được một số dạng chức năng và vì vậy giới hạn về cả độ chính xác của mô hình. Mẫu ví dụ như trong Hình 2. 2 cho thấy ảnh hưởng của một ngưỡng áp dụng cho biến thu nhập đối với tập dữ liệu khách hàng vay vốn. Rõ ràng việc sử dụng một ngưỡng đơn giản như thế đã hạn chế việc phân lớp với đường biên chính xác hơn mà ta có thể nhìn thấy được. Nếu mở rộng không gian của mô hình để cho phép có nhiều mô tả hơn(ví dụ như các mặt siêu phẳng đa biến(multivariate hyperplane) tại các góc ngẫu nhiên) thì mô hình sẽ dự đoán tốt hơn nhưng lại rất khó hiểu. Cho đến nay, đã có rất nhiều giải thuật suy diễn sử dụng các luật và cây quyết định được áp dụng trong học máy và trong thống kê(Breiman et al. 1984; Quinlan 1992) .

Đối với quy mô lớn, người ta dựa trên các phương pháp đánh giá mô hình theo xác suất với các mức độ mô hình phức tạp khác nhau. Các phương pháp tìm kiếm "tham lam", liên quan đến việc tăng và rút gọn các luật và các cấu trúc cây, chủ yếu được sử dụng để khai thác không gian siêu mũ(super-exponential space) của các mô hình. Cây và luật chủ yếu được sử dụng cho việc mô hình hóa dự đoán, phân lớp(Apte & Hong; Fayyad, Djorgovski, & Wei) và hồi quy. Chúng cũng có thể được áp dụng cho việc tóm tắt và mô hình hóa các mô tả(Agrawal et al.) .

2. 3. 2. 3 Phát hiện các luật kết hợp

Phương pháp này nhằm phát hiện ra các luật kết hợp giữa các thành phần dữ liệu trong cơ sở dữ liệu. Mẫu đầu ra của giải thuật khai phá dữ liệu là tập luật kết hợp tìm được. Ta có thể lấy một ví dụ đơn giản về luật kết hợp như sau: sự kết hợp giữa hai thành phần A và B có nghĩa là sự xuất hiện của A trong bản ghi kéo theo sự xuất hiện của B trong cùng bản ghi đó: $A \Rightarrow B$.

Cho một lược đồ $R=\{A_1, \ldots, A_p\}$ các thuộc tính với miền giá trị {0, 1}, và một quan hệ r trên R. Một luật jết hợp trên r được mô tả dưới dạng $X \Rightarrow B$ với $X \subseteq R$ và $B \in R \backslash X$. Về mặt trực giác, ta có thể phát biểu ý nghĩa của luật như sau: nếu một bản ghi của bảng r có giá trị 1 tại mỗi thuộc tính thuộc X thì giá trị của thuộc tính B cũng là 1 trong cùng bản ghi đó. Ví dụ như ta có tập cơ sở dữ liệu về các mặt hàng bán trong siêu thị, các dòng tương ứng với các ngày bán hàng, các cột tương ứng với các mặt hàng thì giá trị 1 tại ô(20/10, bánh mì) xác định rằng bánh mì đã bán ngày hôm đó cũng kéo theo sự xuất hiện giá trị 1 tại ô(20/10, bơ) .

Cho W $\subseteq$R, đặt s(W, r) là tần số xuất hiện của W trong r được tính bằng tỷ lệ của các hàng trong r có giá trị 1 tại mỗi cột thuộc W. Tần số xuất hiện của luật X$\Rightarrow$B trong r được định nghĩa là s(X$\cup${B}, r) còn gọi là độ hỗ trợ của luật, độ tin cậy của luật là s(X$\cup${B}, r) /s(X, r) . Ở đây X có thể gồm nhiều thuộc tính, B là giá trị không cố định. Nhờ vậy mà không xảy ra việc tạo ra các luật không mong muốn trước khi quá trìm tìm kiếm bắt đầu. Điều đó cũng cho thấy không gian tìm kiếm có kích thước tăng theo hàm mũ của số lượng các thuộc tính ở đầu vào. Do vậy cần phải chú ý khi thiết kế dữ liệu cho việc tìm kiếm các luật kết hợp.

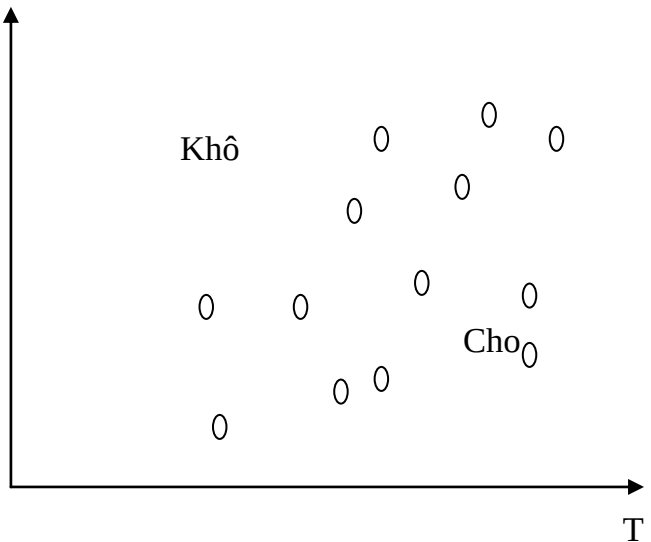
Nhiệm vụ của việc phát hiện các luật kết hợp là phải tìm tất cả các luật X$\Rightarrow$B sao cho tần số của luật không nhỏ hơn ngưỡng σ cho trước và độ tin cậy của luật không nhỏ hơn ngưỡng θ cho trước. Từ một cơ sở dữ liệu ta có thể tìm được hàng nghìn và thậm chí hàng trăm nghìn các luật kết hợp.

Ta gọi một tập con X$\subseteq$R là *thường xuyên* trong r nếu thỏa mãn điều kiện s(X, r) $\geq\sigma$. Nếu biết tất cả các tập *thường xuyên* trong r thì việc tìm kiếm các luật rất dễ dàng. Vì vậy, giải thuật tìm kiếm các luật kết hợp trước tiên đi tìm tất cả các tập *thường xuyên* này, sau đó tạo dựng dần các luật kết hợp bằng cách ghép dần các tập thuộc tính dựa trên mức độ thường xuyên.

Các luật kết hợp có thể là một cách hình thức hóa đơn giản. Chúng rất thích hợp cho việc tạo ra các kết quả có dữ liệu dạng nhị phân. Giới hạn cơ bản của phương pháp này là ở chỗ các quan hệ cần phải thưa theo nghĩa không có tập *thường xuyên*nào chứa nhiều hơn 15 thuộc tính. Giải thuật tìm kiếm các luật kết hợp tạo ra số luật ít nhất phải bằng với số các tập *thường xuyên* và nếu như một tập *thường xuyên* có kích thước K thì phải có ít nhất là 2^K tập *thường xuyên*. Thông tin về các tập *thường xuyên* được sử dụng để ước lượng độ tin cậy của các tập luật kết hợp.

2. 3. 2. 4 Các phương pháp phân lớp và hồi quy phi tuyến

Các phương pháp này bao gồm một họ các kỹ thuật dự đoán để làm cho các kết hợp tuyến tính và phi tuyến của các hàm cơ bản(hàm sigmoid, hàm spline(hàm mành) , hàm đa thức) phù hợp với các kết hợp của các giá trị biến vào. Các phương pháp thuộc loại này như mạng neuron truyền thẳng, phương pháp mành



thích nghi, v. v…(Friedman 1989, Cheng & Titterington 1994, Elder & Pregibon) . Mẫu minh họa trên Hình 2. 7 mô tả một dạng đường biên phi tuyến mà mạng neuron tìm ra từ tập dữ liệu khách hàng vay. Xét về mặt đánh giá mô hình, mặc dù mạng neuron với kích thước tương đối hầu như lúc nào cũng có thể mô phỏng bất kỳ hàm nào gần đúng với một độ chính xác mong muốn nào đó. Nhưng để tìm được một mạng có kích thước tối ưu cho một tập dữ liệu xác định lại là một việc khá công phu và không ai có thể biết chắc có tìm ra được kích thước đó hay không. Các phương pháp sai số bình phương chuẩn(standard squared error) và các hàm entropy(cross entropy loss function) được sử dụng để học có thể được x như các hàm khả năng logarit(log-likelihood functions) khi phân lớp và hồi quy(Gan, Bienentstock & Doursat 1992; Ripley 1994) . Lan truyền ngược sai số là một phương pháp tìm kiếm tham số thực hiện việc giảm gradient trong không gian tham số(ở đây là các trọng số) để tìm một giá trị cực đại cục bộ của hàm xác suất bắt đầu từ các giá trị khởi tạo ngẫu nhiên. Các phương pháp hồi quy phi tuyến mặc dù rất có khả năng diễn tả nhưng lại rất khó diễn giải thành các luật. Ví dụ như đường biên phân lớp mô tả trong Hình 2. 6 chính xác hơn đường biên đơn giản dựa trên ngưỡng như mẫu trên Hình 2. 2 nhưng đường biên dựa trên ngưỡng lại có thuận lợi là mô hình có thể dễ dàng diễn giải thành một luật đơn giản với một độ chính xác nào đó: "*nếu thu nhập của khách hàng lớn hơn t đồng thì có thể cho vay*".

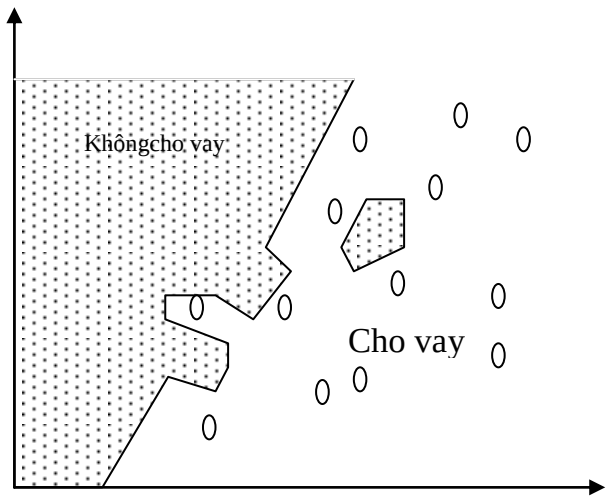
2. 3. 2. 5 Phân nhóm và phân đoạn(clustering and segmentation)

Kỹ thuật phân nhóm và phân đoạn là những kỹ thuật phân chia dữ liệu sao cho mỗi phần hoặc mỗi nhóm giống nhau theo một tiêu chuẩn nào đó. Mối quan hệ thành viên của các nhóm có thể dựa trên mức độ giống nhau của các thành viên và từ đó xây dựng nên các luật ràng buộc giữa các thành viên trong nhóm. Một kỹ thuật phân nhóm khác là xây dựng nên các hàm đánh giá các thuộc tính của các thành phần như là hàm của các tham số của các thành phần. Phương pháp này được gọi là phương pháp phân hoạch tối ưu(optimal partitioning) . Một ví dụ của phương pháp phân nhóm theo độ giống nhau là cơ sở dữ liệu khách hàng, ứng dụng của phương pháp tối ưu ví dụ như phân nhóm khách hàng theo số các tham số và các nhóm thuế tối ưu có được khi thiết lập biểu thuế bảo hiểm.

Mẫu đầu ra của quá trình khai phá dữ liệu sử dụng kỹ thuật này là các tập mẫu chứa các dữ liệu có chung những tính chất nào đó được phân tách từ cơ sở dữ liệu. Khi các mẫu được thiết lập, chúng có thể được sử dụng để tái tạo các tập dữ liệu ở dạng dễ hiểu hơn, đồng thời cũng cung cấp các nhóm dữ liệu cho các hoạt động cũng như công việc phân tích. Đối với cơ sở dữ liệu lớn, việc lấy ra các nhóm này là rất quan trọng.

2. 4 Các phương pháp dựa trên mẫu

Sử dụng các mẫu mô tả từ cơ sở dữ liệu để tạo nên một mô hình dự đoán các mẫu mới bằng cách rút ra những thuộc tính tương tự như các mẫu đã biết trong mô hình. Các kỹ thuật bao gồm phân lớp theo láng giềng gần nhất, các giải thuật hồi quy(Dasarathy 1991) và các hệ thống suy diễn dựa trên tình huống(case-based reasoning)(Kolodner 1993) . Hình 2. 8 minh họa mẫu đầu ra của quá trình khai phá dữ liệu sử dụng kỹ thuật phân lớp theo láng giềng gần nhất đối với tập dữ liệu khách hàng vay vốn. Bất kỳ điểm dữ liệu mới nào nằm gần điểm dữ liệu trong tập học sẽ được xếp chung vào lớp với điểm dữ liệu mẫu đã học đó.



Hình 2. 8. Mẫu kết quả sử dụng kỹ thuật phân lớp theo láng giềng gần nhất

Khuyết điểm của các kỹ thuật này là cần phải xác định được khoảng cách, độ đo giống nhau giữa các mẫu. Mô hình thường được đánh giá bằng phương pháp đánh giá chéo trên các lỗi dự đoán(Weiss & Kulikowski, 1991) . "Tham số" của mô hình được đánh giá có thể bao gồm một số láng giềng dùng để dự đoán và số đo khoảng cách. Giống như phương pháp hồi quy phi tuyến, các phương pháp này khá mạnh trong việc đánh giá xấp xỉ các thuộc tính, nhưng lại rất khó hiểu vì mô hình không được định dạng rõ ràng mà tiềm ẩn trong dữ liệu.

2. 5 Mô hình phụ thuộc dựa trên đồ thị xác suất

Các mô hình đồ thị xác định sự phụ thuộc xác suất giữa các sự kiện thông qua các liên hệ trực tiếp theo các cung đồ thị(Pearl 1988; Whittaker, 1990) . Ở dạng đơn giản nhất, mô hình này xác định những biến nào phụ thuộc trực tiếp vào nhau. Những mô hình này chủ yếu được sử dụng với các biến có giá trị rời rạc hoặc phân loại. Tuy nhiên cũng được mở rộng cho một số trường hợp đặc biệt như mật độ Gaussian hoặc cho các biến giá trị thực.

Trong trí tuệ nhân tạo và thống kê, các phương pháp này ban đầu được phát triển trong khuôn khổ của các hệ chuyên gia. Cấu trúc của mô hình và các tham số(xác suất có điều kiện được gắn với các đường nối của đồ thị) được suy ra từ các chuyên gia. Ngày nay, các phương pháp này đã được phát triển, cả cấu trúc và các tham số mô hình

đồ thị đều có thể học trực tiếp từ cơ sở dữ liệu(Buntine; Heckerman) . Tiêu chuẩn đánh giá mô hình chủ yếu là ở dạng Bayesian. Việc đánh giá tham số là một sự kết hợp các đánh giá dạng đóng(closed form estimate) và các phương pháp lặp phụ thuộc vào việc biến được quan sát trực tiếp hay ở dạng ẩn. Việc tìm kiếm mô hình dựa trên các phương pháp leo đồi trên nhiều cấu trúc đồ thị. Các tri thức trước đó, ví dụ như việc sắp xếp một phần các biến dựa trên mối quan hệ nhân quả, có thể rất có ích trong việc làm giảm không gian tìm kiếm mô hình. Mặc dù phương pháp này mới ở giai đoạn đầu của việc nghiên cứu nhưng nó đã cho thấy nhiều hứa hẹn vì dạng đồ thị dễ hiểu hơn và biểu đạt được nhiều ý nghĩa hơn đối với con người.

2. 6 Mô hình học quan hệ

Trong khi mẫu chiết xuất được bằng các luật suy diễn và cây quyết định gắn chặt với các mệnh đề logic(propositional logic) thì mô hình học quan hệ(còn được gọi là lập trình logic quy nạp – inductive logic programming) sử dụng ngôn ngữ mẫu theo thứ tự logic trước(first-order logic) rất linh hoạt. Mô hình này có thể dễ dàng tìm ra công thức X=Y. Cho đến nay, hầu hết các nghiên cứu về các phương pháp đánh giá mô hình này đều theo logic trong tự nhiên.

2. 7 Khai phá dữ liệu dạng văn bản(Text Mining)

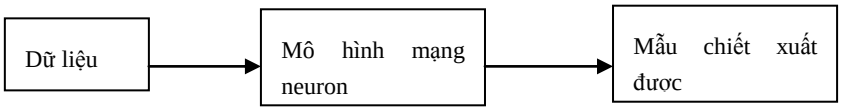
Kỹ thuật này được ứng dụng trong một loạt các công cụ phần mềm thương mại. Công cụ khai phá dữ liệu rất phù hợp với việc tìm kiếm, phân tích và phân lớp các dữ liệu văn bản không định dạng. Các lĩnh vực ứng dụng như nghiên cứu thị trường, thu thập tình báo, v. v… Khai phá dữ liệu dạng văn bản đã được sử dụng để phân tích câu trả lời cho các câu hỏi mở trong khảo sát thị trường, tìm kiếm các tài liệu phức tạp.

2. 8 Mạng neuron

Mạng neuron là tiếp cận tính toán mới liên quan đến việc phát triển các cấu trúc toán học với khả năng học. Các phương pháp là kết quả của việc nghiên cứu mô hình học của hệ thống thần kinh con người. Mạng neuron có thể đưa ra ý nghĩa từ các dữ liệu phức tạp hoặc không chính xác và có thể được sử dụng để chiết xuất các mẫu và phát hiện ra các xu hướng quá phức tạp mà con người cũng như các kỹ thuật máy tính khác không thể phát hiện được.

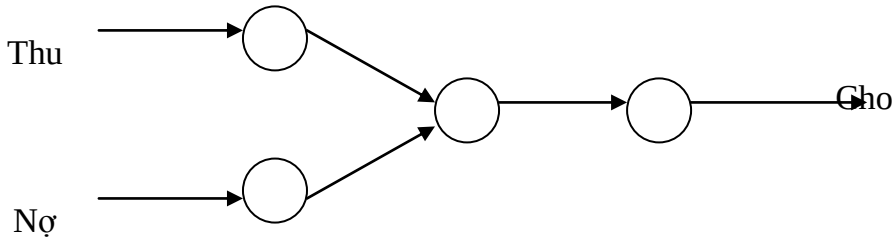
Khi đề cập đến khai thác dữ liệu, người ta thường đề cập nhiều đến mạng neuron. Tuy mạng neuron có một số hạn chế gây khó khăn trong việc áp dụng và triển khai nhưng nó cũng có những ưu điểm đáng kể. Một trong số những ưu điểm phải kể đến của mạng neuron là khả năng tạo ra các mô hình dự đoán có độ chính xác cao, có thể áp

dụng được cho rất nhiều loại bài toán khác nhau, đáp ứng được nhiệm vụ đặt ra của khai phá dữ liệu như phân lớp, phân nhóm, mô hình hóa, dự báo các sự kiện phụ thuộc vào thời gian, v. v…



Hình 2. 9. Sơ đồ quá trình khai phá dữ liệu bằng

Mẫu chiết xuất bằng mạng neuron được thể hiện ở các nút đầu ra của mạng. Mạng neuron sử dụng các hàm số chứ không sử dụng các hàm biểu tượng(symbol functions) để tính mức tích cực của các nút đầu ra và cập nhật các trọng số của nó. Trong mạng lan truyền ngược mà ta sẽ đề cập cụ thể ở phần sau, mỗi nút khái niệm được kết hợp với một ngưỡng, vì vậy trong mạng lan truyền ngược, các mẫu(hay các luật) của một khái niệm là sự kết hợp của các trọng số lớn hơn ngưỡng. Với tập dữ liệu khách hàng vay vốn ngân hàng, ta có bài toán phân lớp áp dụng mạng neuron sẽ cho kết quả là mẫu chiết xuất được như trên Hình 2. 10.



Hình 2. 10. Mẫu kết quả với kỹ thuật mạng neuron

Đặc điểm của mạng neuron là không cần gia công dữ liệu nhiều trước khi bắt đầu quá trình học như các phương pháp khác. Tuy nhiên, để có thể sử dụng mạng neuron có hiệu quả cần phải xác định các yếu tố khi thiết kế mạng như:

- Mô hình mạng là gì?
- Mạng cần có bao nhiêu nút?
- Khi nào thì việc học dừng để tránh bị "học quá"?
- v. v…

Ngoài ra, còn rất nhiều bước quan trọng cần phải làm để tiền xử lý dữ liệu trước khi đưa vào mạng neuron để mạng có thể hiểu được(ví dụ như việc chuẩn hóa dữ liệu, đưa tất cả tiêu chuẩn dự đoán về dạng số) .

Mạng neuron được đóng gói với những thông tin trợ giúp của các chuyên gia đáng tin cậy và được các chuyên gia đảm bảo các mô hình này làm việc tốt. Sau khi học, mạng có thể được coi là một chuyên gia trong lĩnh vực thông tin mà nó vừa được học.

2. 9 Giải thuật di truyền

Giải thuật di truyền, nói theo nghĩa rộng là mô phỏng lại hệ thống tiến hóa trong tự nhiên, chính xác hơn đó là giải thuật chỉ ra tập các cá thể được hình thành, được ước lượng và biến đổi như thế nào. Ví dụ như xác định x làm thế nào để lựa chọn các cá thể tạo giống và lựa chọn các cá thể nào sẽ bị loại bỏ. Giải thuật cũng mô phỏng lại yếu tố gen trong nhiễm sắc thể sinh học trên máy tính để có thể giải quyết nhiều bài toán thực tế khác nhau.

Giải thuật di truyền là một giải thuật tối ưu hóa. Nó được sử dụng rất rộng rãi trong việc tối ưu hóa các kỹ thuật khai phá dữ liệu trong đó có kỹ thuật mạng neuron. Sự liên hệ của nó với các giải thuật khai phá dữ liệu là ở chỗ việc tối ưu hóa cần thiết cho các quá trình khai phá dữ liệu. Ví dụ như trong kỹ thuật cây quyết định, tạo luật. Như đã đề cập ở phần trước, các luật mô hình hóa dữ liệu chứa các tham số được xác định bởi các giải thuật phát hiện tri thức. Giai đoạn tối ưu hóa là cần thiết để xác định x các giá trị tham số nào tạo ra các luật tốt nhất. Và vì vậy mà giải thuật di truyền đã được sử dụng trong các công cụ khai phá dữ liệu. Kỹ thuật này sẽ được tìm hiểu sâu hơn ở chương sau.

Như vậy, nhìn vào các phương pháp giới thiệu ở trên, thấy có rất nhiều các phương pháp khai phá dữ liệu. Mỗi phương pháp có những đặc điểm riêng phù hợp với một lớp các bài toán với các dạng dữ liệu và miền dữ liệu nhất định. Giả sử đối với bài toán dự đoán theo thời gian, trước kia người ta thường đặt nhiệm vụ cho việc khai phá các mẫu dạng này là hồi quy dự đoán hoặc các mô hình hồi quy tự động dựa trên thống kê, v. v… Mới đây, các mô hình khác như các hàm phi tuyến, phương pháp dựa trên mẫu, mạng neuron đã được áp dụng để giải bài toán này.

Như vậy, nặc dù nhìn bề ngoài ta thấy có rất nhiều phương pháp và ứng dụng khai phá dữ liệu nhưng cũng không có gì lạ khi nhận thấy chúng có mộ số thành phần chung. Hiểu quá trình khai phá dữ liệu và suy diễn được mô hình dựa trên những thành phần này là ta đã thực hiện được nhiệm vụ của khai phá dữ liệu.

2. 4 Lợi thế của Khai phá dữ liệu so với các phương pháp cơ bản

Như đã phân tích ở trên, ta thấy khai phá dữ liệu không có gì mới mà hoàn toàn dựa trên các phương pháp cơ bản đã biết. Vậy khai phá dữ liệu có gì khác so với các phương pháp đó? Và tại sao khai phá dữ liệu lại có ưu thế hơn hẳn chúng? Các phân tích sau đây sẽ giải đáp câu hỏi này.

2. 4. 1 Học máy(Machine Learning)

Mặc dù người ta đã cố gắng cải tiến các phương pháp học máy để có thể phù hợp với mục đích khai phá dữ liệu nhưng sự khác biệt giữa cách thiết kế, các đặc điểm của cơ sở dữ liệu đã làm cho phương pháp học máy trở nên không phù hợp với mục đích này, mặc dù cho đến nay, phần lớn các phương pháp khai phá dữ liệu vẫn đựa trên nền tảng cơ sở của phương pháp học máy. Những phân tích sau đây sẽ cho thấy điều đó.

Trong quản trị cơ sở dữ liệu, một cơ sở dữ liệu là một tập hợp được tích hợp một cách logic của dữ liệu được lưu trong một hay nhiều tệp và được tổ chức để lưu trữ có hiệu quả, sửa đổi và lấy thông tin liên quan được dễ dàng. Ví dụ như trong cơ sở dữ liệu quan hệ, dữ liệu được tổ chức thành các tệp hoặc các bảng có các bản ghi có độ dài cố định. Mỗi bản ghi là một danh sách có thứ tự các giá trị, mỗi giá trị được đặt vào một trường. Thông tin về tên trường và giá trị của trường được đặt trong một tệp riêng gọi là thư viện dữ liệu(data dictionary) . Một hệ thống quản trị cơ sở dữ liệu sẽ quản lý các thủ tục(procedures) để lấy, lưu trữ, và xử lý dữ liệu trong các cơ sở dữ liệu đó.

Trong học máy, tuhật ngữ cơ sở dữ liệu chủ yếu đề cập đến một tập các mẫu(instance hay example) được lưu trong một tệp. Các mẫu thường là các vector đặc điểm có độ dài cố định. Thông tin về các tên đặc điểm, dãy giá trị của chúng đôi khi cũng được lưu lại như trong từ điển dữ liệu. Một giải thuật học còn sử dụng tập dữ liệu và các thông tin kèm theo tập dữ liệu đó làm đầu vào và đầu ra biểu thị kết quả của việc học(ví dụ như một khái niệm) .

Với so sánh cơ sở dữ liệu thông thường và CSDL trong học máy như trên, có thể thấy là học máy có khả năng được áp dụng cho cơ sở dữ liệu, bởi vì không phải học trên tập các mẫu mà học trên tệp các bản ghi của cơ sở dữ liệu.

Tuy nhiên, phát hiện tri thức trong cơ sở dữ liệu làm tăng thêm các vấn đề vốn đã là điển hình trong học máy và đã quá khả năng của học máy. Trong thực tế, cơ sở dữ liệu thường động, không đầy đủ, bị nhiễu, và lớn hơn nhiều so với các tập dữ liệu học máy điển hình. Các yếu tố này làm cho hầu hết các giải thuật học máy trở nên không

hiệu quả trong hầu hết các trường hợp. Vì vậy trong khai phá dữ liệu, cần tập trung rất nhiều công sức vào việc vượt qua những khó khăn, phức tạp này trong CSDL.

2. 4. 2 Phương pháp hệ chuyên gia

Các hệ chuyên gia cố gắng nắm bắt các tri thức thích hợp với một bài toán nào đó. Các kỹ thuật thu thập giúp cho việc lấy tri thức từ các chuyên gia con người. Mỗi phương pháp đó là một cách suy diễn các luật từ các ví dụ và giải pháp đối với bài toán chuyên gia đưa ra. Phương pháp này khác với khai phá dữ liệu ở chỗ các ví dụ của chuyên gia thường ở mức chất lượng cao hơn rất nhiều so với các dữ liệu trong cơ sở dữ liệu, và chúng thường chỉ bao được các trường hợp quan trọng. Hơn nữa, các chuyên gia sẽ xác nhận tính giá trị và hữu dụng của các mẫu phát hiện được. Cũng như với các c6ng cụ quản trị cơ sở dữ liệu, ở các phương pháp này đòi hỏi có sự tham gia của con người trong việc phát hiện tri thức.

2. 4. 3 Phát kiến khoa học

Khai phá dữ liệu rất khác với phát kiến khoa học ở chỗ những khai phá trong cơ sở dữ liệu ít có chủ tâm và có điều khiển hơn. Các dữ liệu khoa học có từ thực nghiệm nhằm loại bỏ tác động của một số tham số để nhấn mạnh độ biến thiên của một hay một số tham số đích. Tuy nhiên, các cơ sở dữ liệu thương , ại điển hình lại ghi một số lượng thừa thông tin về các dự án của họ để đạt được một số mục đích về mặt tổ chức. Độ dư thừa này(hay có thể gọi là sự lẫn lộn – confusion) có thể nhìn thấy và cũng có thể ẩn chứa trong các mối quan hệ dữ liệu. Hơn nữa, các nhà khoa học có thể tạo lại các thí nghiệm và có thể tìm ra rằng các thiết kế ban đầu không thích hợp. Trong khi đó, các nhà quản lý cơ sở dữ liệu hầu như không thể xa xỉ đi thiết kế lại các trường dữ liệu và thu thập lại dữ liệu.

2.4.4 Phương pháp thống kê

Một câu hỏi hiển nhiên là khai phá dữ liệu khác gì so với phương pháp thống kê. Từ nhiều năm nay, con người đã sử dụng phương pháp thống kê một cách rất hiệu quả để đạt được những mục đích của mình.

Mặc dù các phương pháp thống kê cung cấp một nền tảng lý thuyết vững chắc cho các bài toàn phân tích dữ liệu nhưng chỉ có tiếp cận thống kê thuần túy thôi chưa đủ. Thứ nhất, các phương pháp thống kê chuẩn không phù hợp đối với các kiểu dữ liệu có cấu trúc trong rất nhiều các cơ sở dữ liệu. Thứ hai, thống kê hoàn toàn theo dữ liệu(data driven) , nó không sử dụng tri thức sẵn có về lĩnh vực. Thứ ba, các kết quả phân tích thống kê có thể sẽ rất nhiều và khó có thể làm rõ được. Cuối cùng, các phương

pháp thống kê cần có sự hướng dẫn của người dùng để xác định phân tích dữ liệu như thế nào và ở đâu.

Sự khác nhau cơ bản giữa khai phá dữ liệu và thống kê là ở chỗ khai phá dữ liệu là một phương tiện được dùng bởi người sử dụng đầu cuối chứ không phải là các nhà thống kê. Khai phá dữ liệu tự động quá trình thống kê một cách có hiệu quả, vì vậy làm nhẹ bớt công việc của người dùng đầu cuối, tạo ra một công cụ dễ sử dụng hơn. Như vậy, nhờ có khai phá dữ liệu, việc dự đoán và kiểm tra rất vất vả trước đây có thể được đưa lên máy tính, được tính, dự đoán và kiểm tra một cách tự động.

2. 5 Lựa chọn phương pháp

Các giải thuật khai phá dữ liệu tự động vẫn mới chỉ ở giai đoạn phát triển ban đầu. Người ta vẫn chưa đưa ra được một tiêu chuẩn nào trong việc quyết định sử dụng phương pháp nào và trong trường hợp thì có hiệu quả.

Hầu hết các kỹ thuật khai phá dữ liệu đều mới đối với lĩnh vực kinh doanh. Hơn nữa lại có rất nhiều kỹ thuật, mỗi kỹ thuật được sử dụng cho nhiều bài toán khác nhau. Vì vậy, ngay sau câu hỏi "*khai phá dữ liệu là gì?*" sẽ là câu hỏi "*vậy thì dùng kỹ thuật nào?*". Câu trả lời tất nhiên là không đơn giản. Mỗi phương pháp đều có điểm mạnh và yếu của nó, nhưng hầu hết các điểm yếu đều có thể khắc phục được. Vậy thì phải làm như thế nào để áp dụng kỹ thuật một cách thật đơn giản, dễ sử dụng để không cảm thấy những phức tạp vốn có của kỹ thuật đó.

Để so sánh các kỹ thuật cần phải có một tập lớn các quy tắc và các phương pháp thực nghiệm tốt. Thường thì quy tắc này không được sử dụng khi đánh giá các kỹ thuật mới nhất. Vi vậy mà những yêu cầu cải thiện độ chính xác không phải lúc nào cũng thực hiện được.

Nhiều công ty đã đưa ra những sản phẩm sử dụng kết hợp nhiều kỹ thuật khai phá dữ liệu khác nhau với hy vọng nhiều kỹ thuật sẽ tốt hơn. Nhưng thực tế cho thấy nhiều kỹ thuật chỉ thêm nhiều rắc rối và gây khó khăn cho việc so sánh giữa các phương pháp và các sản phẩm này. Theo nhiều đánh giá cho thấy, khi đã hiểu được các kỹ thuật và nghiên cứu tính giống nhau giữa chúng, người ta thấy rằng nhiều kỹ thuật lúc đầu thì có vẻ khác nhau nhưng thực chất ra khi hiểu được các kỹ thuật này thì thấy chúng hoàn toàn giống nhau. Tuy nhiên, đánh giá này cũng chỉ để tham khảo vì cho đến nay, khai phá dữ liệu vẫn còn là kỹ thuật mới chứa nhiều tiềm năng mà người ta vẫn chưa khai thác hết.

2. 6 Những thách thức trong ứng dụng và nghiên cứu kỹ thuật Khai phá dữ liệu

Ở đây, ta đưa ra một số khó khăn trong việc nghiên cứu và ứng dụng kỹ thuật khai phá dữ liệu. Tuy nhiên, thế không có nghĩa là việc giải quyết là hoàn toàn bế tắc mà chỉ muốn nêu lên rằng để khai phá được dữ liệu không phải đơn giản, mà phải xem xét cũng như tìm cách giải quyết những vấn đề này. Ta có thể liệt kê một số khó khăn như sau:

2. 6. 1 Các vấn đề về cơ sở dữ liệu

Đầu vào chủ yếu của một hệ thống khai thác tri thức là các dữ liệu thô trong cơ sở dữ liệu. Những vấn đề khó khăn phát sinh trong khai phá dữ liệu chính là từ đây. Do các dữ liệu trong thực tế thường động, không đầy đủ, lớn và bị nhiễu. Trong những trường hợp khác, người ta không biết cơ sở dữ liệu có chứa các thông tin cần thiết cho việc khai thác hay không và làm thế nào để giải quyết với sự dư thừa những thông tin không thích hợp này.

- *Dữ liệu lớn:* Cho đến nay, các cơ sở dữ liệu với hàng trăm trường và bảng, hàng triệu bản ghi và với kích thước đến gigabytes đã là chuyện bình thường. Hiện nay đã bắt đầu xuất hiện các cơ sở dữ liệu có kích thước tới terabytes. Các phương pháp giải quyết hiện nay là đưa ra một ngưỡng cho cơ sở dữ liệu, lấu mẫu, các phương pháp xấp xỉ, xử lý song song(Agrawal et al, Holsheimer et al) .
- *Kích thước lớn:* không chỉ có số lượng bản ghi lớn mà số các trường trong cơ sở dữ liệu cũng nhiều. Vì vậy mà kích thước của bài toán trở nên lớn hơn. Một tập dữ liệu có kích thước lớn sinh ra vấn đề làm tăng không gian tìm kiếm mô hình suy diễn. Hơn nữa, nó cũng làm tăng khả năng một giải thuật khai phá dữ liệu có thể tìm thấy các mẫu giả. Biện pháp khắc phục là làm giảm kích thước tác động của bài toán và sử dụng các tri thức biết trước để xác định các biến không phù hợp.
- *Dữ liệu động:* Đặc điểm cơ bản của hầu hết các cơ sở dữ liệu là nội dung của chúng thay đổi liên tục. Dữ liệu có thể thay đổi theo thời gian và việc khai phá dữ liệu cũng bị ảnh hưởng bởi thời điểm quan sát dữ liệu. Ví dụ trong cơ sở dữ liệu về tình trạng bệnh nhân, một số giá trị dữ liệu là hằng số, một số khác lại thay đổi liên tục theo thời gian(ví dụ cân nặng và chiều cao) , một số khác lại thay đổi tùy thuộc vào tình huống và chỉ có giá trị được quan sát mới nhất là đủ(ví dụ nhịp đập của mạch) . Vậy thay đổi dữ liệu nhanh chóng có thể làm cho các mẫu khai thác được trước đó mất giá

trị. Hơn nữa, các biến trong cơ sở dữ liệu của ứng dụng đã cho cũng có thể bị thay đổi, bị xóa hoặc là tăng lên theo thời gian. Vấn đề này được giải quyết bằng các giải pháp tăng trưởng để nâng cấp các mẫu và coi những thay đổi như là cơ hội để khai thác bằng cách sử dụng nó để tìm kiếm các mẫu bị thay đổi.

- *Các trường không phù hợp:* Một đặc điểm quan trọng khác là tính không thích hợp của dữ liệu, nghĩa là mục dữ liệu trở thành không thích hợp với trọng tâm hiện tại của việc khai thác. Một khía cạnh khác đôi khi cũng liên quan đến độ phù hợp là tính ứng dụng của một thuộc tính đối với một tập con của cơ sở dữ liệu. Ví dụ trường số tài khoản Nostro không áp dụng cho các các nhân.
- *Các giá trị bị thiếu:* Sự có mặt hay vắng mặt của giá trị các thuộc tính dữ liệu phù hợp có thể ảnh hưởng đến việc khai phá dữ liệu. Trong hệ thống tương tác, sự thiếu vắng dữ liệu quan trọng có thể dẫn đến việc yêu cầu cho giá trị của nó hoặc kiểm tra để xác định giá trị của nó. Hoặc cũng có thể sự vắng mặt của dữ liệu được coi như một điều kiện, thuộc tính bị mất có thể được coi như một giá trị trung gian và là giá trị không biết.
- *Các trường bị thiếu:* Một quan sát không đầy đủ cơ sở dữ liệu có thể làm cho các dữ liệu có giá trị bị x như có lỗi. Việc quan sát cơ sở dữ liệu phải phát hiện được toàn bộ các thuộc tính có thể dùng để giải thuật khai phá dữ liệu có thể áp dụng nhằm giải quyết bài toán. Giả sử ta có các thuộc tính để phân biệt các tình huống đáng quan tâm. Nếu chúng không làm được điều đó thì có nghĩa là đã có lỗi trong dữ liệu. Đối với một hệ thống học để chuẩn đoán bệnh sốt rét từ một cơ sở dữ liệu bệnh nhân thì trường hợp các bản ghi của bệnh nhân có triệu chứng giống nhau nhưng lại có các chuẩn đoán khác nhau là do trong dữ liệu đã bị lỗi. Đây cũng là vấn đề thường xảy ra trong cơ sở dữ liệu kinh doanh. Các thuộc tính quan trọng có thể sẽ bị thiếu nếu dữ liệu không được chuẩn bị cho việc khai phá dữ liệu.
- *Độ nhiễu và không chắc chắn:* Đối với các thuộc tính đã thích hợp, độ nghiêm trọng của lỗi phụ thuộc vào kiểu dữ liệu của các giá trị cho phép. Các giá trị của các thuộc tính khác nhau có thể là các số thực, số nguyên, chuỗi và có thể thuộc vào tập các giá trị định danh. Các giá trị định danh

này có thể sắp xếp theo thứ tự từng phần hoặc đầy đủ, thậm chí có thể có cấu trúc ngữ nghĩa.

Một yếu tố khác của độ không chắc chắn chính là tính kế thừa hoặc độ chính xác mà dữ liệu cần có, nói cách khác là độ nhiễu của dữ liệu. Dựa trên việc tính toán trên các phép đo và phân tích có ưu tiên, mô hình thống kê mô tả tính ngẫu nhiên được tạo ra và được sử dụng để định nghĩa độ mong muốn và độ dung sai của dữ liệu. Thường thì các mô hình thống kê được áp dụng theo cách đặc biệt để xác định một cách chủ quan các thuộc tính để đạt được các thống kê và đánh giá khả năng chấp nhận của các(hay tổ hợp các) giá trị thuộc tính. Đặc biệt là với dữ liệu kiểu số, sự đúng đắn của dữ liệu có thể là một yếu tố trong việc khai phá. Ví dụ như trong việc đo nhiệt độ cơ thể, ta thường cho phép chênh lệch 0. 1 độ. Nhưng việc phân tích theo xu hướng nhạy cảm nhiệt độ của cơ thể lại yêu cầu độ chính xác cao hơn. Để một hệ thống khai thác có thể liên hệ đến xu hướng này để chuẩn đoán thì lại cần có một độ nhiễu trong dữ liệu đầu vào.

- *Mối quan hệ phức tạp giữa các trường:* các thuộc tính hoặc các giá trị có cấu trúc phân cấp, các mối quan hệ giữa các thuộc tính và các phương tiện phức tạp để diễn tả tri thức về nội dung của cơ sở dữ liệu yêu cầu các giải thuật phải có khả năng sử dụng một cách hiệu quả các thông tin này. Ban đầu, kỹ thuật khai phá dữ liệu chỉ được phát triển cho các bản ghi có giá trị thuộc tính đơn giản. Tuy nhiên, ngày nay người ta đang tìm cách phát triển các kỹ thuật nhằm rút ra mối quan hệ giữa các biến này.

2. 6. 2 Một số vấn đề khác

- *"Quá phù hợp"(Overfitting) :* Khi một giải thuật tìm kiếm các tham số tốt nhất cho một mô hình nào đó sử dụng một tập dữ liệu hữu hạn, nó có thể sẽ bị tình trạng "quá độ" dữ liệu(nghĩa là tìm kiếm quá mức cần thiết gây ra hiện tượng chỉ phù hợp với các dữ liệu đó mà không có khả năng đáp ứng cho các dữ liệu lạ) , làm cho mô hình hoạt động rất kém đối với các dữ liệu thử. Các giải pháp khắc phục bao gồm đánh giá chéo(cross-validation) , thực hiện theo nguyên tắc nào đó hoặc sử dụng các biện pháp thống kê khác.
- *Đánh giá tầm quan trọng thống kê:* Vấn đề(liên quan đến overfitting) xảy ra khi một hệ thống tìm kiếm qua nhiều mô hình. Ví dụ như nếu một hệ thống kiểm tra N mô hình ở mức độ quan trọng 0, 001 thì với dữ liệu ngẫu nhiên trung bình sẽ có N/1000 mô hình được chấp nhận là quan trọng. Để xử lý vấn

đề này, ta có thể sử dụng phương pháp điều chỉnh thống kê trong kiểm tra như một hàm tìm kiếm, ví dụ như điều chỉnh Bonferroni đối với các kiểm tra độc lập.

- *Khả năng biểu đạt của mẫu:* Trong rất nhiều ứng dụng, điều quan trọng là những điều khai thác được phải cáng dễ hiểu với con người càng tốt. Vì vậy, các giải pháp thường bao gồm việc diễn tả dưới dạng đồ họa, xây dựng cấu trúc luật với các đồ thị có hướng(Gaines) , biểu diễn bằng ngôn ngữ tự nhiên(Matheus et al.) và các kỹ thuật khác nhằm biểi diễn tri thức và dữ liệu.
- *Sự tương tác với người sử dụng và các tri thức sẵn có:* rất nhiều công cụ và phương pháp khai phá dữ liệu không thực sự tương tác với người dùng và không dễ dàng kết hợp cùng với các tri thức đã biết trước đó. Việc sử dụng tri thức miền là rất quan trọng trong khai phá dữ liệu. Đã có nhiều biện pháp nhằm khắc phục vấn đề này như sử dụng cơ sở dữ liệu suy diễn để phát hiện tri thức, những tri thức này sau đó được sử dụng để hướng dẫn cho việc tìm kiếm khai phá dữ liệu hoặc sử dụng sự phân bố và xác suất dữ liệu trước đó như một dạng mã hóa tri thức có sẵn.

2. 7 Tình trạng ứng dụng dữ liệu

Mặc dù còn rất nhiều vấn đề mà khai phá dữ liệu cần phải tiếp tục nghiên cứu để giải quyết nhưng tiềm năng của nó đã được khẳng định bằng sự ra đời của rất nhiều ứng dụng.

Khai phá dữ liệu được ứng dụng rất thành công trong "cơ sở dữ liệu thị trường"(database marketing) , đây là một phương pháp phân tích cơ sở dữ liệu khách hàng, tìm kiếm các mẫu trong số các khách hàng và sử dụng các mẫu này để lựa chọn các khách hàng trong tương lai. Tạp chí Business Week của Mỹ đã đánh giá hơn 50% các nhà bán lẻ đang và có ý định sử dụng "cơ sở dữ liệu thị trường" cho hoạt động kinh doanh của họ(Berry 1994) . Kết quả ứng dụng cho thấy số lượng thẻ tín dụng American Express bán ra đã tăng 15% - 20%(Berry 1994) . Các ứng dụng khác của khai phá dữ liệu trong kinh doanh như phân tích chứng khoán và các văn kiện tài chính; phân tích và báo cáo những thay đổi trong dữ liệu, bao gồm Coverstory của IRI(Schmitz, Armstrong, & Little 1990) , Spotlight của A. C Nielsen(nand & Kahn 1992) đối với các dữ liệu bán hàng trong siêu thị, KEFIR của GTE cho cơ sở dữ liệu y tế(Matheus, Piatetsky-Shapiro, & McNeil) ; phát hiện và phòng chống gian lận cũng thường là bài toán của khai phá dữ liệu và phát hiện tri thức. Ví dụ như hệ thống phát hiện gian lận trong dịch vụ y tế đã được Major và Riedinger phát triển tại Travelers insurance năm 1992. Internal Revenue

Service đã phát triển một hệ thống chọn thuế thu để kiểm toán. Nestor FDS(Blanchard 1994) được phát triển dựa trên mạng neuron để phát hiện ra gian lận trong thẻ tín dụng.

Các ứng dụng của khai phá dữ liệu trong khoa học cũng được phát triển. Ta có thể đưa ra một số ứng dụng trong khoa học như:

- Thiên văn học: Hệ thống SKICAT do JPL/Caltech phát triển được sử dụng cho các nhà thiên văn để tự động xác định các vì sao và các dải thiên hà trong một bản khảo sát lớn để có thể phân tích và phân loại(Fayyad, Djorgovski, & Weir) .
- Phân tử sinh học: Hệ thống tìm kiếm các mẫu trong cấu trúc phân tử(Conklin, Fortier, và Glasgow 1993) và trong các dữ liệu gen(Holder, Cook, và Djoko 1994) .
- Mô hình hóa những thay đổi thời tiết: các mẫu không thời gian như lốc, gió xoáy được tự động tìm thấy trong các tập lớn dữ liệu mô phỏng và quan sát được(Stolorz et al. 1994) .

Chương 3: Bài Toán Ứng Dụng

Trên cơ sở nghiên cứu và tìm hiểu R cũng như một số kỹ thuật trong khai phá dữ liệu, trong khuôn khổ đồ án tốt nghiệp, tôi xin trình bày một bài toán về khai phá dữ liệu sử dụng ngôn ngữ R.

Nghiên cứu này đề cập đến vấn đề cố gắng để xây dựng một hệ thống giao dịch chứng khoán dựa trên mô hình dự báo thu được với dữ liệu báo giá cổ phiếu hàng ngày. Nên sẽ áp dụng các mô hình khác nhau để dự đoán lợi nhuận của cổ phiếu IBM tại New York Stock Exchange. Những dự đoán này sẽ được sử dụng cùng với một quy tắc giao dịch mà sẽ tạo ra tín hiệu mua bán. Ở đây đề cập một số dữ liệu mới khai thác các vấn đề: làm thế nào để sử dụng R để phân tích các dữ liệu được lưu trữ trong một cơ sở dữ liệu, như thế nào xử lý các vấn đề dự đoán, nơi có một thời gian đặt hàng trong trường hợp đào tạo(thường được biết đến như là một chuỗi thời gian).

3. 1 Mô tả bài toán

Thị trường giao dịch chứng khoán là một lĩnh vực ứng dụng với một tiềm năng lớn để khai thác dữ liệu. Trong thực tế, sự tồn tại của một số lượng lớn lịch sử dữ liệu cho thấy rằng việc khai thác dữ liệu có thể cung cấp một lợi thế cạnh tranh so với kiểm tra của con người trên bộ dữ liệu này. Mặt khác, các thị trường thích ứng nhanh chóng về điều chỉnh giá là không có mặt để có được lợi nhuận một cách nhất quán. Điều này thường được gọi là giả thuyết hiệu quả thị trường. Mục tiêu chung của giao dịch chứng khoán là để duy trì một danh mục đầu tư cổ phiếu dựa vào lệnh mua và bán. Mục tiêu dài hạn là để đạt được lợi nhuận nhiều nhất có thể từ những hành động kinh doanh.

Với bảo mật này và vốn ban đầu, bằng việc cố gắng tối đa hóa lợi nhuận trong thời gian thử nghiệm trong tương lai bằng phương tiện của các hành động kinh doanh(Mua, Bán) . Chiến lược kinh doanh sẽ sử dụng như là cơ sở cho việc ra quyết định các chỉ dẫn được cung cấp bởi kết quả của một quá trình khai thác dữ liệu. Quá trình này sẽ cố gắng để dự đoán lợi nhuận của cổ phiếu trong tương lai dựa trên một mô hình thu được với lịch sử dữ liệu. Vì vậy, mô hình dự đoán sẽ được đưa vào một hệ thống kinh doanh tạo ra quyết định của mình dựa trên các dự đoán của mô hình. Tổng thể các tiêu chí đánh giá sẽ được hệ thống thực hiện giao dịch này, tức là lợi nhuận / lỗ từ các hành động của hệ thống. Điều này có nghĩa tiêu chí đánh giá chính là kết quả của việc áp dụng các kiến thức phát hiện của quá trình khai thác dữ liệu và tính chính xác của các mô hình phát triển trong quá trình này.

3. 2 Các dữ liệu cần thiết

Trong khuôn khổ đồ án tốt nghiệp sẽ tập trung nghiên cứu hoạt động kinh doanh cổ phiếu “IBM" từ thị trường New York Stock Exchange(NYSE). Trong phần này, sẽ minh họa làm thế nào để tải dữ liệu này vào R với tập dữ liệu CSV có sẵn. Báo giá cổ phiếu hàng ngày bao gồm các thông tin liên quan đến các thuộc tính sau:

- ngày của phiên giao dịch chứng khoán(Index).
- mở cửa Giá vào lúc bắt đầu của phiên họp(Open).
- Giá cao nhất trong phiên(High).
- Giá thấp nhất(Low).
- Giá đóng cửa phiên giao dịch(Close).
- Khối lượng giao dịch(Volume).
- Mã chứng khoán(AdjClose).

với IBM, bao gồm các trích dẫn từ 02-Jan-1970 17-May-2002.

3. 3 Chuỗi thời gian dự đoán

Các kiểu dữ liệu nghiên cứu trường hợp này biết đến như là một chuỗi thời gian. Các tính năng phân biệt chính của loại dữ liệu này là sự tồn tại của một đánh dấu thời gian gắn liền với mỗi quan sát, có nghĩa là thứ tự giữa các trường hợp vấn đề. Nói chung một chuỗi thời gian là một tập hợp các quan sát của một biến Y,

$$Y_1, y_2, \ldots, y_t{-}1, y_t, y_t{+}1, \ldots, y_n \qquad (3.\ 1)$$

Ở đây, yt là giá trị của Y loạt biến mục tiêu thời gian t. Phân tích chuỗi thời gian là để có được một mô hình dựa trên những quan sát trong quá khứ của biến y1, y2, . . . , Yt-1, yt, cho phép đưa ra dự đoán về quan sát của biến, yt +1, trong tương lai. . . , Yn.

Trong trường hợp cổ phiếu dữ liệu , đã thường được biết đến như là một đa biến chuỗi thời gian, bởi vì có một số biến được ghi lại cùng một lúc , cụ thể là Open, High, Low, Close và Volume. Trong đồ án này, sẽ sử dụng dữ liệu đa biến để có được mô hình.

Tuy nhiên, sẽ bắt đầu với mô hình đơn giản có thể xử lý một biến chuỗi thời gian duy nhất. Phương pháp thông thường trong phân tích chuỗi thời gian tài chính để tập trung vào dự đoán giá đóng cửa của một cổ phiếu. Hơn nữa, để tránh hiệu ứng xu hướng, được phổ biến sử dụng sự khác biệt tỷ lệ phần trăm của giá đóng cửa là chuỗi

thời gian cơ bản được mô hình hóa, thay vì các giá trị tuyệt đối. Dưới đây định nghĩa hai biến mục tiêu thay thế,

$$Yt = \frac{Closet - Closet-1}{Closet-1} \qquad (3.\ 3)$$

$$Yt = \log\frac{Closet}{Closet-1} \qquad (3.\ 4)$$

Sẽ áp dụng thay thế đầu tiên bởi vì nó có một sự giải thích trực quan hơn cho người sử dụng. Cụ thể, Sẽ tập trung vào các mô hình phát triển cho một chuỗi thời gian bao gồm lợi nhuận hàng-ngày giá đóng cửa được định nghĩa là,

$$Rh(t) = \frac{Closet - Closet-h}{Closet-h}$$

Tóm lại, dữ liệu sẽ bao gồm các quan sát,

$$Rh(1)\ ,\ Rh(2)\ ,\ \ldots\ ,\ Rh(t-1)\ ,\ Rh(t)\ ,\ Rh(t+1)\ ,\ \ldots\ ,\ Rh(n) \qquad (3.\ 5)$$

Các chức năng R sau đây có được trả lại hàng ngày của một vector các giá trị(ví dụ giá đóng cửa của cổ phiếu) ,

```
> h. returns <- function(x, h=1) {
+ diff(x, lag=h) /x[1:(length(x) -h) ]
+ }
> h. returns(c(45, 23, 4, 56, -45, 3) , h=2)
[1] -0. 9111111 1. 4347826 -12. 2500000 -0. 9464286
```

Để tạo ra chức năng này, sử dụng các chức năng khác() . Chức năng này R tính toán sự khác biệt độ trễ của một véc tơ, tức là xt - xt-lag.

Tạo ra một tập dữ liệu bằng cách sử dụng chức năng này, sau đó sẽ được sử dụng để có được một mô hình để dự đoán hàng-ngày trả lại tương lai của giá đóng cửa của cổ phiếu IBM. Phương pháp phổ biến nhất để có được mô hình để dự đoán các giá trị tương lai của một biến chuỗi thời gian là sử dụng giá trị trong quá khứ của loạt bài gần đây nhất là các biến đầu vào của mô hình. Vì vậy, mô hình xây dựng được sẽ cố gắng để

dự đoán lợi nhuận hàng ngày của giá đóng cửa của cổ phiếu IBM dựa trên các giá trị gần đây nhất trong số này trả về. Chuẩn bị dữ liệu Kỹ thuật này thường được gọi là nhúng thời gian trễ.

Có thể sử dụng tham số mờ để thiết lập kích thước của địa chỉ mạng. Sử dụng chức năng này cùng với các hàm h. returns() , có thể xây dựng một chức năng mới để tạo ra một khung dữ liệu lưu trữ một nhúng trả lại hàng ngày,

```
> beded. dataset <- function(data, quote=’Close’, hday=1, b=10) {
+ ds <- data. frame(bed(h. returns(data[, quote], h=hday) , b+hday))
+ ds <- ds[, c(1, (1+hday) :(hday+b)) ]
+ names(ds) <- c(paste(’r’, hday, ’. f’, hday, sep="") ,
+ paste(’r’, hday, ’. t’, 0:(b-1) , sep=""))
+ ds$Date <- data[(hday+b) :(nrow(data) -hday) , ’Date’]
+ ds
+ }
> ibm. data <- beded. dataset(ibm, hday=1)
> names(ibm. data)
[1] "r1. f1" "r1. t0" "r1. t1" "r1. t2" "r1. t3" "r1. t4" "r1. t5" "r1. t6" "r1. t7"
[10] "r1. t8" "r1. t9" "Date"
> ibm. data[1:2, ]
r1. f1 r1. t0 r1. t1 r1. t2 r1. t3 r1. t4
1-0. 05352267-0. 03070408 0. 02052944 -0. 001079331 0. 017572762 -0. 003829322
2 0. 01904212 -0. 05352267 -0. 03070408 0. 020529444 -0. 001079331 0. 017572762
r1. t5 r1. t6 r1. t7 r1. t8 r1. t9 Date
1-0. 001092896 0. 002190581 0. 0005479452 0. 0005482456 0. 0099667774 1970-01-16
2-0. 003829322 -0. 001092896 0. 0021905805 0. 0005479452 0. 0005482456 1970-01-19
```

Chức năng nhận được một khung dữ liệu phải có một cấu trúc như một trong những bảng đã tạo ra khi đọc dữ liệu báo giá, điều đó có nghĩa là nó cần phải có ít nhất

một cột Date và nếu giá trị mặc định của các báo tham số được sử dụng, cũng cần một cột tên là Close. bed. dataset() chức năng tạo ra một khung dữ liệu bằng cách sử dụng một địa chỉ mạng của lợi nhuận hàng ngày. Khung dữ liệu cũng sẽ nhận được "thích hợp" tên cột, cột đầu tiên là "mục tiêu", tức là giá trị của lợi nhuận hàng ngày hàng ngày trước, trong khi giá trị trước đó của các lợi nhuận.

Ví dụ nhỏ để hiểu rõ hơn những gì các chức năng được làm. Với một loạt giống như một trong những phương trình(3, 5) và giá trị của h, cần phải cẩn thận khi lựa chọn các biến mục tiêu. Ví dụ nếu h = 3 và đang sử dụng một địa chỉ mạng của 2, trường hợp đầu tiên là thu được bằng cách beded. dataset chức năng() sẽ bao gồm các yếu tố sau đây của chuỗi thời gian R3(t) :

Target(r3. f3) 1stVar. (r3. t0) 2ndVar. (r3. t1)

R3(8) R3(5) R3(4)

Đây là trường hợp đầu tiên được thu được tại thời điểm . "Date" này, muốn để dự đoán trả về 3 ngày trước, tức là giá trị của R3. Như đang sử dụng một địa chỉ mạng của 2 sử dụng như là các biến dự đoán giá trị hiện tại của lợi nhuận(R3(5)) và giá trị vào ngày trước khi(R3(4)) . Chú ý rằng trường hợp đầu tiên được thu được tại thời điểm 5 vì trước "Date" sẽ bao hàm trả về tính toán mà không thể có được. Trong thực tế, nếu muốn có được một trường hợp cho thời gian 4, cần:

Target(r3. f3) 1stVar. (r3. t0) 2ndVar. (r3. t1)

R3(7) R3(4) R3(3)

Tuy nhiên, để tính toán R3(3) ta sẽ cần mức giá đóng cửa lúc đầu tiên(x phương trình(3. 4)) , đó là không khả thi với giả định báo đầu tiên .

3. 3. 1 Lấy mô hình chuỗi thời gian dự đoán

Với một khung dữ liệu hiển thị trước đây, có thể sử dụng phương pháp hồi quy để có được một mô hình dự đoán giá trị của lợi nhuận trong tương lai quan sát trả lại qua. Trước khi có mô hình này cho phép nghiên cứu một số thuộc tính của tập dữ liệu này.

Từ 8166 hàng của khung ibm. data dữ liệu, chỉ có 68 là các biến thể tuyệt đối của hơn 5%. Điều này sẽ không là một vấn đề nếu không phải là những tình huống mà thực sự quan tâm !Một trong những tính năng thú vị hơn của tên miền này là hầu hết các lần thị trường thay đổi theo một cách mà không phải là thú vị để kinh doanh vì những thay đổi quá nhỏ để bù đắp cho các chi phí kinh doanh. Như vậy, phần lớn các dữ liệu có không phải là thú vị từ một quan điểm kinh doanh. Các trường hợp thú vị là chính xác

những người với những biến đổi lớn. Muốn mô hình là cực kỳ chính xác dự đoán những chuyển động lớn . Các đặc tính của lĩnh vực này là vấn đề đối với hầu hết các công cụ mô hình hóa. Tiêu chuẩn phương pháp hồi quy cố gắng giảm thiểu một số ước tính lỗi, ví dụ, trung bình bình phương lỗi. Trong khi không có gì là sai với điều này, trong lĩnh vực này, sẽ sẵn sàng từ bỏ một số chính xác về chuyển động nhỏ trong trao đổi cho các dự đoán chính xác của các phong trào thị trường rộng lớn. Hơn nữa, Sẽ không đặc biệt quan tâm đến chính xác định theo giá trị chính xác của một sự thay đổi lớn, miễn là dự đoán một giá trị mà dẫn đến quyết định kinh doanh đúng(mua, bán hay nắm giữ) .

Nếu quan tâm trong việc có được các mô hình với mục tiêu kinh doanh thực sự, có ít nhất hai con đường khác nhau mà có thể làm theo: Một là để có được một mô hình hồi quy dự báo lợi nhuận và sau đó sử dụng các dự đoán để làm cho quyết định kinh doanh , phần khác là để có được một loại khác nhau của các mô hình với mục tiêu là để dự đoán các hành động kinh doanh chính xác trực tiếp. Cách tiếp cận thứ hai có ngụ ý rằng sẽ có một biến mục tiêu có giá trị có thể là 'bán', 'giữ' và 'mua', mà thường được biết đến như là một vấn đề phân loại.

Đồ án được trình bày ở trên là một minh chứng cho sự khác biệt giữa dự đoán và quyết định của họ vào các quyết định. Trong thực tế, có thể có một mô hình dự đoán rất chính xác được tốt hơn về kết quả giao dịch bằng một mô hình ít chính xác hơn(Deboeck, 1994) .

Tập dữ liệu bao gồm các giá trị trả về hàng ngày như các thông tin duy nhất mà sẽ được sử dụng để có được một mô hình để dự đoán giá trị tiếp theo của loạt thời gian trước đây. Có thể có một ý tưởng về sự tương quan giữa lợi nhuận 1-ngày với các giá trị của những trở lại trong những ngày trước đó bằng cách sử dụng khái niệm về tương quan tự động. Khái niệm thống kê nắm bắt các giá trị của một biến chuỗi thời gian như thế nào có tương quan với các giá trị về các bước thời gian trước . Các hàm tính toán acf() và mặc định tương quan tự động của một biến cùng một tập hợp các độ trễ thời gian,

> acf(ibm. data$r1. f1, main=’’, ylim=c(-0. 1, 0. 1))

Công thức 3. 3 cho thấy tương quan tự động của biến r1. f1 '. Đại diện cho một ngưỡng tin cậy 95% về ý nghĩa của các giá trị tương quan tự động. Do đã hạn chế trục Y để loại trừ ảnh hưởng rộng của một giá trị tương quan rất lớn cho 0 lag.

Như đã thấy có rất ít ý nghĩa giá trị tương quan tự động , mà không cung cấp chỉ tốt về khả năng dự đoán biến 'r1. f1' bằng cách sử dụng sự trả về độ trễ. Tuy nhiên, những tương quan tự động đề cập đến mối tương quan tuyến tính và sẽ không sử dụng

các mô hình tuyến tính cho Đồ án này mà vẫn có thể nhận được kết quả hợp lý chỉ sử dụng các thông tin này.

Trong phạm vi đồ án này, sẽ đề cập đến các giả định tuyến tính đằng sau những giá trị tương quan, thông tin này chủ yếu cho các mô hình tuyến tính. Trên cơ sở đó, có thể tạo ra một tập dữ liệu với địa chỉ mạng như sau:

> ibm. data <- beded. dataset(ibm, hday=1, b=24)

Neural Networks

Mô hình mục tiêu của dự đoán trước 1 ngày- lợi nhuận của giá đóng cửa sẽ là một mạng lưới. Các mạng là một trong các mô hình được sử dụng thường xuyên nhất trong các thí nghiệm dự đoán tài chính(Deboeck, 1994) , bởi vì khả năng của họ để đối phó với vấn đề phi tuyến tính. Nnet gói thực hiện nuôi chuyển tiếp mạng lưới trong R. Đây là loại mạng là một trong những thường xuyên nhất.

Một mạng lưới được hình thành bởi một mạng lưới các đơn vị máy tính(các tế bào) liên kết với nhau. Mỗi phòng trong những kết nối này có một trọng lượng liên quan. Xây dựng một mạng lưới bao gồm sử dụng một thuật toán để tìm thấy trọng lượng của các kết nối giữa các tế bào. Một mạng lưới có các tế bào của nó được tổ chức trong lớp. Lớp đầu tiên có chứa các tế bào đầu vào của mạng. Các vấn đề đang giải quyết được trình bày vào mạng thông qua các tế bào đầu vào. Lớp cuối cùng gồm có các dự đoán của các mạng lưới đối với trường hợp được trình bày tại các đầu vào của nó. Ở giữa có một hoặc nhiều "ẩn" lớp tế bào. Trọng lượng cập nhật thuật toán, ví dụ như phương pháp lan truyền ngược, cố gắng để có được trọng lượng kết nối tối ưu hóa một tiêu chuẩn lỗi nhất định, đó là trọng lượng đảm bảo rằng sản lượng mạng là phù hợp với các trường hợp trình bày mô hình.

Bây giờ sẽ minh họa làm thế nào để có được một mạng lưới trong R, và cũng có thể làm thế nào để sử dụng các mô hình này để có được những dự đoán. Để đạt được những mục tiêu này, sẽ chia nhỏ dữ liệu trong hai cửa sổ thời gian, được sử dụng để có được mạng lưới và các khác để đánh giá nó, tức là để kiểm tra mô hình dự đoán các giá trị biến mục tiêu. Trong phần trước, sẽ giải quyết triệt để các vấn đề đánh giá các mô hình chuỗi thời gian. Bây giờ hãy chỉ đơn giản là tách dữ liệu trong hai cửa sổ thời gian, bao gồm 20 năm đầu tiên của dữ liệu, và khác với 12 năm còn lại

> ibm. train <- ibm. data[ibm. data$Date < ’1990-01-01’,]

> ibm. test <- ibm. data[ibm. data$Date > ’1989-12-31’,]

Các mạng thường có được kết quả tốt hơn với các dữ liệu bình thường. Tuy nhiên, trong ứng dụng cụ thể, tất cả các biến có quy mô tương tự và không có nghĩa là có khoảng một phân phối bình thường. Vì vậy, sẽ tránh bước này bình thường.

Để có được một mạng lưới để dự đoán 1-ngày trả lại trước tương lai, có thể sử dụng nnet chức năng() như sau,

```
> library(nnet)

> nn <- nnet(r1. f1 ~. , data=ibm. train[, -ncol(ibm. train) ],

+ linout=T, size=10, decay=0. 01, maxit=1000)
```

Hàm này sẽ xây dựng một mạng lưới với một lớp ẩn duy nhất, trong trường hợp này được hình thành bởi 10 đơn vị ẩn(kích thước tham số) . Hơn nữa, các trọng lượng sẽ được học với tỷ lệ trọng lượng cập nhật 0, 01(sự phân rã tham số) . Tham số chỉ ra rằng các biến mục tiêu là liên tục. Tham số maxit thiết lập số lượng tối đa lặp đi lặp lại của các thuật toán hội tụ trọng lượng. Chú ý rằng đã gỡ bỏ các cột cuối cùng của khung dữ liệu ibm. data trong đó có các thông tin ngày. Loại thông tin này là vô ích để xây dựng mô hình như giá trị của nó là khác nhau cho tất cả các trường hợp. Tuy nhiên, cuối cùng có thể xét sử dụng một phần của thông tin này, như số lượng tháng hoặc ngày trong tuần, để cố gắng để nắm bắt các hiệu ứng theo mùa cuối cùng.

Các nnet() chức năng sử dụng các thuật toán lan truyền ngược là cơ sở của một quá trình lặp đi lặp lại của việc cập nhật các trọng số của mạng neural, lên đến tối đa là chu kỳ maxit. Điều này lặp đi lặp lại quá trình này có thể mất một thời gian dài để tính toán cho các bộ dữ liệu lớn.

Hàm trên tạo ra một mạng lưới với 24 đơn vị đầu vào(số lượng các biến dự đoán của vấn đề này) kết nối với 10 đơn vị ẩn, sau đó sẽ được liên kết với một đơn vị đầu ra duy nhất. Điều này dẫn đến một số trong tổng số 261 kết nối. có thể thấy trọng lượng cuối cùng của những kết nối này bằng cách phát hành,

```
> summary(nn)
```

Mạng nơron này có thể được sử dụng để đưa ra dự đoán cho giai đoạn thử nghiệm:

```
> nn. preds <- predict(nn, ibm. test)
```

Các mã sau đây cho là một đồ thị với những tiên đoán vẽ lên biểu đồ đúng giá trị,

```
> plot(ibm. test[, 1], nn. preds, ylim=c(-0. 01, 0. 01),

+ main='Neural Net Results', xlab='True', ylab='NN predictions')
```

> abline(h=0, v=0); abline(0, 1, lty=2)

Dự báo theo đuổi hồi quy

Hãy thử một kỹ thuật hồi quy nhiều, cụ thể là chiếu theo đuổi hồi quy(Friedman, 1981). Một mô hình chiếu theo đuổi có thể được xem như một loại mô hình phụ(Hastie và Tibshirani, 1990) từng kỳ hạn phụ gia là một sự kết hợp tuyến tính của các biến gốc của vấn đề. Điều này có nghĩa rằng một mô hình theo đuổi chiếu là một sự bổ sung từ ngữ được hình thành bởi sự kết hợp tuyến tính của các biến ban đầu. Có thể có được một mô hình với các mã sau đây,

> library(modreg)

> pp <- ppr(r1. f1 ~ . , data=ibm. train[, -ncol(ibm. train)], nterms=5)

> pp. preds <- predict(pp, ibm. test)

Có thể có được một mô hình theo đuổi chiếu bằng cách sử dụng các hàm ppr(). Chức năng này có một số thông số. Nterms tham số thiết lập số lượng các tổ hợp tuyến tính sẽ được bao gồm trong mô hình theo đuổi dự. Bạn cũng có thể sử dụng max. terms tham số để thiết lập một số lượng tối đa các thuật ngữ sẽ được sử dụng. Sau khi thêm con số này lặp đi lặp lại cho đến khi đạt đến một mô hình với nterms.

Như thường lệ với tất cả các mô hình trong R, có thể sử dụng chức năng dự đoán() để có được những dự đoán của mô hình.

Có thể có một cái nhìn tại các hệ số của các tổ hợp tuyến tính hình thành các điều khoản của các mô hình phụ bằng cách phát hành,

> summary(pp)

Call:

ppr. formula(formula = r1. f1 ~ . , data = ibm. train[, -ncol(ibm. train)],

nterms = 5)

Goodness of fit:

5 terms

0

Projection direction vectors:

term 1 term 2 term 3 term 4 term 5

r1. t0 -0. 274593836 -0. 003047629 0. 000000000 0. 000000000 0. 000000000

r1. t1 -0. 232145592 0. 303103072 0. 000000000 0. 000000000 0. 000000000

r1. t2 -0. 125882592 0. 042927734 0. 000000000 0. 000000000 0. 000000000

r1. t3 -0. 277068070 -0. 015256559 0. 000000000 0. 000000000 0. 000000000

r1. t4 0. 237179308 0. 137114309 0. 000000000 0. 000000000 0. 000000000

r1. t5 0. 100427485 -0. 119630099 0. 000000000 0. 000000000 0. 000000000

r1. t6 0. 193712788 -0. 255578319 0. 000000000 0. 000000000 0. 000000000

. . .

. . .

r1. t20 0. 008716963 -0. 014020849 0. 000000000 0. 000000000 0. 000000000

r1. t21 -0. 287156429 0. 138665388 0. 000000000 0. 000000000 0. 000000000

r1. t22 -0. 196584111 0. 181707111 0. 000000000 0. 000000000 0. 000000000

r1. t23 0. 271527937 0. 224461896 0. 000000000 0. 000000000 0. 000000000

Coefficients of ridge terms:

term 1 term 2 term 3 term 4 term 5

0. 001805114 0. 001025505 0. 000000000 0. 000000000 0. 000000000

Như có thể thấy từ kết quả của hàm summary(), ppr() thường xuyên có được một mô hình chỉ có duy nhất hai thuật ngữ. Những thuật ngữ này là sự kết hợp tuyến tính của các biến ban đầu như chúng ta có thể nhìn thấy từ đầu ra này. Điều này có nghĩa rằng các mô hình đã thu được có thể được mô tả bởi phương trình,

r1. f1 = 0. 001805 ×(−0. 2746 × r1. t0 − 0. 23215 × r1. t1 . . .)+

+0. 001026 ×(−0. 00305 × r1. t0 + 0. 30310 × r1. t1 . . .)

3. 3. 2 Đánh giá các mô hình chuỗi thời gian

Do sự phụ thuộc thời gian giữa các quan sát các thủ tục đánh giá cho các mô hình thời gian dự đoán loạt khác nhau từ các phương pháp tiêu chuẩn. Sau này thường dựa trên chiến lược resampling(ví dụ bootstrap hoặc xác nhận chéo) , mà làm việc bằng cách lấy mẫu ngẫu nhiên từ không có thứ tự dữ liệu gốc. Việc sử dụng các phương pháp luận với chuỗi thời gian có thể dẫn đến tình huống không mong muốn như sử dụng quan sát của biến trong tương lai cho việc mục đích đào tạo, và đánh giá các mô hình với dữ liệu quá khứ. Để tránh những vấn đề này, thường chia các dữ liệu chuỗi thời gian có sẵn vào

cửa sổ thời gian, có các mô hình với dữ liệu quá khứ và thử nghiệm nó trên các cắt lớp thời gian tiếp theo.

Mục đích chính của chiến lược đánh giá là để có được một giá trị đáng tin cậy của dự đoán chính xác của một mô hình. Nếu ước tính là đáng tin cậy, có thể được hợp lý tự tin rằng hiệu suất dự đoán của các mô hình sẽ không chệch nhiều từ ước tính khi áp dụng mô hình dữ liệu mới từ cùng một tên miền.

Trong Đồ án này, dữ liệu từ năm 1970 đến giữa năm 2002. Sẽ thiết lập một thời gian t là năm khởi đầu của giai đoạn thử nghiệm. Dữ liệu trước khi t sẽ được sử dụng để có được những mô hình dự báo, trong khi dữ liệu xảy ra sau khi t sẽ chỉ được sử dụng để kiểm tra chúng.

Trong tình huống mô tả ở trên, có một số lựa chọn thay thế có thể xem xét. Đầu tiên là để có được một mô hình duy nhất bằng cách sử dụng các dữ liệu trước khi thời gian tand thử nghiệm nó trên từng trường hợp xảy ra sau khi t. Ngoài ra, có thể sử dụng các chiến lược cửa sổ. Chiến lược cửa sổ đầu tiên mô tả là cửa sổ đang phát triển,

1. Với một loạt Rh(1), Rh(2), . . . , Rh(n) và một thời gian t(<n)

2. Có được một mô hình dự đoán với dữ liệu huấn luyện Rh(1), Rh(2), . . . , Rh(t-1)

3. Lặp lại

4. Có được một dự đoán cho sự quan sát Rh(t)

5. Ghi lại các lỗi dự đoán

6. Rh(t) dữ liệu huấn luyện

7. Có được một mô hình mới với tập huấn luyện mới

8. Hãy t = t + 1

9. Đến khi t = n

Một cách khác là sử dụng một cửa sổ trượt,

1. Với một loạt Rh(1), Rh(2), . . . , Rh(n), thời gian t và kích thước một cửa sổ w

2. Có được một mô hình dự đoán với dữ liệu huấn luyện Rh(t-w-1), . . . , Rh(t-1)

3. Lặp lại

4. Có được một dự đoán cho Rh(t)

5. Ghi lại các lỗi dự đoán

6. Thêm Rh(t) dữ liệu huấn luyện và loại bỏ Rh(- w - t 1)

7. Có được một mô hình dự đoán mới với dữ liệu huấn luyện mới

8. Hãy t = t + 1

9. Đến khi t = n

Các phương pháp tiếp cận cửa sổ có vẻ hợp lý hơn khi họ kết hợp thông tin mới vào mô hình(bằng cách sửa đổi nó) theo thời gian. Sẽ áp dụng chiến lược mô hình duy nhất trong trường hợp này để đơn giản hóa các thí nghiệm. Tuy nhiên, kết quả tốt hơn cuối cùng có thể được thu được với chiến lược cửa sổ, do đó cho các ứng dụng thực tế những điều này không nên bỏ đi.

Thí nghiệm với tập dữ liệu này bao gồm có một mô hình dự đoán với các dữ liệu lên đến 1-Jan-1990, và thử nghiệm nó với các dữ liệu còn lại. Điều này có nghĩa rằng sẽ có được mô hình bằng cách sử dụng khoảng 20 năm dữ liệu và thử nghiệm 12 năm tới. Điều này thiết lập thử nghiệm lớn(khoảng 3100 buổi) đảm bảo một mức độ hợp lý có ý nghĩa thống kê cho các ước tính chính xác. Hơn nữa, khoảng thời gian 12 năm này bao gồm một lượng lớn các điều kiện thị trường, làm tăng sự tự tin về độ chính xác ước tính.

3. 3. 3 Mô hình lựa chọn

Sẽ chia 20 năm của dữ liệu huấn luyện trong hai phần. Phần đầu tiên sẽ bao gồm trong 12 năm đầu(1970-1981) và sẽ được sử dụng để có được các biến thể của mô hình. còn lại 8 năm(1982-1989) sẽ được sử dụng để lựa chọn mô hình "tốt nhất" theo thời gian này lựa chọn. Cuối cùng, mô hình "tốt nhất" sẽ được lấy một lần nữa bằng cách sử dụng đủ 20 năm, dẫn đến kết quả của quá trình khai thác dữ liệu.

> first12y <- nrow(ibm. data[ibm. data$Date < ’1982-01-01’,])

> train <- ibm. train[1:first12y,]

> select <- ibm. train[(first12y+1):nrow(ibm. train),]

Đầu tiên sẽ minh họa điều này chiến lược lựa chọn mô hình bằng cách sử dụng nó để tìm tốt nhất cài đặt cho một kỹ thuật khai thác dữ liệu đặc biệt. Cụ thể, sẽ xem xét các vấn đề về lựa chọn "tốt nhất" thiết lập mạng lưới. Điều chỉnh các mạng lưới thần kinh trước đây đã thu được sử dụng các thiết lập tham số mạng nhất định. Công cụ khai thác dữ liệu nhất có một số loại thông số có thể điều chỉnh để đạt được kết quả tốt hơn. sẽ cố gắng thiết lập nhau cho các thông số kích thước và phân hủy các mạng lưới. Các mã sau cố gắng 12 kết hợp khác nhau của kích thước và sâu, lưu trữ trung bình bình phương lỗi và kết quả tỷ lệ hit trên một khung dữ liệu,

```
> res <- expand. grid(Size=c(5, 10, 15, 20),
+ Decay=c(0. 01, 0. 05, 0. 1),
+ MSE=0,
+ Hit. Rate=0)
> for(i in 1:12) {
+ nn <- nnet(r1. f1 ~ . , data=train[, -ncol(train)], linout=T,
+ size=res[i, 'Size'], decay=res[i, 'Decay'], maxit=1000)
+ nn. preds <- predict(nn, select)
+ res[i, 'MSE'] <- mean((nn. preds-select[, 1])^2)
+ res[i, 'Hit. Rate'] <- hit. rate(nn. preds, select[, 1])
+ }
```

Thử nghiệm này có thể mất một thời gian để chạy trên máy tính kể từ khi đào tạo mạng lưới thường là tính toán chuyên sâu.

Một vài ý kiến trên mã đưa ra ở trên. Các hàm expand. grid() có thể được sử dụng để tạo ra tất cả các kết hợp của các yếu tố khác nhau hoặc các giá trị vector. Đối với mỗi người trong số các biến thể tham số chúng ta có được mạng lưới tương ứng và đánh giá nó về thời gian thời gian lựa chọn mô hình. có thể sử dụng hàm annualized. timeseries. eval() để thực hiện đánh giá này. Tuy nhiên, do số lượng các biến thể đang cố gắng này sẽ lộn xộn trên đầu ra.

Như vậy, đã quyết định sử dụng hai số liệu thống kê đại diện cho các tính năng quan trọng nhất đánh giá là chức năng, cụ thể là tính chính xác hồi quy và dấu hiệu của tính chính xác thị trường. Đối với trước đây đã sử dụng các lỗi trung bình bình phương. đã không được sử dụng hệ số Theil vì đang so sánh các lựa chọn thay thế và do đó so với một yếu tố dự báo cơ bản(như được cung cấp bởi hệ số này) không phải là rất thú vị. Cuối cùng bạn có thể kiểm tra kết quả bằng cách gõ,

```
> res
Size Decay MSE Hit. Rate
1 5 0. 01 0. 0002167111 0. 5005149
2 10 0. 01 0. 0002167079 0. 5025747
3 15 0. 01 0. 0002167094 0. 5025747
```

4 20 0. 01 0. 0002167162 0. 5036045

5 5 0. 05 0. 0002171317 0. 5087539

6 10 0. 05 0. 0002171317 0. 5087539

7 15 0. 05 0. 0002171318 0. 5087539

8 20 0. 05 0. 0002171317 0. 5087539

9 5 0. 10 0. 0002171317 0. 5087539

10 10 0. 10 0. 0002171317 0. 5087539

11 15 0. 10 0. 0002171317 0. 5087539

12 20 0. 10 0. 0002171316 0. 5087539

Các kết quả về độ chính xác hồi quy là khá giống nhau. Tuy nhiên, có thể quan sát cho cùng một sự phân rã thường thu được kết quả tốt nhất với kích thước lớn hơn. Do thực tế là giá trị lớn hơn dẫn đầu phân rã để tăng tỷ lệ nhấn có thể được nghiêng để chọn một mạng lưới với 20 đơn vị thành viên ẩn và sâu 0, 1 như các thiết lập tốt nhất cho dữ liệu .

bây giờ sẽ minh họa điều này cùng một quá trình lựa chọn để tìm ra mô hình "tốt nhất" từ ba lựa chọn thay thế mà đã xem xét. Đối với các mạng lưới thần kinh sẽ sử dụng các thiết lập mà là kết quả của quá trình điều chỉnh. Đối với các mô hình khác sẽ sử dụng các thiết lập tham số như trước để đơn giản hóa mã sau experiments. The có được ba mô hình bằng cách sử dụng 12 năm dữ liệu đầu tiên,

> nn <- nnet(r1. f1 ~ . , data=train[, -ncol(train)], linout=T, size=20, decay=0. 1, maxit=1000)

> nn. preds <- predict(nn, select)

> pp <- ppr(r1. f1 ~ . , data=train[, -ncol(train)], nterms=5)

> pp. preds <- predict(pp, select)

> m <- mars(train[, 2:20], train[, 1])

> m. preds <- predict(m, select[, 2:20])

> naive. returns <- c(train[first12y, 1], select[1:(nrow(select)-1), 1])

After obtaining the models we can compare their performance on the 8 years

left for model selection,

> annualized. timeseries. eval(nn. preds, naive. returns, select)

N Theil HitRate PosRate NegRate Perc. Up Perc. Down

avg 2022 0. 684 0. 509 1 0 1 0

1982 253 0. 687 0. 506 1 0 1 0

1983 253 0. 666 0. 551 1 0 1 0

1984 253 0. 695 0. 492 1 0 1 0

1985 252 0. 760 0. 543 1 0 1 0

1986 253 0. 696 0. 492 1 0 1 0

1987 253 0. 671 0. 528 1 0 1 0

1988 253 0. 670 0. 490 1 0 1 0

1989 252 0. 699 0. 467 1 0 1 0

> annualized. timeseries. eval(pp. preds, naive. returns, select)

N Theil HitRate PosRate NegRate Perc. Up Perc. Down

avg 2022 0. 713 0. 504 0. 490 0. 519 0. 488 0. 512

1982 253 0. 709 0. 506 0. 492 0. 522 0. 498 0. 502

1983 253 0. 687 0. 519 0. 500 0. 541 0. 490 0. 510

1984 253 0. 727 0. 463 0. 450 0. 476 0. 482 0. 518

1985 252 0. 780 0. 490 0. 444 0. 545 0. 452 0. 548

1986 253 0. 722 0. 561 0. 521 0. 600 0. 455 0. 545

1987 253 0. 703 0. 524 0. 562 0. 483 0. 545 0. 455

1988 253 0. 700 0. 485 0. 449 0. 520 0. 474 0. 526

1989 252 0. 748 0. 484 0. 500 0. 469 0. 508 0. 492

> annualized. timeseries. eval(m. preds, naive. returns, select)

N Theil HitRate PosRate NegRate Perc. Up Perc. Down

avg 2022 0. 745 0. 516 0. 495 0. 539 0. 476 0. 524

1982 253 0. 687 0. 545 0. 517 0. 574 0. 482 0. 518

1983 253 0. 664 0. 572 0. 545 0. 606 0. 462 0. 538

1984 253 0. 699 0. 504 0. 533 0. 476 0. 522 0. 478

1985 252 0. 767 0. 465 0. 436 0. 500 0. 456 0. 544

1986 253 0. 709 0. 492 0. 455 0. 528 0. 462 0. 538

1987 253 0. 828 0. 549 0. 538 0. 560 0. 486 0. 514

1988 253 0. 705 0. 519 0. 483 0. 553 0. 470 0. 530

1989 252 0. 707 0. 488 0. 447 0. 523 0. 464 0. 536

Các mô hình mạng lưới thần kinh rõ ràng là tốt nhất trong thời hạn chính xác hồi quy tiên đoán. Tuy nhiên, điều này chính xác hồi quy không được chuyển dịch thành những điểm số tốt về dự đoán là dấu hiệu của sự thay đổi thị trường cho ngày hôm sau. Trên thực tế, mạng lưới thần kinh luôn luôn dự đoán lợi nhuận tích cực dẫn đến độ chính xác 100% vào những chuyển động tích cực của thị trường. Tuy nhiên, từ quan điểm của dự đoán các dấu hiệu của thị trường thay đổi hiệu suất tốt nhất đạt được bằng cách mô hình MARS. Tuy nhiên, đây không phải là kết thúc về đánh giá mô hình. Như chúng ta sẽ thấy trong phần tiếp theo nếu chúng ta dịch các dự đoán của các mô hình kinh doanh thành hành động, chúng ta có thể nhận được thêm thông tin về "giá trị" của những dự đoán.

3. 4 Từ dự đoán kinh doanh thành hành động

Để xây dựng một hệ thống thương mại dựa trên các dự đoán của các mô hình khai thác dữ liệu. Vì vậy, tiêu chí đánh giá cuối cùng sẽ có kết quả kinh doanh thu được từ hệ thống này.

Bước đầu tiên để có được một hệ thống như vậy là để biến đổi các dự đoán của các mô hình kinh doanh thành hành động. Có ba hành động có thể: mua, bán, hoặc giữ . Sử dụng các dự đoán của các mô hình là thông tin duy nhất để quyết định hành động để. Với các loại dữ liệu đang sử dụng để có được mô hình , nó không có ý nghĩa để sử dụng một chiến lược giao dịch trong ngày. Do đó, các quyết định sẽ được thực hiện vào cuối mỗi phiên thị trường, và các đơn đặt hàng cuối cùng sẽ được đăng sau khi phiên làm việc này, nhưng trước khi khai mạc vào ngày hôm sau.

Lý tưởng nhất là sẽ được sử dụng một số mô hình dự đoán với các kịch bản dự báo khác nhau(1 ngày trước 5 ngày trước . . .). Điều này có thể cung cấp cho một dự báo trung hạn của sự phát triển của thị trường và do đó cho phép ra các quyết định liên quan đến hành động tốt nhất vào cuối mỗi ngày.

Sẽ đơn giản hóa cách tiếp cận này, và sẽ sử dụng chỉ 1 dự đoán trước ngày để tạo ra các tín hiệu. Tuy nhiên, một số quyết định phải được thực hiện cho một tập hợp các

dự đoán trước 1 ngày trở lại. Cụ thể, chúng ta phải quyết định khi nào mua hay bán. Các chức năng R sau đây là một phương pháp đơn giản có sử dụng một số ngưỡng trên thị trường dự đoán các biến thể để tạo ra một vector của các tín hiệu kinh doanh,

```
> buy. signals <- function(pred. ret, buy=0. 05, sell=-0. 05) {
+ sig <- ifelse(pred. ret < sell, 'sell',
+ ifelse(pred. ret > buy, 'buy', 'hold'))
+ factor(sig, levels=c('sell', 'hold', 'buy'))
+ }
```

3. 4. 1 Đánh giá các tín hiệu kinh doanh

Các tín hiệu kinh doanh có thể so sánh chúng với các phong trào thị trường hiệu quả để kiểm tra việc thực hiện các hành động khi so sánh với thực tế. Sau đây là một ví dụ nhỏ của sự so sánh này với những tiên đoán MARS cho giai đoạn lựa chọn

```
> mars. actions <- buy. signals(m. preds, buy=0. 01, sell=-0. 01)
> market. moves <- buy. signals(select[, 1], buy=0. 01, sell=-0. 01)
> table(mars. actions, market. moves)
market. moves
mars. actions sell hold buy
sell 7 3 7
hold 377 1200 415
buy 3 8 2
```

Trong ví dụ nhỏ này đã quyết định để tạo ra một tín hiệu mua bất cứ khi nào mô hình dự đoán trước ngày 1-trả về lớn hơn 1%, trong khi tín hiệu bán, nơi tạo ra cho giảm 1%. đã sử dụng hàm table() để có được một bảng dự phòng cho thấy kết quả của các hành động khi so sánh với các phong trào thị trường hiệu quả. Có thể thấy rằng từ 17(= 7 +3 +7) bán hành động được tạo ra bởi mô hình, chỉ có 7 tương ứng với một động thái xuống hiệu quả của thị trường. 3 lần thị trường đóng cửa ở cùng một mức giá, trong khi vào ngày 7 lần thị trường đóng cửa ở mức giá cao hơn, điều đó có nghĩa là nếu đã đăng các lệnh bán sẽ mất tiền!Rõ ràng những con số này chỉ cung cấp cho một nửa của hình ảnh, bởi vì nó cũng có liên quan để quan sát số tiền tương ứng với mỗi đoán(hoặc không chính xác) chính xác. Điều này có nghĩa rằng ngay cả nếu tỷ lệ liên quan đến

biến động thị trường không phải là rất tốt có thể kiếm được tiền. Hai chức năng sau đây cung cấp thêm thông tin về các kết quả của tín hiệu kinh doanh

```
> signals. eval <- function(pred. sig, true. sig, true. ret) {
+ t <- table(pred. sig, true. sig)
+ n. buy <- sum(t['buy', ])
+ n. sell <- sum(t['sell', ])
+ n. sign <- n. buy+n. sell
+ hit. buy <- round(100*t['buy', 'buy']/n. buy, 2)
+ hit. sell <- round(100*t['sell', 'sell']/n. sell, 2)
+ hit. rate <- round(100*(t['sell', 'sell']+t['buy', 'buy'])/n. sign, 2)
+ ret. buy <- round(100*mean(as. vector(true. ret[which(pred. sig=='buy')])), 4)
+ ret. sell <- round(100*mean(as. vector(true. ret[which(pred. sig=='sell')])), 4)
+ data. frame(n. sess=sum(t), acc=hit. rate, acc. buy=hit. buy, acc. sell=hit. sell,
+ n. buy=n. buy, n. sell=n. sell, ret. buy=ret. buy, ret. sell=ret. sell)
+ }
> annualized. signals. results <- function(pred. sig, test) {
+ true. signals <- buy. signals(test[, 1], buy=0, sell=0)
+ res <- signals. eval(pred. sig, true. signals, test[, 1])
+ years <- unique(substr(test[, 'Date'], 1, 4))
+ for(y in years) {
+ idx <- which(substr(test[, 'Date'], 1, 4)==y)
+ res <- rbind(res, signals. eval(pred. sig[idx], true. signals[idx], test[idx, 1]))
+ }
+ row. names(res) <- c('avg', years)
+ res
+ }
> annualized. signals. results(mars. actions, select)
n. sess acc acc. buy acc. sell n. buy n. sell ret. buy ret. sell
avg 2022 46. 67 46. 15 47. 06 13 17 -0. 0926 0. 9472
1982 253 0. 00 0. 00 NaN 1 0 -1. 0062 NaN
1983 253 NaN NaN NaN 0 0 NaN NaN
1984 253 NaN NaN NaN 0 0 NaN NaN
```

1985 252 NaN NaN NaN 0 0 NaN NaN

1986 253 NaN NaN NaN 0 0 NaN NaN

1987 253 40. 00 42. 86 37. 50 7 8 -0. 2367 2. 2839

1988 253 50. 00 33. 33 60. 00 3 5 0. 0254 0. 4773

1989 252 66. 67 100. 00 50. 00 2 4 0. 6917 -1. 1390

Các hàm signals. eval() tính toán tỷ lệ tăng của hành động kinh doanh, số lượng của các loại khác nhau của các hành động, và cũng có tỷ lệ phần trăm lợi nhuận trung bình cho mua và tín hiệu bán. Các hàm annualized. signals. results() cung cấp trung bình một kết quả hàng năm cho những số liệu thống kê. Khi áp dụng chức năng này để hành động MARS, quan sát một hiệu suất rất nghèo. Độ chính xác của các tín hiệu là khá đáng thất vọng và lợi nhuận trung bình thậm chí còn tồi tệ hơn(giá trị âm cho những hành động mua và tích cực cho các tín hiệu bán). Hơn nữa, có là những tín hiệu rất ít tạo ra mỗi năm. Việc thực hiện chỉ chấp nhận được đã đạt được trong năm 1989, mặc dù số lượng thấp của tín hiệu. Nếu áp dụng chức năng này đánh giá các tín hiệu được tạo ra với cùng một ngưỡng từ việc theo đuổi chiếu và dự đoán mạng lưới thần kinh, có được kết quả sau đây,

> annualized. signals. results(buy. signals(pp. preds, buy=0. 01, sell=-0. 01), select)

n. sess acc acc. buy acc. sell n. buy n. sell ret. buy ret. sell

avg 2022 45. 61 44. 12 47. 83 34 23 -0. 4024 -0. 2334

1982 253 42. 86 50. 00 0. 00 6 1 -0. 4786 1. 4364

1983 253 100. 00 NaN 100. 00 0 1 NaN -1. 8998

1984 253 100. 00 NaN 100. 00 0 1 NaN -1. 8222

1985 252 NaN NaN NaN 0 0 NaN NaN

1986 253 0. 00 0. 00 0. 00 4 2 -0. 7692 0. 1847

1987 253 48. 00 47. 37 50. 00 19 6 -0. 7321 -0. 1576

1988 253 70. 00 75. 00 66. 67 4 6 1. 7642 -0. 7143

1989 252 28. 57 0. 00 33. 33 1 6 -0. 8811 0. 2966

> annualized. signals. results(buy. signals(nn. preds, buy=0. 01, sell=-0. 01), select)

n. sess acc acc. buy acc. sell n. buy n. sell ret. buy ret. sell

avg 2022 NaN NaN NaN 0 0 NaN NaN

1982 253 NaN NaN NaN 0 0 NaN NaN

1983 253 NaN NaN NaN 0 0 NaN NaN

1984 253 NaN NaN NaN 0 0 NaN NaN

1985 252 NaN NaN NaN 0 0 NaN NaN

1986 253 NaN NaN NaN 0 0 NaN NaN

1987 253 NaN NaN NaN 0 0 NaN NaN

1988 253 NaN NaN NaN 0 0 NaN NaN

1989 252 NaN NaN NaN 0 0 NaN NaN

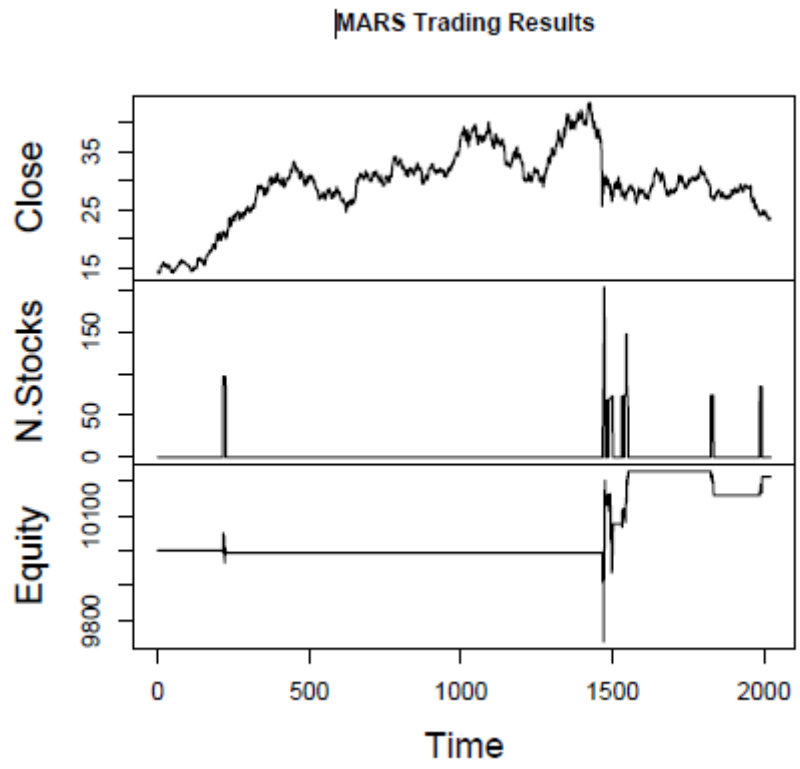
3. 4. 2 Mô phỏng thương mại

Một khó khăn đầu tiên là vốn sẵn có vào lúc bắt đầu của thời kỳ kinh doanh. Trong thí nghiệm, sẽ bắt đầu với số vốn 10. 000 đơn vị tiền tệ. Một vấn đề quan trọng khi thiết kế một chính sách kinh doanh là thời gian của các đơn đặt hàng gửi đến thị trường. Sẽ gửi các lệnh mua vào cuối mỗi phiên mỗi khi một tín hiệu mua được tạo ra bởi mô hình. Số vốn sẽ được đầu tư một tỷ lệ phần của vốn chủ sở hữu(được định nghĩa là số tiền có sẵn cộng với giá trị của danh mục đầu tư cổ phiếu hiện tại). Sẽ giả định rằng lệnh mua này sẽ luôn luôn được thực hiện khi thị trường mở cửa. Điều này có nghĩa rằng đơn đặt hàng sẽ được thực hiện ở mức giá mở cửa của ngày tiếp theo. Ngay sau khi gửi bài này lệnh mua, sẽ gửi một lệnh bán của các cổ phiếu vừa mua. Lệnh này sẽ được đăng với một mức giá bán mục tiêu, mà sẽ được thiết lập như là một tỷ lệ phần trăm(sẽ sử dụng 3% như là mặc định) ở trên giá đóng cửa của ngày hôm nay của cổ phiếu. Điều này có thể được xem như là mục tiêu lợi nhuận . Lệnh bán cũng sẽ được đi kèm với một ngày hết hạn(mặc định là 10 ngày). Nếu giá cổ phiếu đạt mức giá bán mục tiêu cho đến ngày đáo hạn các cổ phiếu được bán ở mức giá đó. Nếu không, các cổ phiếu được bán với giá đóng cửa của phiên giao dịch ngày hết hạn. Cuối cùng, sẽ thiết lập một chi phí giao dịch 5 Euro cho mỗi lệnh. Chú ý rằng chiến lược này là chỉ sử dụng các tín hiệu mua của các mô hình , bởi vì ngay khi mua sẽ tự động gửi một lệnh bán liên quan.

Các thành phần kinh doanh là một khung dữ liệu với các hàng như nhiều như có các phiên họp trong giai đoạn thử nghiệm. Đối với mỗi hàng (phiên), chúng tôi lưu trữ ngày, giá đóng cửa của các cổ phiếu vào cuối phiên, số tiền có sẵn tại cuối ngày, số lượng cổ phiếu hiện tại danh mục đầu tư và vốn chủ sở hữu (tiền + giá trị của các cổ phiếu). Có thể sử dụng khung dữ liệu để kiểm tra chặt chẽ hơn

(ví dụ như thông qua đồ thị) thực hiện của thương mại. Các mã sau đây là một ví dụ kiểm tra đồ họa như vậy, thể hiện trong hình:

```
> plot(ts(t$trading[, c(2, 4, 5)]), main='MARS Trading Results')
```



Hình 3. 1 Một số kết quả kinh doanh của các hành động kéo theo bởi Neural Network dự đoán.

Thật hấp dẫn khi nhìn vào các kết quả hàng năm:

```
> annualized. trading. results <- function(market, signals, . . . ) {
+
+ res <- data. frame(trader. eval(market, signals, . . . )[-1])
+
+ years <- unique(substr(market[, 'Date'], 1, 4))
+ for(y in years) {
+ idx <- which(substr(market[, 'Date'], 1, 4)==y)
+ res <- rbind(res, data. frame(trader. eval(market[idx, ], signals[idx], . . . )[-1]))
+ }
+ row. names(res) <- c('avg', years)
+ round(res, 3)
+ }
```

If we apply this function to the trading actions of MARS we get the following results,

```
> annualized. trading. results(market, mars. actions)
N. trades N. profit Perc. profitable N. obj Max. L Max. P PL Max. DrawDown
avg 13 9 69. 231 9 6. 857 5. 797 211. 818 312. 075
1982 1 0 0. 000 0 0. 359 0. 000 -7. 120 85. 440
1983 0 0 NaN 0 0. 000 0. 000 0. 000 0. 000
1984 0 0 NaN 0 0. 000 0. 000 0. 000 0. 000
```

1985 0 0 NaN 0 0. 000 0. 000 0. 000 0. 000
1986 0 0 NaN 0 0. 000 0. 000 0. 000 0. 000
1987 7 5 71. 429 5 6. 857 5. 797 85. 087 266. 203
1988 3 3 100. 000 3 0. 000 2. 875 149. 341 38. 580
1989 2 1 50. 000 1 3. 462 2. 625 -17. 029 69. 150
Avg. profit Avg. loss Avg. PL Sharpe. Ratio
avg 2. 853 3. 725 0. 829 0. 011
1982 NaN 0. 359 -0. 359 -0. 004
1983 NaN NaN NaN NaN
1984 NaN NaN NaN NaN
1985 NaN NaN NaN NaN
1986 NaN NaN NaN NaN
1987 3. 100 5. 543 0. 631 0. 014
1988 2. 510 NaN 2. 510 0. 087

1989 2. 625 3. 462 -0. 419 -0. 015

Chức năng này có thể mất một thời gian để chạy. Có thể nhận thấy rằng trong một vài năm không có giao dịch ở tất cả. Hơn nữa, trong một số năm qua là một mất mát (1982 và 1989). Kết quả tổng thể là không ấn tượng. Chức năng này sử dụng giả lập kinh doanh để có được kết quả. Có thể cung cấp cho các tham số kinh doanh khác để kiểm tra hiệu quả trên các kết quả

> annualized. trading. results(market, mars. actions, bet=0. 1, exp. prof=0. 02, hold. time=15)
N. trades N. profit Perc. profitable N. obj Max. L Max. P PL Max. DrawDown
avg 13 9 69. 231 10 5. 685 4. 258 34. 672 197. 031
1982 1 1 100. 000 1 0. 000 2. 028 20. 115 5. 000
1983 0 0 NaN 0 0. 000 0. 000 0. 000 0. 000
1984 0 0 NaN 0 0. 000 0. 000 0. 000 0. 000
1985 0 0 NaN 0 0. 000 0. 000 0. 000 0. 000
1986 0 0 NaN 0 0. 000 0. 000 0. 000 0. 000
1987 7 4 57. 143 5 5. 685 4. 258 -22. 738 197. 031
1988 3 3 100. 000 3 0. 000 1. 353 29. 594 21. 560
1989 2 1 50. 000 1 0. 321 1. 108 7. 701 34. 160
Avg. profit Avg. loss Avg. PL Sharpe. Ratio
avg 1. 553 2. 624 0. 268 0. 004
1982 2. 028 NaN 2. 028 0. 045
1983 NaN NaN NaN NaN
1984 NaN NaN NaN NaN

1985 NaN NaN NaN NaN

1986 NaN NaN NaN NaN
1987 1. 959 3. 392 -0. 334 -0. 006
1988 1. 001 NaN 1. 001 0. 051

1989 1. 108 0. 321 0. 393 0. 016

Như chúng ta có thể thấy mọi thứ thậm chí còn tồi tệ hơn với các thiết lập này. Tuy nhiên, những kết quả tổng thể nghèo chỉ củng cố mối quan tâm trước đây về chất lượng của các dự đoán của các mô hình.

3. 5 Các kết quả trên bộ dữ liệu

Khám phá kiến thức là một quá trình tuần hoàn. Các kết quả của các bước khám phá kiến thức khác nhau thường được cho ăn trở lại để cố gắng cải thiện hiệu suất tổng thể.

Các mô hình, đã có được dự đoán trở lại vào ngày hôm sau chỉ được sử dụng tụt lại giá trị của tỉ lệ này. Nó được phổ biến kiến thức (ví dụ như Almeida và Torgo, Năm 2001; Deboeck, 1994)) rằng điều này rõ ràng là không đủ để dự đoán chính xác trong phạm vi của một phi tuyến tính như báo giá cổ phiếu. Như vậy, chúng ta sẽ cố gắng để cải thiện hiệu suất bằng cách làm phong phú thêm các dữ liệu thiết lập thông qua việc sử dụng các thông tin khác nhau từ trở về trước.

Các thiết lập của các biến dự đoán

Một khả năng là cố gắng nắm bắt một số tính năng của hành vi năng động của dòng thời gian. Ngoài ra, người ta cũng có thể cố gắng bao gồm một số thông tin tài chính như là đầu vào cho các mô hình. Một sự lựa chọn phổ biến là việc sử dụng các chỉ số kỹ thuật. Ý tưởng đằng sau Phân tích kỹ thuật là nghiên cứu sự tiến hóa của giá trong những ngày trước và sử dụng thông tin này để tạo ra các tín hiệu. Hầu hết các phân tích thường được thực hiện thông qua biểu đồ. Có một số lượng lớn các chỉ số kỹ thuật có thể được sử dụng để làm phong phú thêm thiết lập của các biến. Trong phần này sẽ sử dụng một vài ví dụ minh họa:

Chỉ số kỹ thuật

Một di chuyển trung bình là một chỉ báo cho thấy giá trị trung bình của giá chứng khoán trong một khoảng thời gian. Phương pháp phổ biến nhất của việc sử dụng chỉ số này để so sánh di chuyển trung bình của giá chứng khoán với giá cả an ninh của chính nó. Một tín hiệu mua được tạo ra khi giá chứng khoán tăng lên trên mức trung bình trượt của nó và tín hiệu bán được tạo ra khi giá chứng khoán giảm xuống dưới mức trung bình này di chuyển. Một di chuyển trung bình có thể dễ dàng thu được với các chức năng sau đây,

```
> ma <- function(x, lag) {
+ require('ts')
+ c(rep(NA, lag), apply(embed(x, lag+1), 1, mean))
```

+ }

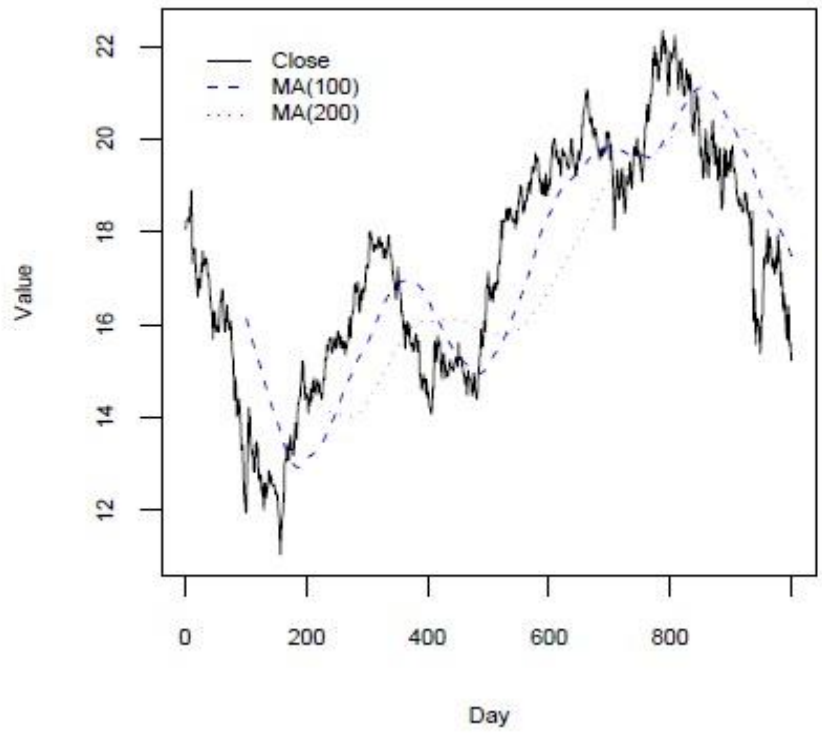
Khi hàm embed() được làm sẵn bởi ts gói. Điều này được thực hiện thông qua hàm require(). .

Nếu muốn để có được một di chuyển trung bình của 20 ngày qua, giá đóng cửa của cổ phiếu IBM, có thể gõ lệnh sau đây:

> ma20. close <- ma(ibm$Close, lag=20)

Có thể có được một tốt hơn giảm việc sử dụng các đường trung bình động như các chỉ số kỹ thuật bằng cách vẽ một vài trong số họ với độ trễ thời gian khác nhau. Một ví dụ hình 3. 2, trong đó cho thấy, giá đóng cửa của IBM cùng với hai đường trung bình khác nhau di chuyển cho 1000 phiên chứng khoán đầu tiên. Các mã để tạo ra các con số là sau đây,

```
> plot(ibm$Close[1:1000], main=’’, type=’l’, ylab=’Value’, xlab=’Day’)
> lines(ma20. close[1:1000], col=’red’, lty=2)
> lines(ma(ibm$Close, lag=100)[1:1000], col=’blue’, lty=3)
> legend(1. 18, 22. 28, c("Close", "MA(20)", "MA(100)"),
+ col = c(’black’, ’red’, ’blue’), lty = c(1, 2, 3))
```



Hình 3. 2 Hai chuyển động trung bình của giá đóng cửa của IBM

Như đã đề cập ở trên có thể so sánh các dòng di chuyển trung bình, giá đóng cửa để có được tín hiệu kinh doanh. Thay vì nhìn vào đồ thị, chúng ta có thể xây dựng một chức năng để có được các tín hiệu,

```
> ma. indicator <- function(x, ma. lag=20) {
+ d <- diff(sign(x-c(rep(NA, ma. lag-1), apply(embed(x, ma. lag), 1, mean))))
+ factor(c(rep(0, ma. lag), d[!is. na(d)]),
+ levels=c(-2, 0, 2), labels=c('sell', 'hold', 'buy'))
        + }
```

Sử dụng chức năng này, Có thể có được những hành động kinh doanh đề nghị của một số chỉ số trung bình động. Các mã sau đây cho thấy một ví dụ của việc sử dụng ngày 20-di chuyển trung bình cho giao dịch trên 1000 phiên đầu tiên, và trình bày ngày 10 tín hiệu mua đầu tiên được tạo ra bởi chỉ số này,

```
> trading. actions <- data. frame(Date=ibm$Date[1:1000],
+ Signal=ma. indicator(ibm$Close[1:1000]))
> trading. actions[which(trading. actions$Signal=='buy'), ][1:10, ]
Date Signal
29 1970-02-11 buy
58 1970-03-25 buy
69 1970-04-10 buy
104 1970-05-29 buy
116 1970-06-16 buy
138 1970-07-17 buy
147 1970-07-30 buy
162 1970-08-20 buy
207 1970-10-23 buy
210 1970-10-28 buy
```

Có thể đánh giá các loại tín hiệu kinh doanh với hàm annualized. trading. results() như thực hiện trước khi cho các tín hiệu được tạo ra bởi một mô hình MARS. Điều này được minh họa bằng đoạn mã dưới đây, cho các tín hiệu được tạo ra bởi ngày 30-di chuyển trung bình trong khoảng thời gian lựa chọn mô hình,

```
> ma30. actions <- data. frame(Date=ibm[ibm$Date < '1990-01-01', 'Date'],
+ Signal=ma. indicator(ibm[ibm$Date < '1990-01-01', 'Close'], ma. lag=30))
> ma30. actions <- ma30. actions[ma30. actions$Date > '1981-12-31', 'Signal']
> annualized. trading. results(market, ma30. actions)
N. trades N. profit Perc. profitable N. obj Max. L Max. P PL Max. DrawDown
avg 96 38 39. 583 2 9. 195 8. 608 -1529. 285 1820. 19
1982 10 5 50. 000 2 7. 194 8. 608 207. 665 243. 28
1983 15 5 33. 333 0 4. 585 3. 563 -300. 700 385. 66
1984 13 5 38. 462 0 4. 246 3. 222 -375. 140 391. 56
```

1985 12 2 16. 667 0 9. 168 3. 520 -399. 420 428. 33
1986 10 4 40. 000 0 6. 218 1. 895 -223. 870 295. 71
1987 11 6 54. 545 0 4. 015 5. 043 23. 170 179. 33
1988 11 5 45. 455 0 5. 054 1. 777 -140. 190 226. 79
1989 14 7 50. 000 0 6. 827 4. 608 -358. 330 481. 30
Avg. profit Avg. loss Avg. PL Sharpe. Ratio
avg 2. 003 2. 738 -0. 861 -0. 059
1982 5. 261 3. 112 1. 074 0. 047
1983 1. 594 2. 332 -1. 023 -0. 087
1984 1. 389 3. 272 -1. 479 -0. 152
1985 1. 932 2. 433 -1. 706 -0. 124
1986 1. 475 2. 880 -1. 138 -0. 078
1987 2. 180 2. 360 0. 117 0. 007
1988 0. 770 1. 823 -0. 644 -0. 038

1989 1. 363 3. 962 -1. 299 -0. 111

Đã sử dụng các dữ liệu trước khi bắt đầu giai đoạn lựa chọn mô hình (gọi lại nó là từ 1981/12/31 đến 1990/01/01), để chúng ta có thể bắt đầu tính toán di chuyển trung bình từ khi bắt đầu của thời kỳ đó.

Đúng như dự đoán chỉ số này rất đơn giản, không tạo ra kết quả kinh doanh rất tốt tổng thể. Tuy nhiên, nên nhớ lại rằng mục tiêu khi giới thiệu các chỉ số kỹ thuật để sử dụng chúng như là các biến đầu vào cho mô hình hồi quy của chúng tôi như là một phương tiện để cải thiện độ chính xác dự báo. Có ba cách để làm điều này: Có thể sử dụng các giá trị của di chuyển trung bình như là một biến đầu vào, có thể sử dụng các tín hiệu được tạo ra bởi các chỉ báo như là một biến đầu vào, hoặc có thể sử dụng cả hai giá trị và các tín hiệu như là các biến đầu vào . Bất kỳ của các lựa chọn thay thế có tiềm năng hữu ích. Các quan sát tương tự cũng được áp dụng cho việc sử dụng của bất kỳ chỉ số kỹ thuật.

ema_(Xt) = _Xt + (1 − _) × ema_(Xt−1)
ema_(X1) = X1 (3. 9)

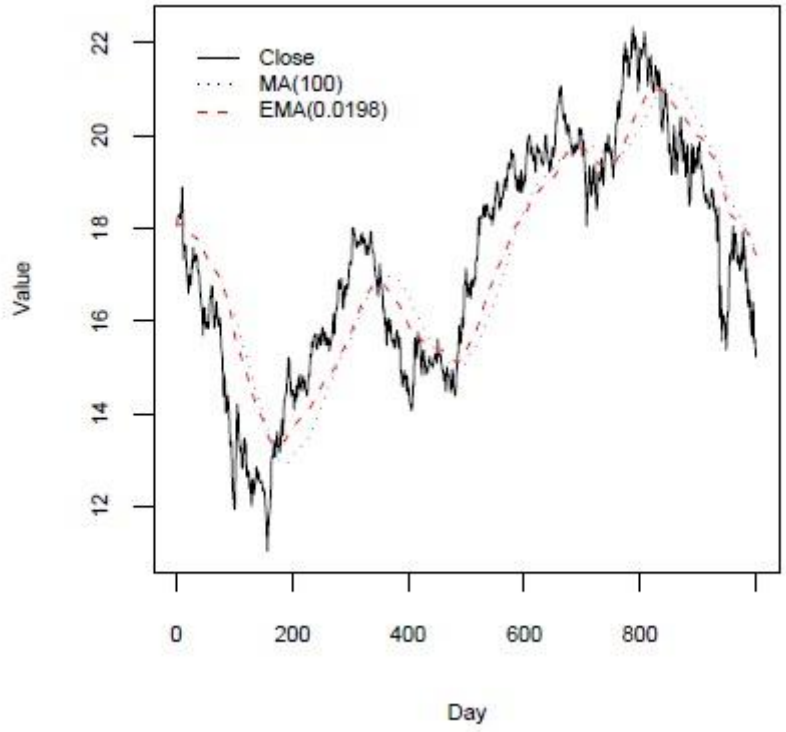
Trung bình hàm mũ chuyển động có thể dễ dàng thực hiện trong R với mã sau đây,

> ema <- function (x, beta = 0. 1, init = x[1])
+ {
+ require(’ts’)
+ filter(beta*x, filter=1-beta, method="recursive", init=init)

+ }

Đôi khi người ta thích sử dụng số ngày để thiết lập khoảng thời gian của một di chuyển trung bình, thay vì sử dụng yếu tố này.

Sự khác biệt giữa trung bình động 100-ngày và trung bình theo cấp số nhân qua cửa sổ cùng một lúc, được thu được với các mã sau đây,



Hình 3. 3 100 ngày và trung bình theo cấp số nhân

```
> plot(ibm$Close[1:1000], main=’’, type=’l’, ylab=’Value’, xlab=’Day’)
> lines(ma(ibm$Close[1:1000], lag=100), col=’blue’, lty=3)
> lines(ema(ibm$Close[1:1000], beta=0. 0198), col=’red’, lty=2)
> legend(1. 18, 22. 28, c("Close", "MA(100)", "EMA(0. 0198)"),
    + col = c(’black’, ’blue’, ’red’), lty = c(1, 3, 2), bty=’n’)
```

Như có thể thấy, không có sự khác biệt lớn giữa hai đường trung bình, mặc dù có thể nói rằng trung bình hàm mũ là "phản ứng" với các biến động gần đây của giá đóng cửa.

Có một số cách để giải thích các giá trị MACD với mục tiêu chuyển đổi chúng thành tín hiệu giao dịch. Một cách là sử dụng sự giao nhau của chỉ số chống lại một đường tín hiệu. Sau đó là thường là một trung bình 9-ngày theo cấp số nhân di chuyển. Các quy tắc giao dịch kết quả là như sau: Bán khi MACD giảm xuống dưới đường tín hiệu của nó và mua khi MACD tăng trên đường tín hiệu của nó. Điều này có thể được thực hiện bởi các mã:

```
> macd. indicator <- function(x, long=26, short=12, signal=9) {
+ v <- macd(x, long, short)
```

```
+ d <- diff(sign(v-ema(v, lambda=1/(signal+1))))
+ factor(c(0, 0, d[-1]), levels=c(-2, 0, 2), labels=c('sell', 'hold', 'buy'))
        + }
```

Một chỉ số khác phổ biến là Relative Strength Index (RSI). giá sau đây dao động khoảng từ 0 đến 100. Nó được tính như,

$$\text{RSI} = 100 - \left(\frac{100}{1 + \frac{U}{D}}\right) \qquad (3.\ 6)$$

Ở đây, U là tỷ lệ phần trăm lợi nhuận tích cực trong một khoảng thời gian nhất định, và D là tỷ lệ phần trăm lợi nhuận tiêu cực trong thời gian đó. Chỉ số này có thể được thực hiện bởi các đoạn code sau đây:

```
> moving. function <- function(x, lag, FUN, . . . ) {
+ require('ts')
+ FUN <- match. fun(FUN)
+ c(rep(NA, lag), apply(embed(x, lag+1), 1, FUN, . . . ))
+ }
> rsi. aux <- function(diffs, lag) {
+ u <- length(which(diffs > 0))/lag
+ d <- length(which(diffs < 0))/lag
+ ifelse(d==0, 100, 100-(100/(1 + u/d)))
+ }
> rsi <- function(x, lag=20) {
+ d <- c(0, diff(x))
+ moving. function(d, lag, rsi. aux, lag)
        + }
```

RSI-kinh doanh dựa trên quy tắc là như sau: Khi RSI vượt qua giá trị 70 và từ phía, tạo ra một tín hiệu bán khi chỉ số này vượt qua giá trị 30 đến từ bên dưới tạo ra một tín hiệu bán. Quy tắc này được thực hiện bởi các chức năng sau đây,

```
> rsi. indicator <- function(x, lag=20) {
+ r <- rsi(x, lag)
+ d <- diff(ifelse(r > 70, 3, ifelse(r<30, 2, 1)))
+ f <- cut(c(rep(0, lag), d[!is. na(d)]), breaks=c(-3, -2, -1, 10),
+ labels=c('sell', 'buy', 'hold'), right=T)
+ factor(f, levels=c('sell', 'hold', 'buy'))
        + }
```

Chỉ số này bao gồm các thông tin về khối lượng giao dịch bên cạnh việc giá cổ phiếu. Nó có thể được tính toán với hai chức năng sau đây,

```
> ad. line <- function(df) {
```

```
+ df$Volume*((df$Close-df$Low) - (df$High-df$Close))/(df$High-df$Low)
+ }
> chaikin. oscillator <- function(df, short=3, long=10) {
+ ad <- ad. line(df)
+ ewma(ad, lambda=1/(short+1))-ewma(ad, lambda=1/(long+1))
+ }
```

Các chỉ số kỹ thuật cung cấp một số thông tin tài chính theo định hướng xây dựng mô hình. Tuy nhiên, cũng có thể sử dụng các biến cố gắng nắm bắt các tính năng khác của sự năng động của chuỗi thời gian. Ví dụ, có thể sử dụng không chỉ 1-ngày trở lại mà còn thông tin trên lớn hơn lợi nhuận bị tụt hậu, giống như 5, 10 - và 20-ngày trở lại. Có thể tính toán sự thay đổi của giá đóng cửa phiên qua. Một biến khác mà đôi khi hữu ích là một sự khác biệt về sự khác biệt. Ví dụ, có thể tính toán sự khác biệt của lợi nhuận . Cuối cùng, cũng có thể tính toán xu hướng tuyến tính của giá trong những phiên gần đây.

R mã sau thực hiện một số các biến

```
> d5. returns <- h. returns(ibm[, 'Close'], h=5)
> var. 20d <- moving. function(ibm[, 'Close'], 20, sd)
> dif. returns <- diff(h. returns(ibm[, 'Close'], h=1))
```

KẾT LUẬN

Trong Đồ án này, em vận dụng Khai phá dữ liệu với R xây dựng lên chương trình Predicting Stock Market Returns. Kết quả đạt được bao gồm:

- Tìm hiểu về Ngôn Ngữ R.

- Tìm hiểu Khai Phá Dữ Liệu.

- Dự đoán Trả lại thị trường chứng khoán.

Xây dựng chương trình và cài đặt thử nghiệm với một số dữ liệu chạy thông suốt, cho ra kết quả.

Qua quá trình làm Đồ án, em đã học thêm nhiều kiến thức thực tế và biết vận dụng kiến thức đã học để giải quyết bài toán đặt ra. Tuy nhiên kết quả còn rất hạn chế, cần có sự hỗ trợ nhiều của thầy cô giáo. Để có khả năng làm việc tốt vận dụng lý thuyết vào thực hành và có kỹ thuật nhất định, em thấy cần phải thực hành và vận dụng kiến thức nhiều hơn nữa.

TÀI LIỆU THAM KHẢO

- Tài liệu Tiếng Việt

1. Phân tích số liệu và biểu đồ bằng R cuả Nguyễn Văn Tuấn

2. Phát hiện tri thức và khai phá dữ liệu, Phan Đình Diệu Khoa CNTT, ĐHQG Hà nội, 12/2/1998

- Tài liệu Tiếng Anh

3. Data Mining with Rattle and R by Graham Williams (25 Feb 2011)

4. Data Mining with R by Torgo and Luis (11 Sep 2010)

5. Data Mining with R: Learning with Case Studies by Luis Torgo (19 Nov 2010)

- Tài Liệu Internet

6. http://vietsciences. free. fr/nhipcaubandoc/diemsach/phantichsolieu-bieudoR. htm

7. http://www. liaad. up. pt/~ltorgo/DataMiningWithR/

8. http://www. amazon. co. uk/Mining-Chapman-Knowledge-Discovery-ebook/dp/B005HOIINK/ref=sr_1_fkmr0_3?s=books&ie=UTF8&qid=1353918127&sr=1-3-fkmr0