

Київський національний університет імені Тараса Шевченка
Кафедра теорії ймовірностей, статистики та актуарної математики

Р. Майборода

Методичні рекомендації по курсу

“Комп’ютерна статистика”

Для студентів магістратури за напрямом “статистика”

*Індивідуальні завдання
та рекомендації по виконанню*
Робоча версія від 21.09.2020

Київ — 2020

Вступ

Для виконання завдань потрібно встановити R та RStudio на своєму комп'ютері.

Щоб встановити R для Windows зайдіть на сторінку

<http://cran.r-project.org/bin/windows/base/>

і виберіть **Download R 3.4.1 for Windows** (номер версії, скоріше за все, буде вже іншим). Після цього запустіть програму, яка буде завантажена на ваш комп'ютер і відповідайте на її запити.

Якщо вам потрібна версія R для іншої операційної системи, зайдіть на сторінку

<http://www.r-project.org/>

і виберіть там варіант, який вас влаштовує.

Для того, щоб встановити RStudio, зайдіть на сторінку

www.rstudio.com

і виберіть там варіант для завантаження. Встановлювати RStudio треба після того, як буде встановлено R.

Мої книжки по регресійному аналізу та комп'ютерній статистиці можна знайти на моїй персональній сторінці на сайті кафедри теорії ймовірностей та математичної статистики. Там само — поточна версія цього документу за адресою:

<http://probability.univ.kiev.ua/userfiles/mre/tasksCSmag5.pdf>

Теоретичні питання курсу

1. Метод найменших квадратів у лінійній регресії

Поняття регресійної моделі. Структурна і функціональна регресія. Умови L_2 регресії і гауссова регресія. Метод найменших квадратів для лінійної регресії, його геометричний зміст. Прогноз за методом найменших квадратів. Алгебраїчне зображення оцінок методу найменших квадратів. Незміщеність МНК-оцінок і їх коваріаційна матриця ([2] п. 2.1-2.3; [3] п. 10.1).

2. Відбір регресорів на основі тесту Фішера в \mathbb{R}

Загальне поняття статистичного тесту, основна та альтернативна гіпотези, ймовірності помилок першого і другого роду, досягнутий рівень значущості тесту. ([3] п. 9.1; [2] додаток С.5.) Тест Фішера для перевірки загальної лінійної гіпотези і його застосування для перевірки залежності від хоча б одного з регресорів або від конкретного регресора ([2] п. 2.4, приклад 2.4.3; [3] п. 10.2).

3. Діагностика лінійних регресійних моделей. Залишки, важелі і відстані Кука

Залишки (residuals), прогноз (prediction), діаграма прогноз-залишки (residual-prediction scatterplot), QQ-діаграма для залишків, викиди (outlier), студентизовані залишки (Studentized residuals), важелі (leverage), відстані Кука (Cook distance) — [9] п. 10.2, 10.6; [3] п. 10.2.; [2] додаток D.2.

4. Поняття про мультиколінеарність

Строга і нестрога мультиколінеарність ([2], 3.4)

5. Спектральний та сингулярний розклад матриці. Матриця Мура-Пенроуза

Лінійні простори, лінійні оператори та їх матриці, власні числа і власні вектори матриці (оператора). Спектральний розклад симетричної матриці. Сингулярний розклад довільної матриці. Знаходження функцій від матриці з використанням спектрального розкладу. Псевдообернена матриця Мура-Пенроуза. ([2] додаток А.1; [9] додаток А.10, А.12; [7] п. 3.4.1).

6. Теорема про функції, що дозволяють оцінку. Діагностика нестрогої мультиколінеарності

[2], 3.4.

7. Теорема про оптимальну проекцію. Поняття головних компонент вибірки

Простір перших k власних векторів матриці інерції як розв'язок задачі оптимального вибору простору проекції. Головні напрямки вибірки за коваріаційною та кореляційною матрицями. ([2], 3.5.)

8. Вибір вимірності простору проєкції. Регресія на головні компоненти

Частка збереженої дисперсії, діаграма власних чисел. Техніка регресії на головні компоненти. ([2], 3.5; [7] п. 3.5.1.)

9. Поняття про рідж-регресію. Теорема про покращення МНК-оцінок

Рідж-регресія як метод покращення МНК оцінок у випадку мульти-колінеарності. Середньоквадратична похибка рідж-оцінок. ([2], 3.6; [7] п. 3.3.4.1)

10. Крос-валідація для рідж-регресії. Візуалізація результатів рідж-регресії

Поняття про техніку крос-валідації. Крос-валідація для рідж-регресії. Реалізація в R і візуалізація результатів. ([2], 3.6; [7] п. 3.4.1)

11. Рідж- і лассо- регресія у формі Лагранжа

([7] п. 3.4.2)

12. Відбір оптимальної множини регресорів на основі критерію Мелоуза

Теоретичний та емпіричний критерії Мелоуза. Відбір регресорів на основі діаграми Мелоуза. ([2] п. 3.7)

13. Алгоритми регресії назад і вперед

([2], 3.7; [7] п.3.3)

14. Навантажений метод найменших квадратів у гетероскедастичній регресії

Поняття про гетероскедастичну регресію. Оптимальність навантаженого методу найменших квадратів для гетероскедастичної регресії. Двокроковий МНК. ([2], 3.1)

15. Регресія з похибками, що утворюють процес авторегресії ([2], 3.1 - підрозділ “авторегресія залишків”).

Завдання 1. Аналіз впливу у лінійній регресії.

Перша частина

1. Отримайте файл з даними про котирування на американських фондових біржах акцій компаній, що входять до індексу S&P 500. Файл-каталог можна завантажити з сайту компанії Quantquote:

`quantquote.com/files/quantquote_daily_sp500_83986.zip`

Розпакуйте цей архів у зручний для вас каталог (теку). Дані по кожній компанії містяться в окремому файлі. Імена файлів містять скорочені назви компаній, наприклад, `table_ibm.csv` — файл з даними про котирування акцій компанії IBM. Список компаній з їх скороченими та повними назвами і сферою їх діяльності можна подивитись тут:

`en.wikipedia.org/wiki/List_of_S%26P_500_companies`

Кожен файл містить таблицю у форматі csv з семи стовпчиків:

- дата біржових торгів (формат rrrrmmdd) — `dat`
- індикатор — `z`,
- ціна відкриття — `orp`,
- максимальне ціна — `mx`,
- мінімальна ціна — `mn`,
- ціна закриття — `clo`,
- обсяг продаж — `vol`.

2. Знайдіть дані по компаніях, що відповідають вашому варіанту: це компанії, які нумерації за скороченими назвами в алфавітному порядку мають номери від $10N-9$ до $10N$ де N — номер вашого варіанту. (За бажанням, можна вибрати інші компанії, які вас цікавлять і узгодити список з викладачем). Виділіть відповідні файли у окремий каталог. Надалі ви будете працювати тільки з ними.

3. Далі Ви будете працювати зі змінною `clo` для цих компаній. За бажанням, можна спробувати включити у аналіз інші змінні для компаній з Вашого варіанту, або перетоврення цих змінних, наприклад, `log-returns`.

4. Виберіть одну компанію, `clo` якої ви будете прогнозувати і використайте `clo` інших компаній з лагом -1 для прогнозування. (Тобто Ви будете прогнозувати ціну закриття компанії на одну біржову сесію вперед). Побудуйте прогноз використовуючи лінійну регресійну модель. Розгляньте два варіанти:

- використання всіх наявних даних крім останніх 10 сесій (повні дані);

— використання даних по 50 сесіях, що передують 10-ти останнім (останні данні).

5. Виконайте підгонку за методом найменших квадратів регресійної моделі для прогнозування обраної вами змінної-відгуку за повними і за останніми даними. Перевірте, наскільки значущою є залежність відгуку від всіх регресорів, оцініть якість прогнозу на основі коефіцієнта детермінації. Визначіть, для яких регресорів спостерігається значуща залежність відгуку від них у обраній моделі регресії. При потребі вилучіть не впливові регресори і повторіть підгонку.

Друга частина.

6. Проведіть аналіз залишків та аналіз впливу у ваших моделях. Відобразіть бульбашкову діаграму впливу. При потребі, вилучіть з даних впливові спостереження і повторіть підгонку.

7. Перевірте якість прогнозу на 20-ти останніх даних, порівняйте результати підгонки моделі за повними і останніми даними.

Результати і висновки опишіть у звіті.

Рекомендації по виконанню завдання 1.

перша частина

Прочитати дані з різних файлів і зібрати стовпчики `сlo` в один фрейм даних можна наступним чином:

```
> # у filenames --- повні імена всіх файлів, що лежать у каталозі C:\\rem\\d
> filenames=list.files(path="C:\\rem\\d", full.names=TRUE)
> # читаємо файли, виймаємо 1-ший і 6-й стовпчики і кладемо в окремий фрейм
> datalist = lapply(filenames,
+ function(x){x0<-read.csv(file=x,header=F)[,c(1,6)];
+ colnames(x0)<-c("data",
+ unlist(strsplit(x,"[_.]"))[2]);# назва компанії стає назвою стовчика
+ x0})
> # зливаємо фрейми в один:
> y<-Reduce(function(x,y) {merge(x,y,by="data")}, datalist)
```

(Аналогічно можна вибрати інші змінні і включити їх у той же фрейм даних, або створити для них окремий фрейм).

Створюємо файл з даними для підгонки регресійної моделі для ціни компанії `adi` за даними по всіх інших компаніях:

```
> Data<-y[-nrow(y),-1]
> Data$adi<-y$adi[-1]
```

Підганяємо модель за даними по останніх 50 сесіях, виводимо таблицю результатів:

```
> nn<-nrow(Data)
> model1<-lm(adi ~ . - adi, data=Data[(nn-50):nn,])
> summary(model1)
```

Call:

```
lm(formula = adi ~ . - adi, data = Data[(nn - 50):nn, ])
```

Residuals:

| | Min | 1Q | Median | 3Q | Max |
|--|----------|----------|---------|---------|---------|
| | -1.66667 | -0.36419 | 0.01634 | 0.43573 | 1.02146 |

Coefficients:

| | Estimate | Std. Error | t value | Pr(> t) |
|-------------|-----------|------------|---------|------------|
| (Intercept) | 1.291750 | 8.911019 | 0.145 | 0.88544 |
| aapl | 0.024811 | 0.008355 | 2.970 | 0.00491 ** |
| abbv | 0.374864 | 0.200480 | 1.870 | 0.06849 . |
| abc | 0.238507 | 0.119236 | 2.000 | 0.05196 . |
| abt | 0.117824 | 0.225605 | 0.522 | 0.60424 |
| ace | 0.016992 | 0.087651 | 0.194 | 0.84722 |
| acn | -0.095824 | 0.061724 | -1.552 | 0.12806 |
| act | -0.019595 | 0.044026 | -0.445 | 0.65854 |
| adbe | 0.203669 | 0.147952 | 1.377 | 0.17594 |

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.6067 on 42 degrees of freedom

Multiple R-squared: 0.8987, Adjusted R-squared: 0.8795

F-statistic: 46.6 on 8 and 42 DF, p-value: < 2.2e-16

Як бачимо за таблицею результатів, при підгонці моделі виявлена значуща залежність відгуку від регресорів ($p\text{-value} < 2.2e-16$), коефіцієнт детермінації (Multiple R-squared: 0.8987) достатньо високий для практичного застосування прогнозу. Значуща залежність відгуку на рівні $\alpha = 0.05$ виявлена лише від регресора aapl.

Можна спробувати вилучити всі інші регресори і подивитись, як це вплине на якість прогнозування:

```
> model2<-lm(adi~aapl,data=Data[(nn-50):nn,])
> summary(model2)
```

Call:

```
lm(formula = adi ~ aapl, data = Data[(nn - 50):nn, ])
```

Residuals:

| | Min | 1Q | Median | 3Q | Max |
|--|---------|---------|--------|--------|--------|
| | -3.0190 | -1.1500 | 0.2514 | 1.3650 | 2.0429 |

Coefficients:

| | Estimate | Std. Error | t value | Pr(> t) | |
|-------------|----------|------------|---------|----------|-----|
| (Intercept) | 24.60980 | 4.82548 | 5.100 | 5.49e-06 | *** |
| aapl | 0.05187 | 0.01121 | 4.628 | 2.74e-05 | *** |

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 1.472 on 49 degrees of freedom

Multiple R-squared: 0.3042, Adjusted R-squared: 0.2899

F-statistic: 21.42 on 1 and 49 DF, p-value: 2.735e-05

Як бачимо, у цьому випадку залежність від регресора виявляється, але якість прогнозу різко знижується (коефіцієнт детермінації 0.3042). Отже можливо, що у відкинутих регресорах була інформація, важлива для побудови прогнозу. Але можливо також, що використання зайвих регресорів дало ефект “перепідгонки”.

Друга частина роботи

Для графічного аналізу результатів підгонки скористайтесь

```
plot(model1)
```

Для виведення бульбашкової діаграми впливу —

```
> library(car)
> influencePlot(model1)
```

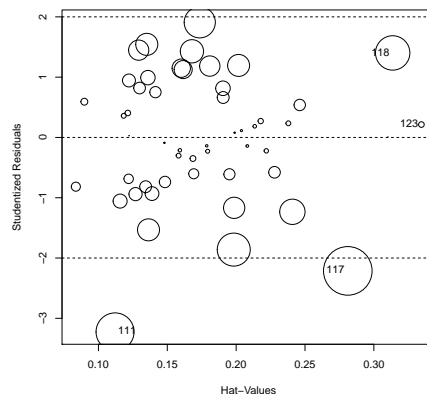



Рис. 1: Діаграма впливу

| | StudRes | Hat | CookD |
|-----|------------|-----------|-------------|
| 111 | -3.2251296 | 0.1120708 | 0.119190052 |
| 117 | -2.2132836 | 0.2810901 | 0.194738756 |
| 118 | 1.4019047 | 0.3136802 | 0.097563208 |
| 123 | 0.2138565 | 0.3345383 | 0.002614004 |

(Результат див. на рис. 1).

Для порівняння якості прогнозів на “нових даних” потрібно використати тестову частину вибірки - у завданні це 20 останніх елементів набору даних, які не враховувались при підгонці прогнозу. Підраховуємо різниці між справжніми значеннями відгуку і його прогнозами на основі моделі і відображаємо їх на одному рисунку для різних моделей. Наприклад, для моделей `model1`, `model2`, підігнаних вище, це можна зробити так:

```
> U1 <- Data$adi[(nn-60):(nn-51)]-predict(model1,Data[(nn-60):(nn-51),])
> U2 <- Data$adi[(nn-60):(nn-51)]-predict(model2,Data[(nn-60):(nn-51),])
> plot(U1,type="l",col="red",ylim=c(min(U1,U2),0))
> lines(U2,col="blue")
```

(див. рис. 2) Тут тестова частина даних це десять спостережень перед тими, за якими проводилась підгонка моделей. На рисунку помітно, що друга модель (синім кольором) дає гірші результати, ніж перша (червоним).

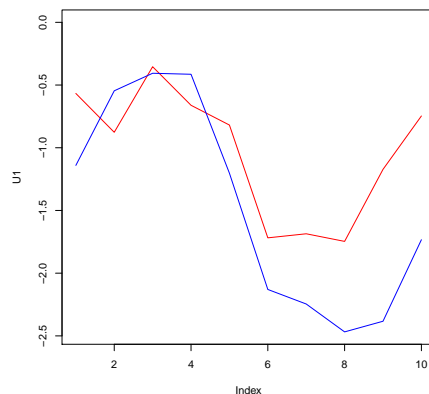


Рис. 2: Діаграма впливу

Завдання 2. Регресія на головні компоненти.

1. Це завдання можна виконувати на даних, отриманих у п. 1-2 завдання 1.

Як і у завданні 1, відділіть 20 останніх спостережень для перевірки якості підгнаних моделей (тестова частина вибірки). Для підгонки `PCmodel1` використовуйте всі спостереження крім останніх 20. Для підгонки моделі `PCmodel1` — 50 спостережень, що передують 20-ти останнім.

2. За даними, виділеними для підгонки, потрібно підігнати лінійну регресійну модель з проекцією на головні компоненти для прогнозування з лагом 1 ціни одних акцій з вашого набору даних (відгук) за цінами всіх інших акцій з цього набору.

3. Спочатку проведіть аналіз головних компонент набору регресорів, виберіть вимірність простору проекції. При цьому можна використовувати головні компоненти як коваріаційної, так і кореляційної матриці.

4. Використовуючи обрані головні компоненти як регресори, проведіть підгонку лінійної регресійної моделі для прогнозування відгуку.

5. Пункти 3-4 виконайте для `PCmodel1` і `PCmodel11`. Порівняйте результати підгонки на тестовій частині вибірки. Виберіть модель, яку ви вважаєте найкращою — `PCbest`.

6. Порівняйте модель `PCbest` з найкращою моделлю, отриманою у завданні 1.

Зробіть висновки.

У звіті потрібно отриманий найкращий прогноз з проекцією на головні компоненти записати безпосередньо через початкові регресори (тобто через ціни акцій, які використовуються для прогнозування).

Рекомендації по виконанню завдання 2.

Для проведення аналізу головних компонент можна скористатись функцією `princomp()`. Наприклад, тут проводиться аналіз головних компонент набору 7-ми змінних з набору `Data`, отриманого у завданні 1, вміщеного у фреймі даних `X`:

```
> X<-Data[,2:8]
> PC<-princomp(X,cor=T)
> summary(PC)
```

Importance of components:

| | Comp.1 | Comp.2 | Comp.3 | Comp.4 | Comp.5 |
|------------------------|-------------|-------------|------------|------------|------------|
| Standard deviation | 2.3793874 | 0.8622529 | 0.51488820 | 0.39101270 | 0.30420317 |
| Proportion of Variance | 0.8087835 | 0.1062114 | 0.03787284 | 0.02184156 | 0.01321994 |
| Cumulative Proportion | 0.8087835 | 0.9149949 | 0.95286774 | 0.97470930 | 0.98792924 |
| | Comp.6 | Comp.7 | | | |
| Standard deviation | 0.251939463 | 0.144989034 | | | |
| Proportion of Variance | 0.009067642 | 0.003003117 | | | |
| Cumulative Proportion | 0.996996883 | 1.000000000 | | | |

Як бачимо, перша компонента пояснює 80% розкиду даних, друга додатково — іще 10%, третя — 3.8% а всі інші — значно менше. Тому розглядати проекцію на простір більше 3-х компонент не доцільно.

Нарисуємо діаграму власних чисел:

```
> plot(PC)
```

(див. рис. 3). Злам на ній відбувається після першого власного числа.

Отже, можна розглянути моделі з однією, або з трьома першими компонентами.

Навантаження на головні компоненти можна подивитись так:

```
> loadings(PC)
```

Loadings:

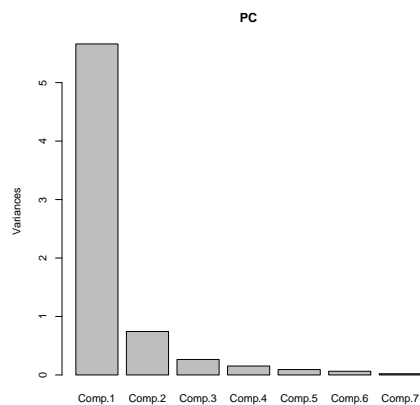


Рис. 3: Діаграма власних чисел

| | Comp.1 | Comp.2 | Comp.3 | Comp.4 | Comp.5 | Comp.6 | Comp.7 |
|------|--------|--------|--------|--------|--------|--------|--------|
| abbv | 0.409 | | | | | 0.893 | 0.162 |
| abc | 0.408 | -0.230 | | | 0.223 | | -0.854 |
| abt | 0.376 | 0.355 | -0.142 | 0.796 | | -0.264 | |
| ace | 0.386 | -0.208 | -0.508 | -0.252 | -0.682 | -0.122 | |
| acn | 0.298 | 0.782 | | -0.518 | 0.120 | -0.125 | |
| act | 0.372 | -0.174 | 0.830 | | -0.291 | -0.184 | 0.148 |
| adbe | 0.386 | -0.367 | -0.179 | -0.165 | 0.619 | -0.258 | 0.456 |

| | Comp.1 | Comp.2 | Comp.3 | Comp.4 | Comp.5 | Comp.6 | Comp.7 |
|----------------|--------|--------|--------|--------|--------|--------|--------|
| SS loadings | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 |
| Proportion Var | 0.143 | 0.143 | 0.143 | 0.143 | 0.143 | 0.143 | 0.143 |
| Cumulative Var | 0.143 | 0.286 | 0.429 | 0.571 | 0.714 | 0.857 | 1.000 |

Завдання 3. Рідж-егресія.

1. Це завдання можна виконувати на даних, отриманих у п. 1-2 завдання 1.

Як і у завданні 1, відділіть 20 останніх спостережень для перевірки якості підігнаних моделей (тестова частина вибірки). Для підгонки `RidgeModel1` використайте всі спостереження крім останніх 20. Для підгонки моделі `RidgeModel2` — 50 спостережень, що передують 20-ти останнім.

2. За даними потрібно підігнати модель регресії для прогнозування ціни акцій однієї фірми (відгук) за даними про інші фірми (регресори) з лагом 1 використовуючи техніку рідж-регресії.

3. Використайте функціонал крос-валідації для вибору оптимального значення параметра регуляризації. Відобразіть графік залежності функціоналу крос-валідації від значення параметра регуляризації. На графіку вертикальною лінією відмітьте обране оптимальне значення.

4. Відобразіть графіки залежності оцінок коефіцієнтів від параметра регуляризації. Опишіть помічені вами особливості їх поведінки.

5. Побудуйте прогнози на основі останніх 20 сесій і перевірте їх якість, порівнявши зі справжніми значеннями прогнозованих цін.

6. Порівняйте результати з найкращими моделями, отриманими при виконанні завдань 1 і 2.

Зробіть висновки.

У звіті потрібно вказати отримані коефіцієнти оптимальної моделі рідж-регресії, значення параметра регуляризації, що був використаний при її підгонці і відповідне значення функціоналу крос-валідації.

Рекомендації по виконанню завдання 3.

Для підгонки лінійної регресійної моделі за методом рідж-регресії можна скористатись функцією `lm.ridge()` з бібліотеки `MASS`.

Наприклад, на даних, описаних у завданні 1, підгонка за даними по 50 сесіях, що передують 20 останнім, може виглядати так:

```
> last<-nrow(Data)-21 # номер останньої сесії для підгонки
> library(MASS)
> # підгонка за методом рідж-регресії,
> #       у lambda --- набір значень параметра регуляризації:
> fit <- lm.ridge(adi ~ ., data=Data[(last-50):last,],
+ lambda = seq(0.001, 50, .01))
```

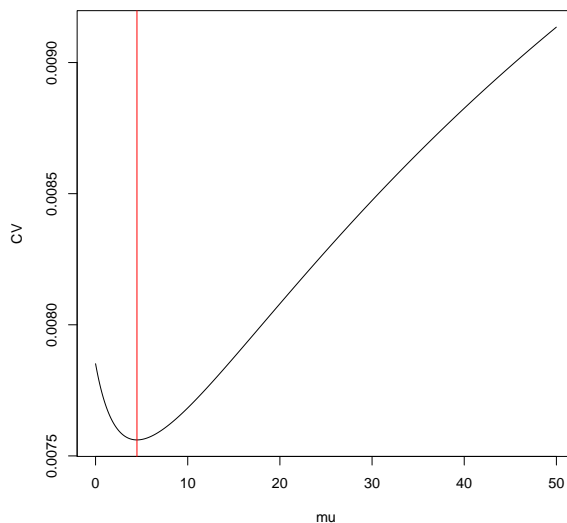


Рис. 4: CV як функція від параметра регуляризації.

```
> # графік значень крос-валідації
> plot(fit$lambda, fit$GCV, type="l", xlab="mu", ylab="CV")
> # номер мінімального значення CV:
> i<-which.min(fit$GCV)
> abline(v=fit$lambda[i], col="red")
```

(див. рис. 4).

Для рисування графіка залежності коефіцієнтів від параметра регуляризації можна скористатись функцією `matplot()`:

```
> matplot(fit$lambda, t(fit$coef), type="l", col=1:8, lty=1:8,
+         xlab="mu", ylab="coefficients")
> legend("topright", col=1:9, legend=colnames(Data)[1:8], lty=1:8)
```

(див. рис. 5).

Для підрахунку прогнозу можна використовувати матричне множення `%*%`. При цьому слід враховувати, що у моделі використовується адитивна константа, яку потрібно розраховувати і додавати окремо. Крім того, регресори при побудові моделі нормуються (середньоквадратичним відхиленням). Значення нормуючих констант знаходяться у атрибуті `$scales`

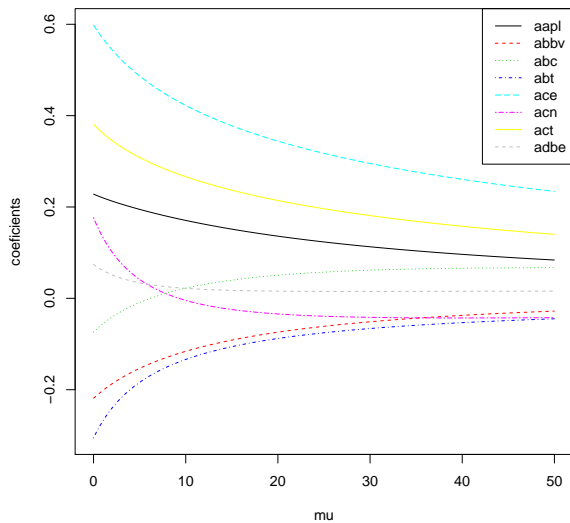


Рис. 5: Коефіцієнти як функції від параметра регуляризації.

результату виконання функції `lm.ridge()`. Для того, щоб отримати коефіцієнти у моделі з не нормованими регресорами, потрібно розділити значення атрибуту `$coef` на значення `$scales`. Наприклад, при виконанні наступних дій:

```
coefs<-matrix(fit$coef[,i]/fit$scales,ncol=1)
```

отримуємо значення коефіцієнтів у не моделі з не нормованими регресорами, які відповідають i -тому значенню параметра регуляризації. Значення коефіцієнтів тут записані у вигляді вектора-стовпчика (матриці з одного стовпчика).

Завдання 4. Вибір оптимального набору регресорів.

1. Це завдання можна виконувати на даних, отриманих у п. 1-2 завдання 1.

Як і у завданні 1, відділіть 20 останніх спостережень для перевірки якості підігнаних моделей (тестова частина вибірки). Для підгонки `SrModel1` використайте всі спостереження крім останніх 20. Для підгонки моделі `SrModel2` — 50 спостережень, що передують 20-ти останнім.

2. За даними потрібно підігнати модель регресії для прогнозування ціни акцій однієї фірми (відгук) за даними про інші фірми (регресори) з лагом 1, відібравши оптимальний набір регресорів з використанням критерія Ср Меллоуза (Mallows C_p).

3. Для вибору оптимального набору регресорів нарисуйте діаграму $p-C_p$ для наборів кращих моделей із заданою кількістю регресорів. Виберіть оптимальний набір, виходячи з мінімізації C_p з урахуванням його надійності як оцінки для теоретичного критерія Меллоуза.

4. Підрахуйте значення коефіцієнтів для вибраних вами оптимальних моделей `SrModel1` і `SrModel2`. Знайдіть прогнози для відгуку на тестовій частині вибірки на основі цих моделей. Виведіть діаграми для порівняння прогнозів зі справжніми значеннями відгуку.

5. Порівняйте моделі `SrModel1` і `SrModel2` з отриманими у попередніх завданнях. Чи мають вони якісь переваги?

Зробіть висновки.

Рекомендації по виконанню завдання 4.

Для вибору оптимальних наборів регресорів можна скористатись функцією `regsubsets()` з пакета `leaps`. Цій функції потрібно передати як параметри:

`formula` — формула, що описує специфікацію моделі регресії з максимальним набором регресорів;

`data` — фрейм даних зі значеннями відгука та регресорів;

`nbest` — кількість найкращих наборів, які функція виведе для кожної заданої кількості регресорів;

`nvmax` — максимальна кількість регресорів, які можуть бути включені до набору.

Результат роботи функції записується у об'єкт, який потрібно обробити функцією `summary()`, щоб отримати звіт з наступними атрибутами:

`$which` — матриця логічних значень. Кожен рядочок цієї матриці від-

повідляє одному вибраному набору. Довжина дорівнює довжині рядочка у фреймі даних (+1, якщо у модель включено вільний член). Значення TRUE відповідають змінним з фрейму, які включені до даного набору регресорів. Назви рядочків матриці `$which` дорівнюють кількості регресорів у відповідному наборі.

`$cp` — вектор значень C_p Меллуза, що відповідають наборам з матриці `$which`.

Якщо потрібні значення оцінок коефіцієнтів моделі, яка відповідає i -тому рядочку матриці `which`, їх можна отримати викликавши функцію `coef()` —

```
coef(object, i),
```

де `object` — результат виконання функції `regsubsets()`

Наприклад:

```
> library(leaps)
> Data<-y[, -1]
> res<-regsubsets(adi~., data=Data, nbest=3, nvmax=9)
> p<-apply(summary(res)$which, 1, sum)+1
> Cp<-summary(res)$cp
> plot(p, Cp)
> abline(0, 1, col="red")
> plot(p, Cp, ylim=c(5, 15))
> abline(0, 1, col="red")
> #index<-identify(p, Cp)
> index<-c(10, 13) # замість identify
> text(p[index], Cp[index], labels=index, col="blue", adj=1)
> summary(res)$which[index,] # регресори, що входять у моделі
```

```
(Intercept) aapl abbv abc abt ace acn act adbe
4          TRUE TRUE TRUE FALSE FALSE TRUE FALSE TRUE FALSE
5          TRUE TRUE TRUE FALSE TRUE TRUE FALSE TRUE FALSE
```

```
> Cp[index]
```

```
[1] 5.845019 5.470112
```

```
> coef(res, 10) # коефіцієнти 10-ї моделі
```

```
(Intercept)          aapl          abbv          ace          act
-2.234622164  0.008892002 -0.400491604  0.593619333  0.072031246
```

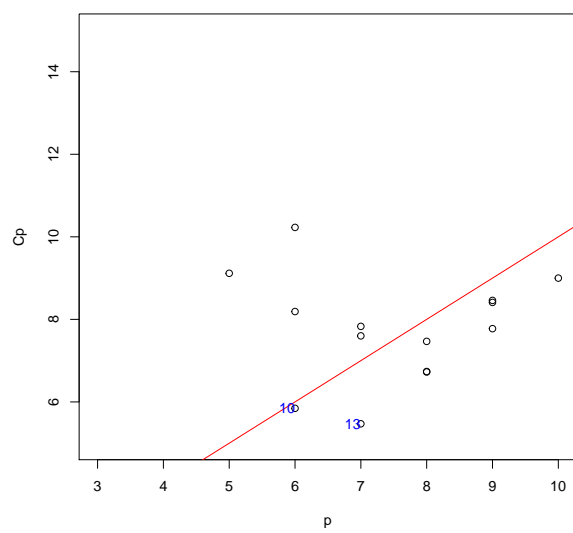
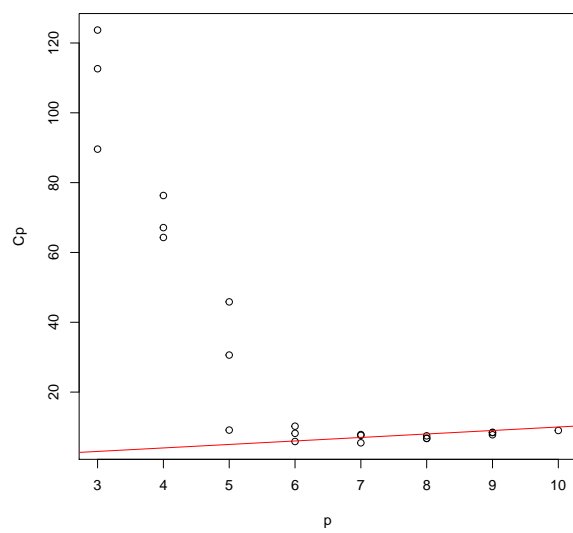


Рис. 6: Графік Cp Меллуза.

Результат виконання — на рис. 6. На першому (верхньому) рисунку точки, що розмістились вгорі (недопідігнані моделі) заважають аналізувати моделі, які можна вибрати як найкращі. Тому на нижньому рисунку відображені лише точки з $5 < C_p < 15$. Хорошими кандидатами для вибору є моделі 13 (з найменшим можливим $C_p = 5.470112$) і 10 — з $C_p = 5.845019$. 13-та модель відрізняється від 10-ї лише включенням змінної **abt**.

Оскільки точка 13 лежить помітно нижче прямої $p = C_p$, можна запідозрити, що для неї C_p не є хорошою оцінкою для теоретичного критерію Мелуза. Тому як остаточну модель приймемо 10-ту, у якій C_p не на багато більше, але вона знаходиться практично на лінії $p = C_p$.

Завдання 5. Гетероскедастична регресія

1. Згенерувати дані обсягу $n=1000$ спостережень, що описуються гетероскедастично. регресійною моделлю $y_j = a + bx_j + \varepsilon_j$, де ε_j – незалежні похибки з нормальним розподілом з нульовим математичним сподіванням і дисперсією $g(x)$, x має нормальний розподіл з математичним сподіванням m і дисперсією s .

2. Знайти оцінки параметрів a і b за даними, згенерованими у п. 1, використовуючи адаптивний двокроковий метод найменших квадратів у якому (1) на роль пілотної оцінки (оцінки першого кроку) використовується оцінка методу найменших квадратів (2) для підгонки залежності дисперсій похибок від регресора x використовується поліноміальна регресія другого порядку.

3. Вивести діаграму розсіювання даних, діаграми розсіювання прогноз-залишки для оцінок першого і другого кроку та порівняти значення отриманих оцінок зі справжніми значеннями параметрів.

4. Повторити кроки 2-3 1000 разів, отримати вибірки з 1000 оцінок першого і другого кроку для параметрів. Підрахувати зміщення та дисперсії оцінок, зробити висновок про те, наскільки оцінки 2-го кроку точніші, ніж оцінки 1-го кроку. Провести аналогічне дослідження для інших обсягів вибірки: $n=25, 50, 100, 500, 2000$. Зробити висновок про доцільність використання двокрокових оцінок при різних обсягах вибірки.

Дані для індивідуального завдання вибрати з таблиці.

| Варіант | a | b | $g(x)$ | m | s |
|---------|-----|------|----------------|-----|-----|
| 1 | 1 | 2 | $(x - 1)^2$ | 1 | 1 |
| 2 | -2 | 1 | $(x + 1)^2$ | -1 | 1 |
| 3 | 1 | 0.5 | $0.5(x - 1)^2$ | 1 | 2 |
| 4 | -2 | 0.5 | $0.5(x - 1)^2$ | -1 | 2 |
| 5 | -1 | 2 | $(x - 1)^2$ | 1 | 1 |
| 6 | 2 | -1 | $(x + 1)^2$ | -1 | 1 |
| 7 | -1 | 0.5 | $0.5(x - 1)^2$ | 1 | 2 |
| 8 | 2 | -0.5 | $0.5(x - 1)^2$ | -1 | 2 |
| 9 | 0.5 | -0.5 | $(x - 1)^2$ | -1 | 2 |
| 10 | -1 | -0.5 | $0.5(x + 1)^2$ | -1 | 1 |

Литература

- [1] Карташов М.В. "Імовірність, процеси, статистика".— К: ВПЦ "Київський університет", 2007. — 494 с.
- [2] Майборода Р.Є. Регресія: Лінійні моделі.— К.: ВПЦ "Київський університет" 2007. — 296с.
- [3] Майборода Р.Є. "Комп'ютерна статистика".— К.: ВПЦ "Київський університет", 2019. — 589 с.
- [4] Майборода Р.Є., Сугакова О.В. "Аналіз даних за допомогою пакета R". , 2015. — 65 с.
- [5] Себер Дж. Линейный регрессионный анализ.— М.: Мир, 1980.— 456с.
- [6] Турчин В.М. Теорія ймовірностей і математична статистика.- Дніпропетровськ, ІМА-пресс, 2014. — 566 с.
- [7] Hastie T., Tibshirani R., Friedman J.The Elements of Statistical Learning.—Springer, 2013.— 764 p.
- [8] James G., Witten D., Hastie T., Tibshirani R. An Introduction to Statistical Learning with Applications in R.— Springer NY 2013.— 440p.
- [9] Seber G., Lee A. Linear regression analysis.—Wiley, 2003. — 558p.