МИРЭА
Российский технологический университет

Институт технологий управления

Программные средства анализа данных

01.03.05 Статистика

Профиль «Анализ данных в бизнесе и экономике»

Квалификация «бакалавр»

Бурцева Татьяна Александровна

Профессор кафедры статистики и математических методов в управлении

Burceva_t@mirea.ru

Москва, 2022

МИРЭА
Российский технологический университет

Тема 5. Тесты на гетероскедастичность и мультиколинеарность. Метод главных компонент в R.

План лекции

1. Тесты на гетероскедастичность в R.
2. Тест на мультиколинеарность в R.
3. Пример реализации метода главных компонент и визуализации его результатов.

Предпосылки МНК (условия Гаусса–Маркова)

1 . Математическое ожидание случайного отклонения ε_i равно нулю: $M(\varepsilon_i) = 0$ для всех наблюдений.

Данное условие означает, что случайное отклонение в среднем не оказывает влияния на зависимую переменную. В каждом конкретном наблюдении случайный член может быть либо положительным, либо отрицательным, но он не должен иметь систематического смещения.

2 . Дисперсия случайных отклонений ε_i постоянна:

$D(\varepsilon_i) = D(\varepsilon_j) = \sigma^2$ *для любых наблюдений i и j.*

Данное условие подразумевает, что несмотря на то, что при каждом конкретном наблюдении случайное отклонение может быть либо большим, либо меньшим, не должно быть некой априорной причины, вызывающей большую ошибку (отклонение).

Выполнимость данной предпосылки называется *гомоскедастичностью (постоянством дисперсии отклонений).* Невыполнимость данной предпосылки называется *гетероскедастичностью (непостоянством дисперсий отклонений).*

3 . Случайные отклонения ε_i и ε_j являются независимыми друг от друга для $i \neq j$.

Выполнимость данной предпосылки предполагает, что отсутствует систематическая связь между любыми случайными отклонениями. Другими словами, величина и определенный знак любого случайного отклонения не должны быть причинами величины и знака любого другого отклонения.

если данное условие выполняется, то говорят об отсутствии *автокорреляции.*

4 . Случайное отклонение должно быть независимо от объясняющих переменных.

Обычно это условие выполняется автоматически при условии, что объясняющие переменные не являются случайными в данной модели.

Следует отметить, что выполнимость данной предпосылки не столь критична для эконометрических моделей.

5 . Модель является линейной относительно параметров.

Теорема Гаусса–Маркова. Если предпосылки *1* – *5* выполнены, то оценки, полученные по МНК, обладают следующими свойствами:

1. Оценки являются несмещенными, т. е. $M(b_0) = в_0, M(b_1) = в_1$. Это вытекает из того, что $M(e_i) = 0$ и говорит об отсутствии систематической ошибки в определении положения линии регрессии.
2. Оценки состоятельны, т. к. дисперсия оценок параметров при возрастании числа n наблюдений стремится к нулю: $D(b_0) \xrightarrow[n\to\infty]{} 0$, $D(b_1) \xrightarrow[n\to\infty]{} 0$. Другими словами, при увеличении объема выборки надежность оценок увеличивается (b_0 наверняка близко к β_0, b_1 – близко к β_1).
3. Оценки эффективны, т. е. они имеют наименьшую дисперсию по сравнению с любыми другими оценками данных параметров, линейными относительно величин y_i.

В англоязычной литературе такие оценки называются *BLUE (Best Linear Unbiased Estimators) – наилучшие линейные несмещенные оценки.*

Если предпосылки *2* и *3* нарушены, т. е. дисперсия отклонений непостоянна и (или) значения e_i, e_j связаны друг с другом, то свойства несмещенности и состоятельности сохраняются, но свойство эффективности – нет.

Наряду с выполнимостью указанных предпосылок при построении классических линейных регрессионных моделей делаются еще некоторые предположения :

- объясняющие переменные не являются случайными величинами;
- случайные отклонения имеют нормальное распределение;
- число наблюдений существенно больше числа объясняющих переменных;
- отсутствуют ошибки спецификации;
- отсутствует мультиколлинеарность.

МИРЭА
Российский технологический университет

1. Тесты на гетероскедастичность в R

Гетероскедастичность остатков

- это **непостоянство дисперсии остатков**, которое также приводит к снижению эффективности применения уравнения регрессии.

Для её выявления используются различные критерии - критерий Голдфелда-Квандта, тест ранговой корреляции Спирмена и д.р.

Гомоскедастичность остатков – предпосылка МНК

- Для каждого x дисперсия остатков одинакова

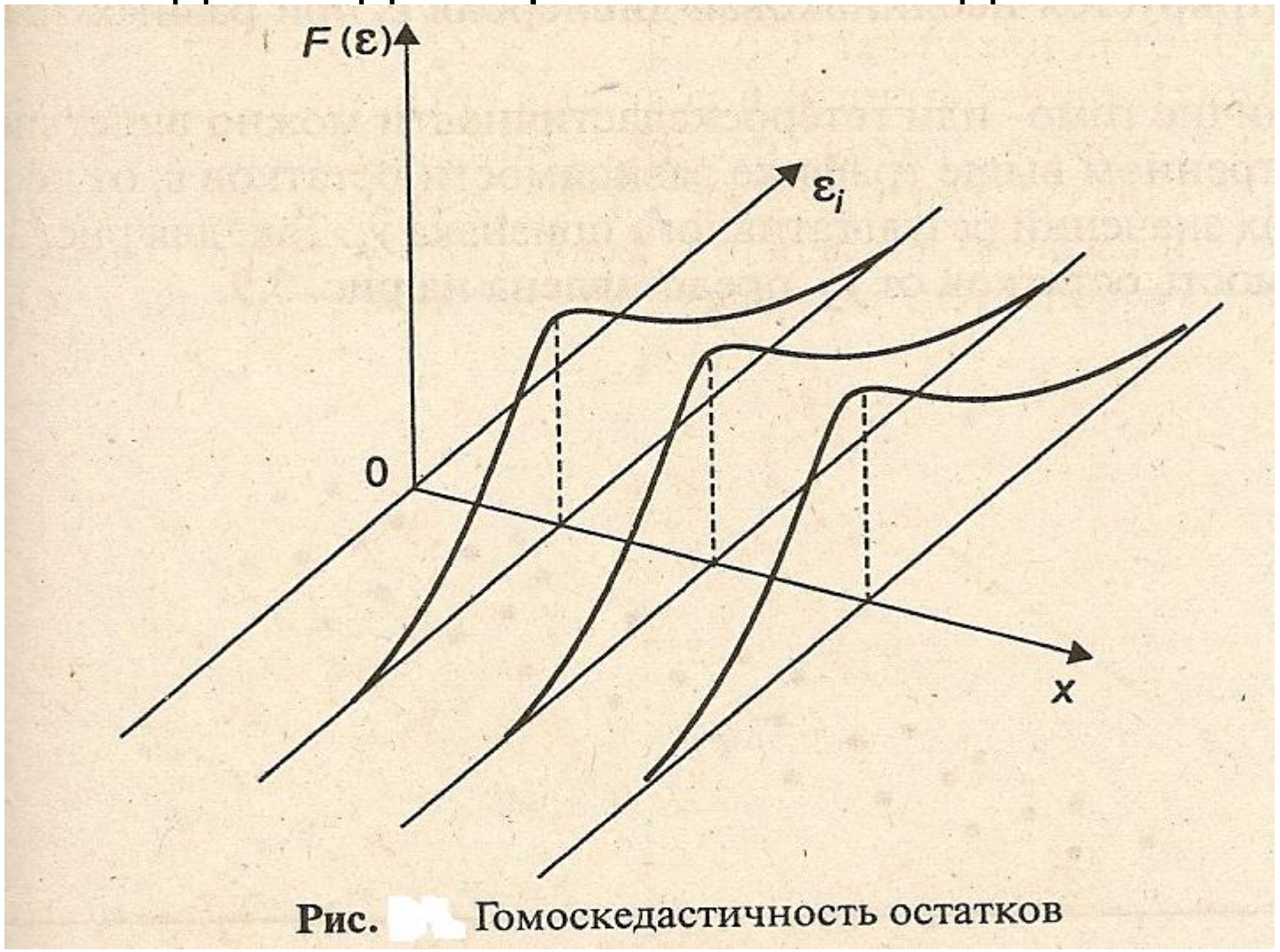


Рис. Гомоскедастичность остатков

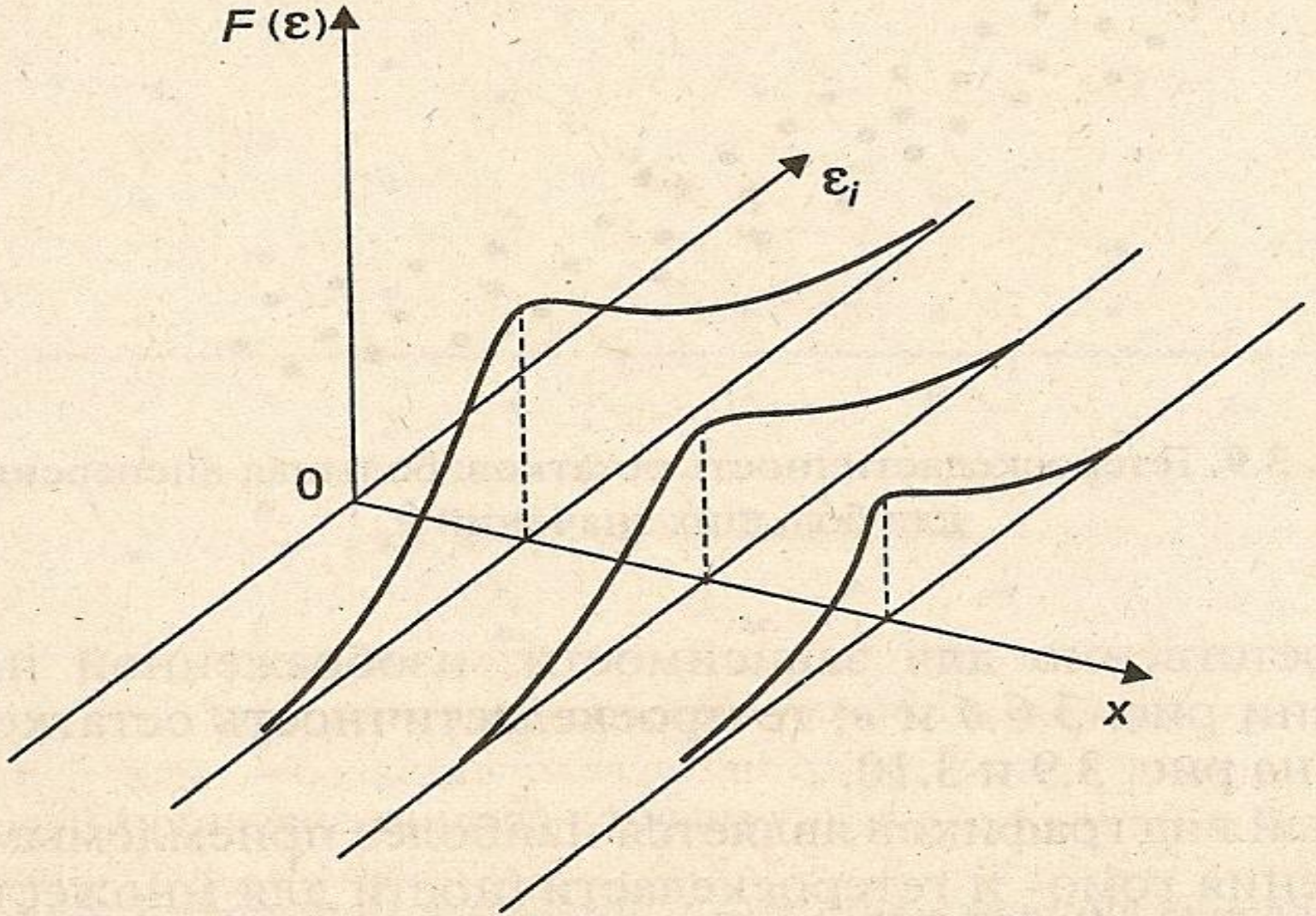


Рис. Гетероскедастичность остатков

Тест ранговой корреляции Спирмена

- рассчитывается коэффициент Спирмена между модулями остатков и значениями факторов, если коэффициент Спирмена значим, то гетероскедастичность остатков доказана и уравнение регрессии ненадежно

Тест Голдфелда-Квандта

Рассматривается связь величин вида $y = a + bx$. Предполагается, что стандартное отклонение $\sigma_i = \sigma(\varepsilon_i)$ пропорционально значению переменной x в этом наблюдении: $\sigma_i^2 = \sigma^2 x_i^2$, $i = 1, \ldots, n$, n — число наблюдений. Также предполагается, что ε_i имеет нормальное распределение и отсутствует автокорреляция (будет рассмотрена в дальнейшем). Все n наблюдений упорядочиваются по величине x. Эта упорядоченная выборка делится на три примерно равные части объемов k, $n - 2k$ и k соответственно. При $n = 30$ $k = 11$, при $n = 60$ $k = 22$.

Для каждой из выборок объема k оценивается свое уравнение регрессии и находятся суммы квадратов отклонений $S_1 = \sum_{i=1}^{k} e_i^2$ и $S_3 = \sum_{i=n-k+1}^{n} e_i^2$ соответственно.

Зададим доверительную вероятность p. $\alpha = 1 - p$. По F-таблицам находим граничную точку $F_{\alpha;k-m-1;k-m-1}$, где m — число факторов модели.

Статистика $F = S_3/S_1$.

Если $F < F_{\alpha;k-m-1;k-m-1}$, то на уровне значимости α принимается гипотеза об отсутствии гетероскедастичности. Иначе гипотеза об отсутствии гетероскедастичности отклоняется. Для множественной регрессии тест обычно проводится для того фактора, который в максимальной степени связан

Гомоскедастичность остатков

МИРЭА
Российский технологический университет

- В пакете car также реализованы функции для обнаружения неоднородности дисперсии остатков.
- Функция ncvTest() позволяет проверить гипотезу о постоянстве дисперсии остатков как альтернативу тому, что дисперсия остатков изменяется в зависимости от подобранных значений. Статистически значимый результат свидетельствует о гетероскедастичности (неоднородности дисперсии остатков).
- Функция spreadLevelPlot() создает диаграмму рассеяния для абсолютных значений стандартизованных остатков и подобранных значений с наложенной регрессионной прямой.

Гомоскедастичность остатков

- Проверка на гомоскедастичность

```
> library(car)
> ncvTest(fit)

Non-constant Variance Score Test
Variance formula: ~ fitted.values

Chisquare=1.7     Df=1      p=0.19

> spreadLevelPlot(fit)


Suggested power transformation:  1.2
```

Результат теста незначим (p = 0.19), что свидетельствует о выполнении условия однородности дисперсии. Это также можно увидеть на диаграмме Точки беспорядочно располагаются в виде горизонтальной полосы вдоль горизонтальной регрессионной прямой. Если бы требование гомоскедастичности не выполнялось, мы увидели бы не горизонтальную прямую.

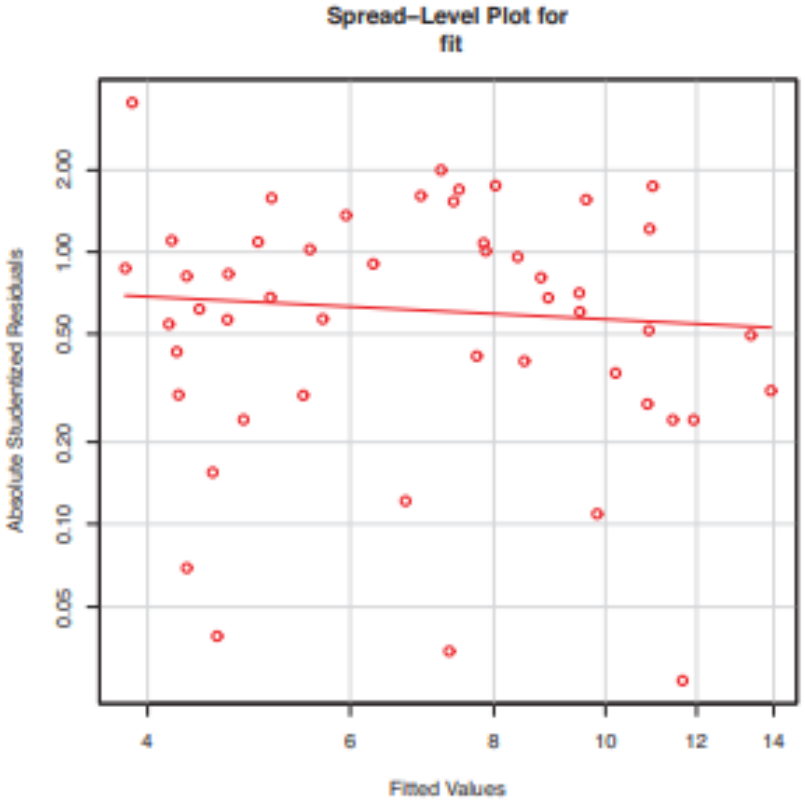


Диаграмма для проверки однородности дисперсии остатков

Проверка остатков модели на гетероскедастичность с помощью теста Спирмена

Проверить на основе теста Спирмена наличие гетероскедастичности в остатках

```
#посмотрим структуру переменной fit
str(fit)
# в переменную yt поместим теоретические значения y
yt<-fitted.values(fit)
#построим их график зависимости остатков от теоретических значений y
plot(~ yt+fit$residuals)

# в fit$residuals сохранены остатки модели, сохраним их модули в переменную absres
absres<-abs(fit$residuals)
#найдем коэффициент Спирмена между модулями остатков модели и переменной x5
cor.test(absres, tmod$x5, method = "spearman")
#p-value больше 5%, то взаимосвязь незначима и
#гетероскедастичности остатков нет по переменной x5
#найдем коэффициент Спирмена между модулями остатков модели и переменной x6
cor.test(absres, tmod$x6, method = "spearman")
#p-value больше 5%, то взаимосвязь незначима и
#гетероскедастичности остатков нет по переменной x6
```

МИРЭА
Российский
технологический
университет

2. Тест на мультиколинеарность в R

ИССЛЕДОВАНИЕ МУЛЬТИКОЛЛИНЕАРНОСТИ

Наличие существенной линейной связи между независимыми переменными – мультиколлинеарности - ведёт к ненадёжности оценок уравнения регрессии и прогнозов на их основе. Для оценки её наличия используют определитель матрицы парных линейных коэффициентов корреляции, например для 3 факторов:

$$\det|R| = \begin{vmatrix} r_{x1x1} & r_{x2x1} & r_{x3x1} \\ r_{x1x2} & r_{x2x2} & r_{x3x2} \\ r_{x1x3} & r_{x2x3} & r_{x3x3} \end{vmatrix}$$

Чем ближе значение определителя к нулю, тем сильнее мультиколлинеарность и ненадёжней результаты множественной регрессии

Проверка гипотезы об отсутствии мультиколлинеарности методом Ферарра-Глобера

H_0: Det|R|=1, то есть мультиколлинеарности нет

H_1:Det|R|=0 , то есть она есть

Если $\chi^2_{расч} > \chi^2(\alpha;0,5(m(m-1))$, то H_0 отклоняется и мультиколлинеарность факторов доказана

$$X^2_{расч} = -[n-1-(1/6)(2p+5)\ln DetR]$$

Причины мультиколинеарности

МИРЭА
Российский технологический университет

- Выбор в качестве независимых переменных показателей, которые являются характеристиками одного и того же признака
- Ошибки измерения в независимых переменных, связанных между собой линейно
- Неоднородность совокупности данных
- Наличие периодичности и цикличности в данных
- Использование удельных весов как независимых переменных, сумма которых постоянное число
- Использование удельных величин как независимых переменных с одинаковой базой сравнения

УСТРАНЕНИЕ МУЛЬТИКОЛИНЕАРНОСТИ

МИРЭА
Российский технологический университет

- Исключение из модели наиболее мультиколлинеарных факторов (строят множественную регрессию относительно каждого фактора и исключают фактор с максимальным R^2)
- Преобразование факторов через их объединение или изменение (Δ)
- Совмещённые уравнения регрессии (при коэффициенте регрессии стоит не один, а произведение факторов)
- Использование уравнений регрессии приведённой формы

Стандартный подход проверки моделей в R

МИРЭА
Российский технологический университет

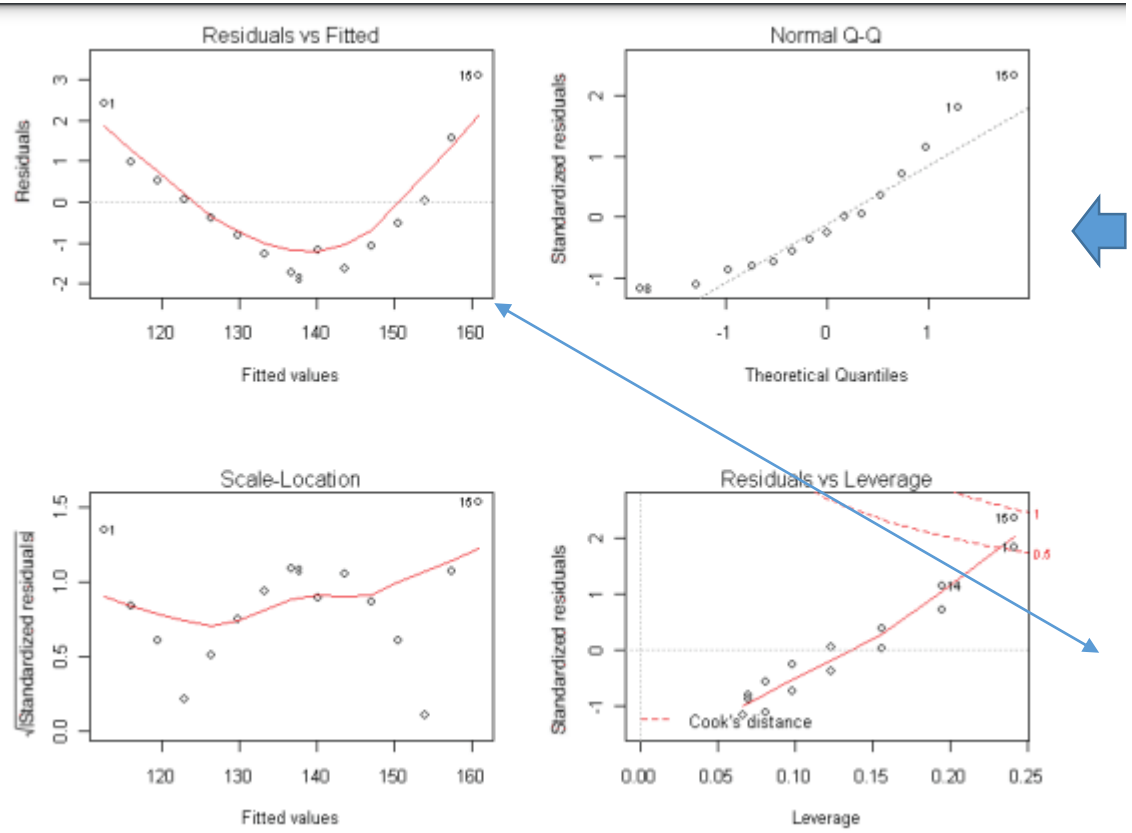
- В базовой версии программы R реализованы многочисленные методы проверки выполнения статистических допущений. Наиболее распространенный подход – применить функцию plot() к объекту, представляющему собой результат действия функции lm(). В результате появляются четыре диаграммы, полезные для оценки адекватности модели регрессии.

Пример: зависимость веса от роста женщин

fit <- lm(weight ~ height, data=women) – построение модели

par(mfrow=c(2,2)) – задание матрицы диаграмм

plot(fit) – построение диагностических диаграмм модели регрессии

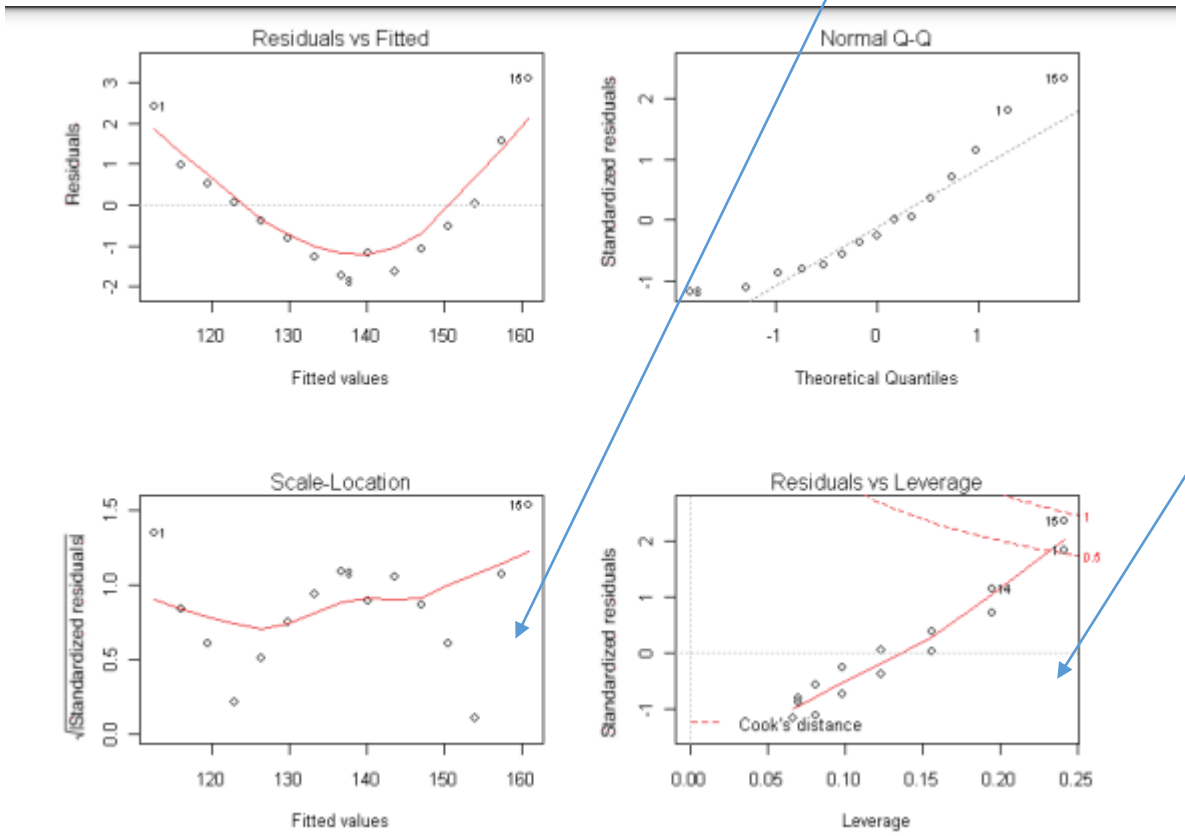


Нормальность. Если значения зависимой переменной нормально распределены при постоянных значениях независимых переменных, тогда остатки должны быть нормально распределены со средним значением 0. Графическая проверка данных на нормальность (Normal Q-Q plot – справа сверху) – это построение графика распределения вероятностей, сопоставляющего стандартизованные остатки и значения, которые ожидаются при нормальном распределении. Если допущение о нормальном распределении выполняется, то точки на этой диаграмме должны ложиться на прямую с углом наклона в 45°. Поскольку здесь это не наблюдается, это допущение не выполняется.

Линейность. Если зависимая переменная линейно связана с независимой, то связь между остатками и предсказанными (то есть подогнанными) значениями отсутствует. Другими словами, модель должна отражать всю закономерную изменчивость в данных, учитывая все, кроме белого шума. На диаграмме зависимости остатков от предсказанных значений (сверху слева) вы ясно видите нелинейную зависимость, что позволяет задуматься о добавлении квадратного члена в уравнение регрессии.

Пример: зависимость веса от роста женщин

Гомоскедастичность. Если допущение о постоянной изменчивости выполняется, то точки на нижней левой диаграмме должны располагаться в форме полосы вокруг горизонтальной линии. Похоже, что это допущение выполняется.



fit <- lm(weight ~ height, data=women) – построение модели

par(mfrow=c(2,2)) – задание матрицы диаграмм

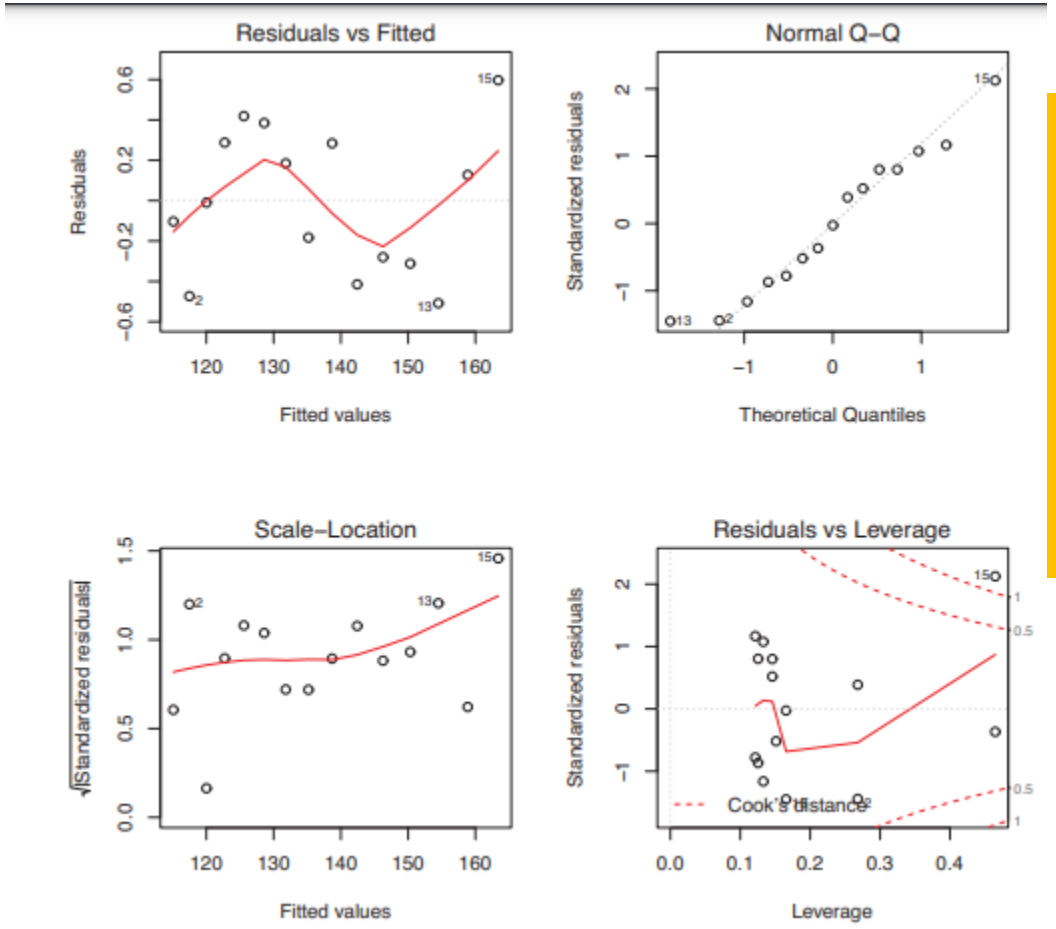
plot(fit) – построение диагностических диаграмм модели регрессии

Диаграмма зависимости остатков от «показателя напряженности» (англ. leverage) (слева внизу) содержит информацию о наблюдениях, на которые вам, возможно, следует обратить внимание. Диаграмма выявляет выбросы, точки высокой напряженности и влиятельные наблюдения:

- Выброс – это значение, которое плохо предсказывается подобранной моделью (то есть имеет большой положительный или отрицательный остаток).
- Влиятельное наблюдение – это значение, которое вносит непропорциональный вклад в расчет параметров модели. Влиятельные наблюдения выявляются при помощи статистики, называемой расстоянием Кука (Cook's distance, Cook's D).
- Значение с высоким значением напряженности описывается необычной комбинацией независимых переменных. Таким образом, это выброс в пространстве независимых переменных.

Диагностические диаграммы для квадратичной регрессии в данном примере

fi t2 <- lm(weight ~ height + I(height^2), data=women) par(mfrow=c(2,2)) plot(fi t2)



Этот набор диаграмм свидетельствует о том, что полиномиальная регрессия подходит лучше, поскольку учитывает требования линейности, нормального распределения остатков (за исключением наблюдения 13) и гомоскедастичности (постоянной дисперсии остатков). Наблюдение 15 можно отнести к влиятельным (на основе высокого значения расстояния Кука), его удаление повлияет на оценку параметров модели.

Усовершенствованный подход проверки выполнения предпосылок регрессионных моделей

Функция	Назначение
qqPlot()	Диаграмма сравнения квантилей
durbinWatsonTest()	Тест Дарбина-Уотсона (Durbin-Watson test) на авто-корреляцию в остатках
crPlots()	Диаграмма компонент и остатков
ncvTest()	Тест на неоднородность дисперсии остатков

Функция	Назначение
spreadLevelPlot()	Диаграмма для обнаружения неоднородности дис-персии остатков (spread-level plot)
outlierTest()	Тест Бонферрони на выбросы
avPlots()	Диаграммы добавленных переменных
influencePlot()	Диаграмма влияния наблюдений на регрессию
scatterplot()	Усовершенствованная диаграмма рассеяния
scatterplotMatrix()	Усовершенствованная матрица диаграмм рассеяния
vif()	Фактор инфляции дисперсии

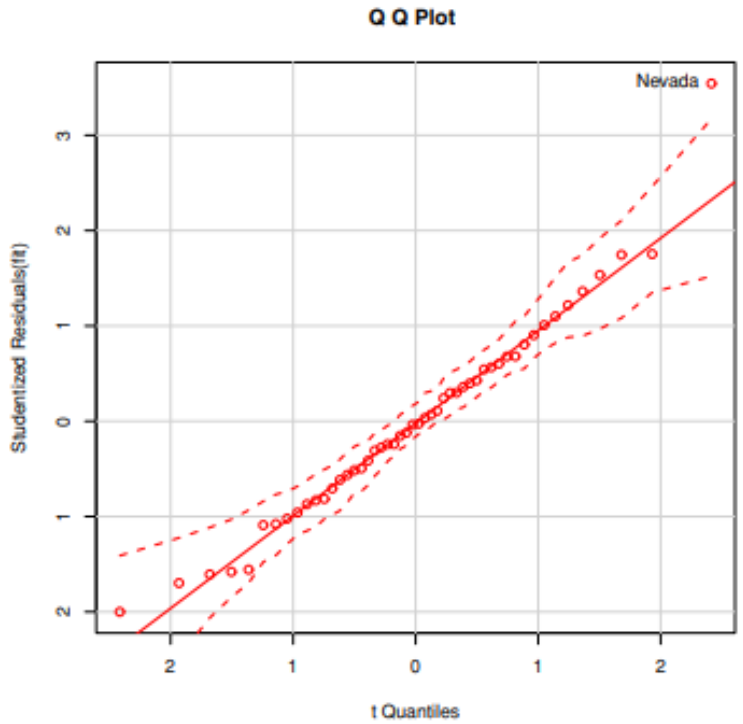
- В пакете car реализован ряд функций, которые значительно расширяют возможности подгонки и оценки регрессионных моделей
- В пакете gvlma реализован общий тест на соответствие условиям линейной модели

Нормальность

Функция `qqPlot()` – это более аккуратный метод проверки предположения о нормальности, по сравнению с функцией `plot()` из базовой версии программы. Она изображает связь между *остатками Стьюдента* (также называемыми *исключенными остатками Стьюдента*, или *остатками, вычисленными методом последовательного исключения значений* – jackknifed residuals) и квантилями распределения Стьюдента с $n - p - 1$ степенями свободы, где n – это объем выборки, а p – число параметров регрессии (включая свободный член). Программный код таков:

```
library(car)
fit <- lm(Murder ~ Population + Illiteracy + Income + Frost, data=states)
qqPlot(fit, labels=row.names(states), id.method="identify",
  simulate=TRUE, main="Q-Q Plot")
```

Здесь вы видите, что уровень преступности равен 11.5%, а модель предсказывает 3.9%.

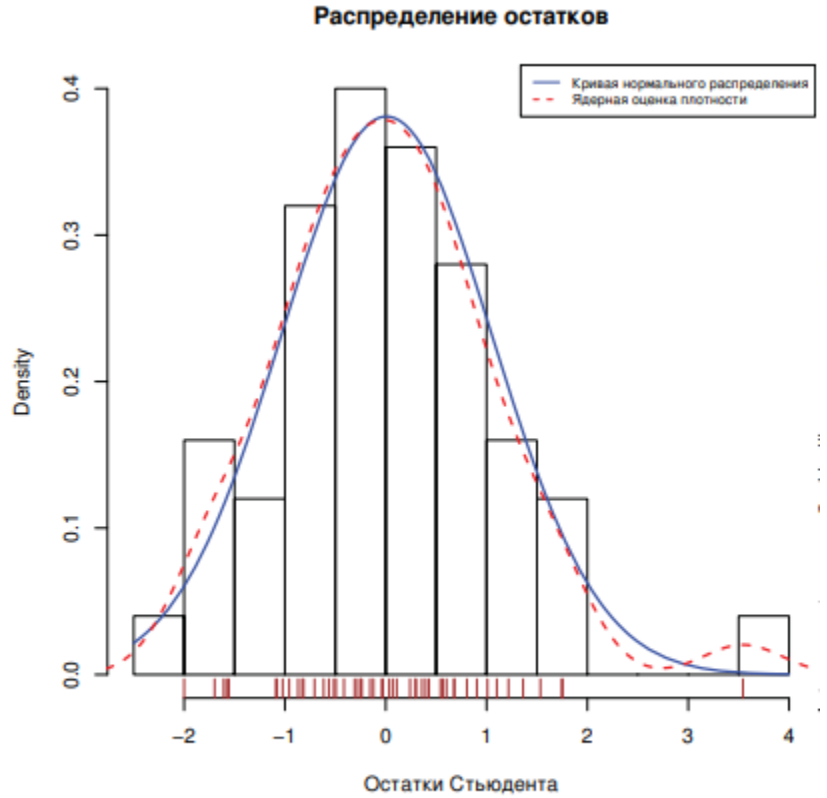


Графическая проверка нормальности распределения (Q-Q-диаграмма)

Функция residplot() создает гистограмму остатков Стьюдента с наложенными кривой нормального распределения, кривой ядерной оценки функции плотности и графиком-щеткой. Для использования этой функции пакет car не нужен.

Функция для изображения остатков Стьюдента

```
residplot <- function(fit, nbreaks=10) {
              z <- rstudent(fit)
              hist(z, breaks=nbreaks, freq=FALSE,
              xlab="Остатки Стьюдента",
              main="Распределение остатков")
              rug(jitter(z), col="brown")
              curve(dnorm(x, mean=mean(z), sd=sd(z)),
                    add=TRUE, col="blue", lwd=2)
              lines(density(z)$x, density(z)$y,
                    col="red", lwd=2, lty=2)
              legend("topright",
                    legend = c( "Кривая нормального распределения",
                                ➥"Ядерная оценка плотности"),
                    lty=1:2, col=c("blue","red"), cex=.7)
        }
residplot(fit)
```



Распределение остатков Стьюдента, изображенное при помощи функции residplot()

Как вы можете видеть, остатки достаточно хорошо соответствуют нормальному распределению, за исключением большого выброса.

Независимость остатков

- В пакете car реализована функция для проведения теста Дарбина-Уотсона на наличие автокорреляции в остатках.

```
> durbinWatsonTest(fit)
 lag Autocorrelation D-W Statistic p-value
   1          -0.201           2.32   0.282
 Alternative hypothesis: rho != 0
```

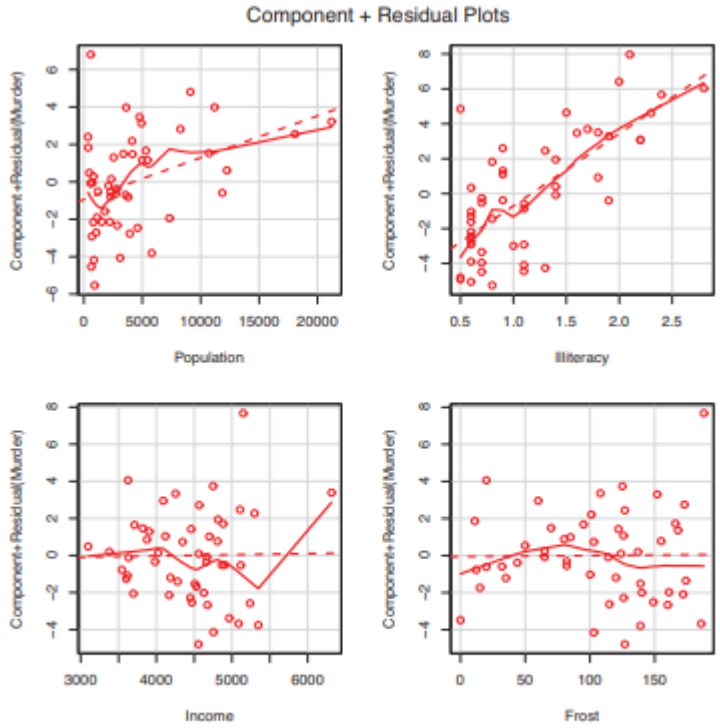
- Высокое значение вероятности статистической ошибки первого рода (p = 0.282>0,05) свидетельствует об отсутствии автокорреляции и, следовательно, о независимости остатков. Значение интервала (lag, в данном случае 1) говорит о том, что каждое значение сравнивается с соседним, следующим за ним значением в наборе данных

Линейность модели регрессии

- Наличие нелинейной связи между зависимой и независимыми переменными можно проверить при помощи диаграмм компонент и остатков (также известных под названием диаграммы частных остатков). Диаграммы создаются при помощи функции crPlots() из пакета car

  - library(car)
  - crPlots(fi t)

Нелинейность на любой из них свидетельствует о том, что вы могли некорректно смоделировать функциональную форму этой независимой переменной в уравнении. В этом случае может потребоваться добавить нелинейные составляющие, такие как полиномиальные члены, преобразовать одну или более переменных (например, использовать log(X) вместо X) или же отказаться от линейной регрессии в пользу какой-нибудь другой ее разновидности



Диаграммы компонент и остатков для регрессии уровня преступности по характеристикам штатов

Функция gvlma() из пакета gvlma (Pena, Slate, 2006) осуществляет общую проверку выполнения требований, предъявляемых к линейным моделям

МИРЭА Российский технологический университет

Общая проверка выполнения требований, предъявляемых к линейным моделям

```
> library(gvlma)
> gvmodel <- gvlma(fit)
> summary(gvmodel)
ASSESSMENT OF THE LINEAR MODEL ASSUMPTIONS
USING THE GLOBAL TEST ON 4 DEGREES-OF-FREEDOM:
Level of Significance= 0.05
Call:
 gvlma(x=fit)
                    Value p-value                Decision
Global Stat         2.773   0.597 Assumptions acceptable.
Skewness            1.537   0.215 Assumptions acceptable.
Kurtosis            0.638   0.425 Assumptions acceptable.
Link Function       0.115   0.734 Assumptions acceptable.
Heteroscedasticity 0.482   0.487 Assumptions acceptable.
```

Из результатов теста (строка Global Stat) следует, что данные удовлетворяют всем статистическим допущениям, лежащим в основе МНК-регрессии ($p = 0.597$). Если бы в этой строке было указано, что эти требования нарушены (то есть $p < 0.05$), вам бы пришлось исследовать данные описанными выше методами, чтобы определить, какие именно требования были нарушены.

Мультиколлинеарность

- Мультиколлинеарность можно выявить при помощи статистики, называемой фактором инфляции дисперсии. Квадратный корень, извлечённый из этой статистики для любой независимой переменной, указывает на степень увеличения доверительного интервала для параметра регрессии данной переменной по сравнению с моделью без скоррелированных независимых переменных (отсюда название статистики).
- Фактор инфляции дисперсии можно вычислить при помощи функции vif() из пакета car. Обычно значения квадратного корня этой статистики, превышающие 2, указывают на наличие мультиколлинеарности.

```
>library(car)
> vif(fit)
Population Illiteracy     Income      Frost
       1.2        2.2        1.3        2.1
> sqrt(vif(fit)) > 2 # проблема?
Population Illiteracy     Income      Frost
     FALSE      FALSE      FALSE      FALSE
```

Мультиколлинеарность

- # Получить корреляционную матрицу и на основе теста Ферарра-Глоббера проверить гипотезу о наличии мультиколлинеарности переменных X
- #получим корреляционную матрицу по переменным x5 и x6
- #определим ее определитель
- c<-tmod[, c(6,7)]
- c[,1]<-(c[,1]-mean(c[,1])/sd(c[,1]))
- c[,2]<-(c[,2]-mean(c[,2])/sd(c[,2]))
- r<-det(cor(c))
- #определим статистику Хи расчетную
- xi<-(79-1-(1/6)*(2*2+5)*log(r))
- # xI=78, что больше табличного 3,84, то есть факторы мультиколлинеарны
- # применим гребневую регрессию, чтобы убрать мультиколлинеарность x5 и x6
- library(MASS)
- fitridge<-lm.ridge(formula = tmod$y ~ tmod$x5+tmod$x6, lambda = 0)
- fitridge<-lm.ridge(formula = tmod$y ~ tmod$x5+tmod$x6, lambda = 1)

МИРЭА
Российский технологический университет

3. Пример реализации метода главных компонент и визуализации его результатов

Метод главных компонент

Метод главных компонент (Principal Components Analysis) основан на определении минимального числа факторов, вносящих наибольший вклад в дисперсию данных. Они называются **главными компонентами**.

- Метод основан на допущении, что характерности всех признаков равны нулю, а число общих факторов F равно числу исходных признаков X.

- Главные компоненты **независимы**, т.е. в геометрическом плане ортогональны. Выделение первой главной компоненты по максимальному вкладу в дисперсию признаков означает, что мы находим такое направление в пространстве признаков, которому соответствует максимальная дисперсия, т.е. наибольшая дифференциация, разброс объектов. Затем находится вторая главная компонента, ортогональная первой и дающая вновь наибольшую дифференциацию объектов, не объясненную первой компонентой и т.д.

- После построения всех главных компонент (число которых равно числу признаков X) остаточная дисперсия равна 0, т.е. задача имеет точное математическое решение. Обычно суммарная дисперсия признаков раскладывается таким образом, что первые несколько компонент уже объясняют большую долю этой дисперсии, а остальные почти ничего не добавляют, поэтому совсем не обязательно выделение всех компонент.

Пример

Переход к новым переменным

- **Исходные переменные** (центрированные) x_1 и x_2
- **Новые переменные** (главные компоненты)

$$pc_1 = \frac{1}{\sqrt{2}}x_1 + \frac{1}{\sqrt{2}}x_2$$

$$pc_2 = \frac{1}{2}x_1 - \frac{\sqrt{3}}{2}x_2$$

Сумма квадратов весов равна 1

- В каждой главной компоненте

- Новые переменные не коррелируют, нет мультиколлинеарности

- Центрированные или нормированные, среднее 0, дисперсия =1

PC – главная компонента

МИРЭА
Российский технологический университет

Новые переменные

- переменная pc_1 имеет максимальную выборочную дисперсию $sVar(pc_1)$
- переменная pc_2 некоррелирована pc_1 с и имеет максимальную $sVar(pc_2)$
- переменная pc_3 некоррелирована с pc_1, pc_2 и имеет максимальную $sVar(pc_3)$
- и т. д.

Свойства главных компонент

$$pc_1 = v_{11} \cdot x_1 + v_{21} \cdot x_2 + \ldots + v_{k1} \cdot x_k$$

$$pc_2 = v_{12} \cdot x_1 + v_{22} \cdot x_2 + \ldots + v_{k2} \cdot x_k$$

$$\ldots$$

$$sCorr(pc_j, pc_m) = 0$$

$$sVar(x_1) + sVar(x_2) + \ldots + sVar(x_k) =$$

$$= sVar(pc_1) + sVar(pc_2) + \ldots + sVar(pc_k)$$

Что дают главные компоненты?

- визуализировать сложный набор данных
- увидеть самые информативные переменные
- увидеть особенные наблюдения
- переход к некоррелированным переменным

Подводные камни на практике

- разные единицы измерения
- бездумное применение перед регрессией

Разные единицы измерения

Первая главная компонента «поймает» переменную с самыми мелкими единицами измерения

Вместо самой информативной — самая шумная

Выход: нормировать переменные $x_j = \frac{a_j - \bar{a}_j}{se(a_j)}$

Применение метода главных компонент перед регрессией

Небезопасная процедура:

Шаг 1. Найти главные компоненты, pc_1, pc_2

Шаг 2. Построить регрессию y на pc_1, pc_2

Проблемы:

- Коэффициенты при pc_j сложнее интерпретировать
- Самый изменчивый регрессор x гипотетически может быть наименее связан с y

Способы реализации метода главных компонент

Метод главных компонент позволяет заменить несколько исходных переменных на меньшее количество новых переменных. Новые искусственные переменные называются главными компонентами.

Существует два эквивалентных взгляда на метод главных компонент.

1. Последовательная максимизация выборочного разброса новых переменных:

- Новые переменные являются линейными комбинациями старых
- Новые переменные ортогональны
- Первая главная компонента имеет максимальную выборочную дисперсию
- Вторая главная компонента имеет максимальную выборочную дисперсию при фиксированной первой
- Третья главная компонента имеет максимальную выборочную дисперсию при фиксированной первой и второй
- и т.д.

2. Последовательная минимизация суммы квадратов расстояний от старых точек до новых точек:

- Расположим новые точки на прямой так, чтобы сумма квадратов расстояний до старых была минимальна
- Расположим новые точки на плоскости так, чтобы сумма квадратов расстояний до старых была минимальна
- Расположим новые точки в трехмерном подпространстве так, чтобы сумма квадратов расстояний до старых была минимальна
- и т.д.
- Зададим систему координат, так чтобы они были ортогональны и первая совпадала с прямой, первые две — с плоскостью, первые три с найденным трёхмерным подространством и т.д.

Формально:

Мы хотим вместо k переменных оставить $p < k$. Другими словами, хотим заменить матрицу $X_{n \times k}$ на матрицу $\hat{X}_{n \times k}$ ранга p, так чтобы минимизировать сумму квадратов ошибок:

$$\min \sum_{j=1}^{k} \sum_{i=1}^{n} (x_{ij} - \hat{x}_{ij})^2$$

Метод главных компонент в R

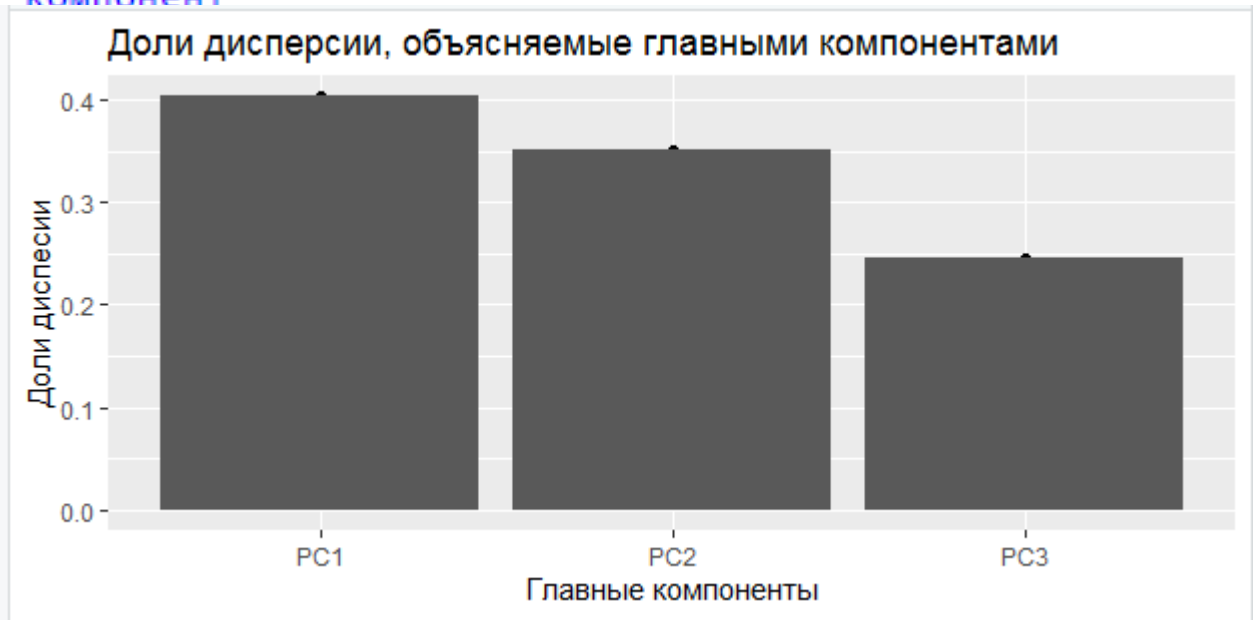
```r
# создание таблицы
library(data.table)
library(readxl)
types = c( rep("numeric", 22))
treg18 <- as.data.frame(read_excel("C:/Users/компьютер/Documents/reg-18.xlsx", 1,
                                   col_types = types))
#смотрим dataframe
str(treg18)

#Загружаем интересующий нас набор данных, без 22 переменной в m
m18<-treg18[,c(-22)]
#проверяем переменные на согласованность с нормальным распределением и заменяем неопределенные
#значения на Мо и Ме

for (i in 1:21){ stat_test  <- shapiro.test(m18[,i]); if (stat_test$p.value > 0.05){
  x<-m18[,i];  x[is.na(x)]  <- mean(x, na.rm = T); m18[,i]<-x

} else{
  #иначе все пропущенные значения заменяем на медиану
  x<-m18[,i];  x[is.na(x)]  <- median(x, na.rm = T); m18[,i]<-x

}  }
for (i in 1:21){ stat_test  <- shapiro.test(m18[, i]);

print(stat_test$p.value)}
#определили для переменных 1, 4, 14 согласуется с нормальным распределением stat_test$p.value>0,05

data18<-cbind(m18[,c(1,4,14)])
mm18[, c(2,3,5,6,7, 8,9,10,11, 12, 13, 15, 16, 17, 18,19, 20, 21)]

```

```
+
+ print(stat_test$p.value)}
[1] 0.1696491
[1] 0.0001149001
[1] 4.59749e-11
[1] 0.512528
[1] 0.006065245
[1] 1.050116e-12
[1] 9.241278e-07
[1] 0.006673937
[1] 1.394338e-11
[1] 2.847376e-07
[1] 3.629405e-10
[1] 2.077406e-10
[1] 2.292362e-07
[1] 0.357586
[1] 2.916363e-05
[1] 1.869477e-05
[1] 0.01555467
[1] 0.002696486
[1] 0.01135612
[1] 0.0002151659
[1] 0.03731344
```

```
> #Корреляции  переменных
> cor(data18)
           x1         x4        x14
x1   1.0000000 -0.1947926 0.0599563
x4  -0.1947926  1.0000000 0.1298827
x14  0.0599563  0.1298827 1.0000000
```

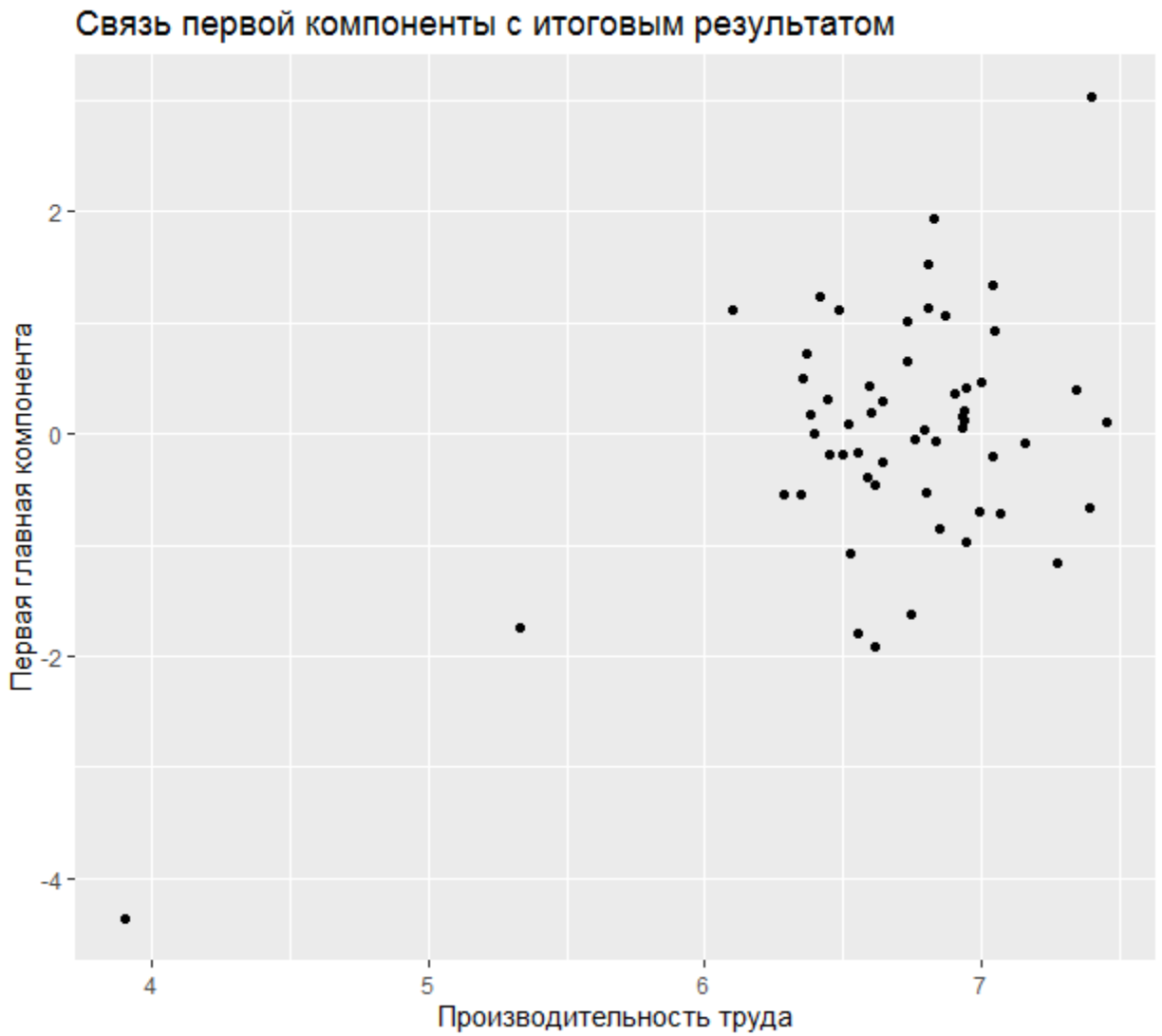
Метод главных компонент в R

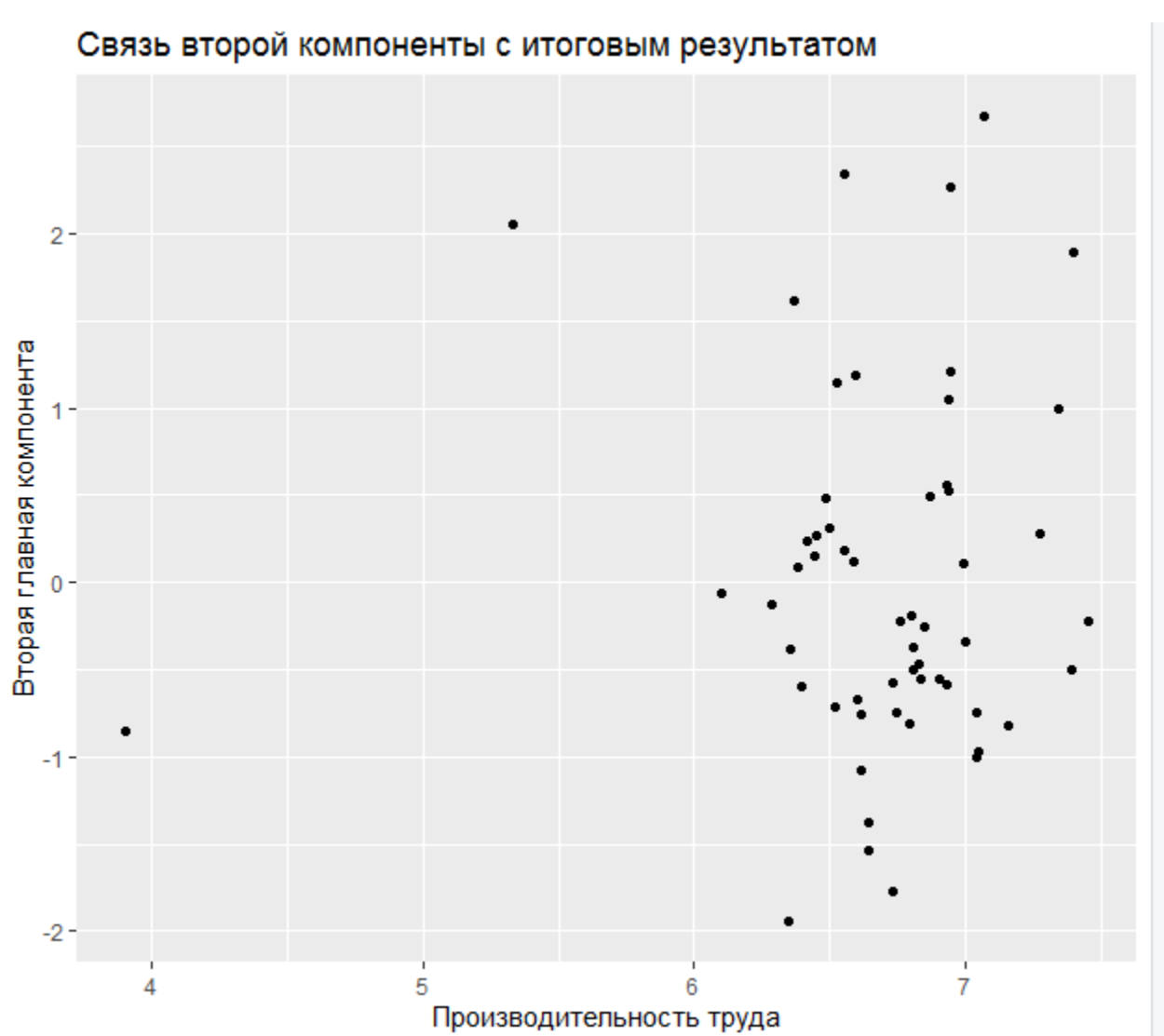
```
> #нормируем переменные, то есть проводим их стандартизацию scale = TRUE
> #и prcomp(m, scale = TRUE)применяем метод главных компонент
> hept_pca <- prcomp(data18, scale = TRUE)
> #сохраним в pc главные компоненты
> pc <- hept_pca$x
> #озаглавим PC
> head(pc)
             PC1        PC2          PC3
[1,]  0.9201003 -0.9688425 -0.560188815
[2,]  0.3190349  0.1565685  0.887175734
[3,] -0.1671923  0.1844432 -0.997798541
[4,] -1.6153744 -0.7464851  0.644752970
[5,]  1.1247339 -0.0664072  0.687516695
[6,] -0.0633836 -0.5542624  0.004634788
```




```
50 #Смотрим какую долю дисперсии объясняют главные компоненты
51 summary(hept_pca)
52 #Для построения графика сначала получим вектор долей дисперсии, объясняемых
53 exp_percent <- summary(hept_pca)$importance[2, ]
54 qplot(y = exp_percent, x = names(exp_percent)) +
55   geom_bar(stat = 'identity') +
56   labs(x = "Главные компоненты",
57        y = "Доли диспесии",
58        title = "Доли дисперсии, объясняемые главными компонентами ")
59
```

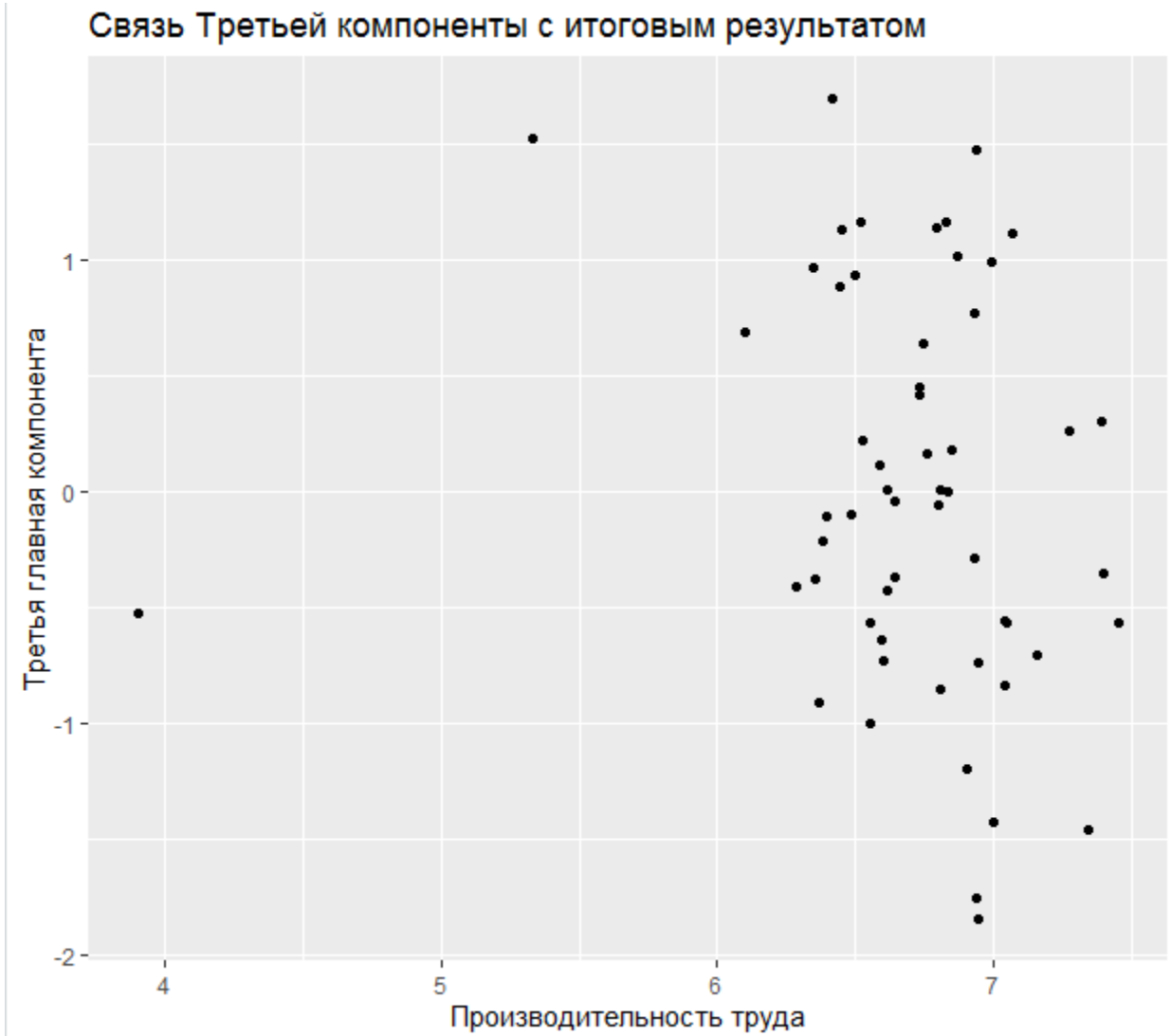
```
> summary(hept_pca)
Importance of components:
                          PC1    PC2    PC3
Standard deviation     1.0998 1.0270 0.8577
Proportion of Variance 0.4032 0.3516 0.2452
Cumulative Proportion  0.4032 0.7548 1.0000
```



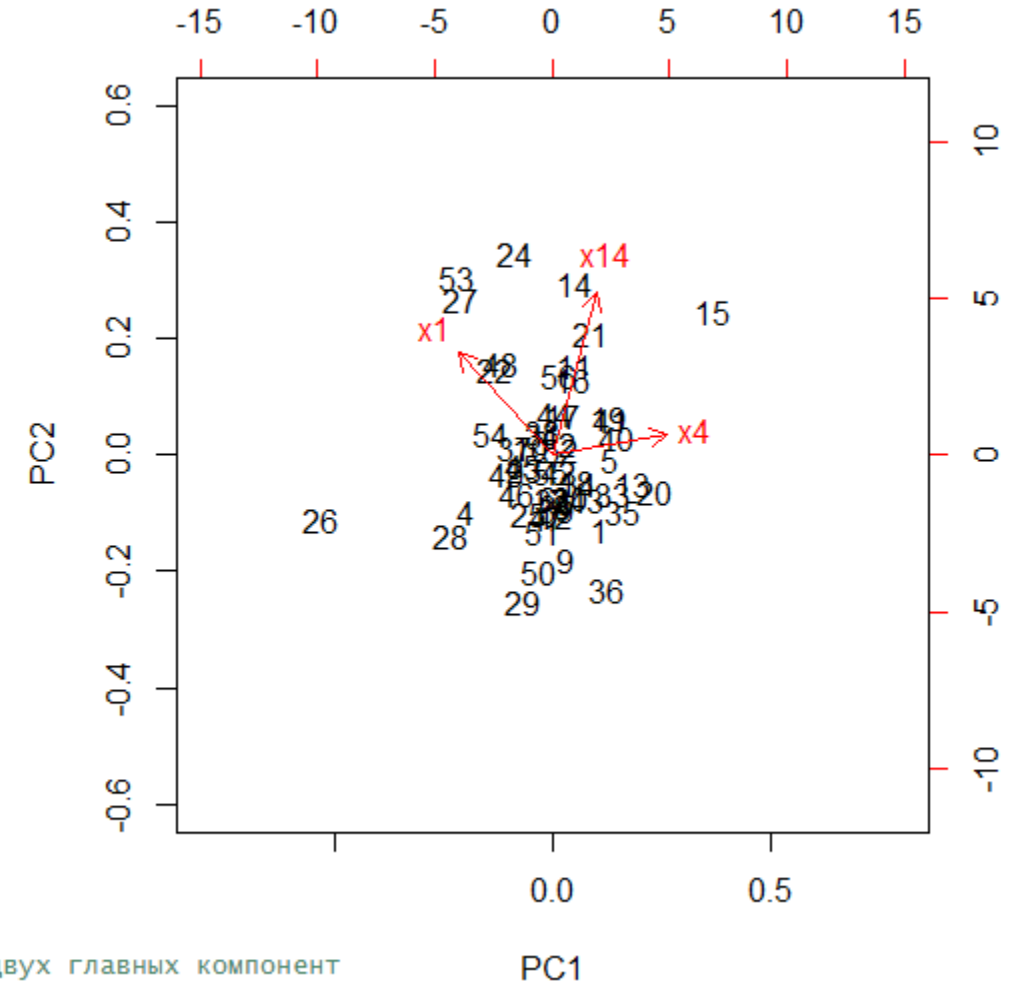



```
77
78  #Посмотрим как первая компонента связана с итоговым результатом
79  qplot(x = (mm18[,21]), y = pc[, 1]) +
80    labs(x = "Производительность труда",
81         y = "Первая главная компонента",
82         title = "Связь первой компоненты с итоговым результатом")
83
```

```
77
78  #Посмотрим как вторая компонента связана с итоговым результатом
79  qplot(x = (mm18[,21]), y = pc[, 2]) +
80    labs(x = "Производительность труда",
81         y = "Вторая главная компонента",
82         title = "Связь второй компоненты с итоговым результатом")
83
```




```
> #вычислим корреляцию между главными компонентами и У
> cor(mm18[,21], pc[, 1])
[1] 0.5127085
> cor(mm18[,21], pc[, 2])
[1] 0.0501533
> cor(mm18[,1], pc[, 2])
[1] 0.5434891
```




```
#График  компонент с вкладом каждой переменной
biplot(hept_pca,
       xlim = c(-0.8, 0.8), ylim = c(-0.6, 0.6))
#Красные векторы – это проекции векторов исходных координат на плоскость первых двух главных компонент
```

#посмотрим веса переменных в главных компонентах v<-hept_pca$rotation

```
> v
             PC1        PC2        PC3
x1   -0.6072540  0.5298291 -0.5920504
x4    0.7413666  0.1099193 -0.6620372
x14   0.2856888  0.8409512  0.4595465
```

#формируем новые факторы

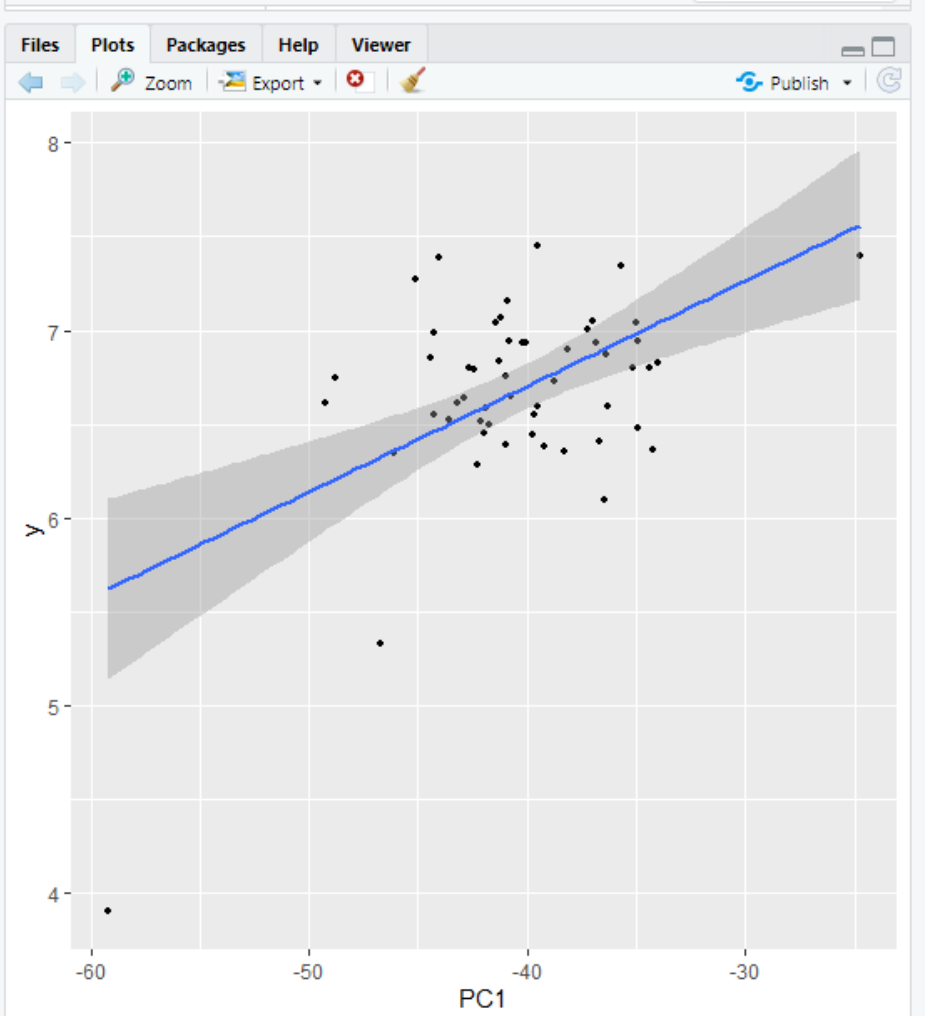
dataPC<-as.matrix(data18) %*% as.matrix(v[1:3, 1:3])

```
109  y<-mm18[,21]
110  mult<-as.data.frame(dataPC)
111  mult <- transform(mult, y = mult$PC1)
112  mult$y<-y
113
114
115  fit  <- lm((y) ~ PC1+PC2+PC3, mult)
116  summary(fit)
117
118
119
120  library(ggplot2)
121  ggplot(mult, aes(PC1,y)) +   geom_point(size = 1)  +   geom_smooth(method = "lm")
122
123
124
```

124:1 (Top Level) R Script

Console Terminal Jobs

```
Call:
lm(formula = (y) ~ PC1 + PC2 + PC3, data = mult)

Residuals:
     Min       1Q   Median       3Q      Max
-1.75467 -0.20967  0.01222  0.25191  0.90638

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept)  9.442202   3.364022   2.807  0.00703 **
PC1          0.058640   0.019121   3.067  0.00343 **
PC2         -0.008269   0.017841  -0.463  0.64496
PC3         -0.005463   0.022178  -0.246  0.80639
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.4498 on 52 degrees of freedom
Multiple R-squared:  0.2927,    Adjusted R-squared:  0.2519
F-statistic: 7.174 on 3 and 52 DF,  p-value: 0.0004045
```



Вопросы по теме

МИРЭА
Российский технологический университет

1. Как проверить в R предпосылки МНК?
2. Что такое гомоскедастичность остатков модели и как ее выявить?
3. Что такое мультиколлинеарность факторов? Как ее выявить в R?
4. Что такое автокорреляция остатков и как ее выявить в R?
5. Почему нужно использовать метод главных компонент? Как это сделать в R?

Литература

1. Эконометрика и эконометрическое моделирование в EXCEL и R: учебник/Л.О. Бабешко, И.В. Орлова. – Москва: ИНФРА-М, 2021. – 300 с.: ил. – (Высшее образование: Магистратура). –DOI 10.12737/1079837.
2. Роберт И. Кабаков R в действии. Анализ и визуализация данных в программе R / пер. с англ. Полины А. Волковой. – М.: ДМК Пресс, 2014. – 588 с.: ил. ISBN 978-5-947060-077-1