ТОМСКИЙ
ПОЛИТЕХНИЧЕСКИЙ
УНИВЕРСИТЕТ

# Статистическое моделирование и прогнозирование
# Лекция 7. Факторный анализ (метод главных компонент).

Семёнов Михаил Евгеньевич
к. ф.-м. н., доцент ОЭФ ИЯТШ

Томский политехнический университет
26 мая 2020 г.

ТОМСКИЙ ПОЛИТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ

Введение

Основные определения
- Постановка задачи
- Определения главных компонент
- Вычисление главных компонент

Лабораторная работа 7

Список использованных источников

В задачах анализа многомерных наблюдений типичными являются ситуации, когда общее число признаков, регистрируемых на каждом из наблюдаемых объектов, очень велико. В этом случае вполне естественным выглядит стремление представить каждое из наблюдений в виде вектора $Z$ с существенно меньшим числом компонент.

Предполагается, что непосредственно наблюдаемые *признаки* (индикаторы) являются функциями гораздо меньшего числа неявных (скрытых), но объективно существующих признаков, называемых обычно *факторами*. Эта фундаментальная идея является основой целого класса статистических методов, называемых факторным анализом, к которому относят и метод главных компонент.

Формально задача перехода к новому набору признаков может быть описана следующим образом. Пусть имеется $p$-мерная величина $X = (x_1, x_2, ..., x_p)$ с вектором средних значений $a = (a_1, a_2, ..., a_p)$ и ковариационной матрицей $\Sigma = (\sigma_{ij})$, где $i = 1, 2, ..., p$ и $j = 1, 2, ..., p$. Определим на множестве признаков в качестве класса допустимых преобразований всевозможные линейные ортагональные нормированные комбинации, то есть будем полагать, что

$$z_j = \sum_{i=1}^{p} l_{ij}(x_i - a_i), \tag{1}$$

где $\sum_{i=1}^{p} {l_{ij}}^2 = 1$ и $\sum_{i=1}^{p} l_{ji} l_{ki} = 0$ для $j = 1, 2, ..., p$ и $k = 1, 2, ..., p$, но $k \neq p$. При этом потребуем, чтобы эти преобразования удовлетворяли условиям монотонности дисперсии $D(z_1) \geqslant D(z_2) \geqslant ... \geqslant D(z_p)$. Полученные таким образом переменные $z_1(X), z_2(X), ..., z_p(X)$ и называются *главными компонентами.*

# Определения главных компонент



*Первой главной компонентой* $z_1(X)$ системы признаков $X$ называется такая нормированно-центрированная линейная комбинация этих показателей, которая среди всех прочих линейных комбинаций такого рода обладает наибольшей дисперсией.

*k-ой главной компонентой* $z_k(X)$ системы $X$ при $k = 2, 3, ..., p$ называется такая линейная комбинация этих показателей, которая не коррелированна с предыдущими $(k-1)$ главными компонентами и среди всех прочих некоррелированных с предыдущими $(k-1)$ главными компонентами линейных комбинаций переменных $x_1, x_2, ..., x_p$ обладает наибольшей дисперсией.

**Замечание**: так как все главные компоненты ранжированы по величине, то это позволяет отбросить часть компонент, созранив для анализа только те $z_1(X), z_2(X), ..., z_m(X)$, где $m \leqslant p$, которые воспроизводят большую часть дисперсии:

$$\sum_{i=1}^{p} D(x_i) \approx \sum_{j=1}^{m} D(z_j). \tag{2}$$

Покажем, что для того, чтобы величина $D(z_1)$ достигала максимума при условии $\sum_{i=1}^{p} {l_{1i}}^2 = 1$, необходимо, чтобы вектор $l_1$ был собственным вектором, соответствующим максимальному собственному значению ковариационной матрицы $\Sigma = (\sigma_{ij})$. Пусть дана матрица исходных стандартизированных данных

$$\mathrm{X}= \begin{pmatrix} x_{11} & x_{12} & \dots & x_{1p} \\ x_{21} & x_{22} & \dots & x_{2p} \\ \dots & \dots & \dots & \dots \\ x_{n1} & x_{n2} & \dots & x_{np} \end{pmatrix}$$

Тогда произведение $R = X^T X$ будет соответствовать корреляционной матрице.

# Вычисление главных компонент



Предположим, что матрица $L = (l_{ij})$, где $i = 1, 2, ..., p$ и $j = 1, 2, ..., p$, позволяющая вычислить координаты объектов в пространстве компонент, известна. Тогда можно определить матрицу $Z$.

$$Z = XL = \begin{pmatrix} z_{11} & z_{12} & \dots & z_{1p} \\ z_{21} & z_{22} & \dots & z_{2p} \\ \dots & \dots & \dots & \dots \\ z_{n1} & z_{n2} & \dots & z_{np} \end{pmatrix} \tag{3}$$

По определению первой главной компоненты вектор $z_1 = (z_{11}, z_{21}, ..., z_{n1})^\top$ должен иметь максимальную дисперсию. В результате получаем задачу условной оптимизации:

$$D(z_1) = \frac{1}{n} \sum_{j=1}^{n} {z_{j1}}^2 \to \max \text{ при } \sum_{i=1}^{p} {l_{i1}}^2 = 1, \tag{4}$$

Для решения задачи условной оптимизации (4) составим функцию Лагранжа

$$\varphi(l_1, \lambda_1) = \frac{1}{n}\sum_{j=1}^{n}(\sum_{k=1}^{p} l_{j1}x_{k1})^2 - \lambda_1(\sum_{j=1}^{p} {l_{j1}}^2). \quad (5)$$

Продифференцируем (5) по компонентам $l_1$ и приравняем полученные выражения к нулю:

$$\frac{\partial\varphi}{\partial l_{j1}} = \frac{1}{n}\sum_{j=1}^{n}(\sum_{k=1}^{p} l_{k1}x_{jk})x_{j1} - \lambda_1 l_{k1} = 0, \quad j = 1, 2, ..., p. \quad (6)$$

Изменяя порядок суммирования в (6) и внося постоянный множитель $\frac{1}{n}$ под знак суммы, получим

$$\sum_{k=1}^{p}((\frac{1}{n}\sum_{j=1}^{n} x_{jl}x_{jk})l_{k1} - \lambda_1 l_{k1})) = 0. \quad (7)$$

Для стандартизированных данных $X$ внутреннюю сумму можно обозначить как $r_{mk} = \frac{1}{n}\sum_{j=1}^{n} x_{jm} z_{jk}$. Тогда уравнение (7) примет вид

$$\sum_{k=1}^{p}(r_{mk} l_{k1} - \lambda_1 l_{k1}) = 0, \quad m = 1, 2, ..., p. \tag{8}$$

В матричной форме уравнение (8) представляет собой характеристическое уравнение для корреляционной матрицы $R$.

$$Rl_1 - \lambda_1 l_1 = 0 \tag{9}$$

Из характеристического уравнения (9) следует, что вектор $l_1$ является собственным вектором матрицы $R$, а множитель $\lambda_1$ – соответствующим собственным значением.

С учетом (3) дисперсия $D(Z) = Z^T Z = (XL)^T XL = L^T X^T XL = L^T RL$. Для первой главной компоненты будем иметь $D(z_1) = l_1^T R l_1$. Умножив уравнение (9) слева на $l_1^T$ и принимая во внимание условие нормировки (4), получим

$$l_1^T R l_1 = l_1^T l_1 \lambda_1 = \lambda_1 D(z_1) \tag{10}$$

Таким образом, для выполнения условий (4) для первой главной компоненты $z_1$, необходимо взять максимальное собственное значение матрицы $R$ и соответствующий собственный вектор. Аналогичным образом находятся остальные главные компоненты.

# Постановка заданий



1. Используя выборочные данные по ирисам Фишера $iris(x_1, x_2, x_3, x_4)$ для $i = 1, 2, ..., 150$ выбрать главные компоненты $z_1$ и $z_2$, обеспечивающие оптимальную визуализацию данных. Сохранить скрипт в файле (*.r).
2. Установите пакет VIM, загрузите экспериментальный набор данных *sleep*. Примените критерий Кайзера-Гуттмана для оценки необходимого числа главных компонент. Выполните построение линейной модели регрессии на главные компоненты. Вычислите значение AIC критерия для полученной модели. Сохранить скрипт в файле (*.r). Сохраните экспериментальные данные sleep в файле (*.csv).
3. Построить ординационную диаграмму методом главных компонент. Сохранить скрипт в файле (*.r).

**Алгоритм 1** Выбор главных компонент, обеспечивающих оптимальную визуализацию данных

**Вход:** *iris* – выборочные данные

```
1: data(iris)
2: iris[c(1:2,75:76,149:150),]
3: pairs(x <- iris[-5], pch = dots <- as.numeric(iris[,5]))
4: summary(pca <- prcomp(x, center = TRUE, scale = TRUE))
```

**Выход:** распределение средних квадратичных отклонений и относительные доли дисперсии по выделенным главным компонентам.

# Результат



```
Importance of components:
                          PC1     PC2     PC3     PC4
Standard deviation     1.7084 0.9560 0.38309 0.14393
Proportion of Variance 0.7296 0.2285 0.03669 0.00518
Cumulative Proportion  0.7296 0.9581 0.99482 1.00000
```

Рисунок 1 – Результат выполнения алгоритма №1. Описательная часть

# Результат



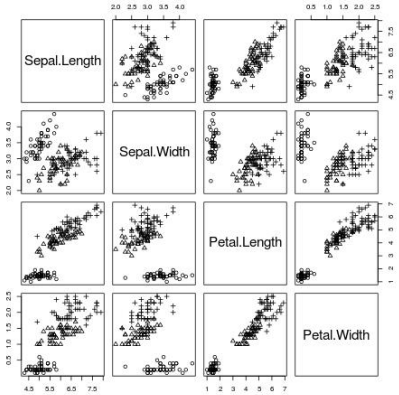


Рисунок 2 – Результат выполнения алгоритма №1. Графическая часть

**Алгоритм 2** Применение критерия Кайзера-Гуттамана, построение линейной модели регрессии на главные компоненты и вычисление значения AIC критерия

**Вход:** *sleep* – экспериментальный набор данных

```
1: load(file="sleep_imp.Rdata")
2: sleep_imp <- as.data.frame(scale(sleep_imp3))
3: sleep.pca <- princomp(~BodyWgt + BrainWgt + Span + Gest + Pred + Exp + Danger,
   data=sleep_imp)
4: summary(sleep.pca)
5: ev <- sleep.pca$sdev #Оценка необходимого числа главных компонент
6: ev[ev > mean(ev)] # Критерий Кайзера-Гуттмана
7: T <- predict(sleep.pca)[,1:3]
8: sleep_PCA <- as.data.frame(cbind(Sleep = sleep_imp3[,5],T))
9: M.PCA <- lm(Sleep~., data=sleep_PCA)
10: summary(M.PCA)
11: AIC(M.PCA)
```

**Выход:** распределение средних квадратичных отклонений и относительные доли дисперсии по выделенным главным компонентам; коэффициенты линейной модели на главные компоненты; значение критерия AIC

# Результат



```
Importance of components:
                          Comp.1    Comp.2    Comp.3    Comp.4     Comp.5     Comp.6      Comp.7
Standard deviation     1.877693 1.4294425 0.8997436 0.5019613 0.41554081 0.226892702 0.180001283
Proportion of Variance 0.511933 0.2966861 0.1175442 0.0365851 0.02507213 0.007474891 0.004704517
Cumulative Proportion  0.511933 0.8086191 0.9261634 0.9627485 0.98782059 0.995295483 1.000000000
```

Рисунок 3 – Результат выполнения алгоритма №2

# Результат



```
Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept)  10.5774     0.4526  23.371  < 2e-16 ***
Comp.1        1.5518     0.2410   6.438 2.56e-08 ***
Comp.2       -0.5103     0.3166  -1.612    0.1125
Comp.3        1.0908     0.5030   2.169    0.0342 *
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 3.564 on 58 degrees of freedom
Multiple R-squared: 0.4567,     Adjusted R-squared: 0.4286
F-statistic: 16.25 on 3 and 58 DF,  p-value: 8.801e-08
AIC(M.PCA)
[1] 339.3941
```

Рисунок 4 – Результат выполнения алгоритма №2

# Скрипт на R [Шитиков and Мастицкий, 2017]



```
F <- as.factor(ifelse(DGlass$Class == 2, 2, 1))
pca.scores <- as.data.frame(summary(mod.pca)$sites[,1:2])
pca.scores <- cbind(pca.scores, F)
# Составляем таблицу для "каркаса" точек на графике
l <- lapply(unique(pca.scores$F), function(c)
        { f <- subset(pca.scores, F == c); f[chull(f), ]})
hull <- do.call(rbind, l)
# Включаем в названия осей доли объясненной дисперсии
axX <- paste("PC1 (",
   as.integer(100*mod.pca$CA$eig[1]/sum(mod.pca$CA$eig)),"%)")
axY <- paste("PC2 (",
   as.integer(100*mod.pca$CA$eig[2]/sum(mod.pca$CA$eig)),"%)")
# Выводим ординационную диаграмму
ggplot() +
  geom_polygon(data=hull,aes(x=PC1,y=PC2, fill=F),
            alpha=0.4, linetype=0) +
  geom_point(data=pca.scores,aes(x=PC1,y=PC2,shape=F,
            colour=F),size=3) +
  scale_colour_manual( values = c('purple', 'blue'))+
  xlab(axX) + ylab(axY) + coord_equal() + theme_bw()
```

Рисунок 5 – Пример построения ординационной диаграммы методом главных компонент

# Результат



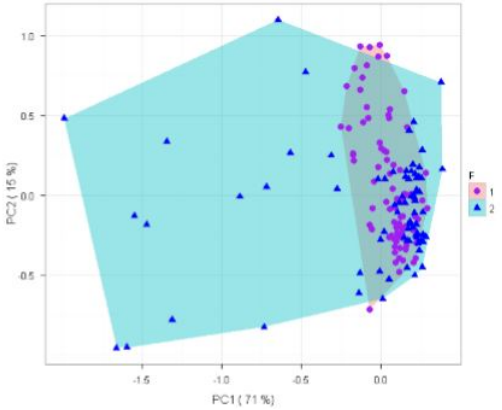


Рисунок 6 – Результат выполнения алгоритма №3

Буховец, and Москалев, (2015).
*Алгоритмы вычислительной статистки в системе R.*

Мастицкий, and Шитиков, (2014).
*Статистический анализ и визуализация данных с помощью R.*

Шитиков, and Мастицкий, (2017).
*Классификация, регрессия и другие алгоритмы Data Mining с использованием R.*