# Генерация и визуализация трёхмерных нормальных подвыборок с различной структурой ковариационной матрицы

Елисеев Данила, 2025, ИС

26 декабря 2025 г.

## 1. Теоретическая часть

Случайный вектор

$$\mathbf{X} = (X_1, X_2, X_3)^\top \sim \mathcal{N}_3(\boldsymbol{\mu}, \boldsymbol{\Sigma})$$

характеризуется вектором средних $\boldsymbol{\mu} \in \mathbb{R}^3$ и симметричной положительно определённой ковариационной матрицей $\boldsymbol{\Sigma} \in \mathbb{R}^{3\times 3}$. Структура $\boldsymbol{\Sigma}$ полностью определяет форму распределения: размеры, ориентацию и степень вытянутости эллипсоидов постоянной плотности.

Если $\mathbf{X} \sim \mathcal{N}_p(\boldsymbol{\mu}, \boldsymbol{\Sigma})$, его плотность:

$$f(\mathbf{x}) = \frac{1}{(2\pi)^{p/2}|\boldsymbol{\Sigma}|^{1/2}} \exp\left\{-\frac{1}{2}(\mathbf{x}-\boldsymbol{\mu})^\top \boldsymbol{\Sigma}^{-1}(\mathbf{x}-\boldsymbol{\mu})\right\}. \tag{1}$$

Множества постоянной плотности — эллипсоиды, задаваемые уровневыми уравнениями

$$(\mathbf{x}-\boldsymbol{\mu})^\top \boldsymbol{\Sigma}^{-1}(\mathbf{x}-\boldsymbol{\mu}) = c^2. \tag{2}$$

Оси этого эллипсоида направлены вдоль собственных векторов $\mathbf{e}_1, \mathbf{e}_2, \mathbf{e}_3$ матрицы $\boldsymbol{\Sigma}$, а длины полуосей равны $c\sqrt{\lambda_1}, c\sqrt{\lambda_2}, c\sqrt{\lambda_3}$, где $\lambda_i$ — соответствующие собственные значения.

Для нормального распределения 95%-эллипсоид (т.е. содержащий 95% вероятностной массы) соответствует уровню $c^2 = \chi^2_{3;0.95} \approx 7.815$, откуда $c = \sqrt{7.815} \approx 2.795$.

$i$-я главная компонента (ГК) популяции имеет вид:

$$Y_i = \mathbf{e}_i^\top \mathbf{X}, \quad \mathrm{Var}(Y_i) = \lambda_i, \quad \mathrm{Cov}(Y_i, Y_j) = 0 \ (i \neq j).$$

Таким образом, ГК — это проекции на оси эллипсоида рассеяния. Форма и ориентация облака напрямую отражают спектр $\{\lambda_i\}$ и собственные векторы.

## 2. Описание подвыборок

Генерируются три подвыборки объёма $n = 100$ каждая из $\mathcal{N}_3(\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)$:

### 2.1. Подвыборка 1: Сферическое распределение

$\boldsymbol{\mu}_1 = (0, 0, 0)^\top$,

$$\boldsymbol{\Sigma}_1 = \begin{pmatrix} 1.0 & 0.0 & 0.0 \\ 0.0 & 1.0 & 0.0 \\ 0.0 & 0.0 & 1.0 \end{pmatrix}.$$

Диагональная матрица, т.е. $X_1, X_2, X_3$ независимы. Все дисперсии равны единице, что даёт сферическое распределение.

### 2.2. Подвыборка 2: Вытянутое распределение

$\boldsymbol{\mu}_2 = (5, 5, 5)^\top$,

$$\boldsymbol{\Sigma}_2 = \begin{pmatrix} 1.0 & 0.0 & 0.0 \\ 0.0 & 1.0 & 0.0 \\ 0.0 & 0.0 & 5.0 \end{pmatrix}.$$

Диагональная матрица с различными дисперсиями. Переменные независимы, но наибольший разброс по оси $Z$ (дисперсия равна 5), что приводит к вытянутому эллипсоиду вдоль оси $Z$.

### 2.3. Подвыборка 3: Коррелированное распределение

$\boldsymbol{\mu}_3 = (10, 0, 0)^\top$,

$$\boldsymbol{\Sigma}_3 = \begin{pmatrix} 1.0 & 0.8 & 0.8 \\ 0.8 & 1.0 & 0.8 \\ 0.8 & 0.8 & 1.0 \end{pmatrix}.$$

Полная взаимная корреляция всех компонент. Корреляция между всеми парами переменных равна 0.8, что приводит к повороту эллипсоида в пространстве. Обратим внимание, что такая матрица положительно определена (все собственные значения положительны).

## 3. Алгоритм генерации

1. Устанавливается `set.seed(123)` для воспроизводимости.
2. С помощью функции `mvrnorm()` из пакета `MASS` генерируются подвыборки.
3. Вычисляется спектральное разложение для каждой ковариационной матрицы:

   $$\boldsymbol{\Sigma}_k = \mathbf{U}_k \, \mathrm{diag}(\lambda_1^{(k)}, \lambda_2^{(k)}, \lambda_3^{(k)}) \mathbf{U}_k^\top,$$

   где $\mathbf{U}_k$ — матрица собственных векторов, $\lambda_i^{(k)}$ — собственные значения.
4. Строится 3D-диаграмма рассеяния с наложенными 95%-эллипсоидами рассеяния.

# 4. Результаты спектрального разложения

Для каждой подвыборки вычислены собственные значения и собственные векторы ковариационной матрицы. Результаты представлены в таблице 1.

Для матрицы $\mathbf{\Sigma}_3$ с корреляцией $\rho = 0.8$ между всеми парами переменных собственные значения можно вычислить аналитически. Характеристическое уравнение:

$$\det(\mathbf{\Sigma}_3 - \lambda \mathbf{I}) = 0.$$

Для матрицы вида

$$\begin{pmatrix} 1 & \rho & \rho \\ \rho & 1 & \rho \\ \rho & \rho & 1 \end{pmatrix}$$

собственные значения: $\lambda_1 = 1 + 2\rho = 2.6$, $\lambda_2 = \lambda_3 = 1 - \rho = 0.2$.

Таблица 1: Собственные значения ковариационных матриц

| Подвыборка | $\lambda_1$ | $\lambda_2$ | $\lambda_3$ |
|---|---|---|---|
| 1 (Сферическая) | 1.0000 | 1.0000 | 1.0000 |
| 2 (Вытянутая) | 5.0000 | 1.0000 | 1.0000 |
| 3 (Коррелированная) | 2.6000 | 0.2000 | 0.2000 |

Доли объяснённой дисперсии главными компонентами:

- Подвыборка 1: $\frac{1.0}{3.0} = 33.3\%$ (все компоненты равнозначны)
- Подвыборка 2: $\frac{5.0}{7.0} \approx 71.4\%$ (первая ГК), $\frac{1.0}{7.0} \approx 14.3\%$ (вторая и третья ГК)
- Подвыборка 3: $\frac{2.6}{3.0} \approx 86.7\%$ (первая ГК), $\frac{0.2}{3.0} \approx 6.7\%$ (вторая и третья ГК)

Для подвыборки 3 первая главная компонента объясняет более 86% дисперсии, что указывает на сильную линейную зависимость между переменными.

# 5. Визуализация

На рис. 1 показаны три подвыборки:

- красные точки и полупрозрачный красный эллипсоид — подвыборка 1;
- синие точки и синий эллипсоид — подвыборка 2;
- тёмно-зелёные точки и зелёный эллипсоид — подвыборка 3.

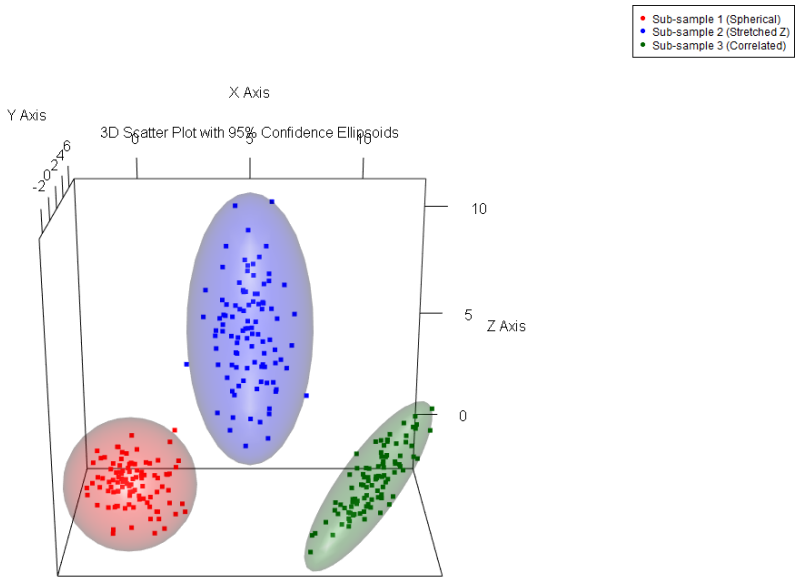


Рис. 1: 3D-облака и 95%-эллипсоиды рассеяния

## 5.1. Подвыборка 1 (диагональная $\Sigma_1$)

- Поскольку $\Sigma_1$ диагональна и имеет равные элементы, главные направления совпадают с координатными осями.
- Эллипсоид имеет сферическую форму: все полуоси равны $c\sqrt{1.0} \approx 2.80$.
- Отсутствие наклона — прямое следствие нулевых ковариаций $s_{12} = s_{13} = s_{23} = 0$.
- Для нормального распределения некоррелированность $\Leftrightarrow$ независимость.

## 5.2. Подвыборка 2 (вытянутая по оси Z)

- Диагональная матрица с различными дисперсиями.
- Эллипсоид вытянут вдоль оси $Z$: $\lambda_1 = 5.0$, полуось длиной $c\sqrt{5.0} \approx 6.25$.
- По осям $X$ и $Y$ полуоси равны $c\sqrt{1.0} \approx 2.80$.
- Переменные независимы, но масштабы различны.

## 5.3. Подвыборка 3 (общая корреляция)

- Собственные векторы уже не совпадают ни с одной координатной осью.
- Ориентация эллипсоида — произвольная в $\mathbb{R}^3$; его оси образуют базис ГК.
- Длины полуосей: $c\sqrt{2.6} \approx 4.52$, $c\sqrt{0.2} \approx 1.25$, $c\sqrt{0.2} \approx 1.25$, где $c \approx 2.795$.

- Большое различие между $\lambda_1 = 2.6$ и $\lambda_2 = \lambda_3 = 0.2$ говорит о сильной вытянутости вдоль первого главного направления.
- Первый собственный вектор для такой матрицы имеет вид $\mathbf{e}_1 = \frac{1}{\sqrt{3}}(1,1,1)^\top$ — направление максимальной дисперсии совпадает с главной диагональю пространства.

Этот случай наиболее сложен для интуитивного восприятия — именно он демонстрирует необходимость PCA: проекция на главные оси позволяет получить некоррелированные координаты $Y_1, Y_2, Y_3$, в которых распределение «распрямляется».

# 6. Выводы

Ковариационная матрица $\mathbf{\Sigma}$ полностью управляет геометрией нормального облака точек:

- диагональные элементы $\Rightarrow$ масштаб по осям;
- недиагональные элементы $\Rightarrow$ поворот и сдвиг в пространстве.

Наличие корреляций ведёт к повороту эллипсоида рассеяния в подпространствах, соответствующих парам связанных переменных. Геометрическая интерпретация главных компонент подтверждается экспериментально: главные направления $\Leftrightarrow$ оси эллипсоидов.

Визуализация в 3D (особенно с наложенными эллипсоидами) является эффективным инструментом для диагностики структуры данных и проверки адекватности многомерных моделей.

# 7. Приложение: Код на R

```r
# Load necessary libraries
# If you don't have these installed, run: install.packages(c("MASS", "plotly", "scatterplot3d", "rgl", "ellipse"))
library(MASS)
library(plotly)
library(scatterplot3d)
library(rgl)
library(ellipse)

# Set seed for reproducibility
set.seed(123)

# --- Configuration & Parameters ---

# Define Sample Volumes (Sample Sizes)
n1 <- 100
n2 <- 100
n3 <- 100

# Define Parameters for Sub-sample 1: Standard Spherical
# Centered at (0,0,0), Uncorrelated, Equal Variance
mu1 <- c(0, 0, 0)
sigma1 <- matrix(c(1, 0, 0,
                   0, 1, 0,
                   0, 0, 1), nrow=3)

# Define Parameters for Sub-sample 2: Axis-Aligned Ellipsoid
# Centered at (5,5,5), Uncorrelated, Unequal Variance (Stretched on Z-axis)
mu2 <- c(5, 5, 5)
sigma2 <- matrix(c(1, 0, 0,
                   0, 1, 0,
                   0, 0, 5), nrow=3) # Variance of Z is 5

# Define Parameters for Sub-sample 3: Rotated Ellipsoid (Correlated)
# Centered at (10,0,0), Correlated variables (Non-diagonal covariance)
mu3 <- c(10, 0, 0)
sigma3 <- matrix(c(1, 0.8, 0.8,
                   0.8, 1, 0.8,
                   0.8, 0.8, 1), nrow=3)

# --- Data Generation ---
```

```r
# Generate the sub-samples using mvrnorm (multivariate normal distribution)
data1 <- mvrnorm(n = n1, mu = mu1, Sigma = sigma1)
data2 <- mvrnorm(n = n2, mu = mu2, Sigma = sigma2)
data3 <- mvrnorm(n = n3, mu = mu3, Sigma = sigma3)

# Convert to data frames with group labels
g1 <- data.frame(
  x = data1[,1],
  y = data1[,2],
  z = data1[,3],
  group = "Sub-sample 1 (Spherical)"
)

g2 <- data.frame(
  x = data2[,1],
  y = data2[,2],
  z = data2[,3],
  group = "Sub-sample 2 (Stretched Z)"
)

g3 <- data.frame(
  x = data3[,1],
  y = data3[,2],
  z = data3[,3],
  group = "Sub-sample 3 (Correlated)"
)

# Combine into a single dataframe
df <- rbind(g1, g2, g3)

# --- Visualization ---

# Create interactive 3D scatter plot using plotly
p <- plot_ly(data = df,
        x = ~x,
        y = ~y,
        z = ~z,
        color = ~group,
        colors = c("red", "blue", "green"),
        type = 'scatter3d',
        mode = 'markers',
        marker = list(size = 3)) %>%
  layout(title = "3D Scatter Plot of Multivariate Normal Sub-samples",
         scene = list(
           xaxis = list(title = 'X Axis'),
           yaxis = list(title = 'Y Axis'),
           zaxis = list(title = 'Z Axis')
         ))

# Display interactive plot
print(p)

# Generate PDF using static plot
pdf("Task1_Results.pdf", width = 8, height = 6)

all_data <- rbind(data1, data2, data3)
colors <- c(rep("red", n1), rep("blue", n2), rep("green", n3))
shapes <- c(rep(16, n1), rep(17, n2), rep(18, n3))

s3d <- scatterplot3d(all_data,
                     color = colors,
                     pch = shapes,
                     main = "3D Scatter Plot of Multivariate Normal Sub-samples",
                     xlab = "X Axis", ylab = "Y Axis", zlab = "Z Axis",
                     grid = TRUE, box = FALSE)

# Add a legend
legend(s3d$xyz.convert(12, 0, 10), legend = c("Sub-sample 1 (Spherical)",
                                              "Sub-sample 2 (Stretched Z)",
                                              "Sub-sample 3 (Correlated)"),
       col =  c("red", "blue", "green"), pch = c(16, 17, 18))

# Close the PDF device to save the file
dev.off()

# --- Spectral Decomposition and PCA Analysis ---

# Function to compute spectral decomposition and save results
analyze_covariance <- function(sigma, mu, name) {
  # Spectral decomposition: Sigma = U * diag(lambda) * U^T
  eigen_result <- eigen(sigma)
  eigenvalues <- eigen_result$values
  eigenvectors <- eigen_result$vectors

  # 95% confidence ellipsoid: c^2 = chi^2(3, 0.95) ≈ 7.815
  c_value <- sqrt(qchisq(0.95, df = 3))

  # Semi-axes lengths
  semi_axes <- c_value * sqrt(eigenvalues)

  # Proportion of variance explained by each PC
  prop_var <- eigenvalues / sum(eigenvalues)
  cum_prop_var <- cumsum(prop_var)

  # Create ellipsoid
  ellipsoid <- ellipse3d(sigma, centre = mu, level = 0.95)

  return(list(
    name = name,
```

```
    eigenvalues = eigenvalues,
    eigenvectors = eigenvectors,
    semi_axes = semi_axes,
    prop_var = prop_var,
    cum_prop_var = cum_prop_var,
    ellipsoid = ellipsoid,
    c_value = c_value
  ))
}

# Analyze all three sub-samples
analysis1 <- analyze_covariance(sigma1, mu1, "Sub-sample 1")
analysis2 <- analyze_covariance(sigma2, mu2, "Sub-sample 2")
analysis3 <- analyze_covariance(sigma3, mu3, "Sub-sample 3")

# --- Print Analysis Results ---
cat("\n=== SPECTRAL DECOMPOSITION ANALYSIS ===\n\n")

for (analysis in list(analysis1, analysis2, analysis3)) {
  cat(sprintf("\n%s:\n", analysis$name))
  cat("Eigenvalues λ():", paste(round(analysis$eigenvalues, 4), collapse = ", "), "\n")
  cat("Semi-axes lengths (√λc):", paste(round(analysis$semi_axes, 4), collapse = ", "), "\n")
  cat("Proportion of variance:", paste(round(analysis$prop_var * 100, 2), "%", collapse = ", "), "\n")
  cat("Cumulative proportion:", paste(round(analysis$cum_prop_var * 100, 2), "%", collapse = ", "), "\n")
  cat("\nEigenvectors (columns):\n")
  print(round(analysis$eigenvectors, 4))
}

# --- 3D Visualization with Ellipsoids using rgl ---
open3d()
par3d(windowRect = c(0, 0, 1200, 800))

# Plot points
points3d(data1, col = "red", size = 5)
points3d(data2, col = "blue", size = 5)
points3d(data3, col = "darkgreen", size = 5)

# Plot ellipsoids
shade3d(analysis1$ellipsoid, col = "red", alpha = 0.2)
shade3d(analysis2$ellipsoid, col = "blue", alpha = 0.2)
shade3d(analysis3$ellipsoid, col = "darkgreen", alpha = 0.2)

# Add axes
axes3d()
title3d("3D Scatter Plot with 95% Confidence Ellipsoids",
        xlab = "X Axis", ylab = "Y Axis", zlab = "Z Axis")

# Add legend
legend3d("topright",
         legend = c("Sub-sample 1 (Spherical)",
                    "Sub-sample 2 (Stretched Z)",
                    "Sub-sample 3 (Correlated)"),
         col = c("red", "blue", "darkgreen"),
         pch = 16)

# Save 3D plot
rgl.snapshot("Task1_3D_with_ellipsoids.png", fmt = "png")
cat("\n3D plot with ellipsoids saved as Task1_3D_with_ellipsoids.png\n")

# --- Save analysis results to file for LaTeX report ---
sink("Task1_analysis_results.txt")
cat("=== COVARIANCE MATRICES ===\n\n")
cat("Sub-sample 1 (Spherical):\n")
print(sigma1)
cat("\nSub-sample 2 (Stretched Z):\n")
print(sigma2)
cat("\nSub-sample 3 (Correlated):\n")
print(sigma3)

cat("\n\n=== SPECTRAL DECOMPOSITION ===\n\n")
for (analysis in list(analysis1, analysis2, analysis3)) {
  cat(sprintf("\n%s:\n", analysis$name))
  cat("Eigenvalues:", paste(round(analysis$eigenvalues, 4), collapse = ", "), "\n")
  cat("Semi-axes:", paste(round(analysis$semi_axes, 4), collapse = ", "), "\n")
  cat("Prop. variance:", paste(round(analysis$prop_var * 100, 2), "%", collapse = ", "), "\n")
}
sink()

print("PDF generated successfully as Task1_Results.pdf")
print("Analysis results saved to Task1_analysis_results.txt")
```