Data learning

Машинное обучение
Лабораторная работа № 2

Mahalanobis distance-based classifier

М25-514 Киргизов Т.К.

Исходные данные

Исходные данные предоставлялись с виде CSV-файла, состоящего из точек и их классов $x1, x2, label$.

При предварительной подготовке было выявлено **150** точек класса **"+1"** и **200** точек класса **"-1"**.

По условиям задания было использовано `Holdout` разделение данных на обучающую и тестовую выборки в пропорциях **70/30**. После чего выявилось количество точек **"+1"** класса **(140)** и количество точек **"-1"** класса **(105)** для обучающей выборки.

График 1. Визуализация тренировочных данных с разделением на классы

Задание 1. Вычисление расстояний Махаланобиса

Для расчета расстояния Махаланобиса для всех 6 случаев матриц (равные скалярные, равные диагональные, равные, различные скалярные, различные диагональные, различные полные) были использованы две функции, отвечающие за создание матриц для того или иного случая:

```
def create_scalar_matrix(data: ArrayLike): ...
def create_diagonal_matrix(data: ArrayLike): ...
```

1. Скалярные равные: $\Sigma_1 = \Sigma_2 = \sigma^2 I = \begin{bmatrix} \sigma^2 & 0 \\ 0 & \sigma^2 \end{bmatrix}$ – одинаковая матрица для обоих классов.

$$\begin{bmatrix} 3.76124419 & 0 \\ 0 & 3.76124419 \end{bmatrix}$$

2. Равные диагональные: $\Sigma_1 = \Sigma_2 = \begin{bmatrix} \sigma_1^2 & 0 \\ 0 & \sigma_2^2 \end{bmatrix}$ – разные дисперсии по $x1, x2$.

$$\begin{bmatrix} 2.3560216 & 0 \\ 0 & 5.1664667 \end{bmatrix}$$

3. Различные скалярные: $\Sigma_1 = \sigma_1^2 I, \quad \Sigma_2 = \sigma_2^2 I$ – свои матрицы для каждого класса.

$$(-1) = \begin{bmatrix} 1.59468766 & 0 \\ 0 & 1.59468766 \end{bmatrix} (+1) = \begin{bmatrix} 2.68760669 & 0 \\ 0 & 2.68760669 \end{bmatrix}$$

4. Различные диагональные: $\Sigma_1 = \begin{bmatrix} \sigma_{11}^2 & 0 \\ 0 & \sigma_{12}^2 \end{bmatrix}, \quad \Sigma_2 = \begin{bmatrix} \sigma_{21}^2 & 0 \\ 0 & \sigma_{22}^2 \end{bmatrix}$ – своя диагональная матрица для каждого класса

$$(-1) = \begin{bmatrix} 0.98241008 & 0 \\ 0 & 2.20696524 \end{bmatrix} (+1) = \begin{bmatrix} 1.77902869 & 0 \\ 0 & 3.59618468 \end{bmatrix}$$

5. Равные полные: $\Sigma_1 = \Sigma_2 = \begin{bmatrix} \sigma_1^2 & \rho\sigma_1\sigma_2 \\ \rho\sigma_1\sigma_2 & \sigma_2^2 \end{bmatrix}$ – оба класса имеют одинаковую полную матрицу, есть корелляция $\rho ! = 0$

$$\begin{bmatrix} 2.35602163 & 2.152103 \\ 2.152103 & 5.16646676 \end{bmatrix}$$ – общая полная матрица

6. Различные полные: $\Sigma_1 \neq \Sigma_2$ (обе полные матрицы)

$$(-1) = \begin{bmatrix} 0.98241008 & 0.48693818 \\ 0.48693818 & 2.20696524 \end{bmatrix} (+1) = \begin{bmatrix} 1.77902869 & 0.71335945 \\ 0.71335945 & 3.59618468 \end{bmatrix}$$

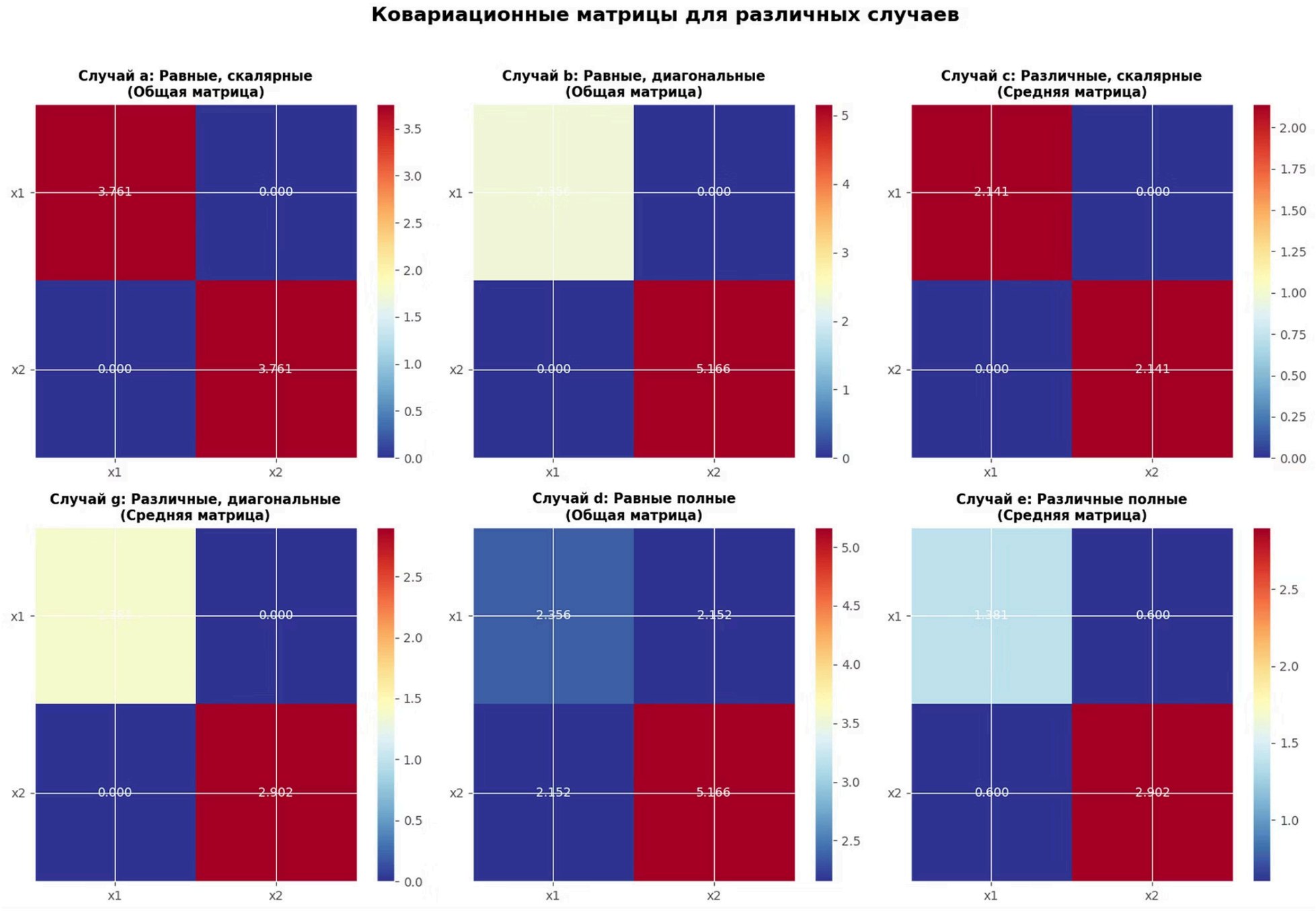

График 2. Визуализация ковариационных матриц

Расстояние Махаланобиса

Расстояние Махаланобиса – это мера расстояния между точкой и распределением данных, которая учитывает:

1. **Ковариацию между признаками** (корреляцию)
2. **Различие в масштабе** (дисперсии по разным осям)

Формула для расстояния от точки x до распределения с центром μ:

$$D^2(x) = (x - \mu)^T \Sigma^{-1} (x - \mu)$$

- x - вектор признаков точки
- μ - вектор средних значений класса (центр распределения)
- Σ - ковариационная матрица (в данном случае $2 * 2$)

С использованием данной формулы и формул для построения матриц были получены данные для визуализации диаграммы рассеяния, а также, была проведена граница для классов.

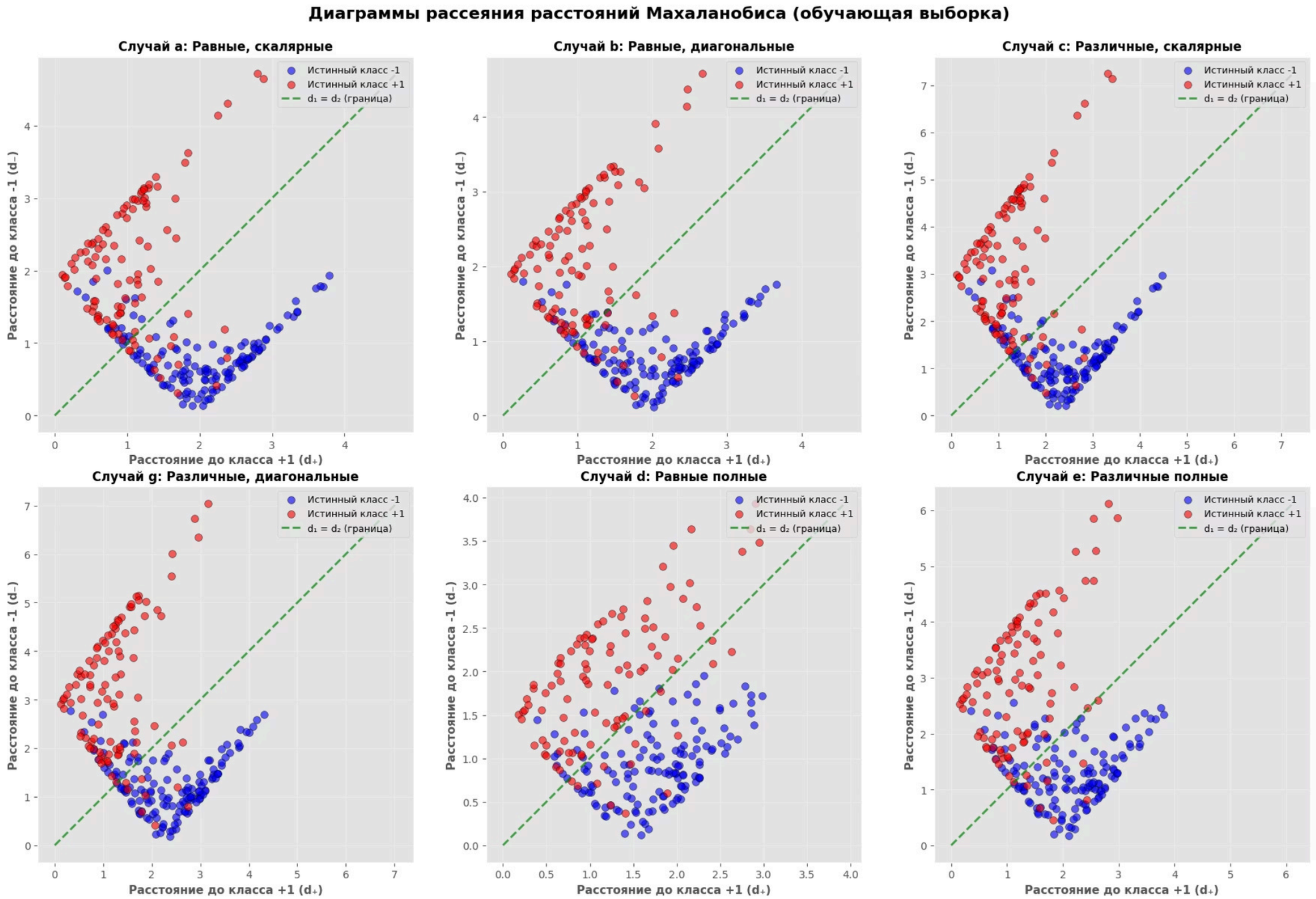

График 3. Визуализация диаграмм рассеяния расстояний Махаланобиса для всех ковариационных матри

Задание 2. Анализ и визуализация границ классов

Границы классов (desicion boundary) $d(x)$ – геометрическое место точек в пространстве признаков, которые равноудалены по расстоянию Махаланобиса от обоих классов.

$$d_+(x) = d_-(x)$$

Для визуализации границ классов для всех случаев матриц были вычислены диапазоны, создана сетка, после чего преобразована в массив точек для вычислений по заданной формуле.

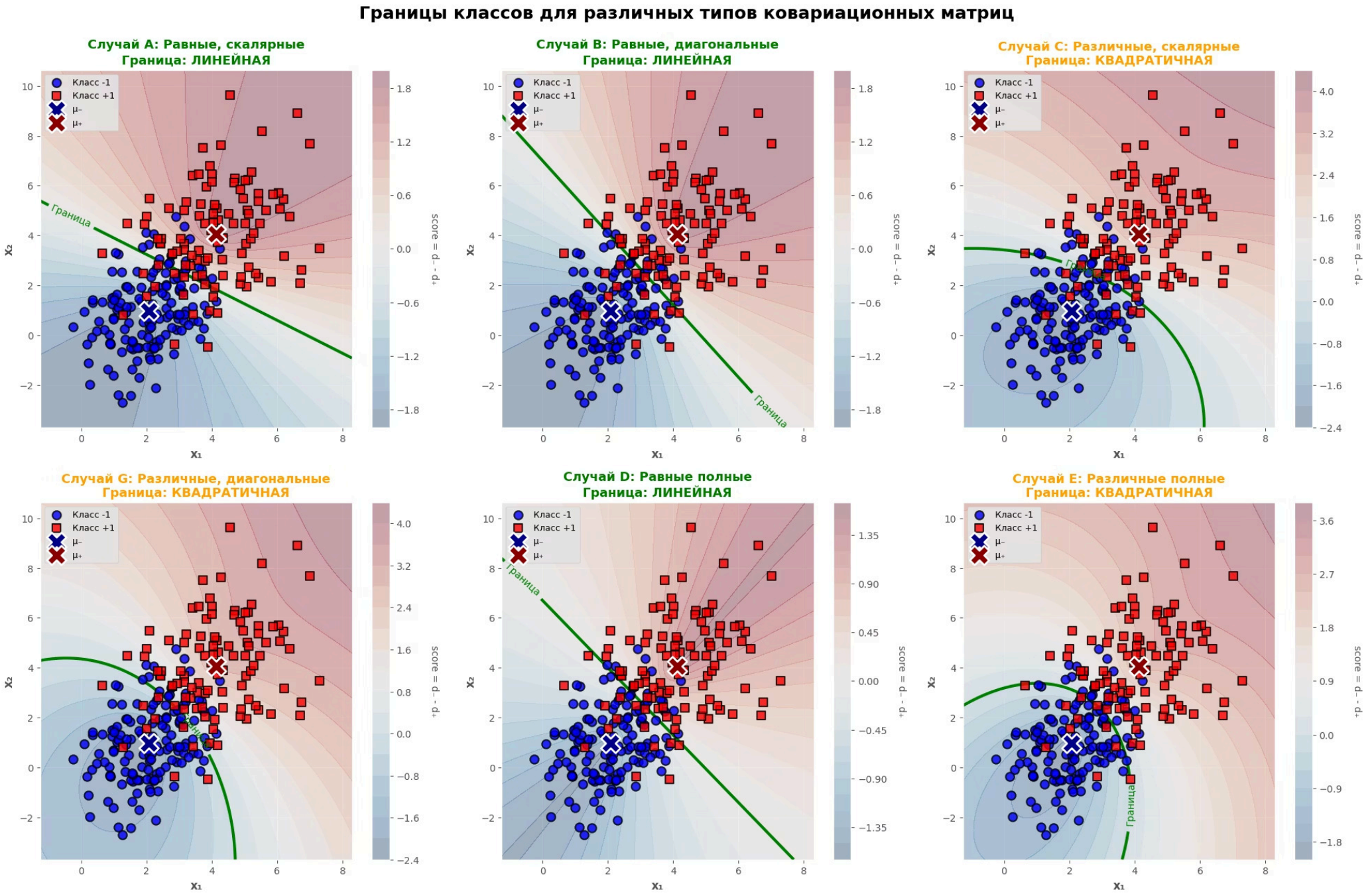

График 4. Визуализация границ классов для различных типов ковариационных матриц

Анализ результатов классификации на тренировочных данных

Ниже представлены результаты работы классификатора, основанного на расстоянии Махаланобиса, для различных конфигураций ковариационных матриц на тренировочном наборе данных.

Тип Матриц	Граница	Сложность	Ошибок	Точность
a) Равные, скалярные	Линейная	0.0120	35 из 245	85.71%
b) Равные, диагональные	Линейная	0.0125	29 из 245	88.16%
c) Различные, скалярные	Квадратичная	0.0146	34 из 245	86.12%
g) Различные, диагональные	Квадратичная	0.0150	33 из 245	86.53%
d) Равные полные	Линейная	0.0087	30 из 245	87.76%
e) Различные полные	Квадратичная	0.0127	32 из 245	86.94%

Примечание: Сложность модели указывает на вычислительные затраты и потенциальную переобучаемость. Чем выше значение, тем сложнее модель.

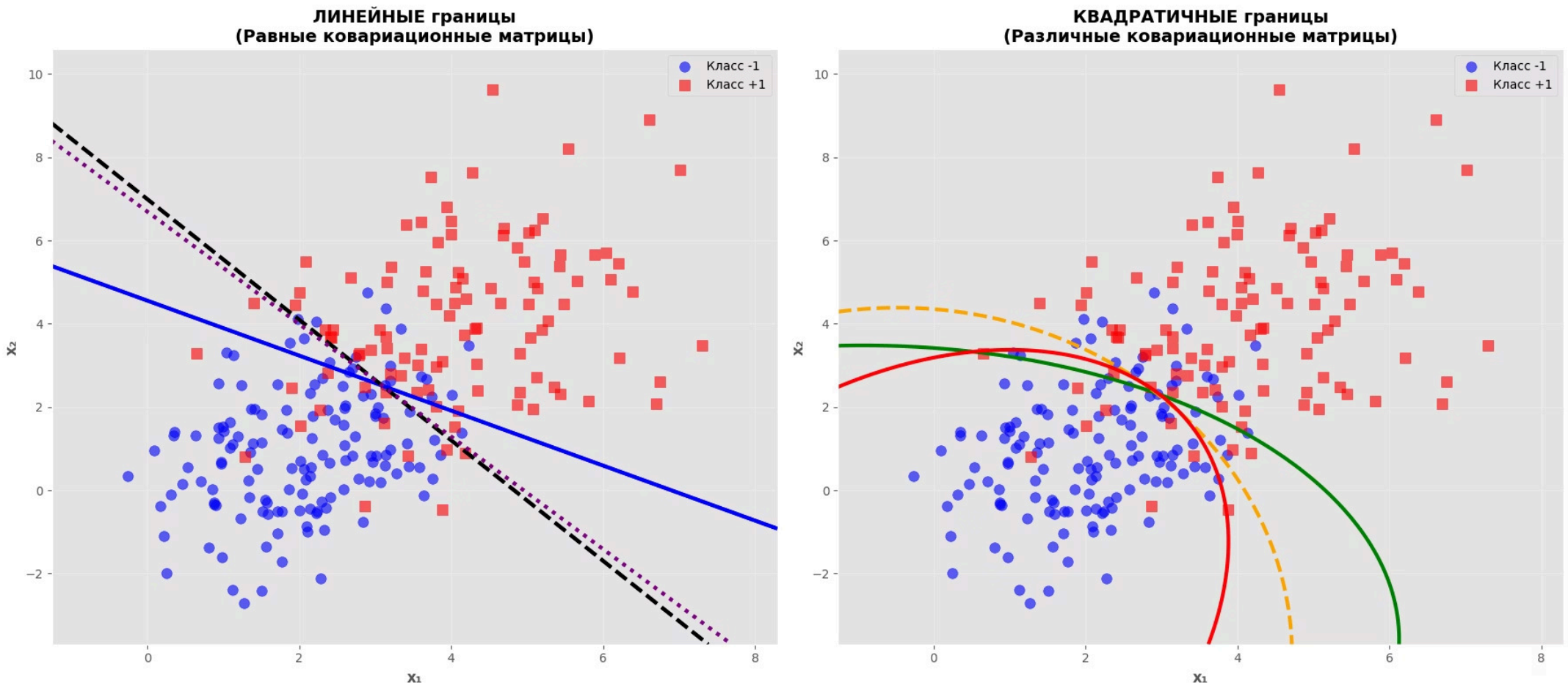

График 5. Визуализация границ для двух типов ковариационных матриц

Задание 3. Расчет показателей точности на train/test

1️⃣ Accuracy **(Точность)** Доля всех правильных предсказаний среди всех предсказаний.

2️⃣ Error Rate **(Доля ошибок)** Доля неправильных предсказаний.

3️⃣ Sensitivity / Recall / TPR **(Чувствительность)** Из всех истинных положительных, сколько мы нашли?

4️⃣ Specificity / TNR **(Специфичность)** Из всех истинных отрицательных, сколько мы правильно определили?

5️⃣ Precision / PPV **(Точность положительных)** Из всех предсказанных положительных, сколько действительно положительные?

6️⃣ Fall-out / FPR **(Доля ложноположительных)** Из всех истинных отрицательных, сколько мы ошибочно назвали положительными?

7️⃣ F1-Score **(Гармоническое среднее)** Баланс между Precision и Recall.

8️⃣ Cohen's Kappa **(Каппа Коэна)** Насколько классификатор лучше случайного угадывания, с учётом распределения классов.

$$Accuracy = \frac{TP + TN}{FP + FN + TP + TN}, Errorrate = \frac{FP + FN}{FP + FN + TP + TN}$$

$$Sensitivity = Recall = TPR = \frac{TP}{FN + TP}, Specificity = TNR = \frac{TN}{TN + FP}$$

$$Precision = PPV = \frac{TP}{TP + FP}, Fallout = FPR = \frac{FP}{TN + FP} = 1 - Specificity$$

$$Fallout = FPR = \frac{FP}{TN + FP} = 1 - Specificity, F_1 = 2 * \frac{Precision * Recall}{Precision + Recall}$$

$$\kappa = \frac{p_0 - p_e}{1 - p_e}$$

- p_0 - наблюдаемая согласованность (Accuracy)
- p_e - ожидаемая случайная согласованность

$$p_e = \frac{(TP + FP)(TP + FN) + (TN + FN)(TN + FP)}{(TP + TP + FP + FN)^2}$$

- $\kappa = 1$ — идеальное согласие

- $\kappa = 0$ — согласие на уровне случайности

- $\kappa < 0$ — хуже случайного (маловероятно)

Анализ и сводки вычисления метрик

Ниже представлена сводная таблица ключевых метрик классификации и анализа переобучения для каждой конфигурации ковариационных матриц на обучающих и тестовых данных.

Тип Матриц	Accuracy (Train)	F1-Score (Train)	Accuracy (Test)	F1-Score (Test)	Разница Accuracy	Разница F1-Score	Переобучение
Равные, скалярные	0.8571	0.8357	0.9333	0.9195	-0.0762	-0.0839	НЕТ
Равные, диагональные	0.8816	0.8612	0.9619	0.9556	-0.0803	-0.0943	НЕТ
Различные, скалярные	0.8612	0.8468	0.9524	0.9462	-0.0912	-0.0994	НЕТ
Различные, диагональные	0.8653	0.8520	0.9619	0.9574	-0.0966	-0.1054	НЕТ
Равные полные	0.8776	0.8571	0.9619	0.9556	-0.0844	-0.0984	НЕТ
Различные полные	0.8694	0.8571	0.9524	0.9474	-0.0830	-0.0902	НЕТ

Из таблицы видно, что ни одна из моделей не демонстрирует переобучения, поскольку разница в метриках Accuracy и F1-Score между обучающей и тестовой выборками значительно меньше 5%.

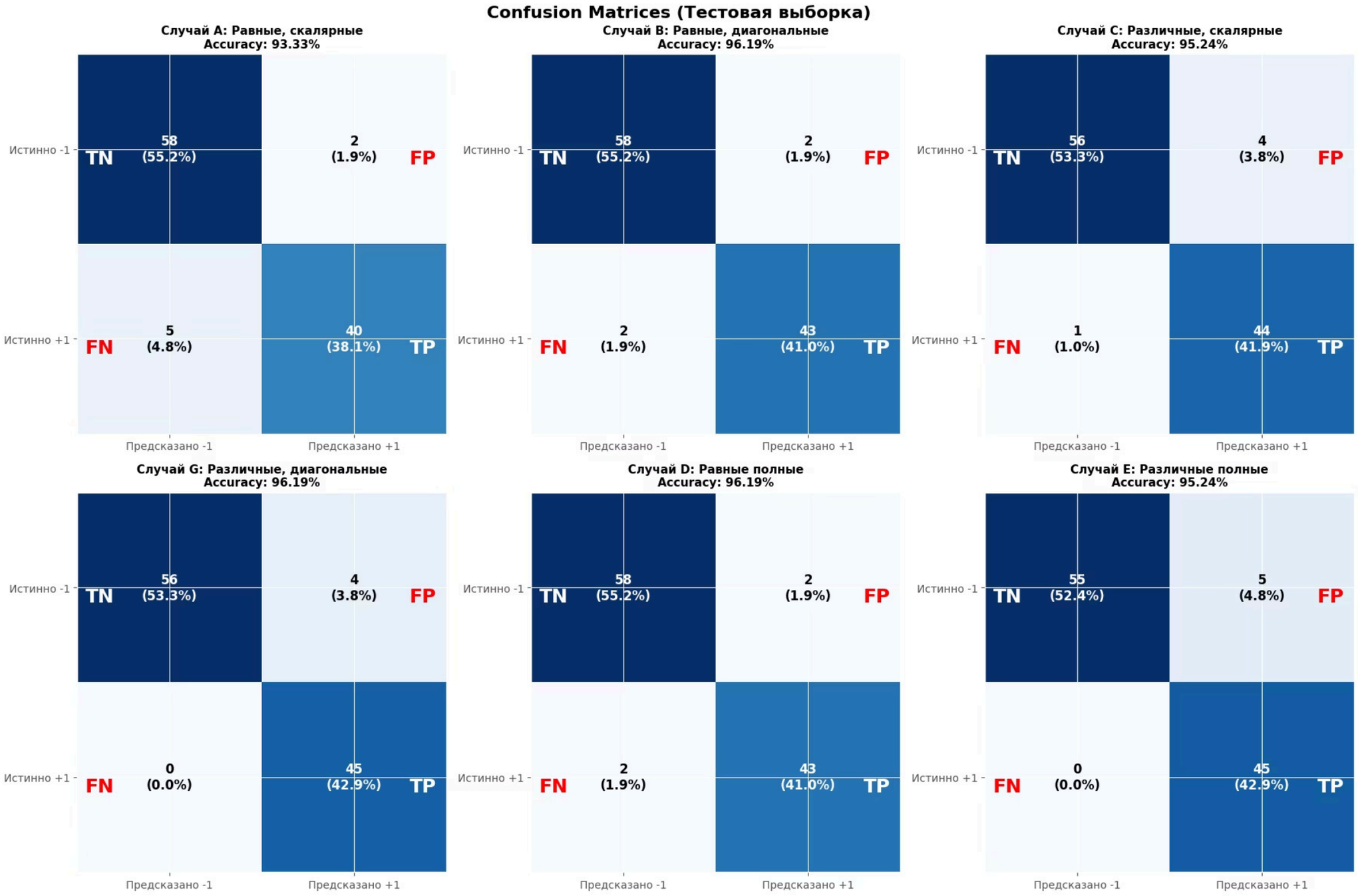

График 6. Матрицы ошибок для разных типов ковариационных матриц

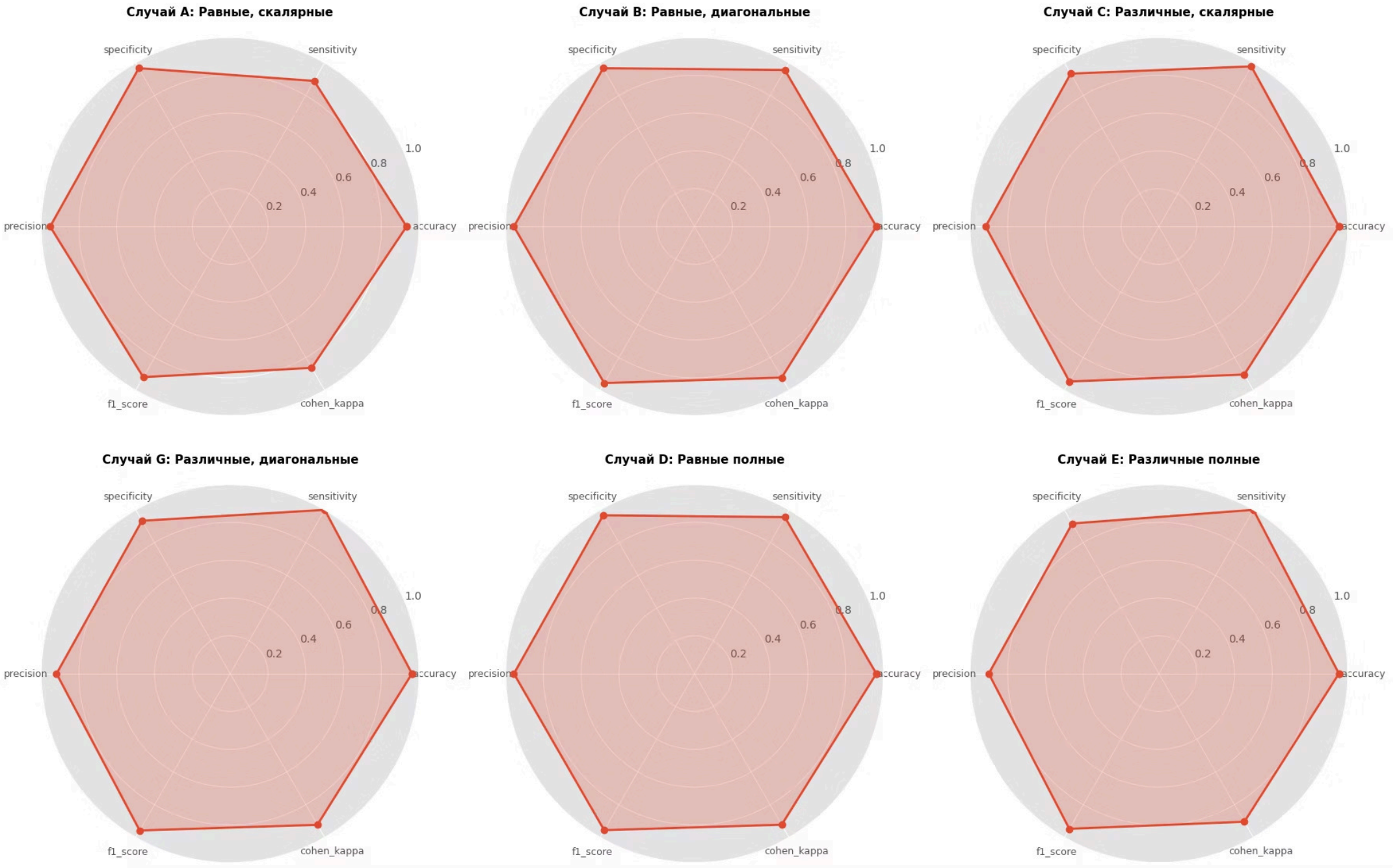

График 7. Радарные диаграммы метрик для различных ковариационных матриц для test

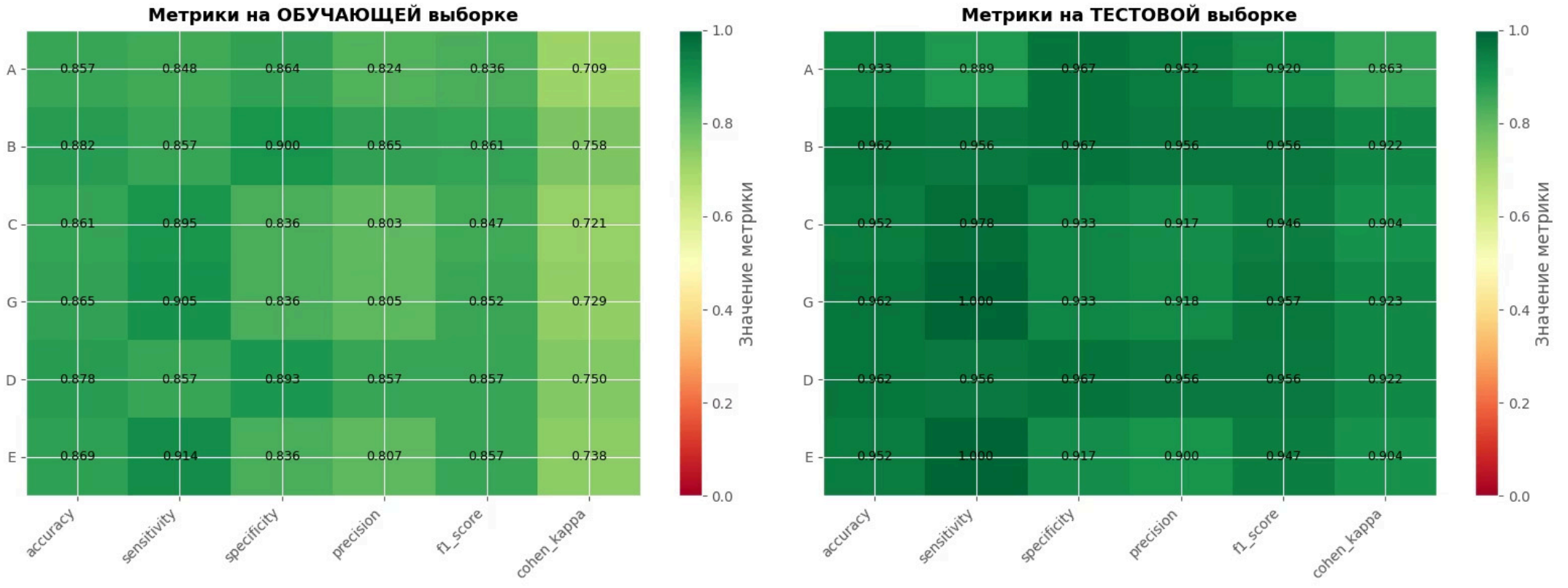

График 8. Тепловая карта метрик для train/test

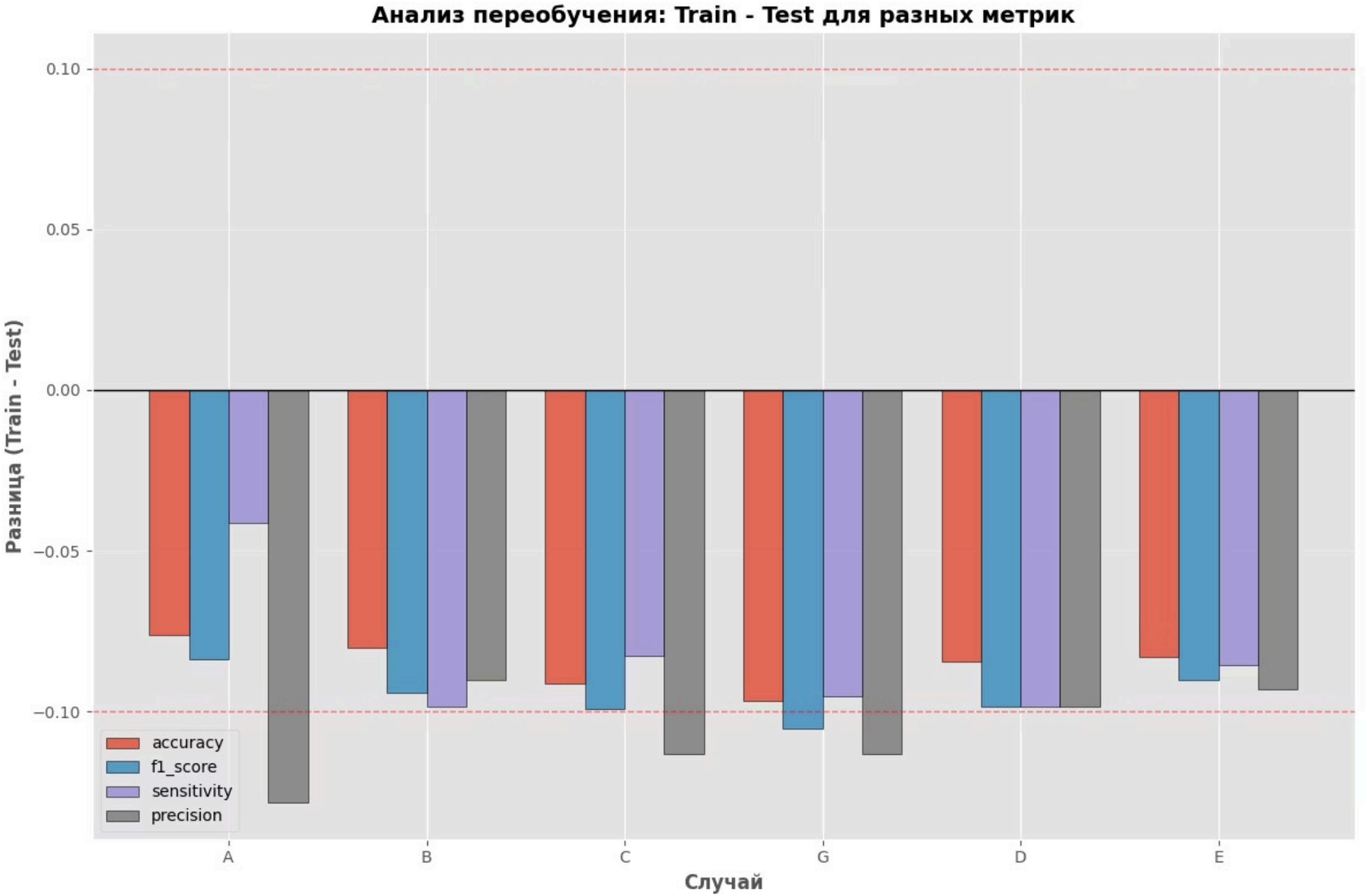

График 9. Анализ переобучения

Итоговое ранжирование моделей по тестовым метрикам

На основе анализа метрик, рассчитанных на тестовых данных, было произведено ранжирование всех конфигураций классификаторов. В таблице представлены значения метрик и их относительные ранги по каждой категории, а также усредненный ранг для комплексной оценки.

№	Название	Accuracy	Sensitivity	Specificity	F1-Score	Rank_Acc	Rank_F1	Rank_Sens	Rank_Spec	Avg_Rank
G	Различные, диагональные	0.961905	1.000000	0.933333	0.957447	1	1	1	4	1.75
B	Равные, диагональные	0.961905	0.955556	0.966667	0.955556	1	2	4	1	2.00
D	Равные полные	0.961905	0.955556	0.966667	0.955556	1	2	4	1	2.00
E	Различные полные	0.952381	1.000000	0.916667	0.947368	4	4	1	6	3.75
C	Различные, скалярные	0.952381	0.977778	0.933333	0.946237	4	5	3	4	4.00
A	Равные, скалярные	0.933333	0.888889	0.966667	0.919540	6	6	6	1	4.75

> ### Лучшая модель по тестовым метрикам ("в лоб")
>
> **Случай:** G
>
> **Название:** Различные, диагональные
>
> - **Accuracy:** 0.9619
> - **F1-Score:** 0.9574
> - **Sensitivity:** 1.0000
> - **Specificity:** 0.9333

Задание 4. Построение ROC (ROC AUC), PR (PR AUC)

1. **ROC кривая** — это график зависимости **TPR** (True Positive Rate) от **FPR** (False Positive Rate) при изменении порога классификации. **ROC AUC** (Area Under Curve) — площадь под ROC кривой.
2. **PR Кривая** (Precision-Recall). PR кривая — это график зависимости **Precision** от **Recall** при изменении порога классификации. **PR AUC** (Area Under PR Curve) — площадь под PR кривой.

По раннее вычисленным данным были построены ROC/PR-кривые для разных ковариационных матриц для тренировочной и тестовой выборок.

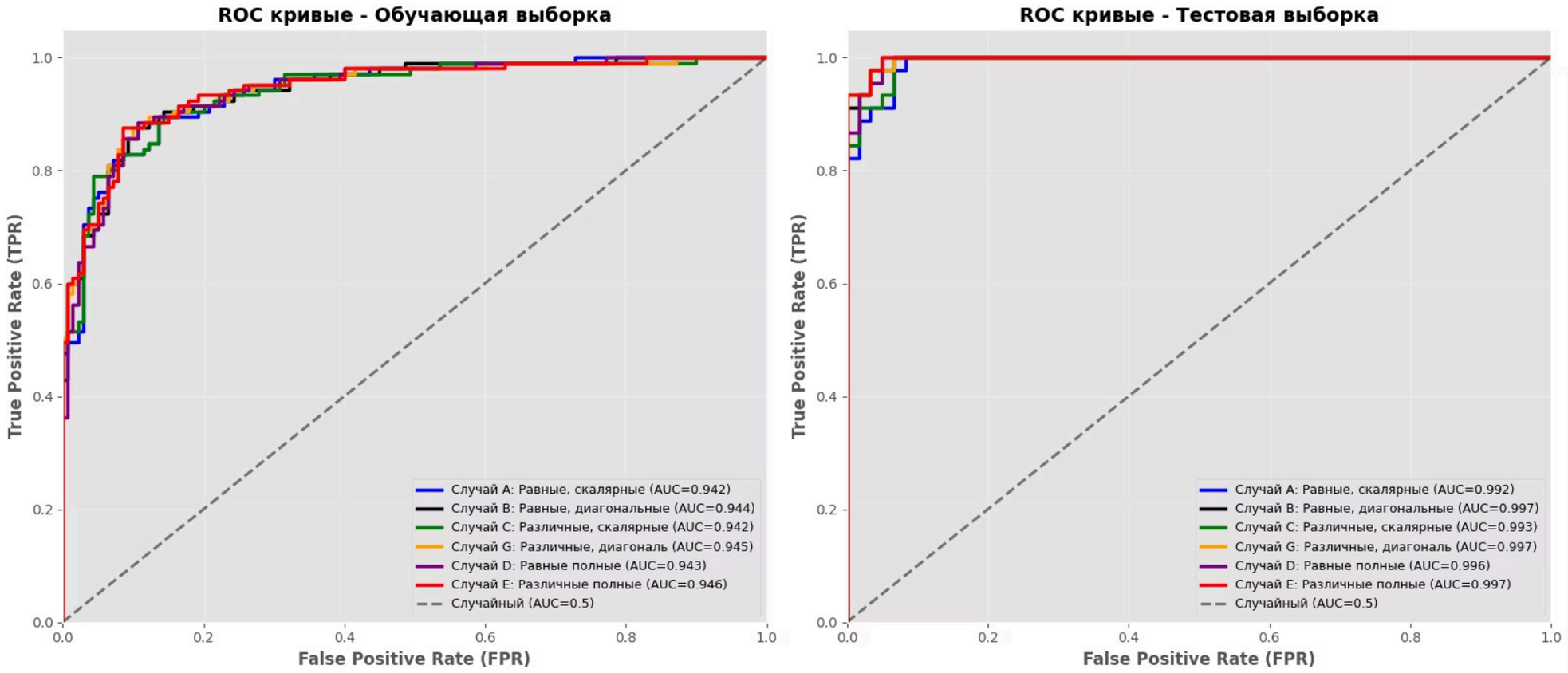

График 10. ROC-кривые для разных ковариационных матриц

PR-кривые. Табличная сводка ROC AUC/PR AUC

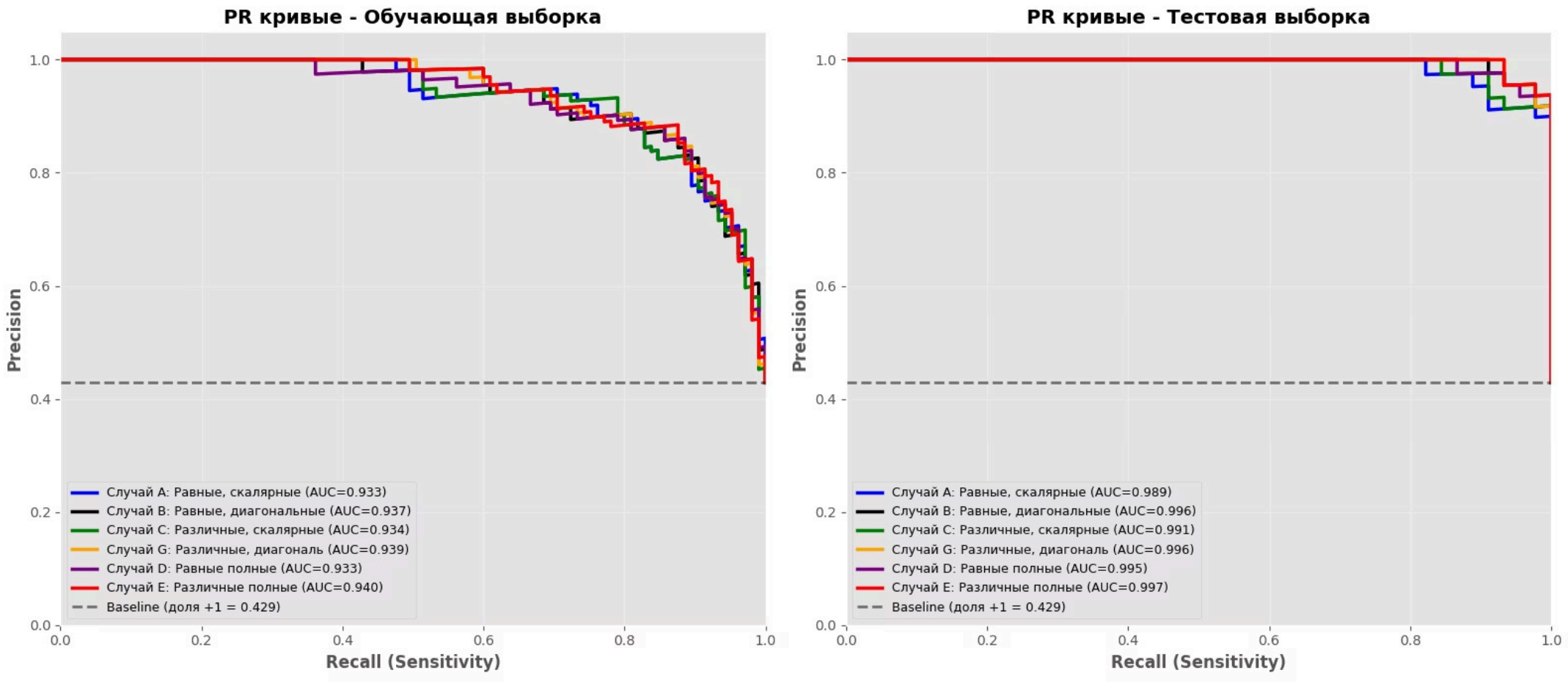

График 11. PR-кривые для разных ковариационных матриц

В данной таблице представлены сводные значения метрик ROC AUC и PR AUC для всех конфигураций ковариационных матриц, рассчитанные как на обучающих, так и на тестовых данных.

№	Название	ROC AUC Train	ROC AUC Test	PR AUC Train	PR AUC Test
A	Равные, скалярные	0.9424	0.9919	0.9330	0.9894
B	Равные, диагональные	0.9444	0.9967	0.9367	0.9957
C	Различные, скалярные	0.9416	0.9933	0.9343	0.9913
G	Различные, диагональные	0.9446	0.9970	0.9390	0.9962
D	Равные полные	0.9433	0.9959	0.9334	0.9946
E	Различные полные	0.9458	0.9974	0.9399	0.9967

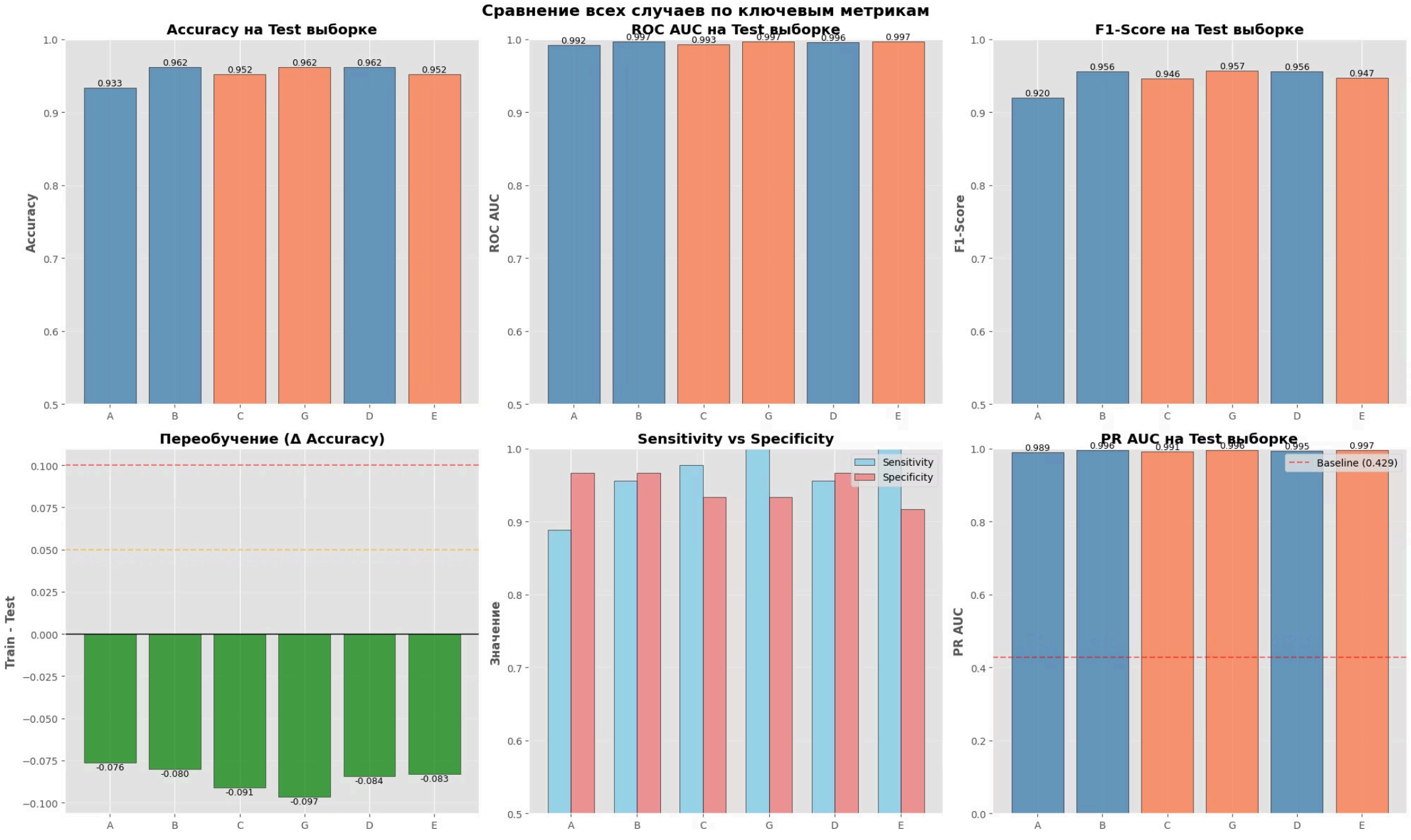

График 12. Заключительное сравнение всех ковариационных матриц по ключевым метрикам

Задание 5. Выводы: Сравнение Линейных и Квадратичных Моделей

На основе проведенного анализа были сравнены линейные и квадратичные модели классификации. Особое внимание уделялось производительности на тестовых данных и признакам переобучения.

Линейные Модели ($\Sigma_1 = \Sigma_2$)

Представляют собой более простые конфигурации с равными ковариационными матрицами, предполагая одинаковую дисперсию данных для разных классов.

- **Средний Accuracy (Test):** 0.9524
- **Средний ROC AUC (Test):** 0.9948
- **Среднее недообучение:** -0.0803 (Отсутствие переобучения)

Квадратичные Модели ($\Sigma_1 \neq \Sigma_2$)

Более сложные конфигурации, позволяющие разным классам иметь различные ковариационные матрицы, что обеспечивает большую гибкость в моделировании данных.

- **Средний Accuracy (Test):** 0.9556
- **Средний ROC AUC (Test):** 0.9959
- **Среднее недообучение:** -0.0902 (Отсутствие переобучения)

Разница в среднем показателе Accuracy между квадратичными и линейными моделями составила всего 0.32% (0.0032), что является статистически незначительной величиной.

Поскольку разница в производительности моделей незначительна (<5%), и обе группы не демонстрируют переобучения, по принципу Оккама следует выбирать **линейную модель**.

Линейные модели являются более простыми, менее требовательными к вычислительным ресурсам и обычно более устойчивыми к шуму, что делает их предпочтительными при сопоставимой точности.