**Лабораторная работа №2. Визуализация данных**

**Тема:** Анализ датасета Global Terrorism Database (GTD).

**Цель:** изучение программных средств для визуализации наборов данных.

**Задачи:**

– Установка и настройка matplotlib, seaborn;

– Изучение основных типов графиков библиотеки matplotlib;

– Изучение основных типов графиков библиотеки seaborn;

– Получение навыков анализа данных по визуальным представлениям данных.

2. Реализация индивидуального задания

2.1. Визуализация количественных признаков

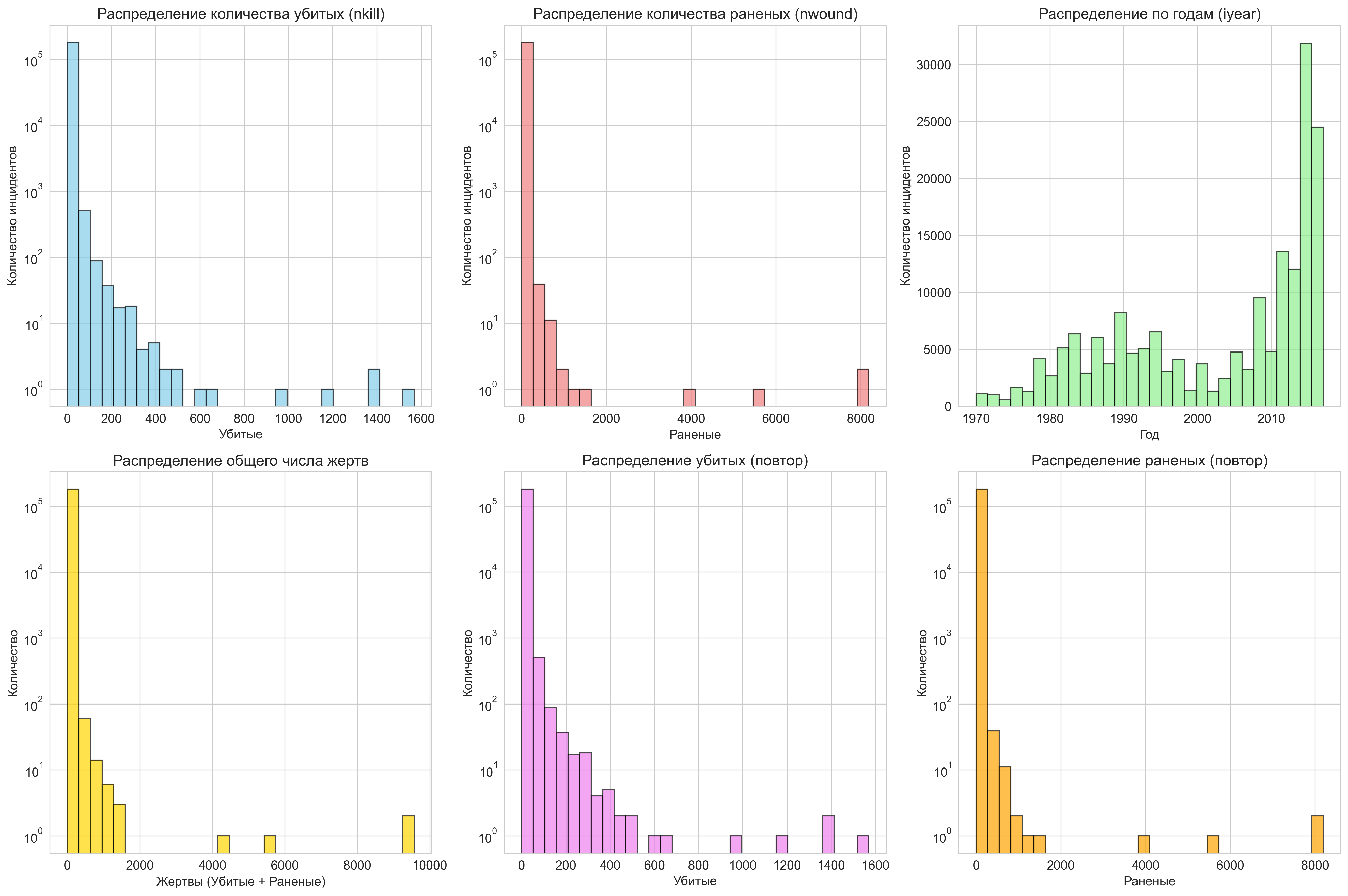


Рисунок 1 – гистограмма распределений

2.1.1. Гистограммы распределения (histograms\_detailed.png)

Анализ распределений:

* Количество убитых (nkill) и раненых (nwound): Распределения имеют экстремально сильную правостороннюю асимметрию (распределение Парето). Подавляющее большинство террористических актов обходится без жертв или с малым их количеством (0–5 человек). Однако присутствуют "тяжелые хвосты" — редкие события с огромным количеством жертв (сотни и тысячи), которые на гистограмме видны как отдельные выбросы справа. Логарифмическая шкала подтверждает степенной закон распределения насилия.
* Годы (iyear): Распределение инцидентов по времени не является равномерным. Наблюдается явный тренд на увеличение активности террористических группировок, начиная с 2010-х годов, с пиком в районе 2014–2015 годов.
* Общее число жертв (Total Victims): Повторяет структуру распределения убитых и раненых — огромное количество мелких инцидентов и единичные катастрофические события.

2.1.2. Графики "Ящик с усами" (boxplots\_detailed.png, boxplot\_by\_fuel.png)

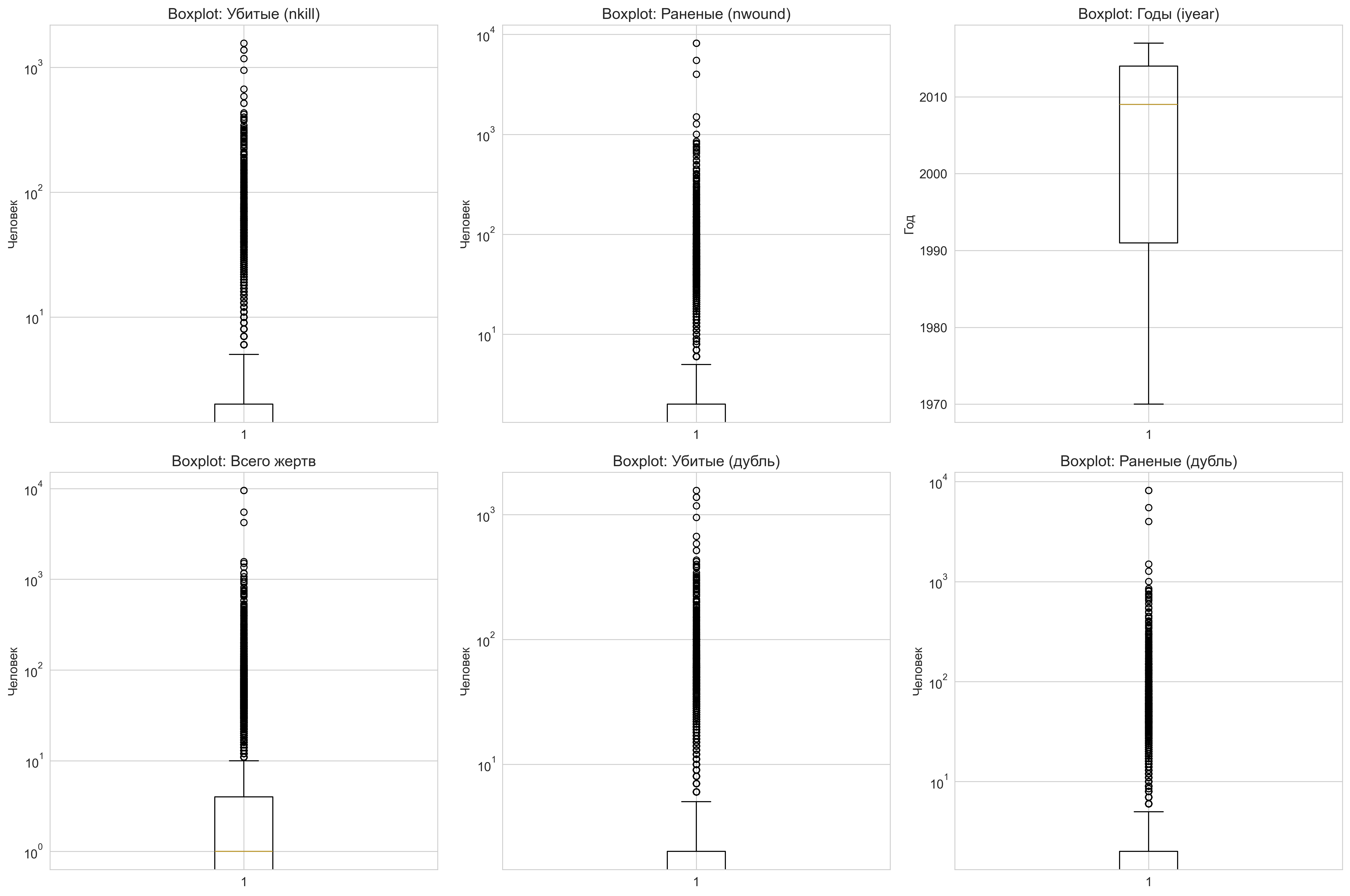


Рисунок 2 – графики ящик с усами



Рисунок 3 – графики ящик с усами по регионам

**Выявленные особенности:**

Графики Boxplot демонстрируют колоссальное количество выбросов (черные точки над "усами"). В контексте терроризма это означает, что "средний" теракт не показателен. Стандартное отклонение очень велико.

Поля nkill и nwound показывают, что медианное значение близко к нулю или единице, но верхние значения достигают тысяч (например, теракт 11 сентября или атаки в Ираке), что сжимает основной "ящик" в узкую линию внизу графика.

**Сравнение по регионам:**

Регионы Middle East & North Africa и Sub-Saharan Africa демонстрируют самые высокие показатели разброса и медианного количества жертв, что говорит о высокой летальности конфликтов в этих зонах.

Western Europe показывает большое количество выбросов (отдельные крупные теракты), но само "тело" ящика находится низко, что говорит о том, что большинство инцидентов там менее летальны по сравнению с горячими точками.

2.2. Визуализация категориальных признаков

2.2.1. Анализ производителей и характеристик (categorical\_analysis.png, countplots\_detailed.png)

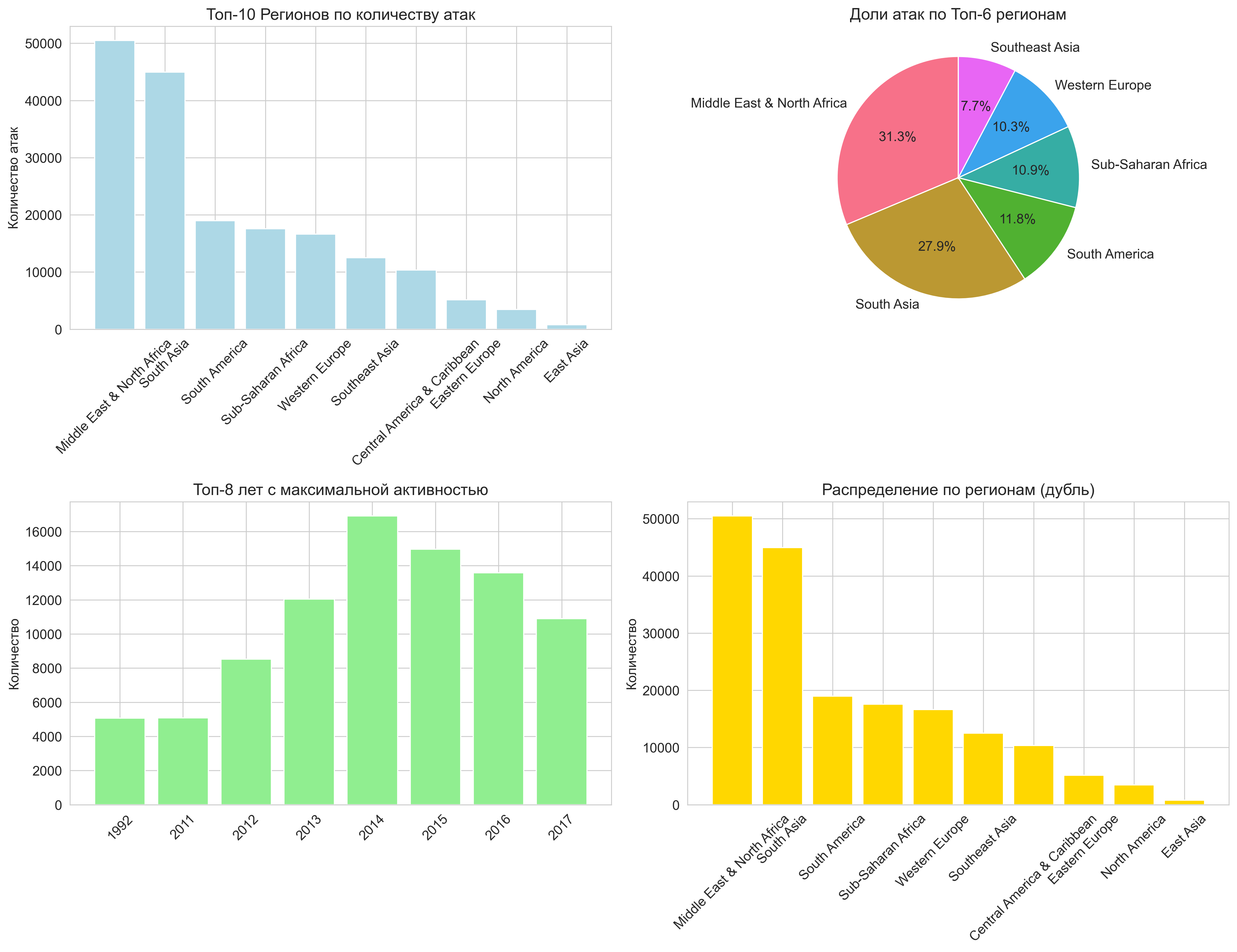


Рисунок 4 – анализ характеристик

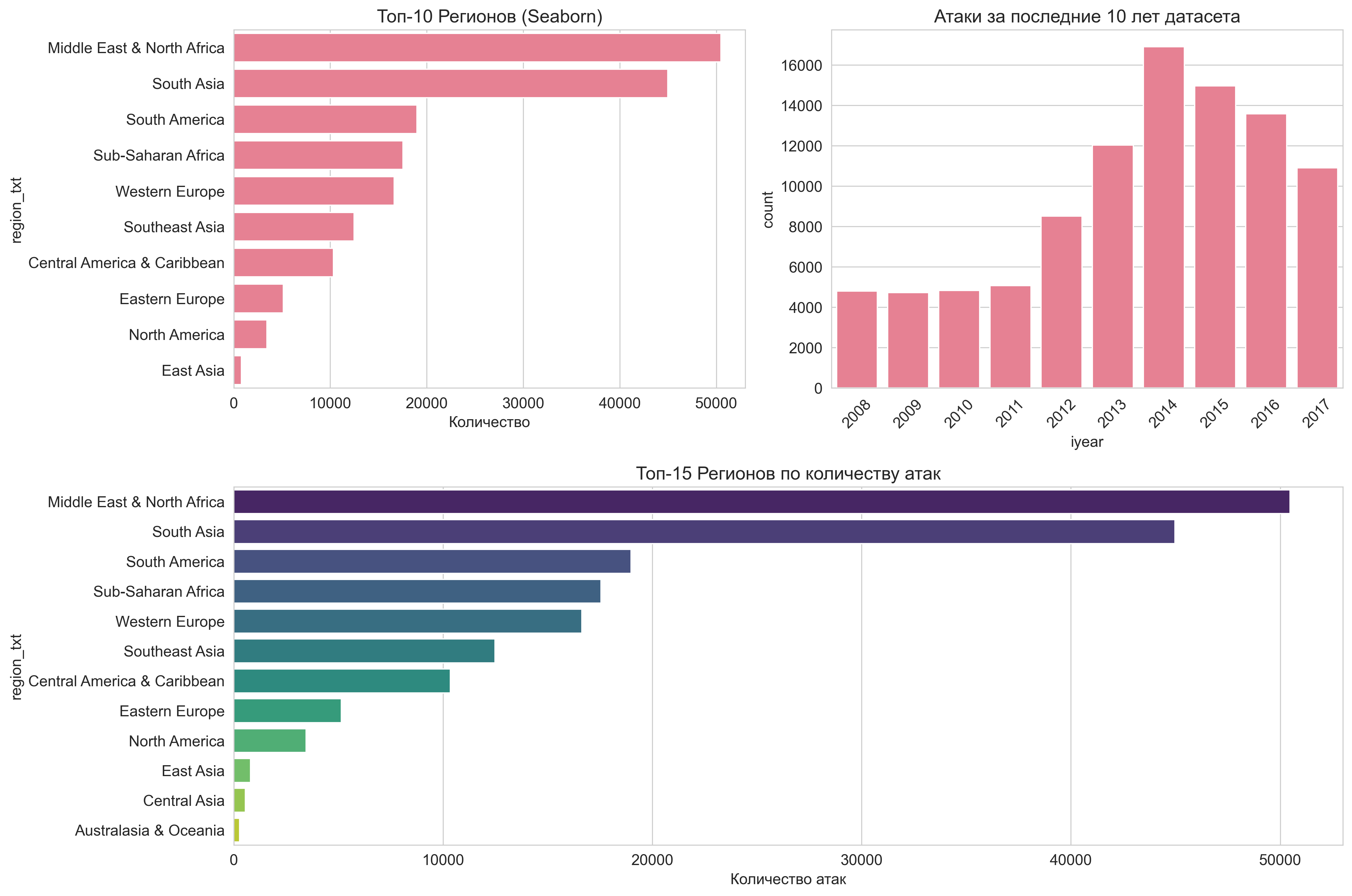


Рисунок 5 – анализ производителей

Выявленные особенности:

География (Pie Chart и Bar Plot): Абсолютным лидером по количеству атак является регион Middle East & North Africa (около 31.3% всех атак), за ним следует South Asia (27.9%). Это указывает на геополитическую концентрацию террористической активности.

Динамика (Bar Plot по годам): График "Атаки за последние 10 лет" показывает резкий рост количества инцидентов с 2011 по 2014 год (период активизации ИГИЛ и конфликта в Сирии), после чего наблюдается постепенный спад к 2017 году.

Региональные различия: В Топ-10 регионов входят также Южная Америка и Африка к югу от Сахары, в то время как Северная Америка и Восточная Азия находятся в хвосте списка.

2.3. Анализ взаимосвязей признаков

2.3.1. Scatter plots (scatter\_advanced.png)

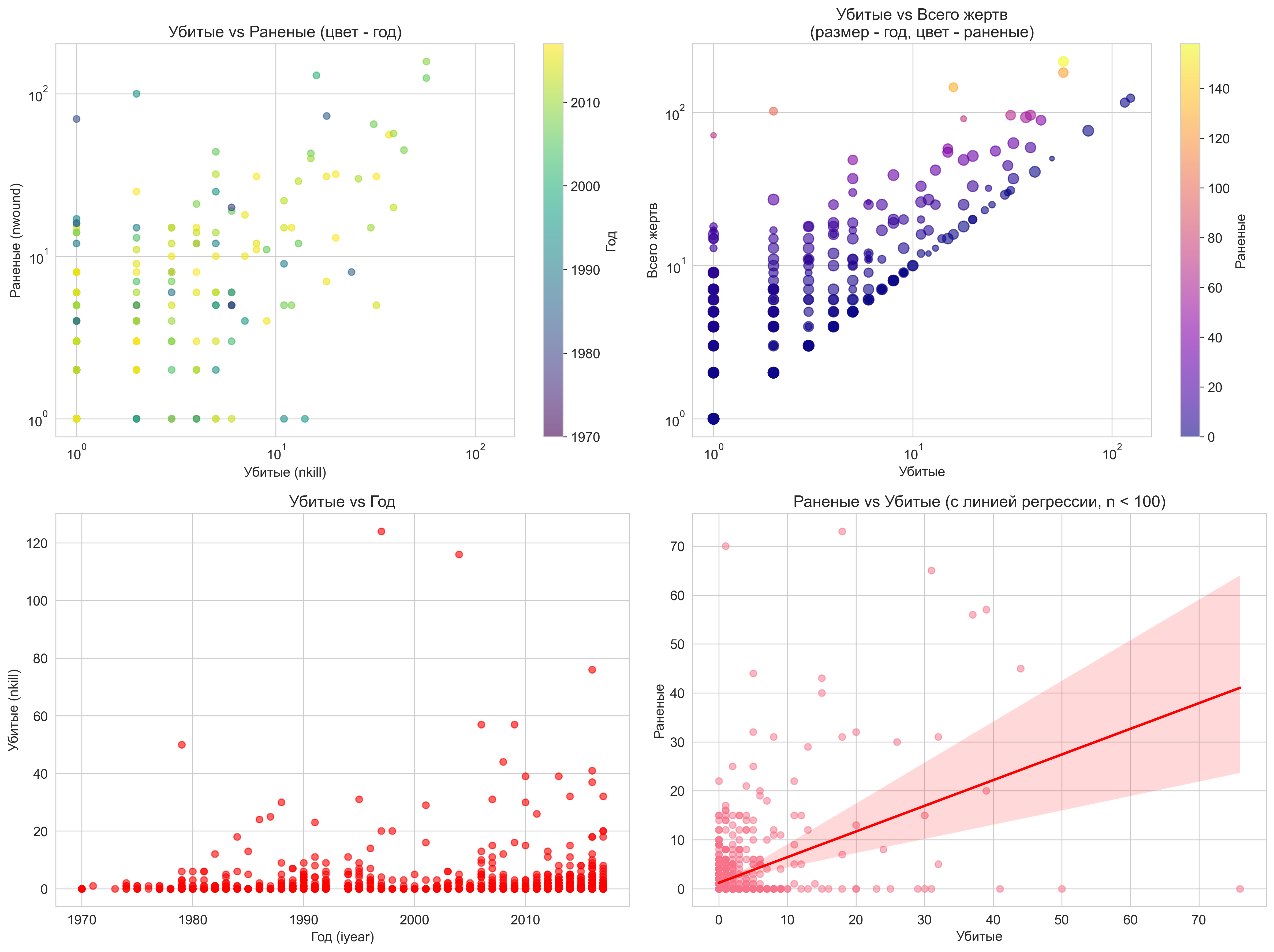


Рисунок 6 – взаимосвязи признаков

Ключевые зависимости:

Убитые vs Раненые (цвет - год): Наблюдается положительная корреляция. Чем больше убитых, тем, как правило, больше и раненых. Цветовая шкала (желтый цвет соответствует более поздним годам) показывает, что наиболее смертоносные атаки часто относятся к современному периоду (2010+ годы).

Убитые vs Год: График показывает "веерное" расширение. Если в 1970-х (фиолетовые точки) количество жертв в одном теракте редко превышало определенный порог, то в 2000-х и 2010-х (красные точки) появляются инциденты с сотнями жертв.

Линейная регрессия (нижний правый угол): Красная линия тренда подтверждает прямую зависимость между числом убитых и раненых, однако широкий разброс точек говорит о том, что тип оружия и тактика сильно влияют на соотношение (например, взрывы дают много раненых, расстрелы — больше убитых).

2.3.2. Многомерный анализ (pairplot\_analysis.png)



Рисунок 7 – многомерный анализ

Pairplot демонстрирует, как различные регионы (цветовая кодировка) распределяются по осям координат. Видно, что точки, соответствующие региону "Middle East & North Africa" (розовые), чаще других занимают области высоких значений по осям nkill и nwound.

2.4. Корреляционный анализ (correlation\_heatmap\_detailed.png)

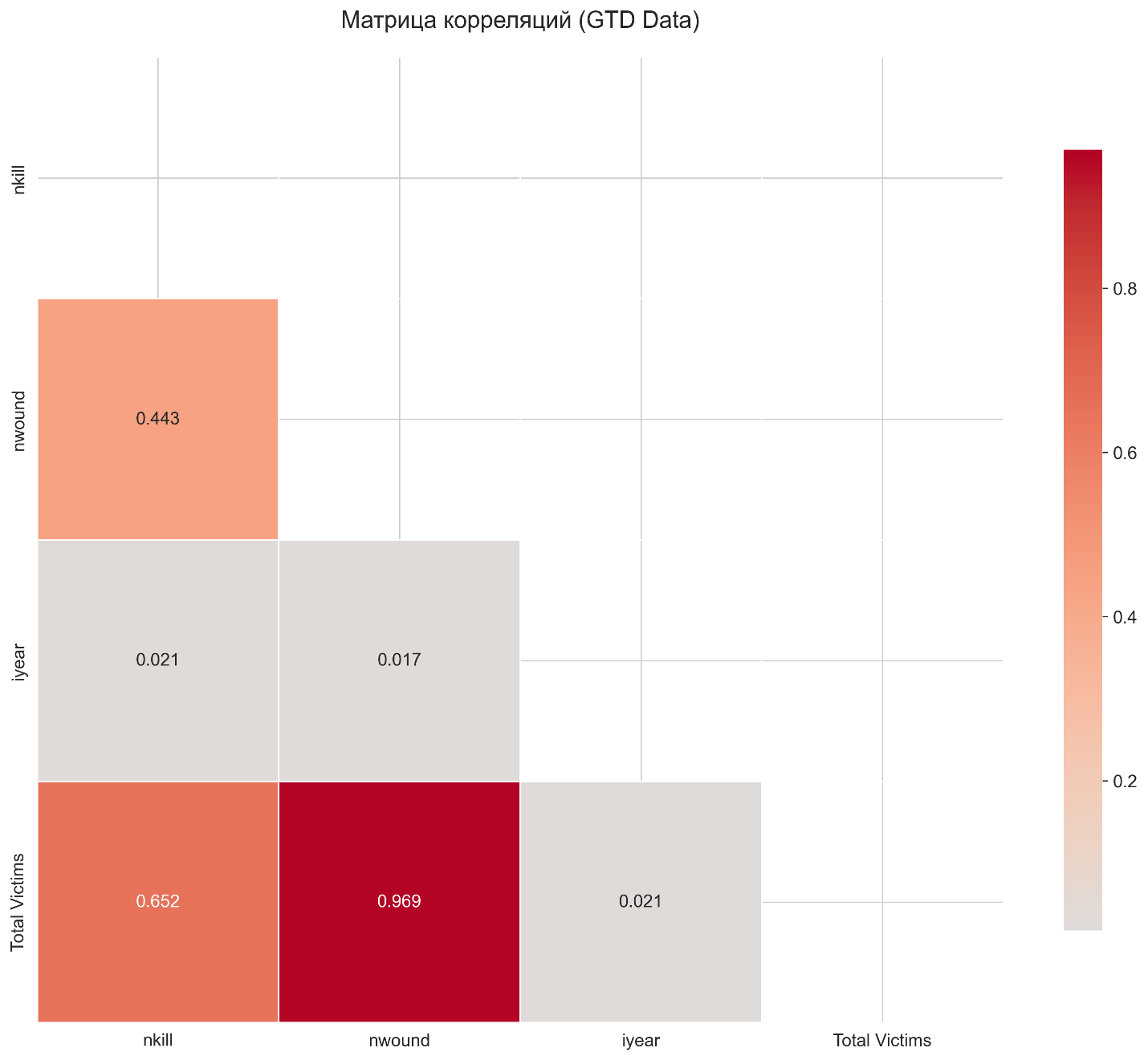


Рисунок 8 – корреляционный анализ

Сильнейшие корреляции (> 0.4):

nkill — Total Victims (0.652): Очевидная зависимость, так как убитые являются частью общего числа жертв.

nwound — Total Victims (0.969): Очень сильная связь. Это говорит о том, что в масштабных терактах число раненых обычно значительно превышает число убитых, поэтому итоговая цифра жертв коррелирует сильнее с ранеными.

nkill — nwound (0.443): Связь средняя. Это подтверждает гипотезу, что не всегда большое число убитых означает огромное число раненых (зависит от метода атаки).

iyear практически не коррелирует с числом жертв (0.02), что значит: сам по себе год не предсказывает число жертв в конкретном отдельном теракте, хотя общее число терактов по годам меняется.

2.5. Специализированные методы визуализации

2.5.1. Violin plot и swarm plot (special\_plots.png)

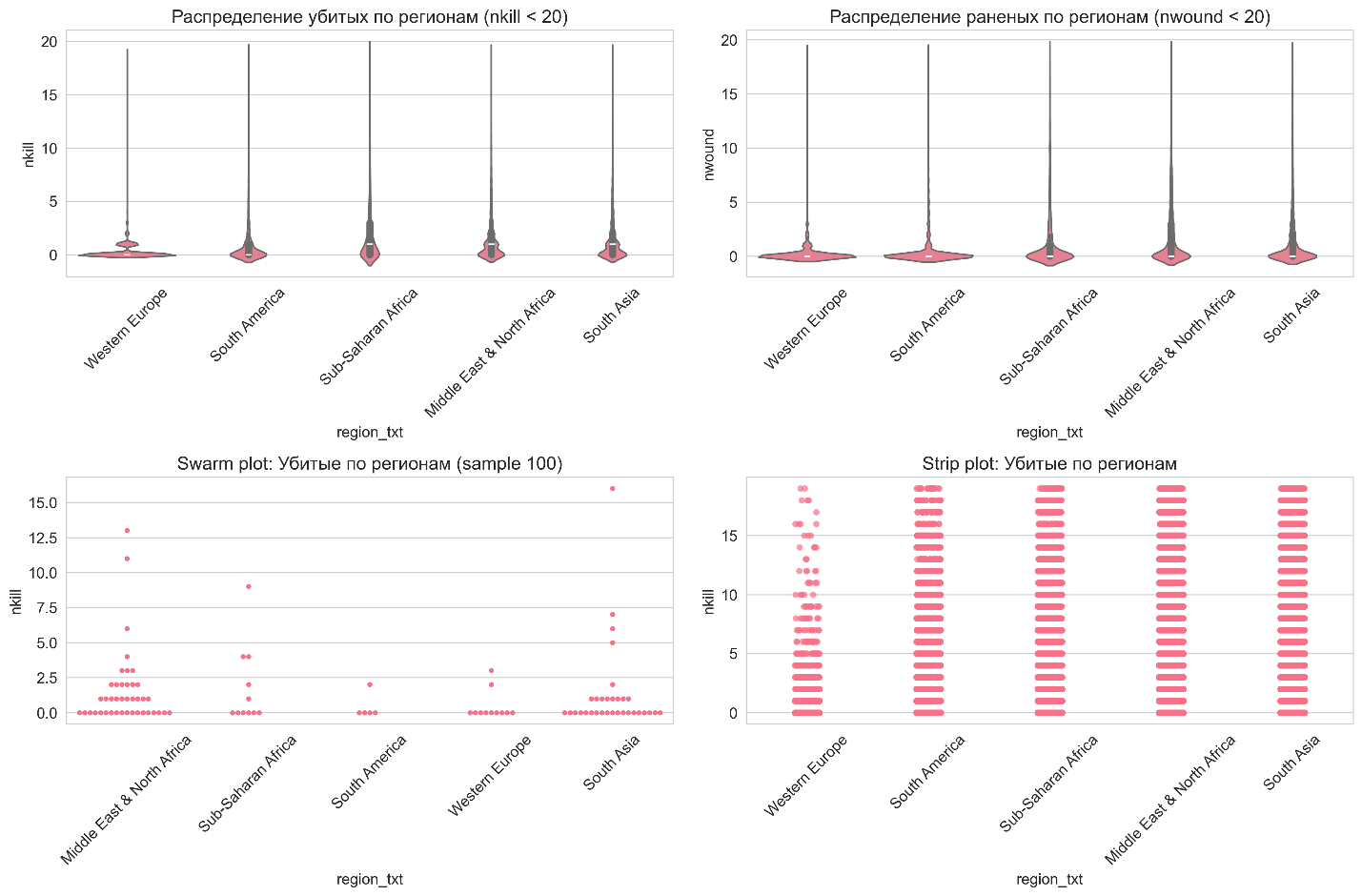


Рисунок 9 – специализированные графики распределений

Violin plot: Показывает плотность распределения. Видно, что у всех регионов очень широкое основание (много атак с малым числом жертв) и очень тонкие длинные пики.

Strip plot: Хорошо визуализирует плотность событий. Видно, что в Южной Азии и на Ближнем Востоке "ковровая" плотность атак значительно выше, чем в Западной Европе.

2.5.2. FacetGrid анализ (facetgrid\_analysis.png)



Рисунок 10 – анализ цены от мощности по типам

Позволяет увидеть различия в паттернах атак:

В South America облако точек более прижато к осям (меньше масштабных жертв).

В Middle East & North Africa и Sub-Saharan Africa облако точек более разреженное и уходит в высокие значения, что свидетельствует о более жестоких конфликтах.

2.5.3. Jointplot анализ (jointplot\_hp\_price.png, jointplot\_hp\_acceleration.png)



Рисунок 11 - зависимость раненых от убитых

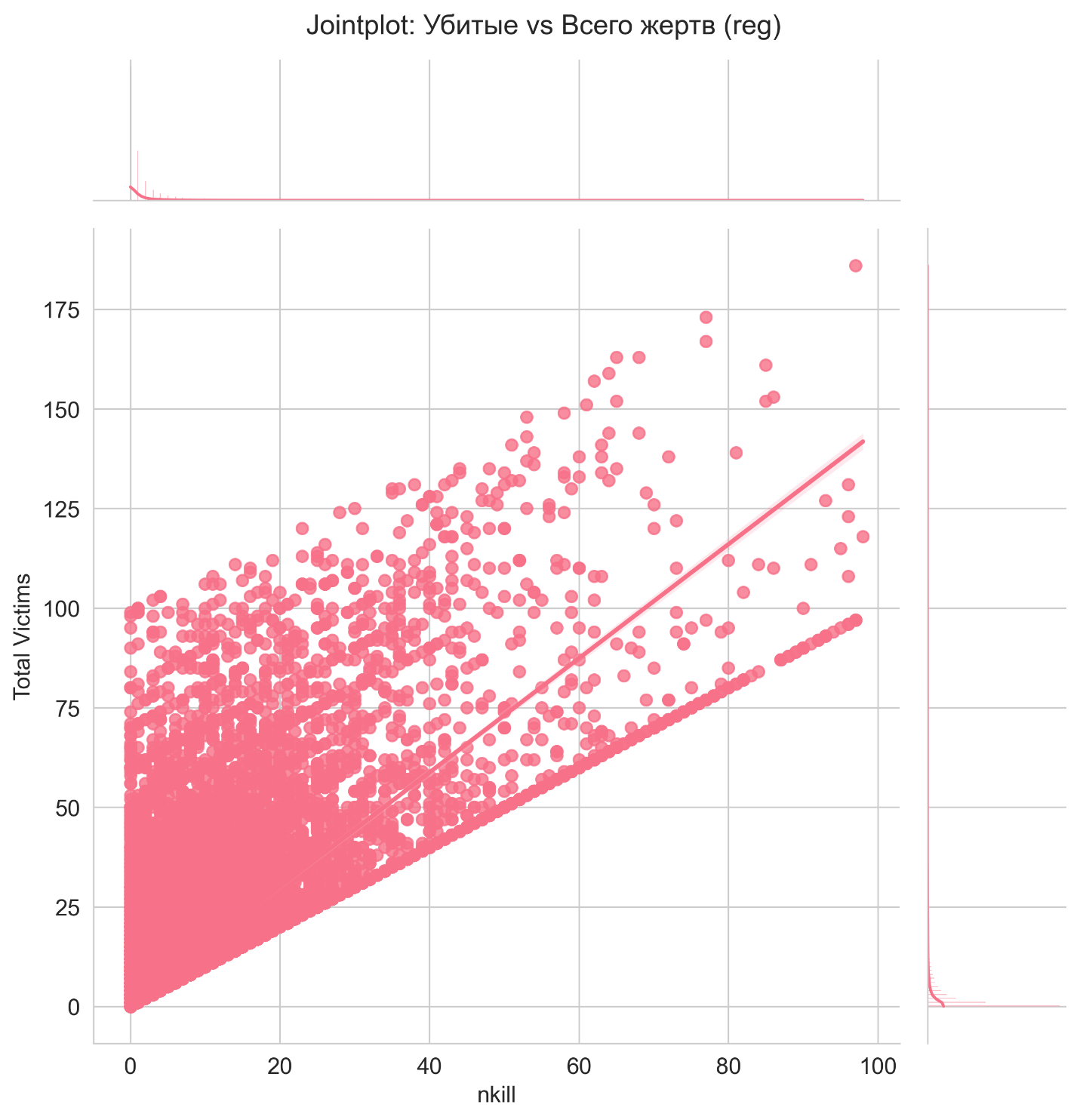


Рисунок 12 – зависимость общего числа жертв от убитых

Эти графики объединяют диаграмму рассеяния и гистограммы, позволяя одновременно оценить и корреляцию двух величин, и их индивидуальные распределения. Четко видна концентрация данных в левом нижнем углу (малые жертвы).

4. Экранные формы и листинг программного кода

4.1. Результаты выполнения программы

RangeIndex: 181691 entries, 0 to 181690

Data columns (total 5 columns):

# Column Non-Null Count Dtype

--- ------ -------------- -----

0 iyear 181691 non-null int64

1 region\_txt 181691 non-null object

2 nkill 181691 non-null int64

3 nwound 181691 non-null float64

4 Total Victims 181691 non-null float64

dtypes: float64(2), int64(2), object(1)

memory usage: 6.9+ MB

Размерность данных: (181691, 5)

Сильнейшие корреляции (> 0.7):

nwound Total Victims 0.968724

Total Victims nwound 0.968724

dtype: float64

ВЫВОДЫ

В ходе лабораторной работы №2 проведен анализ данных глобальной базы терроризма (GTD).

Анализ распределений: Выявлено, что распределение жертв (убитых и раненых) подчиняется степенному закону (распределение Парето) — подавляющее большинство терактов малолетальны, но основной урон наносят редкие, но масштабные события. Данные содержат огромное количество выбросов.

Категориальный анализ: Установлены ключевые "горячие точки" планеты — Ближний Восток, Северная Африка и Южная Азия. Определен временной пик активности терроризма, приходящийся на 2014 год.

Корреляционный анализ: Подтверждена логическая связь между количеством убитых и раненых, однако коэффициент корреляции (0.443) указывает на то, что соотношение убитых/раненых сильно варьируется в зависимости от типа атаки.

Визуализация: Использование библиотек Matplotlib и Seaborn (включая сложные графики типа Violin plot, Jointplot и Pairplot) позволило наглядно представить структуру данных, выявить региональные паттерны и временные тренды, которые невозможно заметить при простом табличном просмотре данных.

3. Ответы на контрольные вопросы

1. Какие инструментальные средства используются для организации рабочего места специалиста Data Science?

Ответ: Стандартный набор инструментов включает язык программирования Python, среды разработки кода (VS Code, Jupyter Notebook), систему контроля версий Git. Также обязательны библиотеки для вычислений и аналитики (NumPy, Pandas), построения графиков (Matplotlib, Seaborn) и средства машинного обучения (Scikit-learn).

2. Какие библиотеки Python используются для работы в области машинного обучения? Дайте краткую характеристику каждой библиотеке.

Ответ:

* Pandas: основной инструмент для манипуляций с табличными данными (чтение, фильтрация, предобработка).
* NumPy: база для научных вычислений, обеспечивает эффективную работу с матрицами и многомерными массивами.
* Scikit-learn: содержит реализации классических алгоритмов ML и инструменты для подготовки данных.
* Matplotlib: фундаментальная библиотека для создания двумерной графики.
* Seaborn: высокоуровневая надстройка над Matplotlib для более эстетичной и простой статистической визуализации.
* TensorFlow / PyTorch: мощные фреймворки для создания и обучения глубоких нейронных сетей.

3. Почему при реализации систем машинного обучения широкое распространение получили библиотеки Python?

Ответ: Ключевые факторы успеха — низкий порог входа благодаря простому синтаксису, огромная база готовых решений (библиотек) для любых задач, активная поддержка мировым сообществом разработчиков и кроссплатформенность языка.

4. Перечислите функции Python, которые были изучены в рамках данной лабораторной работы и которые используются для визуализации данных.

Ответ:

Matplotlib: функции plt.hist (гистограммы), plt.boxplot (ящики с усами), plt.scatter (диаграммы рассеяния), а также инструменты настройки plt.subplot и plt.colorbar.

Seaborn: методы sns.boxplot, sns.countplot (столбчатые диаграммы), sns.scatterplot, sns.pairplot (парные графики), sns.heatmap (тепловые карты), а также сложные графики: sns.violinplot, sns.swarmplot, sns.stripplot, sns.regplot, sns.jointplot и сетка sns.FacetGrid.

Pandas: аналитические методы df.value\_counts() и df.corr().

5. Какая библиотека python предназначена для управления наборами данных: numpy, pandas, sklearn, opencv, matplotlib?

Ответ: Pandas. Именно эта библиотека специально разработана для удобного управления структурированными данными через объекты DataFrame.

6. Какая стратегия является нежелательной при обработке пропусков в данных?

Ответ: Простое удаление строк с отсутствующими значениями. Этот подход нежелателен, так как влечет за собой потерю полезной информации и может исказить статистическое распределение выборки.

7. Обоснуйте ответ на следующую проблему предварительной обработки данных: имеется независимая категориальная переменная y, которая представляет собой категориальный признак, определенный на домене {C#, Java, Python, R}. Нужно ли применять к данному целевому признаку OneHotEncoder?

Ответ: Нет, использование OneHotEncoder в данном случае некорректно. Этот кодировщик применяется для входных признаков (features), чтобы модель могла их интерпретировать. Для целевой переменной в задачах классификации следует использовать LabelEncoder (преобразование категорий в целые числа) либо оставлять метки как есть, если библиотека поддерживает работу со строками.

8. Поясните принцип разбиения набора данных на обучающую и тестовую выборку. Какое соотношение наиболее оптимально?

Ответ: Разделение необходимо для проверки того, как модель справляется с незнакомыми данными (проверка на переобучение). Стандартной практикой считается соотношение 80/20 или 70/30 (обучение/тест). Это позволяет предоставить модели достаточно данных для выявления закономерностей, оставив при этом репрезентативную часть для валидации.

9. Какой код лучше использовать при загрузке данных из csv-файла?

Ответ: Наиболее эффективным и лаконичным является использование библиотеки Pandas:

import pandas as pd

dataset = pd.read\_csv("data.csv")