**МИНОБРНАУКИ РОССИИ**

**Санкт-Петербургский государственный**

**электротехнический университет**

**«ЛЭТИ» им. В.И. Ульянова (Ленина)**

**Кафедра Вычислительной техники**

**отчет**

**по лабораторной работе №1**

**по дисциплине «Основы машинного обучения»**

**Тема: Классификация.**

|  |  |  |
| --- | --- | --- |
| Студентка гр. 3316 |  | Борисов А.М. |
| Преподаватель |  | Турсуков Н. О. |

Санкт-Петербург

2025

Содержание:

[Цель работы: **3**](#_Toc211609293)

[Задачи работы: **3**](#_Toc211609294)

[Краткая теоретическая информация по теме работы: **4**](#_Toc211609295)

[Процесс и результаты работы. **11**](#_Toc211609296)

[Подготовка данных. 11](#_Toc211609297)

[Обучение моделей. 13](#_Toc211609298)

[KNN 13](#_Toc211609299)

[SVM 15](#_Toc211609300)

[Дерево решений. 17](#_Toc211609301)

[Оценка моделей. 20](#_Toc211609302)

[Вывод. 23](#_Toc211609303)

**Цель работы:** закрепить на практике знания о способах предобработки данных и применить алгоритмы машинного обучения для решения задач классификации.

# **Задачи работы:**

Предварительно обработать данные согласно следующим пунктам:

1. Визуализация значимых признаков (диаграммы рассеяния, ящики с усами, гистограммы).
2. Очистка данных (удаление пропусков, нормализация, удаление дубликатов).
3. Корреляция данных (матрица корреляций).

В ходе выполнения лабораторной работы должны быть выполнены следующие этапы:

1. Обучение моделей и подбор параметров:

* K-ближайших соседей (KNN)
* Машина опорных векторов (SVM)
* Дерево решений ИЛИ Случайный лес

1. Оценка моделей:

* Визуализация предсказанных значений
* Оценка качества прогноза (precision/recall/f1-score/ROC-AUC)
* Визуализация дерева решений ИЛИ Визуализация Feature Importance для случайного леса

# **Краткая теоретическая информация по теме работы:**

**Классификация** – это тип задачи обучения с учителем, в которой цель состоит в том, чтобы построить модель, способную предсказывать категориальную метку (класс) для новых, ранее не виденных наблюдений на основе обучающих данных, где каждому объекту уже сопоставлена правильная метка.

Основные характеристики:

* Вход: вектор признаков
* Выход: дискретная метка , где – число классов
* Цель: минимизировать ошибку классификации на новых данных (обобщение)

Типы классификации:

* Бинарная: два класса (например, «да/нет», «заболевание/здоров»)
* Многоклассовая: > двух классов (например, распознавание цифр 0-9)
* Мультилейбл: один объект может принадлежать сразу нескольким классам (редко в базовых задачах)

Классификация лежит в основе множества приложений: диагностика заболеваний, фильтрация спама, распознавание лиц, кредитный скоринг и др.

В рамках лабораторной работы выполняется полный цикл подготовки данных и построения моделей машинного обучения. На этапе предварительной обработки проводится визуализация значимых признаков с помощью гистограмм, диаграмм рассеяния и ящиков с усами.

Визуальный анализ помогает понять распределение данных, выявить паттерны, выбросы и потенциальные зависимости между признаками и целевой переменной.

**Гистограммы:**

* Показывают распределение значений одного числового признака.
* Позволяют оценить: симметричность, наличие мод, хвостов, выбросов.
* Полезны для проверки нормальности распределения (важно для некоторых статистических методов).

**Диаграммы рассеяния (scatter plots):**

* Отображают взаимосвязь между двумя числовыми признаками.
* Если точки образуют облако с наклоном — возможна корреляция.
* При наличии метки класса — точки можно раскрасить по классам, чтобы увидеть, разделимы ли классы в этом пространстве признаков.

**Ящики с усами (boxplots):**

* Визуализируют пять ключевых статистик: минимум, первый квартиль (Q1), медиана (Q2), третий квартиль (Q3), максимум.
* Усы обычно ограничены 1.5 × IQR (межквартильный размах = Q3 – Q1).
* Точки за пределами усов считаются потенциальными выбросами.
* Особенно полезны при сравнении распределений признака по классам (например, boxplot дохода для «дефолт/не дефолт»).

**Корреляция** – мера линейной зависимости между двумя количественными переменными.

Коэффициент корреляции Пирсона:

где:

– ковариация (статистическая мера, которая показывает, насколько две случайные величины изменяются вместе. Другими словами, она характеризует направление линейной зависимости между двумя переменными).

– стандартные отклонения.

Свойства:

: идеальная положительная линейная связь

: идеальная отрицательная линейная связь

: отсутствие линейной связи (но не обязательно независимость!)

**Матрица корреляций.**

Квадратная симметричная матрица, где элемент – корреляция между признаком и признаком. Диагональ всегда = 1 (признак полностью коррелирует сам с собой). Часто визуализируется в виде тепловой карты (heatmap): тёплые цвета – высокая положительная корреляция, холодные – отрицательная.

**K-ближайших соседей (KNN).**

Принцип работы: не строит явную модель; запоминает обучающие данные (ленивое обучение). Для нового объекта:

* 1. Находит K ближайших соседей в пространстве признаков (по евклидову, манхэттенскому или другому расстоянию).
  2. Присваивает класс, который доминирует среди этих K соседей (голосование).

Особенности: чувствителен к масштабу признаков: признак с большим диапазоном «затмит» остальные → обязательна нормализация/стандартизация. Вычислительно затратен на этапе предсказания при больших данных (требует вычисления расстояний до всех точек).

Выбор K:

* + Малое K (например, 1): модель очень гибкая, но склонна к переобучению и шуму.
  + Большое K: сглаживает границы, но может недообучиться, особенно при несбалансированных классах.

Гиперпараметры:

* K – число соседей/
* Метрика расстояния (euclidean, manhattan, cosine и др.).
* Взвешенное голосование (по расстоянию).

**Машина опорных векторов (SVM).**

Принцип работы: ищет гиперплоскость, которая максимально разделяет классы в признаковом пространстве. Опорные векторы – это ближайшие к гиперплоскости точки обучающей выборки; именно они определяют положение границы.

Линейно разделимый случай: идеальная гиперплоскость максимизирует зазор (margin) между классами.

Нелинейно разделимый случай: вводится параметр C (штраф за ошибки):

* + Большой C → модель строго наказывает ошибки → узкий зазор, риск переобучения.
  + Малый C → допускает больше ошибок → широкий зазор, устойчивость к шуму.

Ядра (kernels): позволяют работать в расширенном (часто бесконечномерном) пространстве, где данные становятся линейно разделимыми. *Линейное ядро*: для линейно разделимых данных. *RBF (радиальная базисная функция)*: гибкое, часто используется по умолчанию. Параметр γ: чем больше – тем сложнее граница (риск переобучения).

Особенности:

* Требует нормализации признаков (иначе признаки с большим масштабом доминируют).
* Эффективна при высокой размерности, но плохо масштабируется на очень большие выборки (>100k объектов).

**Дерево решений и Случайный лес.**

**Дерево решений:**

* Рекурсивно разбивает обучающее множество на подмножества, выбирая на каждом шаге наилучший признак и порог для разбиения.
* Критерии качества разбиения:
  + Джини-неоднородность:
  + Энтропия:
* Процесс продолжается, пока не будет достигнуто условие останова (макс. глубина, мин. число объектов в листе и др.).

Преимущества:

* Интерпретируемость (можно визуализировать)
* Не требует нормализации
* Работает с числовыми и категориальными признаками

Недостатки:

* Склонно к переобучению, особенно при большой глубине
* Нестабильно: небольшие изменения в данных → сильно другое дерево

**Случайный лес (Random Forest):**

* Это ансамблевый метод, основанный на бэггинге (bagging) и рандомизации признаков.
* Обучает множество деревьев:
  + Каждое дерево обучается на случайной подвыборке с возвращением (bootstrap sample)
  + На каждом узле разбиение выбирается только из случайного подмножества признаков (обычно для классификации, где — число признаков)

Преимущества:

* Снижает дисперсию → меньше переобучение, чем у одного дерева
* Очень высокая точность «из коробки»
* Устойчив к выбросам и шуму
* Даёт оценку важности признаков

Гиперпараметры:

* Число деревьев (n\_estimators)
* Максимальная глубина (max\_depth)
* Минимальное число объектов в листе (min\_samples\_leaf)
* Количество признаков для разбиения (max\_features)

**Оценка качества прогноза.**

Precision (Точность). Доля корректно предсказанных положительных среди всех, кого модель назвала положительными:

→ Важна, когда ложные срабатывания (FP) дорого стоят.

Recall (Полнота, Sensitivity). Доля найденных положительных среди всех реальных положительных:

→ Важна, когда пропуск положительного случая критичен (например, диагностика рака).

F1-score. Гармоническое среднее precision и recall:

→ Баланс между точностью и полнотой. Особенно полезен при несбалансированных классах.

ROC-AUC (Area Under the ROC Curve).

* ROC-кривая строится по разным порогам классификации.
* AUC = 1 — идеальная разделимость классов
* AUC = 0.5 — модель не лучше случайного угадывания
* Инвариантна к порогу и устойчива к дисбалансу классов, но может вводить в заблуждение при сильном дисбалансе → в таких случаях лучше использовать PR-AUC (Precision-Recall AUC).

# Процесс и результаты работы.

## Подготовка данных.

Датасет: **Rain in Australia**

Размер: 145000 записей, 23 признаков (считая data).

Выполнена проверка на дубликаты (не обнаружено).

Удалено строк со значениями NaN: 89 040

Информация о датасете:

<class 'pandas.core.frame.DataFrame'>

Index: 56420 entries, 6049 to 142302

Data columns (total 23 columns):

# Column Non-Null Count Dtype

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

0 Date 56420 non-null object

1 Loc 56420 non-null object

2 MinT 56420 non-null float64

3 MaxT 56420 non-null float64

4 Rain 56420 non-null float64

5 Evap 56420 non-null float64

6 Sun 56420 non-null float64

7 WGDir 56420 non-null object

8 WGSpeed 56420 non-null float64

9 WD9am 56420 non-null object

10 WD3pm 56420 non-null object

11 WS9am 56420 non-null float64

12 WS3pm 56420 non-null float64

13 H9am 56420 non-null float64

14 H3pm 56420 non-null float64

15 P9am 56420 non-null float64

16 P3pm 56420 non-null float64

17 C9am 56420 non-null float64

18 C3pm 56420 non-null float64

19 T9am 56420 non-null float64

20 T3pm 56420 non-null float64

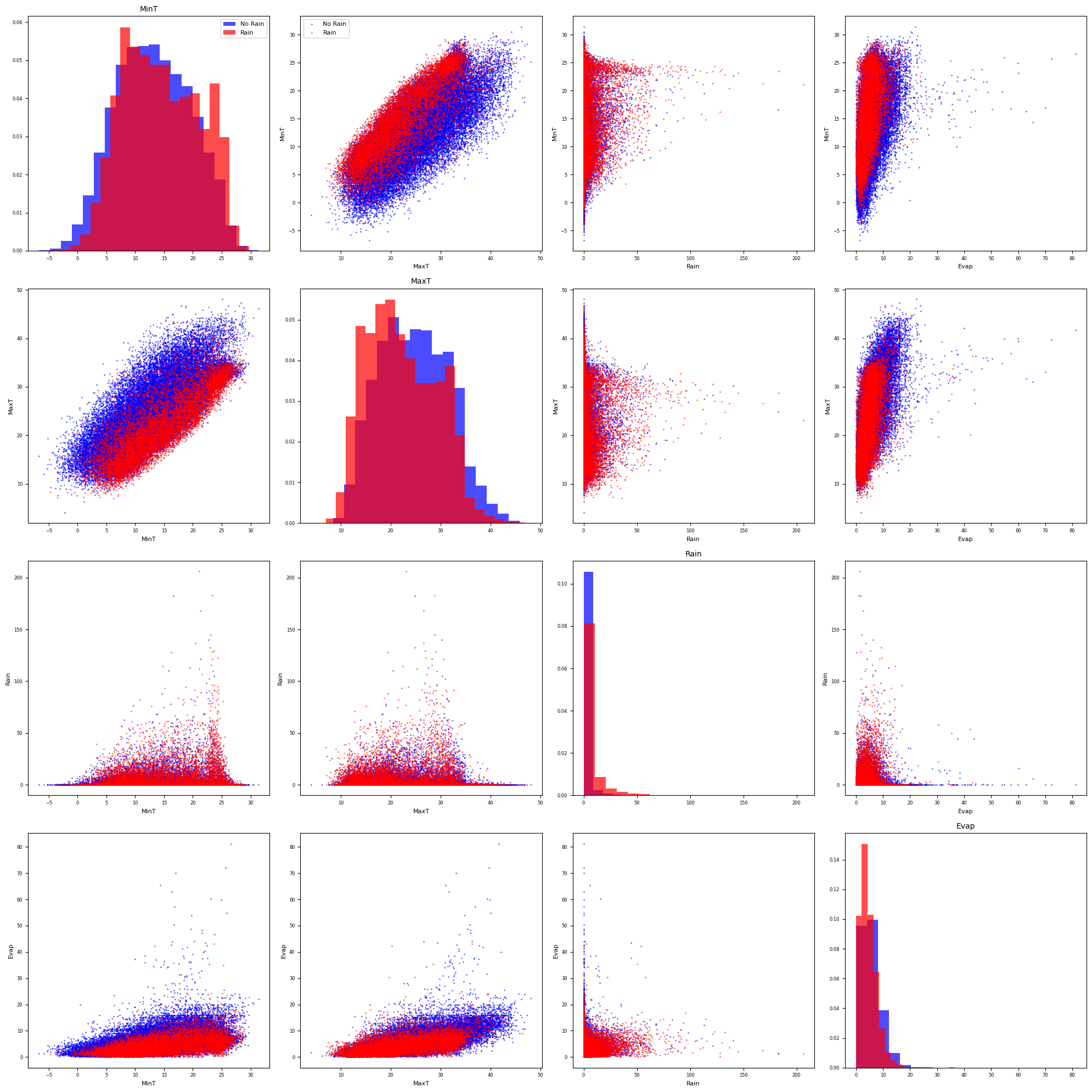
21 RainToday 56420 non-null object

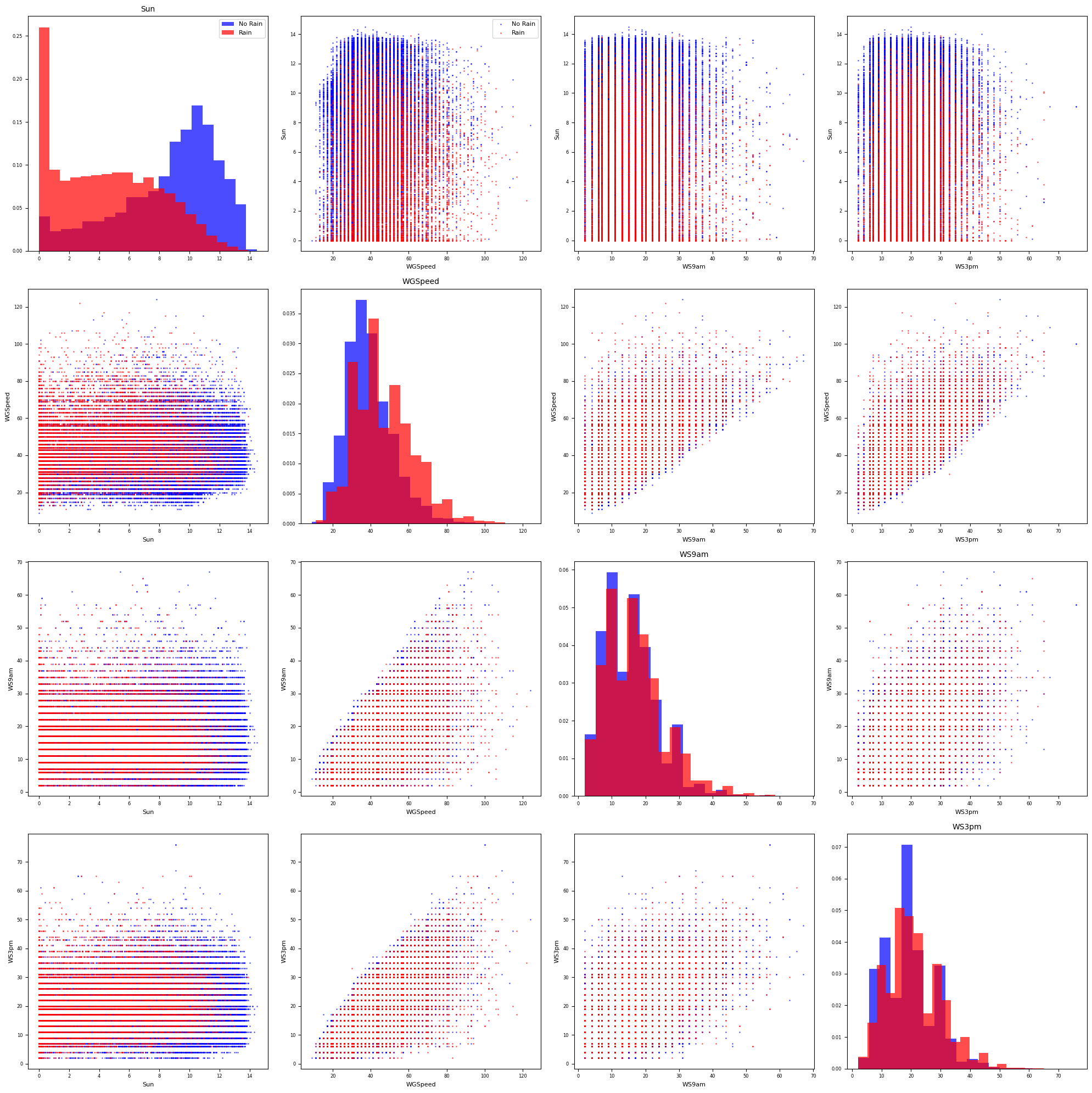
22 RainTomorrow 56420 non-null object

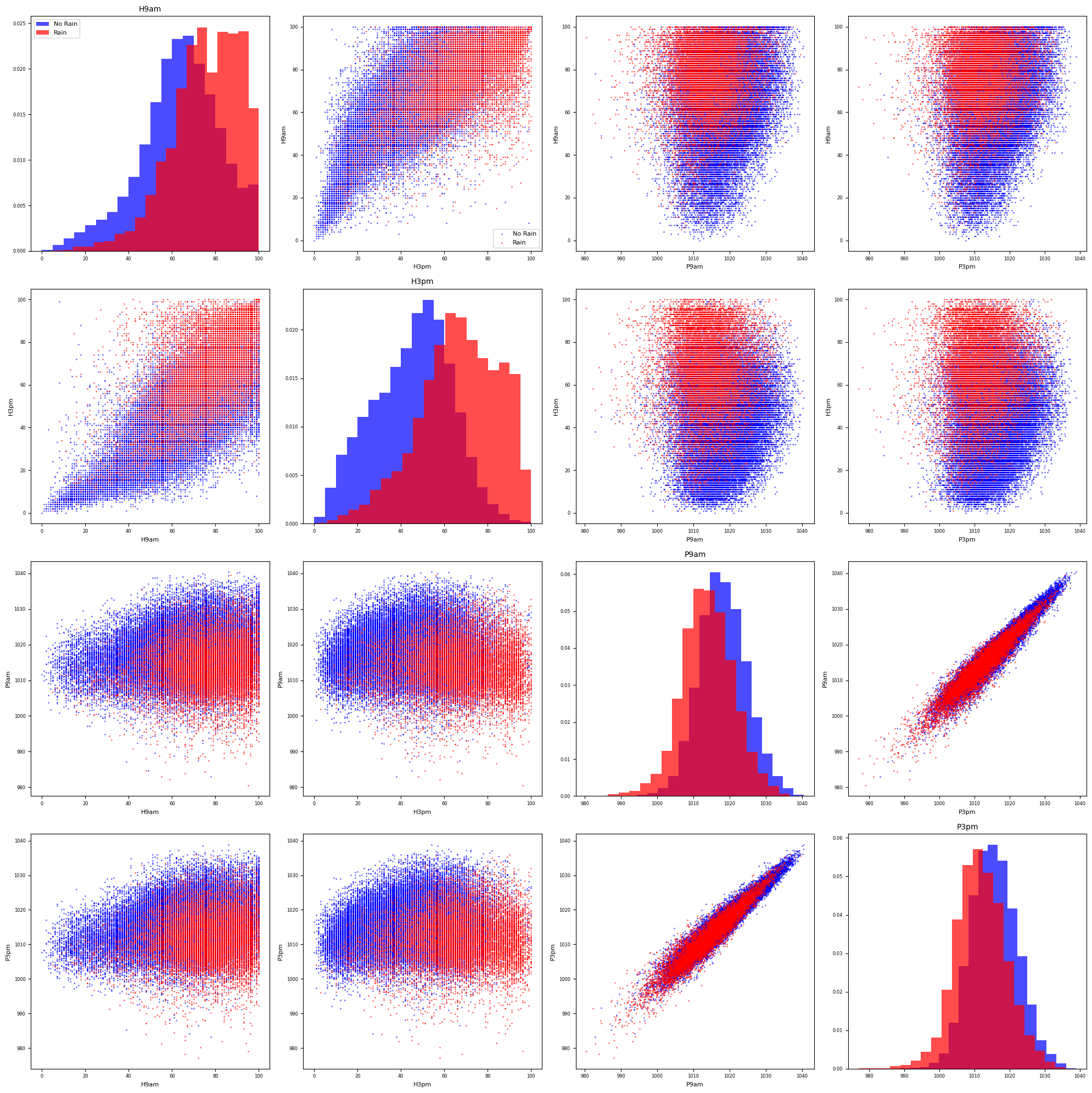
dtypes: float64(16), object(7)

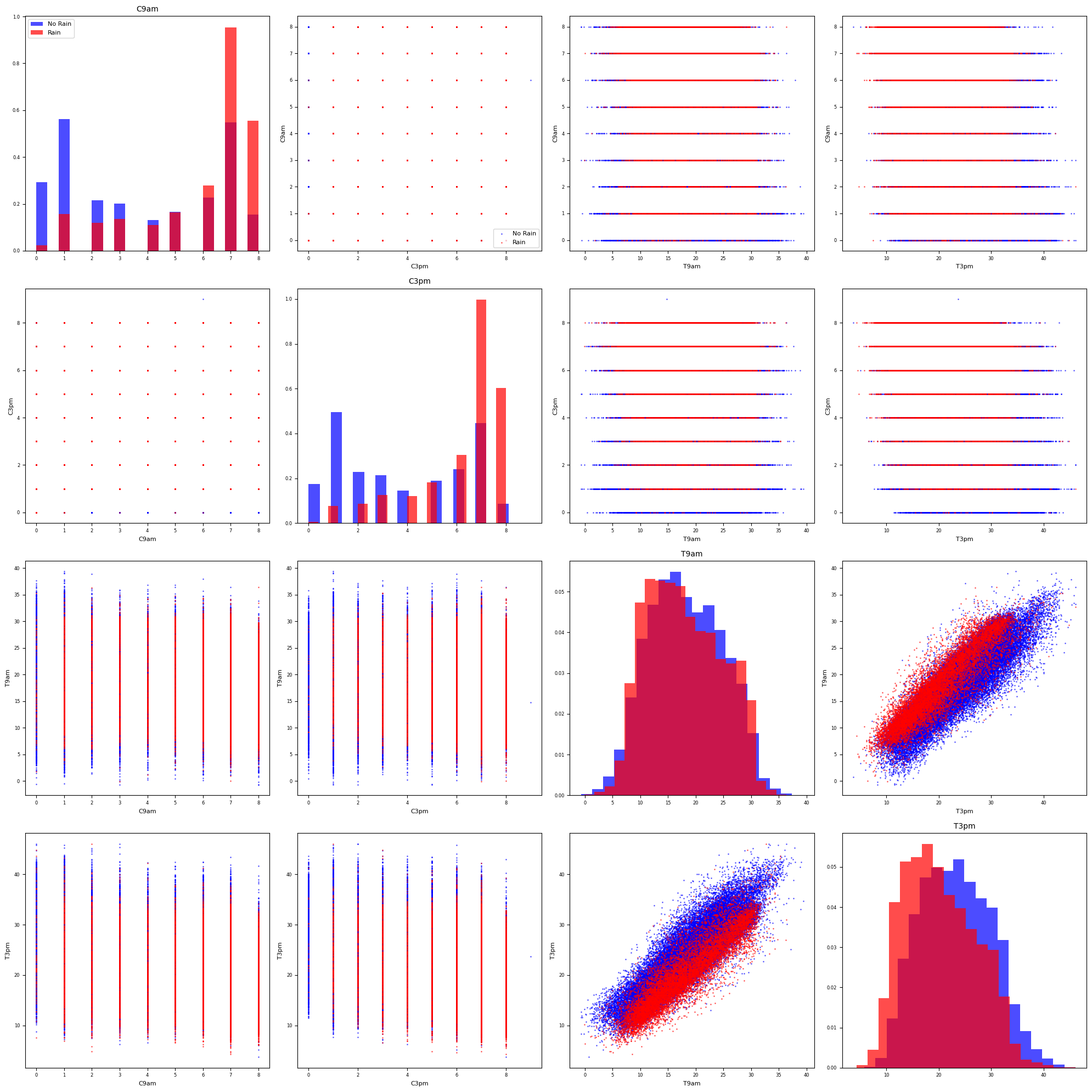
memory usage: 10.3+ MB

Далее будет приведена диаграмма рассеяния числовых признаков. Она показывает взаимосвязь между двумя числовыми переменными в датасете.



**

**

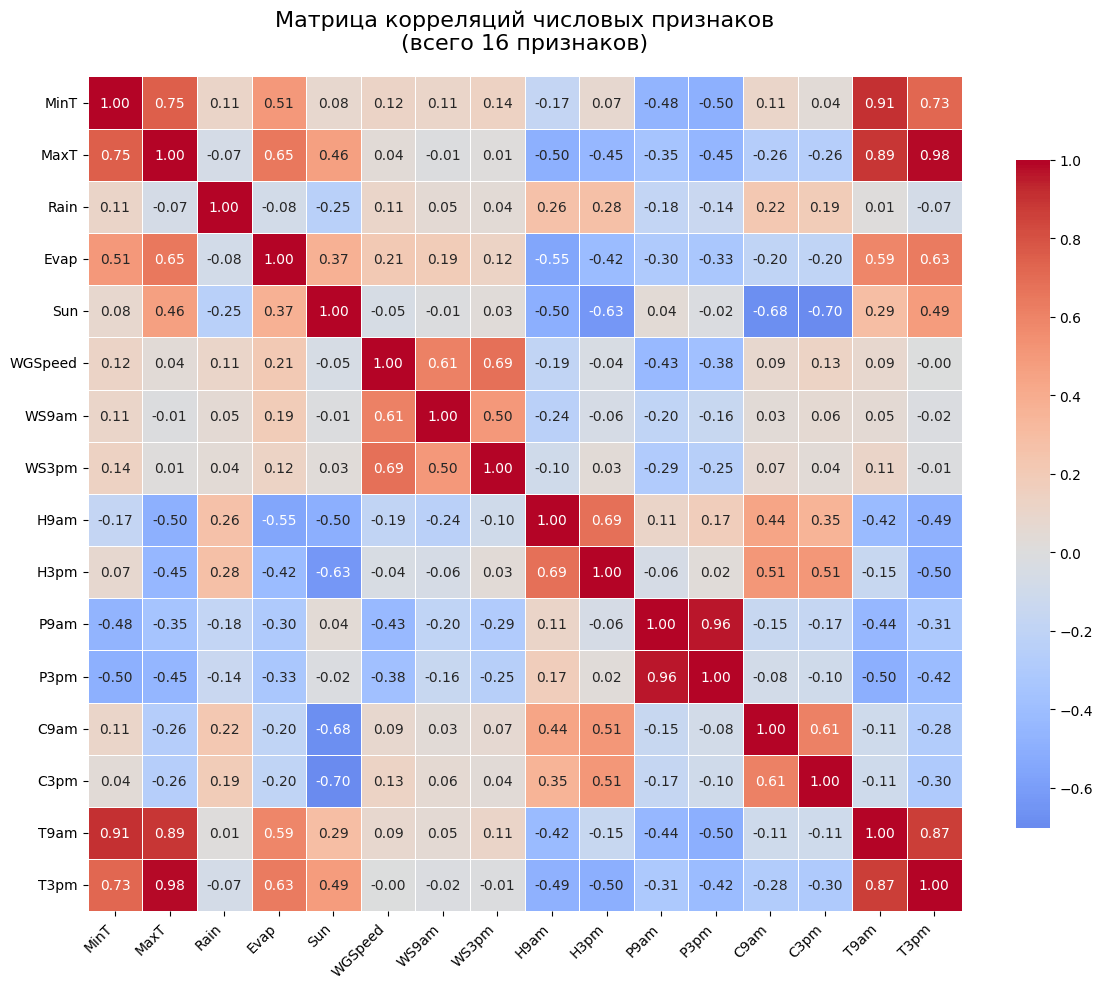
*Рисунок 1-4 – Диаграммы рассеяния.*

Далее будет представлена матрица корреляций признаков. Каждая ячейка матрицы показывает коэффициент корреляции Пирсона между двумя признаками. Коэффициент корреляции — это число от -1 до +1, которое говорит о том, насколько признаки линейно связаны друг с другом.

+1.00 – идеальная положительная линейная связь (если один растёт — другой растёт точно так же)

~0.0–0.2 – очень слабая или нет связи

–1.00 – идеальная отрицательная линейная связь (если один растёт — другой падает).



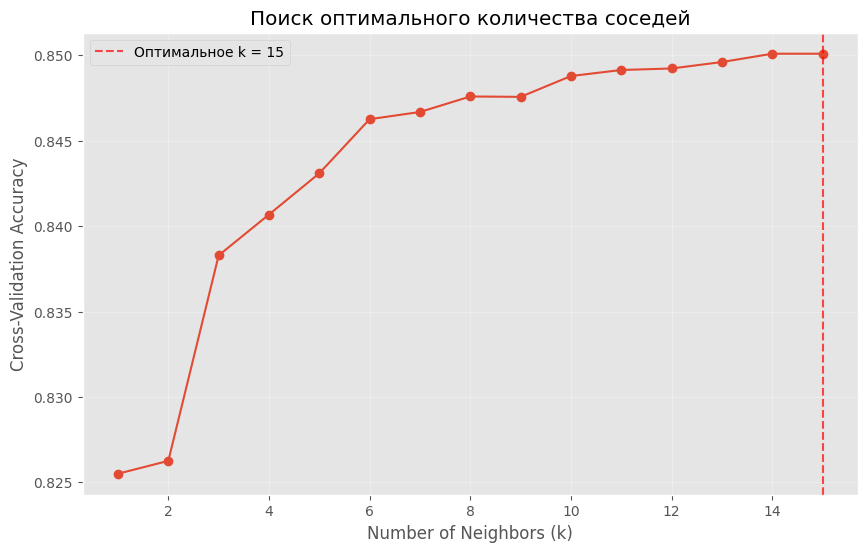
*Рисунок 5 – Матрица корреляций.*

# **Обучение моделей.**

### **KNN**

Оптимальное n\_neighbors: 15

Лучшая точность CV: 0.8501

 Рисунок 6: Точность модели от количества соседей6

=== РЕЗУЛЬТАТЫ KNN ===

Accuracy: 0.8551

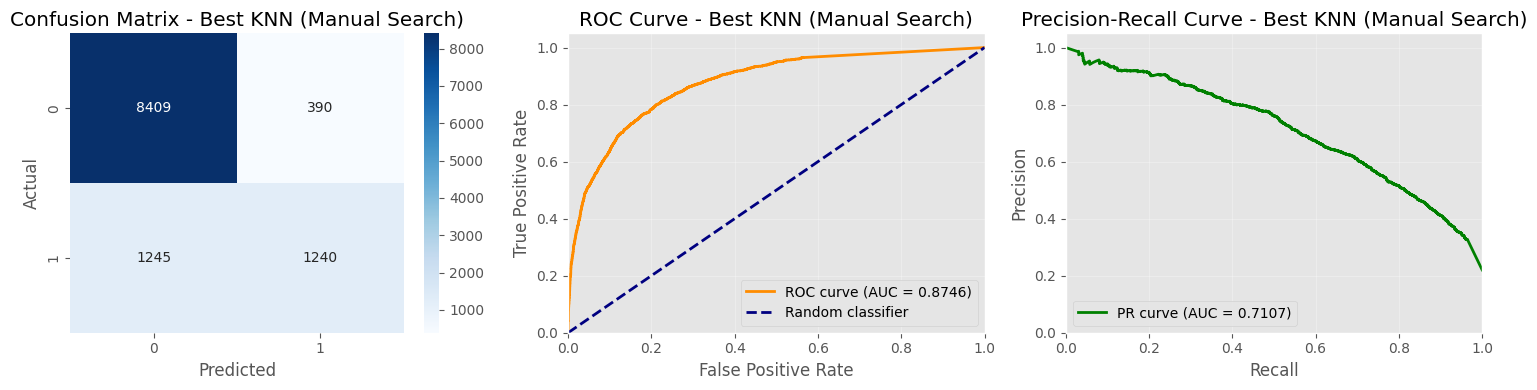
Precision: 0.7607

Recall: 0.4990

F1-Score: 0.6027

ROC-AUC: 0.8746

ROC-AUC: 0.7723

 Рисунок 7: Матрица ошибок KNN и ROC-кривая.

### **SVM**

МЕТРИКИ МОДЕЛИ SVM:

Accuracy: 0.8538

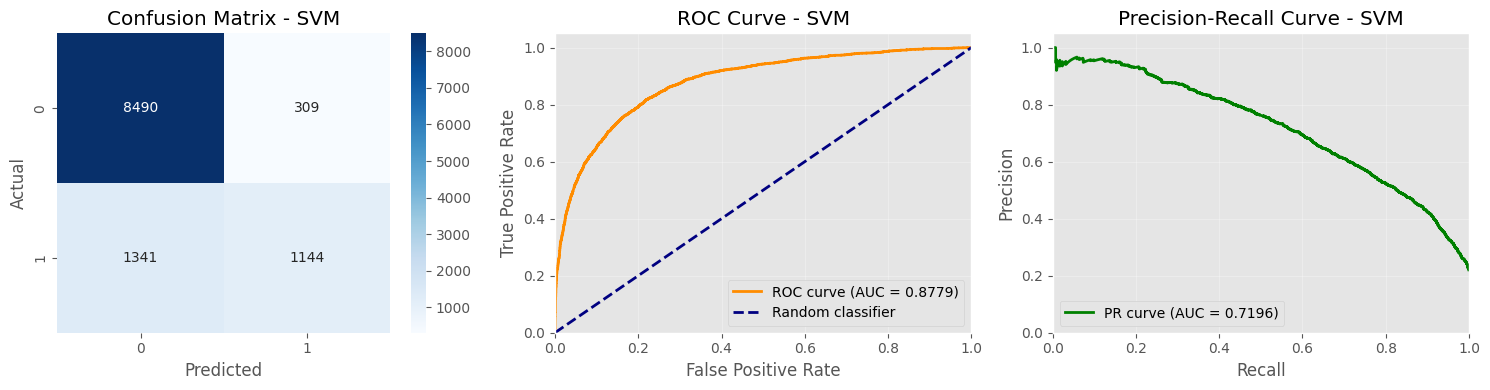
Precision: 0.7873

Recall: 0.4604

F1-Score: 0.5810

ROC-AUC: 0.8779

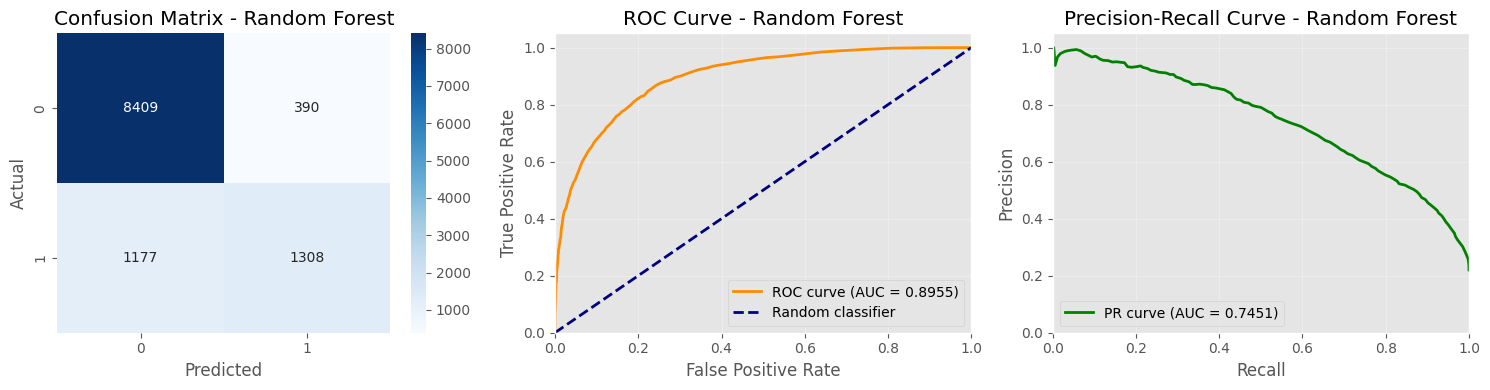
Параметры модели: kernel=rbf, C=1.0, gamma=scale

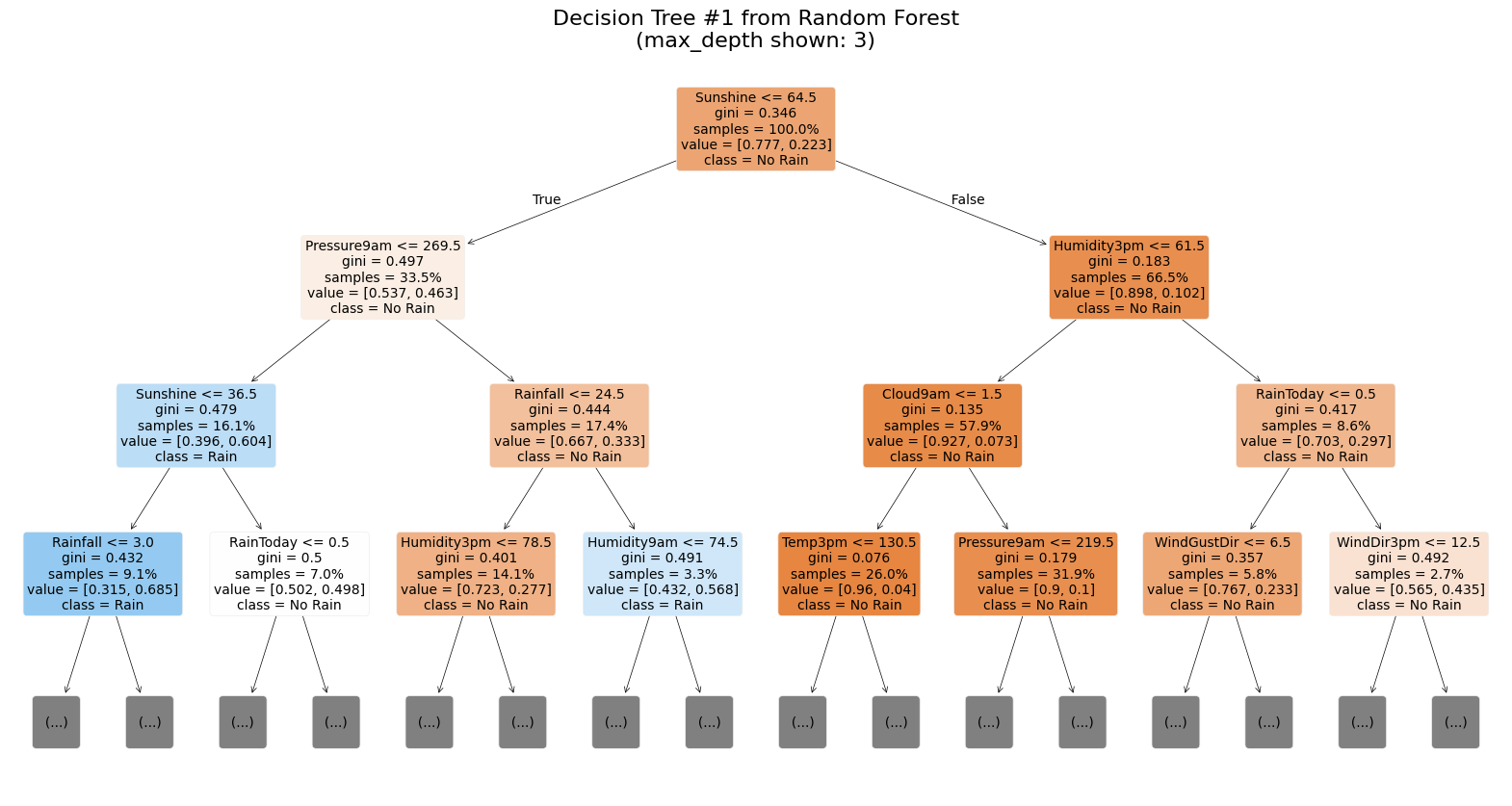
 Рисунок 8: Матрица ошибок и кривые SVM

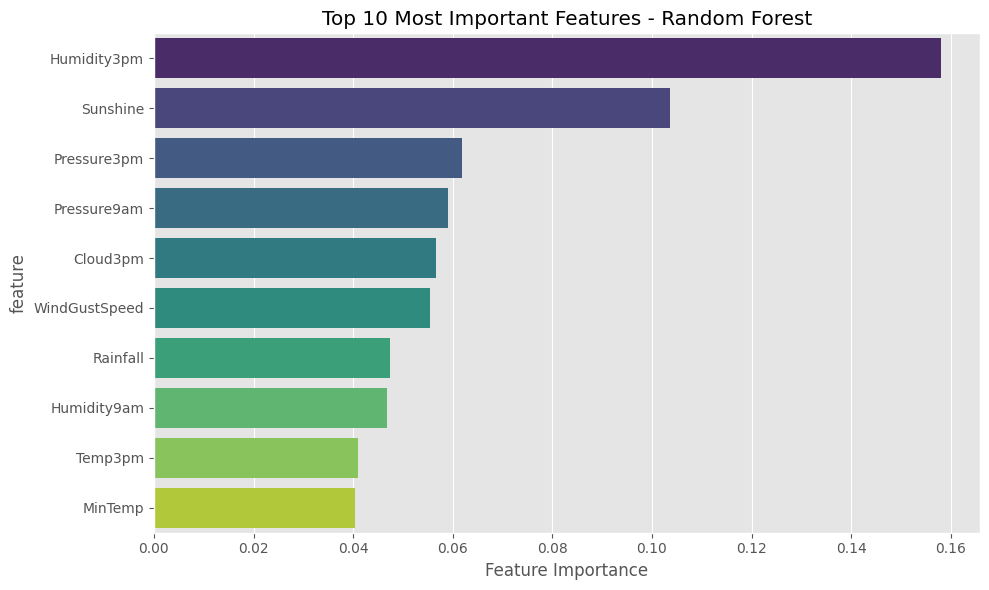
### **Случайный лес****.**

Лучшие параметры: {'criterion': 'entropy', 'max\_depth': 7, 'min\_samples\_leaf': 2, 'min\_samples\_split': 2}

Лучшая точность CV: 0.8467

 Рисунок 9: Матрица и кривые Random forest

 Рисунок 10: Пример одного из деревьев

 Рисунок 11: Топ 10 самых влиятельных параметров

# **Оценка моделей.**

Ниже будут представлены результаты сравнения моделей KNN, SVM, Random Forest.

=== ЛУЧШАЯ МОДЕЛЬ: Random Forest ===

Точность: 0.8611

Лучшие параметры: {'criterion': 'entropy', 'max\_depth': 7, 'min\_samples\_leaf': 2, 'min\_samples\_split': 2}

=== СТАТИСТИКА СРАВНЕНИЯ ===

Разница между лучшей и худшей моделью по точности: 0.0073

Средняя точность всех моделей: 0.8566

СРАВНЕНИЕ МОДЕЛЕЙ ПО МЕТРИКАМ:

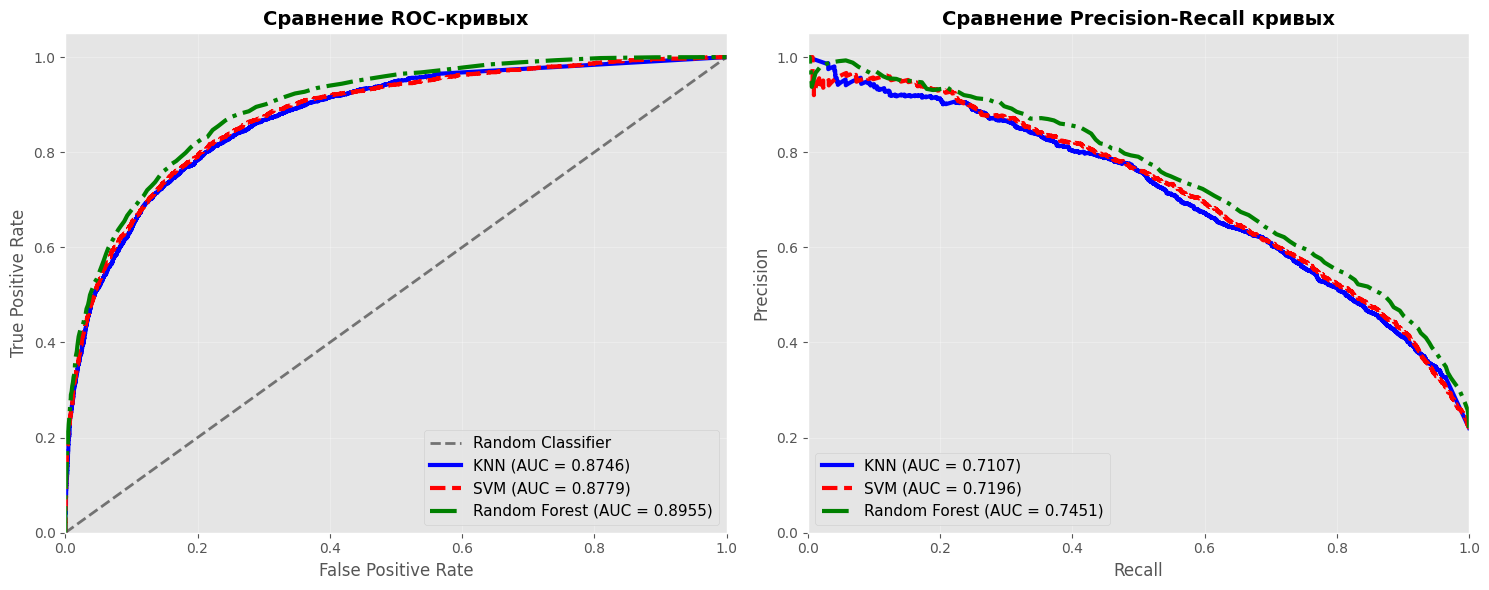
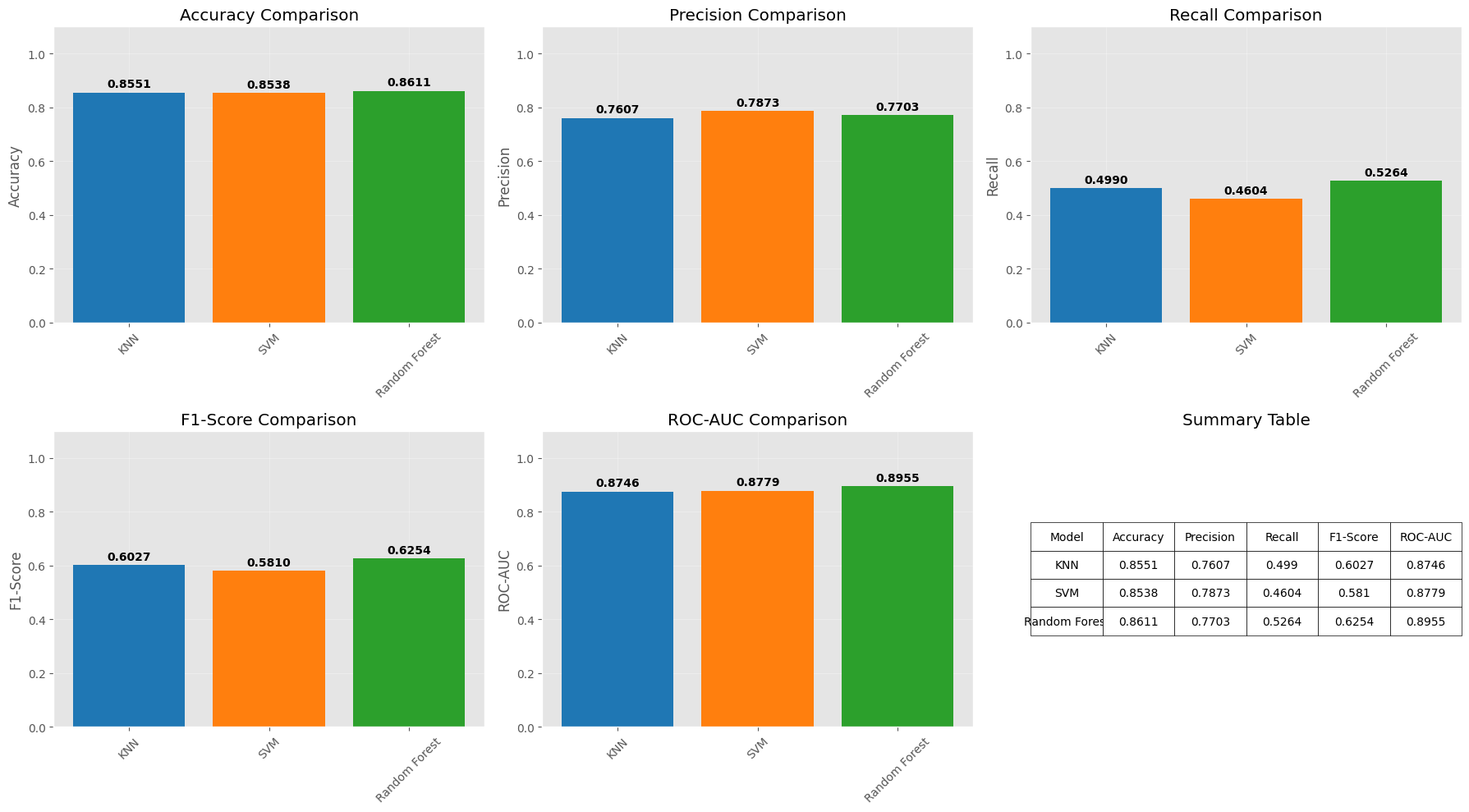
===========================================================

Model Accuracy Precision Recall F1-Score ROC-AUC

KNN 0.8551 0.7607 0.4990 0.6027 0.8746

SVM 0.8538 0.7873 0.4604 0.5810 0.8779

Random Forest 0.8611 0.7703 0.5264 0.6254 0.8955

 *Рисунки 12-13 сравнение моделей по параметрам и кривым*

# **Вывод.**

В ходе выполнения лабораторной работы были успешно реализованы все этапы предобработки данных и построения моделей машинного обучения для решения задачи бинарной классификации на датасете «Погодные условия и вероятность дождя». Проведен комплексный анализ данных, включая визуализацию распределений признаков, диаграммы рассеяния для выявления взаимосвязей, а также построение матрицы корреляций для оценки степени влияния метеорологических параметров на целевой признак.

Результаты моделей классификации:

Были обучены и протестированы три модели классификации:

- K-ближайших соседей (KNN)\*\*: Accuracy = 85.51%, F1-Score = 60.27%, ROC-AUC = 87.46%

- Машина опорных векторов (SVM)\*\*: Accuracy = 85.38%, F1-Score = 58.10%, ROC-AUC = 87.79%

- Случайный лес (Random Forest)\*\*: Accuracy = 86.11%, F1-Score = 62.54%, ROC-AUC = 89.55%

Ключевые выводы:

1. Наилучший результат показала модель Random Forest с точностью 86.11% и F1-Score 62.54%, что демонстрирует ее превосходство над KNN и SVM в данной задаче прогнозирования осадков.

2. Анализ ROC-кривых подтвердил высокую дискриминационную способность всех моделей (AUC > 0.87), при этом Random Forest достиг наивысшего значения 0.8955.

3. Важность признаков, выявленная с помощью Random Forest, показала, что ключевую роль в прогнозировании дождя на следующий день играют следующие метеорологические параметры:

- Влажность (Humidity9am, Humidity3pm)

- Количество осадков (Rainfall)

- Давление (Pressure9am, Pressure3pm)

- Температура (MinTemp, MaxTemp)

4. Сравнение моделей по метрике F1-Score, которая учитывает баланс между precision и recall, показало, что Random Forest лучше справляется с задачей классификации в условиях несбалансированных классов.

Таким образом, цель работы – закрепить навыки предобработки данных, исследования признаков и применения различных алгоритмов машинного обучения для задач бинарной классификации – была успешно достигнута. Модель Random Forest продемонстрировала наилучшие результаты и может быть рекомендована для практического использования в системе прогнозирования осадков.

Полученные результаты свидетельствуют о том, что метеорологические данные обладают достаточной предсказательной силой для построения эффективных моделей прогнозирования дождя, что может найти применение в сельском хозяйстве, логистике и других областях, зависящих от погодных условий.