МОСКОВСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ ИМ. М.В. ЛОМОНОСОВА

ФАКУЛЬТЕТ ВЫЧИСЛИТЕЛЬНОЙ МАТЕМАТИКИ И КИБЕРНЕТИКИ

Отчет по результатам выполнения заданий технологического практикума

Автор:

Ищенко Светлана Сергеевна

2024 г.

Содержание:

1. Описание датасета

Датасет, выбранный мной для анализа, содержит информацию о различных автомобилях, выставленных на продажу на вторичном рынке Индии, составлен в 2020 году. Каждая строка таблицы соответствует одному автомобилю, а столбцы содержат различные характеристики этих автомобилей. Объем датасета - 37 тысяч строк. Датасет содержит следующие столбцы:

1. **brand** : Марка автомобиля (например, "hyundai", "maruti", "tata").
2. **model** : Модель автомобиля (например, "creta", "swift", "nexon").
3. **transmission** : Тип трансмиссии (например, "manual", "automatic").
4. **age** : Возраст автомобиля в годах.
5. **fuel** : Тип топлива, на котором работает автомобиль (например, "petrol", "diesel", "cng").
6. **engine** : Объем двигателя в кубических сантиметрах (например, "1497.0", "1199.0").
7. **km** : Пробег автомобиля в километрах (например, "50015.0", "30203.0").
8. **owner** : Количество предыдущих владельцев автомобиля
9. **price** : Цена автомобиля в рупиях (например, "1231000.0", "786000.0").
10. **location** : Местоположение, где продается автомобиль (например, “mumbai”, “delhi”).
11. **mileage** (расход топлива): Расход топлива автомобиля (например, “19”, “24.4”).
12. **power** (мощность): Мощность двигателя в лошадиных силах (например,).
13. **seats** (сиденья): Количество мест в автомобиле (например, “4”, “7”).
14. **type** (тип): Тип автомобиля (например, “hatchback”, “sedan”).

Посмотрим на описания числовых столбцов:

- **Возраст (age)**: Медиана = 7, мода = 7.119, минимум = 0, максимум = 29

- **Объем двигателя (engine)**: Медиана = 1248 $см^3$, мода = 1489 $см^3$, минимум = 72 $см^3$, максимум = 5998 $см^3$
- **Пробег (km)**: Медиана = 50000 км, мода = 54065 км, максимум = 300000 км (поскольку в изначальном датасете строк, где пробег был свыше этого числа, было 10 штук, а некоторые из них не вызывали доверия (пробег 6 миллионов километров при возрасте автомобиля 7 лет), для выполнения анализа я отбросила эти строки как шум)
- **Число владельцев (owner)**: от 1 до 4, медиана = 1, мода = 1.2
- **Цена (price)**: Цена автомобилей варьируется от 40,000 до 7,066,000 рупий. Средняя цена составляет 734,645 рупий, медиана — 551,000 рупий
- **Расход топлива (mileage):**
  Значения расхода топлива варьируются от 0 до 46.62 (единицы не указаны, вероятно, в км/л). Средний расход составляет 19.31, медиана — 19.01. В 10,819 случаях данные отсутствуют.
- **Мощность двигателя (power):**
  Мощность двигателя варьируется от 34.2 до 600 л.с.. Средняя мощность составляет 103.5, медиана — 88.5. Имеется 10,926 пропусков
- **Количество мест (seats):**
  Значения варьируются от 0 до 10 мест. Среднее количество составляет 5.23, медиана — 5. Имеется 2847 пропусков.

Были удалены строки с пропусками из столбцов, соответствующих пробегу, цене и числу владельцев, так как они использовались для анализа чаще всего.

2. Реализовать аппроксимацию распределений данных с помощью ядерных оценок.

Аппроксимация распределений данных с использованием ядерных оценок плотности (KDE, Kernel Density Estimation) — это статистический метод, позволяющий построить непрерывную оценку плотности вероятности случайной величины на основе выборки. Этот метод используется, когда необходимо восстановить форму распределения данных, не предполагая заранее, что оно следует какому-либо конкретному распределению.

Ядерная оценка плотности строится по следующему принципу: для каждого наблюдения в выборке размещается "ядро" — симметричная и положительная функция, имеющая максимум в точке наблюдения и убывающую по мере удаления от нее. Затем эти ядра суммируются, чтобы получить итоговую функцию плотности.

Оценка плотности задается формулой:

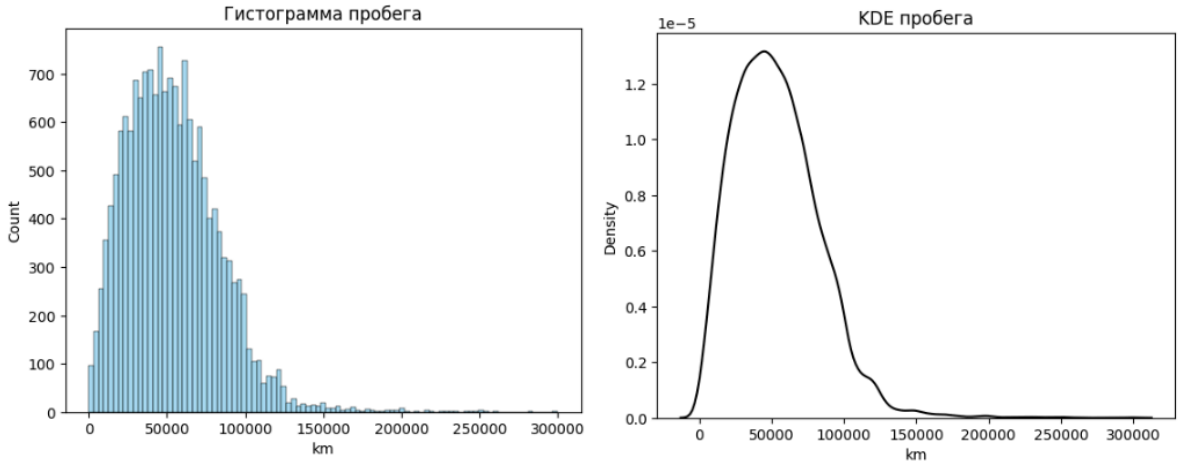
$$f(x) = \frac{1}{nh}\sum_{i=1}^{n} K\left(\frac{x - x_i}{h}\right)$$

где:

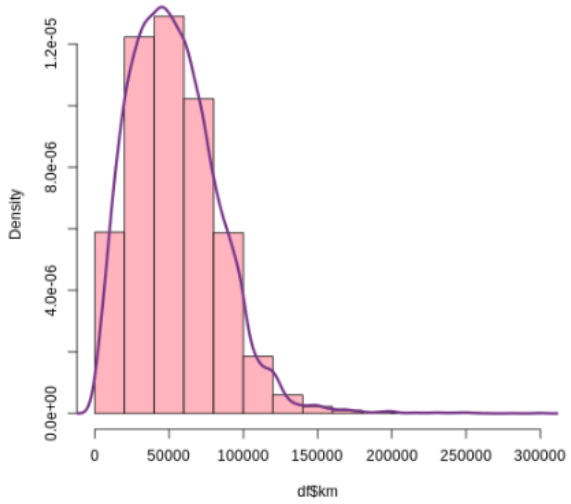
- $x_1, x_2, \ldots, x_n$ — выборка из n точек;
- K(·) — функция ядра (обычно симметричная и интегрируемая, например, гауссовская);
- h>0— ширина окна (или сглаживающий параметр), определяющий степень сглаживания оцененной плотности.

Изобразим гистограмму для пробега и построим её ядерное приближение.

Python:



R:



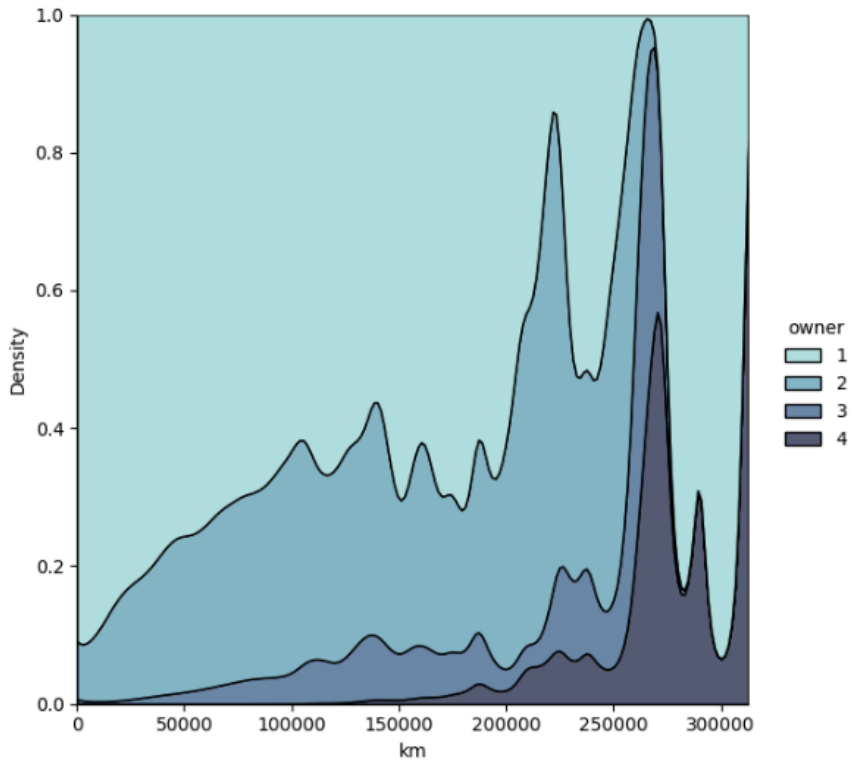
Графики получились одинаковыми, только отметим, что различаются значения ширины столбца для гистограммы по умолчанию. Очевидно, что данные распределены не нормально. Также можно пронаблюдать резкие перепады в высоте столбцов около 50тыс км пробега, 100тыс км и 150тыс км. Заметим, что либо это из-за того, что владельцы решают продавать автомобиль до достижения определенного круглого числа, либо скручивают пробег, чтобы машина попадала под пользовательские фильтры на сайтах для продажи.

3. Реализовать анализ данных с помощью cdplot, dotchart, boxplot и stripchart.

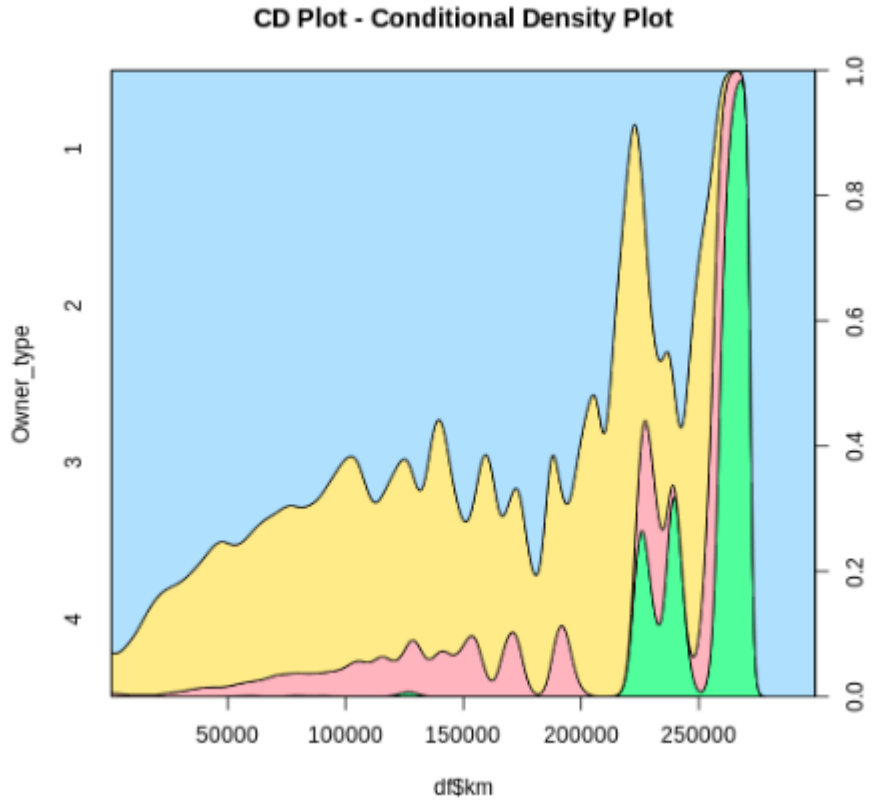
3.1 Cdplot

Построим график зависимости пробега от числа владельцев.

Python:


1.0
0.8
0.6
0.4
0.2
0.0
Density
0
50000
100000
150000
200000
250000
300000
km
owner
1
2
3
4

R:

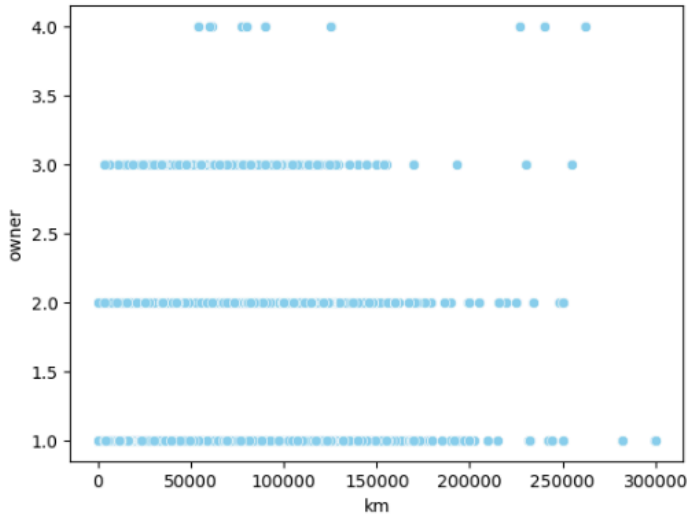

CD Plot - Conditional Density Plot
Owner_type
1
2
3
4
1.0
0.8
0.6
0.4
0.2
0.0
50000
100000
150000
200000
250000
df$km

Заметим, что при одинаковом по смыслу коде, R не строит график до конца оси X. Также на графике R почти не видно график при owner = 4 при пробеге менее 220000 км(т.е. значения, близкие к нулю по оси y). В остальном графики схожие.

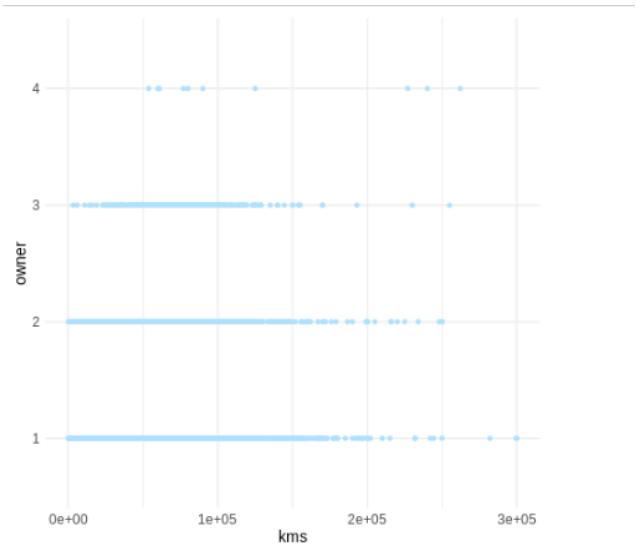
3.2 Scatterplot

Строим ту же зависимость.

Pyton:



R:



Графики идентичны.

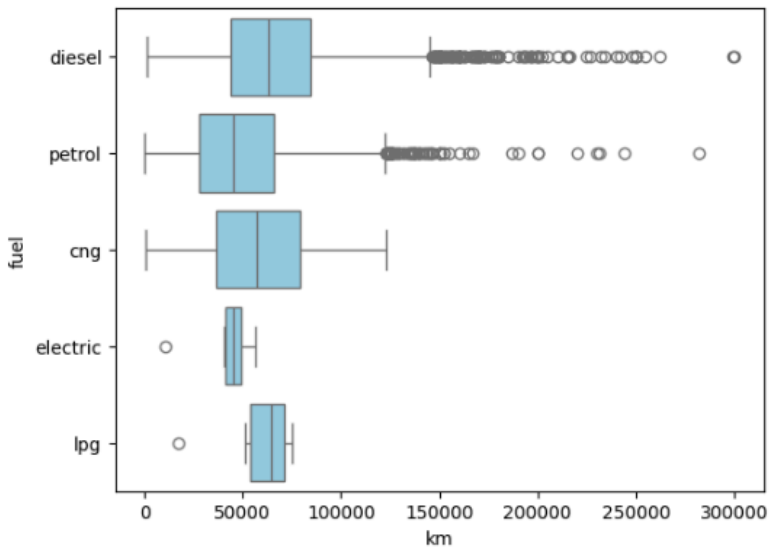
По итогу графиков выше можем сделать следующие выводы:

- Автомобили с одним владельцем имеют наиболее широкое распределение пробега.
- Большая часть таких автомобилей имеет пробег ниже 150,000 км. Плотность резко снижается на высоких значениях пробега, что указывает на то, что автомобили с одним владельцем чаще встречаются с небольшим пробегом.
- Категория автомобилей с двумя владельцами показывает промежуточное распределение: автомобили с двумя владельцами имеют умеренно высокий пробег (100,000–250,000 км).
- Для автомобилей с 3 владельцами распределение смещено к более высоким пробегам.
- Категория автомобилей с 4 владельцами имеет пик в районе 250,000–300,000 км, что говорит о том, что такие автомобили обычно уже прошли значительный пробег.

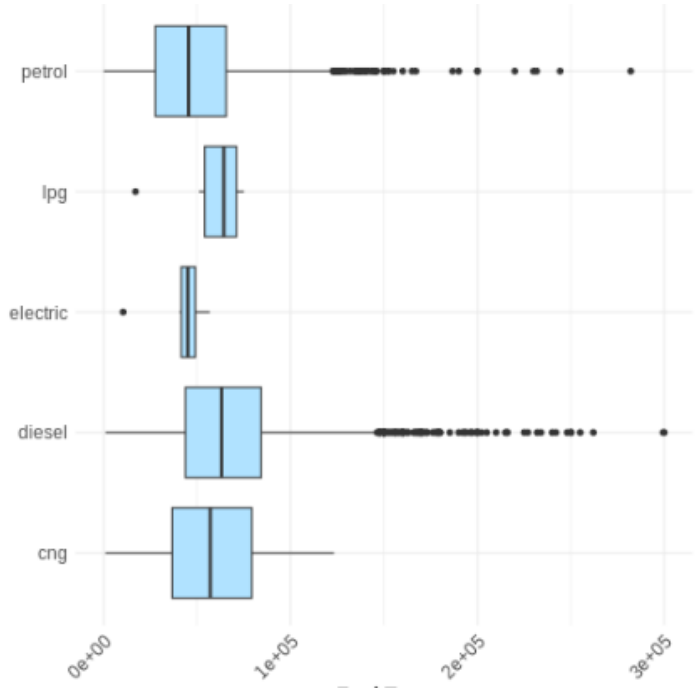
3.3 Boxplot

Построим графики зависимости пробега от типа топлива (дизельное, бензин, электроавтомобили, сжатый природный газ и сжиженный нефтяной газ)

Python:



R:



Графики идентичные.

Проанализируем результат::

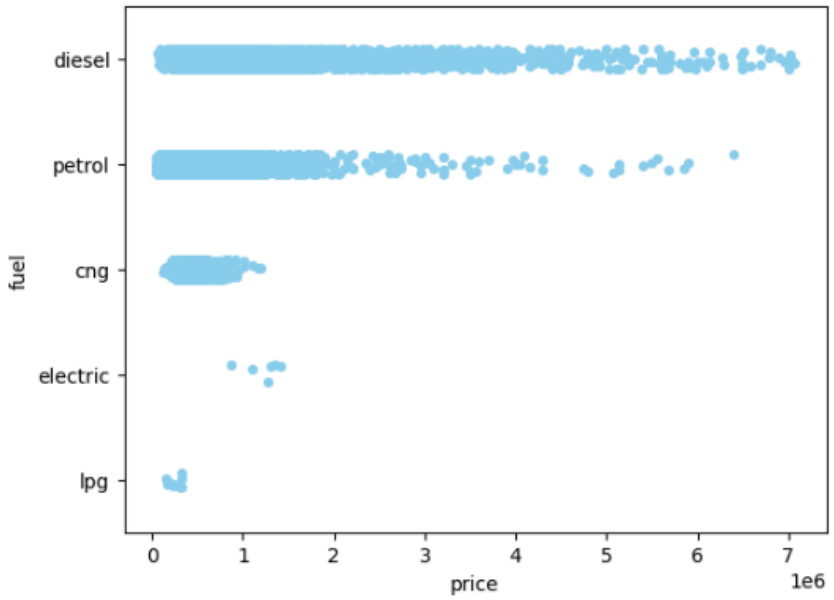
- Petrol (бензин):
  - Медианный пробег у автомобилей на бензине расположен ближе к 50,000–100,000 км.
  - Имеется значительное количество выбросов с пробегом выше 150,000 км, а некоторые даже превышают 250,000 км.
  - Диапазон пробега (межквартильный размах, IQR) варьируется от низких значений до 150,000 км.
- LPG (сжиженный газ):
  - Пробег автомобилей на LPG ниже, чем у других типов топлива.
  - Медиана находится ниже 50,000 км.
  - Имеется всего несколько выбросов,
- Electric (электромобили):
  - Электромобили имеют самый низкий пробег, и почти все значения располагаются в пределах 0–50,000 км.

    - Это может быть связано с относительной новизной электромобилей на рынке, а также с ограниченным запасом хода таких машин.
- Diesel (дизель):
    - Дизельные автомобили демонстрируют самый высокий медианный пробег среди всех типов топлива — он располагается ближе к 100,000 км.
    - Межквартильный размах (IQR) также больше, чем у бензиновых автомобилей.
    - Выбросы наблюдаются даже при значениях выше 300,000 км, что характерно для дизельных автомобилей, которые часто используются для дальних поездок и коммерческого транспорта.
- CNG (сжатый природный газ):
    - Пробег автомобилей на CNG схож с бензиновыми, но медиана чуть ниже (около 50,000 км).
    - Диапазон пробега умеренный

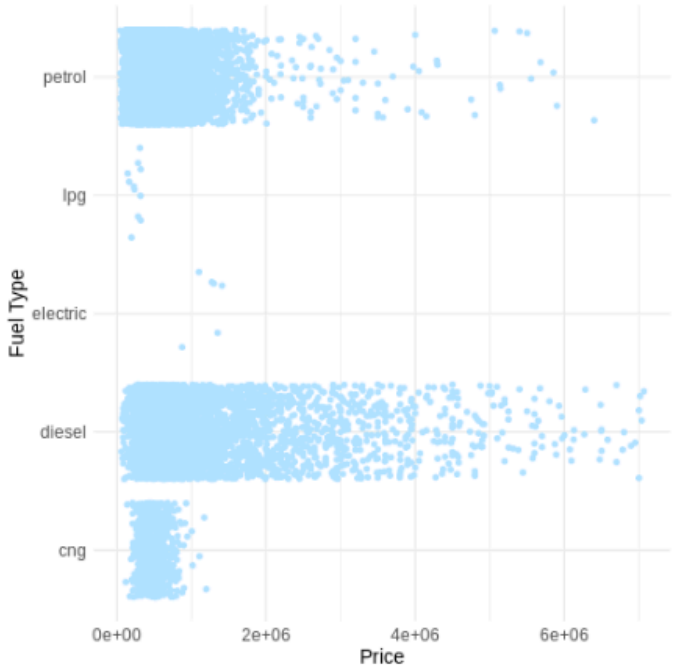
3.4 Stripchart

Визуализируем зависимость цены от типа топлива

Python:



R:



Графики идентичные.

Выводы:

1. **Дешевые сегменты:**

    - LPG и CNG автомобили сосредоточены в низком ценовом диапазоне, что делает их хорошим выбором для экономных водителей.
    - Бензиновые автомобили находятся чуть выше в ценовом диапазоне, но в основном остаются в пределах массового рынка.
2. Премиум сегмент:
    - Электромобили и некоторые дизельные автомобили занимают верхний ценовой сегмент.
    - Это связано с современными технологиями (электромобили) и популярностью дизельных машин в коммерческом или премиальном использовании.
3. Ценовой диапазон:
    - Дизельные автомобили имеют самый широкий диапазон цен, что говорит об их универсальности и применении в разных сегментах рынка.

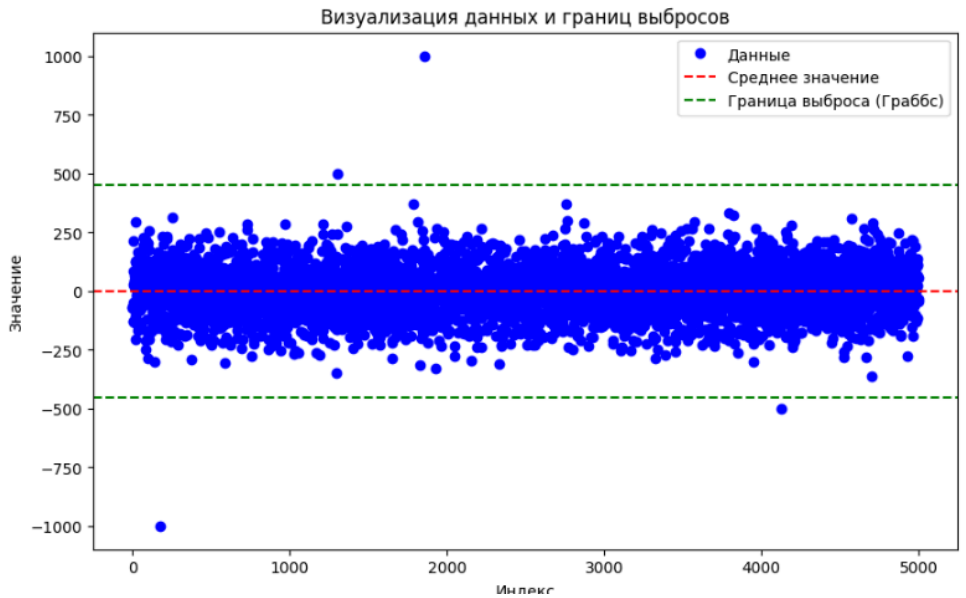
4. Проверить, являются ли наблюдения выбросами с точки зрения формальных статистических критериев Граббса и Q-теста Диксона. Визуализировать результаты.

4.1 Критерий Граббса

Критерий Граббса — это статистический метод для обнаружения выбросов в одномерных данных, которые предполагаются распределенными по нормальному закону. Этот тест особенно полезен для идентификации экстремальных значений, которые значительно отклоняются от остальных данных в выборке. Применю этот тест к сгенерированным нормальным данным, в которые добавлю 4 выброса на случайные индексы вместо оригинальных значений.

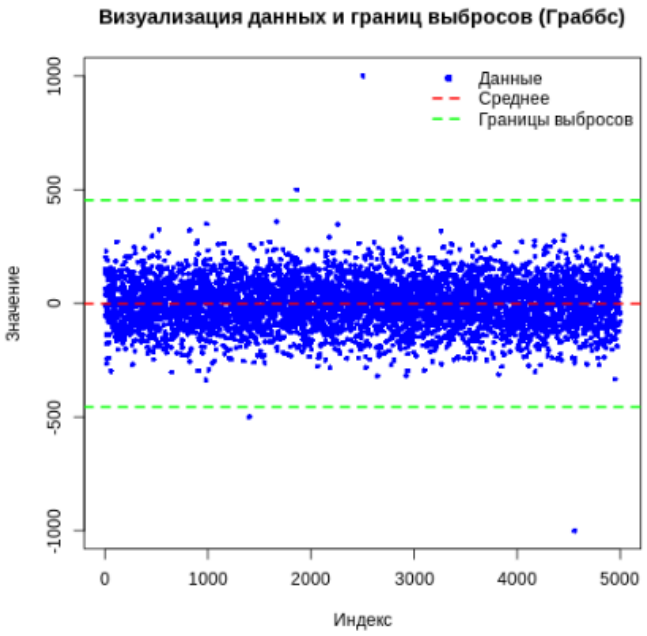
Python:

Статистика 9.8454, critical value 4.4217094



R:

Статистика 9.735067, Critical Value 4.421709



Результаты схожие, оба языка корректно определяют выбросы, значения выдают примерно одинаковые.

4.2 Тест Диксона

Тест Диксона — это статистический метод для обнаружения выбросов в малых выборках, которые предполагаются распределенными по

нормальному закону. Тест особенно полезен, когда данные имеют небольшой размер (до 25 элементов) и требуется проверить, являются ли крайние значения (наименьшее или наибольшее) выбросами.

Генерирую выборку из нормального распределения размером 25, MO = 0, дисперсия = 5 и добавляю один выброс = 50

Результат теста Диксона:

- Python: Q = 0.4376, p-value = 0.277
- R: Q = 0.5267, p-value = 0.2955

На разных языках получаем примерно одинаковые результат, гипотеза о том, что значение = 50 - выброс, подтверждается.

5. Воспользоваться инструментами для заполнения пропусков в данных. Пропуски внести вручную и сравнить результаты заполнения с истинными значениями.

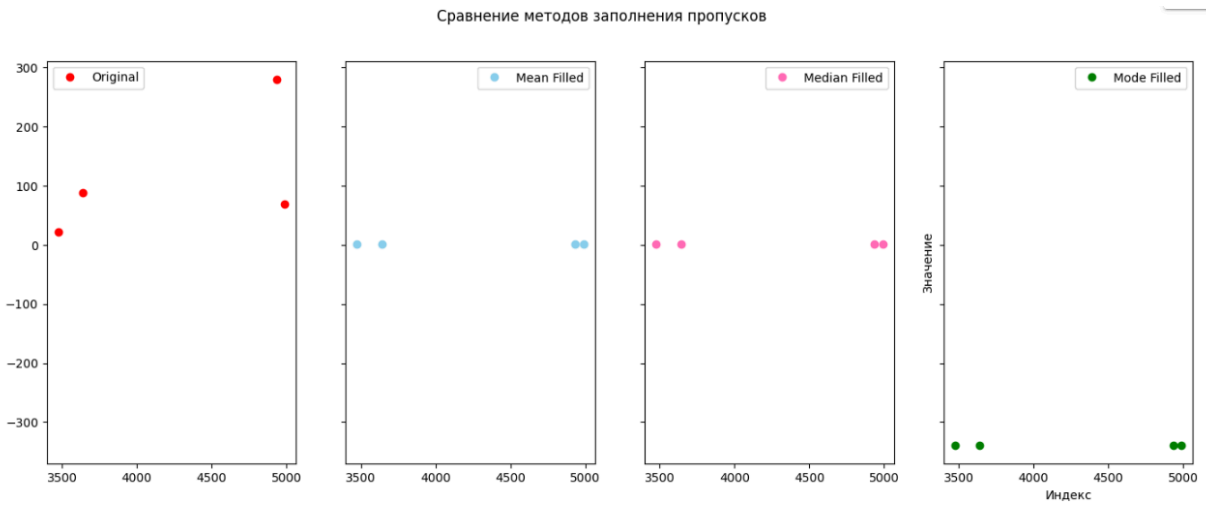
Для выполнения задания пользуюсь то же выборкой, что использовалась в реализации критерия Граббса, но вместо выбросов делаю пропуски в данных, а затем заполняю их медианой, средним или модой

Python:

- Значение, использованное для заполнения средним: 1.0924
- Значение, использованное для заполнения медианой: 1.6708
- Значение, использованное для заполнения модой: -338.6813

R:

- Значение, использованное для заполнения средним: -1.473914
- Значение, использованное для заполнения медианой: -1.223104
- Значение, использованное для заполнения модой: -337.1739



Результаты получились примерно одинаковыми, с учетом того, что выборки генерируются немного разные.

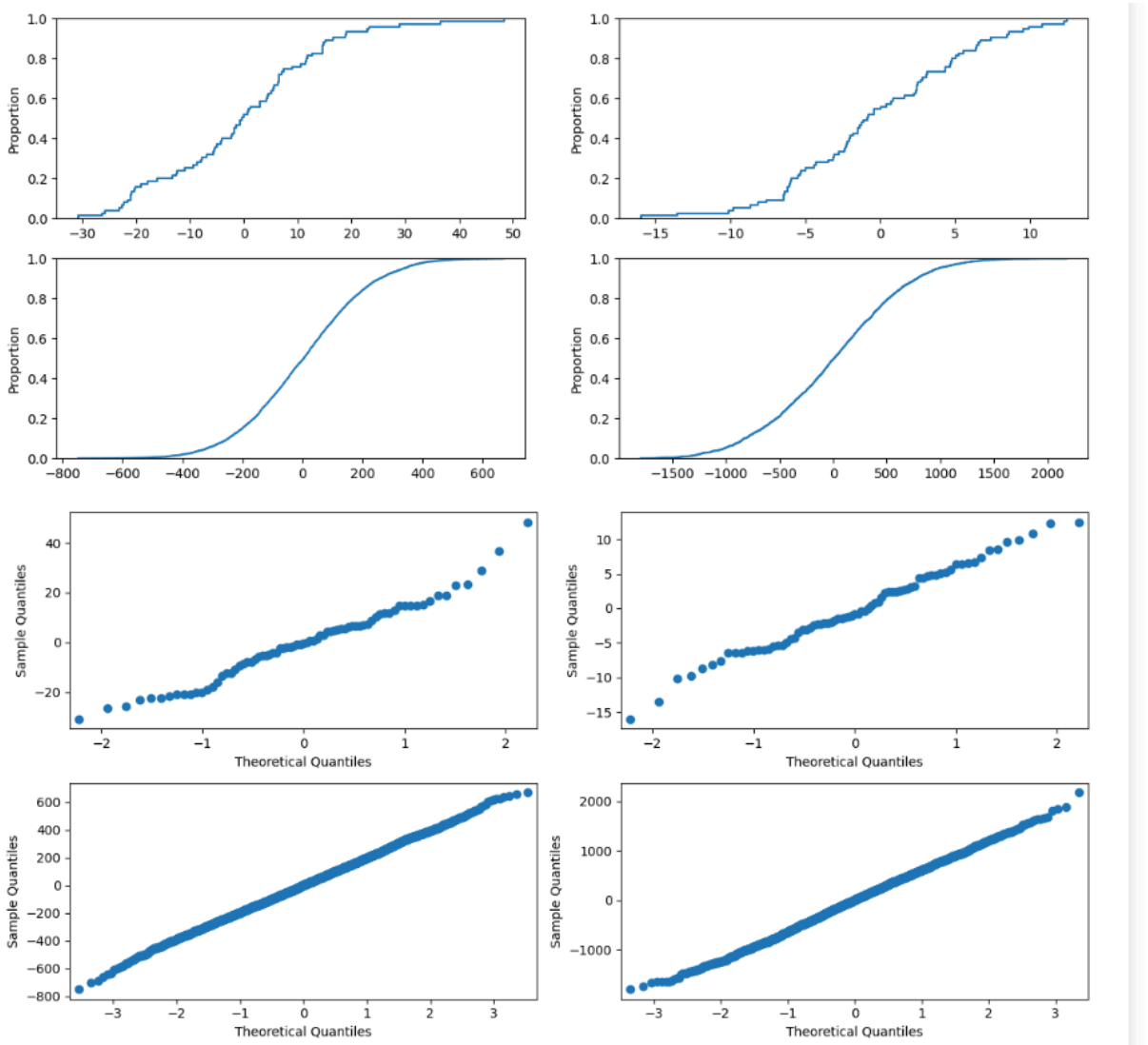
6. Сгенерировать данные из нормального распределения с различными параметрами и провести анализ с помощью графиков эмпирических функций распределений,

квантилей, метода огибающих, а также стандартных процедур проверки гипотез о нормальности (критерии Колмогорова-Смирнова, Шапиро-Уилка, Андерсона-Дарлинга, Крамера фон Мизеса, Колмогорова-Смирнова в модификации Лиллиефорса и Шапиро-Франсия). Рассмотреть выборки малого (не более 50-100 элементов) и умеренного (1000-5000 наблюдений) объемов.

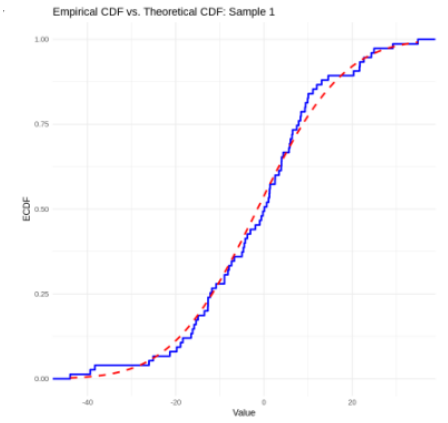
Генерируем 4 выборки:

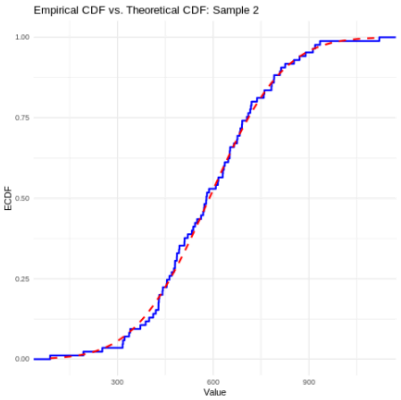
	размер	МО	Дисперсия
data1	75	0	15
data2	75	0	6
data3	5000	0	200
data4	2500	0	600

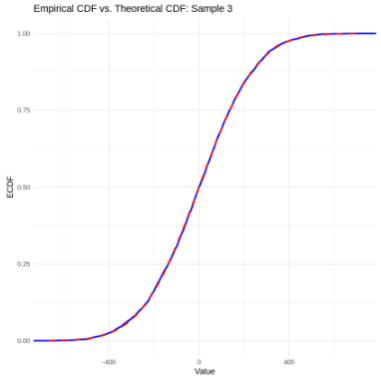
Графики на Python:

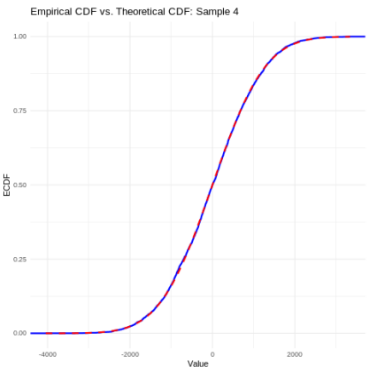


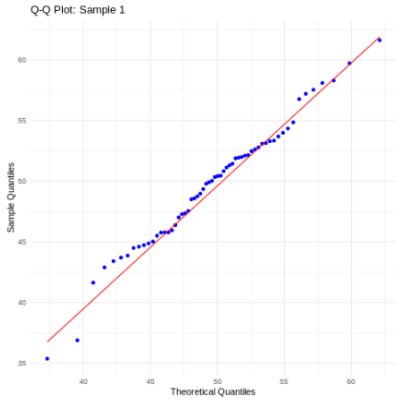
R:

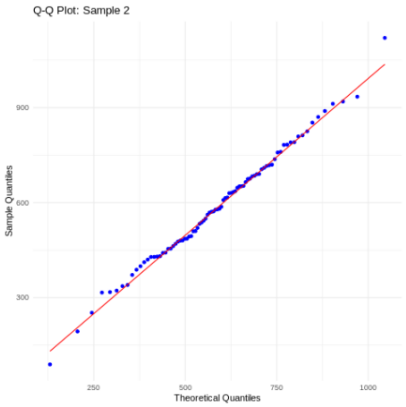

Empirical CDF vs. Theoretical CDF: Sample 1
ECDF
Value

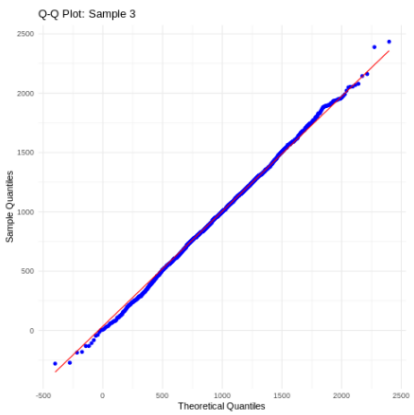

Empirical CDF vs. Theoretical CDF: Sample 2
ECDF
Value

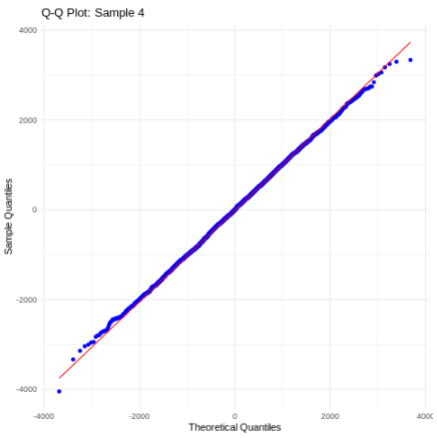

Empirical CDF vs. Theoretical CDF: Sample 3
ECDF
Value


Empirical CDF vs. Theoretical CDF: Sample 4
ECDF
Value


Q-Q Plot: Sample 1
Sample Quantiles
Theoretical Quantiles


Q-Q Plot: Sample 2
Sample Quantiles
Theoretical Quantiles


Q-Q Plot: Sample 3
Sample Quantiles
Theoretical Quantiles


Q-Q Plot: Sample 4
Sample Quantiles
Theoretical Quantiles

Графики несколько различаются, потому что сами выборки не могли сгенерироваться одинаковыми, но заметна тенденция, что чем больше выборка, тем больше распределение схоже с нормальным.

6.1 Критерий Колмогорова-Смирнова:

Сравнивает эмпирическую функцию распределения выборки с ожидаемой (нормальной).

	D-статистика Python	D-статистика R	p-value Python	p-value R
data1	0.4388678	0.064277	1.14e-13	0.8962
data2	0.391359	0.076929	7.35e-11	0.7371
data3	0.499944	0.006872	0.0	0.9722
data4	0.50247	0.0095941	0.0	0.7403

6.2 Критерий Шапиро-Уилка:

Оценивает, насколько данные соответствуют нормальному распределению, вычисляя взвешенную сумму данных, упорядоченных по значению. Высокая мощность для малых выборок, для больших теряет мощность

	статистика Python	статистика R	p-value Python	p-value R

data1	0.5025	0.98313	0.2021	0.4189
data2	0.5024	0.98512	0.82925	0.5275
data3	0.5024	0.99861	0.6964	0.1004
data4	0.999484	0.99948	0.77169	0.3608

6.3 Критерий Андерсона-Дарлинга

Модификация теста Колмогорова-Смирнова, более чувствительная к отклонениям на концах распределения. Хорошо подходит для умеренных выборок

	статистика Python	статистика R	p-value Python	p-value R
data1	0.5755	0.16049	True	0.9453
data2	0.32283	0.2859	True	0.9442
data3	0.57553	0.95132	True	0.0162
data4	0.32283	0.37805	True	0.4076

(На Python критерий Андерсона-Дарлинга не возвращает p-value, только решение, принята ли гипотеза.

6.4 Критерий Крамера фон Мизеса

Похож на тест Андерсона-Дарлинга, но оценивает среднее отклонение эмпирической функции распределения от теоретической на всем диапазоне.

	статистика Python	статистика R	p-value Python	p-value R
data1	6.19336	0.023143	4.8e-11	0.9319
data2	4.04233	0.036201	3.16e-10	0.747
data3	411.617	0.16163	4.88e-11	0.01675
data4	208.262	0.052865	3.16e-10	0.4699

6.5 Критерий Колмогорова-Смирнова в модификации Лиллиефорса

Устраняет недостаток теста K-S, который требует заранее заданных параметров распределения. Параметры нормального распределения (среднее и стандартное отклонение) оцениваются по выборке.

	статистика Python	статистика R	p-value Python	p-value R
data1	0.0628	0.058489	0.26358	0.8774
data2	0.0708	0.048054	0.968150	0.8989
data3	0.0085	0.022247	0.50941	0.02249
data4	0.01677	0.010789	0.09254	0.3126

6.6 Критерий Шапиро-Франсия:

Модификация теста Шапиро-Уилка, оптимизированная для выборок большого объема.

	статистика Python	статистика R	p-value Python	p-value R
data1	0.7308	0.99269	0.0544	0.9419
data2	0.1857	0.98683	0.9032	0.4626
data3	0.1871	0.9985	0.9042	0.06388
data4	0.4359	0.99947	0.2981	0.2837

Для некоторых распределений результаты на R и Python значительно различались. Перечислю возможные причины:

- Сами выборки отличаются, потому что генерируются случайно
- Алгоритмы для тестов могут быть реализованы по-разному, особенно для тестов Шапиро-Уилка и Андерсона-Дарлинга.
- Различия в вычислительных платформах и точности операций (например, округление) могут влиять на результаты.
- Если в данных есть пропуски или выбросы, обработка в R и Python может отличаться.
- В некоторых тестах уровень значимости (alpha) или другие параметры могут быть установлены по-разному.

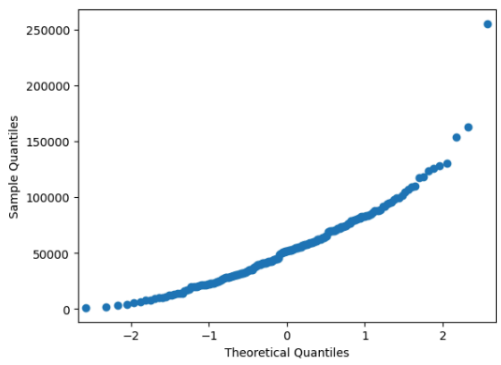
Также некоторые критерии отвергали гипотезу о нормальности данных(хотя данные изначально из нормального распределения). Поскольку происходило это в основном в реализации на Python, а на R нулевых значений не было, только близкие к границе, можно предположить, что выборки на Python генерируются с большими отклонениями от нормального распределения.

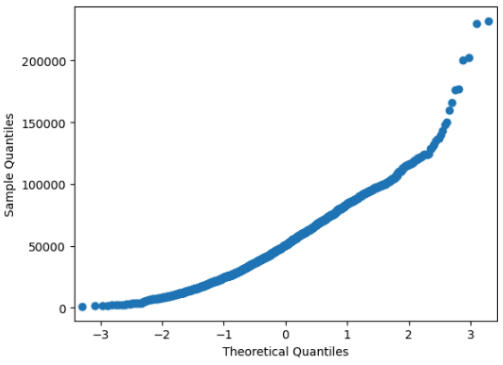
7. Продемонстрировать пример анализа данных с помощью графиков квантилей, метода огибающих, а также стандартных процедур проверки гипотез о нормальности. Рассмотреть выборки малого и умеренного объемов

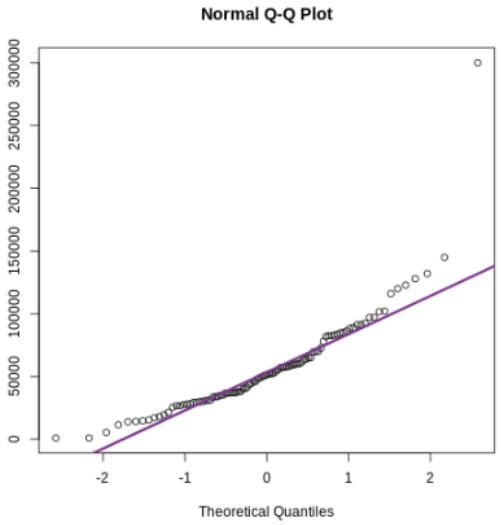
Рассматриваю выборки из столбца со значением пробега размером 200 и 2000 элементов.

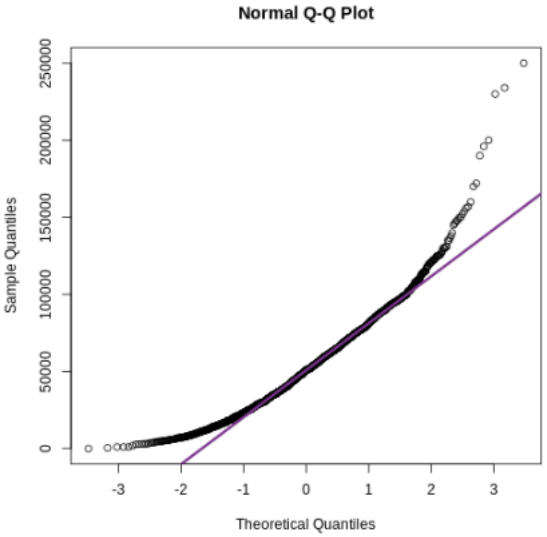
График квантилей:

Python:



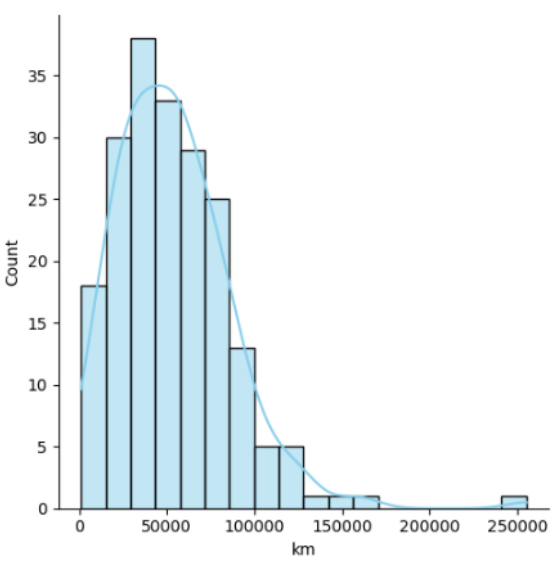


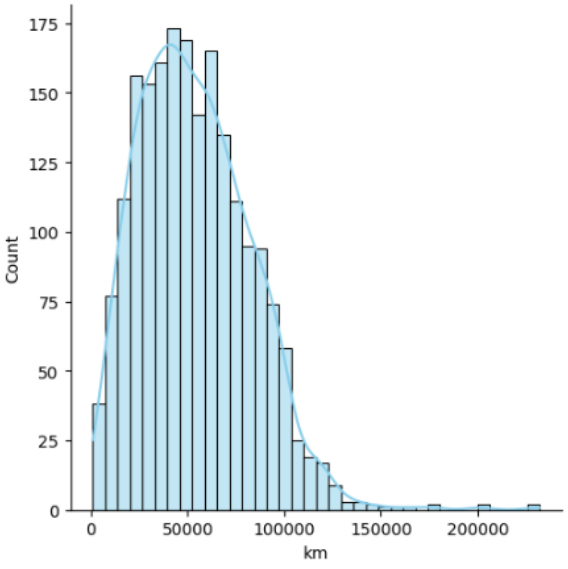




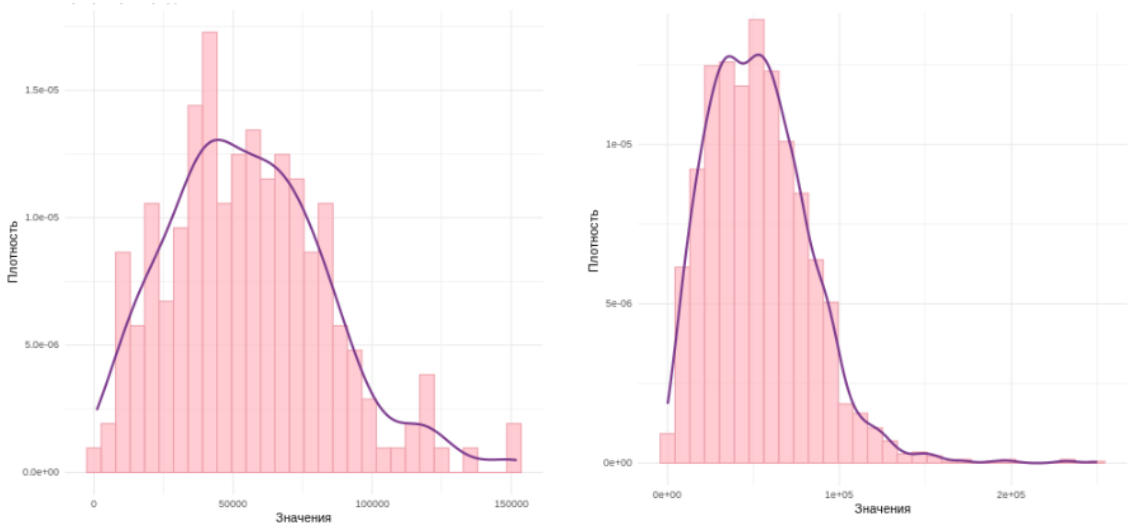
Метод огибающих:

Python:





R:



km_200:

Шапиро-Уилка:

- Python: statistic: 1.0, p-value: 1.2e-09
- R: W = 0.97449, p-value = 0.001052

Крамера-фон-Мизеса:

- Python: statistic=66.67, p-value=0
- R: W = 0.096878, p-value = 0.1232

km_2000:

Шапиро-Уилка:

- Python: D statistic: 1.0, p-value: 0.0
- R: W = 0.95047, p-value < 2.2e-16

Крамера-фон-Мизеса:

- Python: statistic=666.67, p-value=0
- R: A = 11.138, p-value < 2.2e-16

8. Продемонстрировать применение для проверки различных гипотез и различных доверительных уровней (0.9, 0.95, 0.99) следующих критериев

Генерирую для дальнейшего анализа две выборки из нормального распределения:

sample1 - MO = 6 дисперсия = 25 размер = 400

sample2 - MO = 0 дисперсия = 13 размер = 400

8.1 Стьюдента, включая односторонние варианты, когда проверяемая нулевая гипотеза заключается в том, что одно из сравниваемых средних значений больше (или меньше) другого. Реализовать оценку мощности критериев при заданном объеме выборки или определения объема выборки для достижения заданной мощности.

Критерий Стьюдента, также известный как t-тест, — это статистический тест, используемый для проверки гипотез о средних значениях выборок. Он был разработан Уильямом С. Госсетом и широко используется в статистике для сравнения средних значений двух выборок или для проверки гипотезы о среднем значении одной выборки.

two-sided:

- Python: p-value = 3.03e-07
- R: p-value = 0.000292

one-sided(sample2, sample1, ‘less’):

- Python: p-value = 1.0
- R: p-value = 0.99

Для двустороннего теста гипотеза отвергается с любым уровнем уверенности, для одностороннего - принимается для любого.

Python:

- Объем выборки для достижения мощности 0.80 при уровне значимости 0.10: 51
- Объем выборки для достижения мощности 0.80 при уровне значимости 0.05: 64
- Объем выборки для достижения мощности 0.80 при уровне значимости 0.01: 96

R:

- Объем выборки для достижения мощности 0.80 при уровне значимости 0.10: n = 50
- Объем выборки для достижения мощности 0.80 при уровне значимости 0.05: 67
- Объем выборки для достижения мощности 0.80 при уровне значимости 0.010: n = 99

Результаты для двух языков примерно одинаковые.

8.2 Тест Уилкоксона-Манна-Уитни

Тест Уилкоксона-Манна-Уитни, также известный как U-тест Манна-Уитни, — это непараметрический статистический тест, используемый для сравнения двух независимых выборок. Он является альтернативой t-тесту для независимых выборок, когда данные не следуют нормальному распределению или когда выборки малы. Тест Уилкоксона-Манна-Уитни основывается на рангах данных, а не на их фактических значениях, что делает его менее чувствительным к выбросам и отклонениям от нормальности.

Применим критерий Манна-Уитни для сравнения средних значений выборок цен на автомобили марок BMW и Audi. Нулевая гипотеза двустороннего теста - средние не равны, одностороннего - средняя цена на автомобиль марки BMW больше, чем на автомобиль марки Audi.

Python: mean(X) = 2429280.303030303 vs mean(Y) = 2533600.0

- E(X) != E(Y): p-value = 0.09859715011849948
- E(X) < E(Y): p-value = 0.9507648225425709

R:

- E(X) != E(Y): p-value = 0.0986
- E(X) < E(Y): p-value = p-value = 0.9508

Результаты примерно одинаковые. Гипотеза E(X) < E(Y) подтверждается с любым уровнем доверия, E(X) != E(Y) - с уровнем доверия 0.95 и 0.99

8.3 Критерии Фишера, Левене, Бартлетта, Флигнера-Килина (проверка гипотез об однородности дисперсий)

8.3.1 Критерий Фишера

Критерий Фишера для проверки однородности дисперсий (F-test for equality of variances) — это статистический тест, используемый для проверки гипотезы о равенстве дисперсий двух выборок. Он также известен как тест Фишера для сравнения дисперсий. Этот тест полезен, когда необходимо определить, можно ли предположить, что две выборки происходят из распределения с одинаковыми дисперсиями. Критерий Фишера применим только к нормально распределенным данным, поэтому снова используем выборки sample1 и sample2

Python: F-статистика: 3.3649, p-значение: 2.22e-16

R: F-статистика: 3.51279, p-значение: 0

В обоих случаях гипотеза отвергается, что ожидаемо, так как sample1 и sample2 имеют разные дисперсии.

8.3.2 Критерий Левене

Критерий Левена — это статистический тест, используемый для проверки гипотезы о равенстве дисперсий (вариаций) двух или более выборок. Он является альтернативой тесту Фишера для однородности дисперсий, особенно когда данные не принадлежат к нормальному распределению. Критерий Левена менее чувствителен к отклонениям от нормальности и поэтому часто предпочтителен в практических приложениях. Создадим выборку sample3 с такой же дисперсией, как у sample2 и применим Критерий Левене к различным комбинациям выборок:

Sample2 и sample3:

- Python: статистика = 0.1597, p-значение = 0.6894
- R: статистика = 0.0055, p-значение = 0.9411

Sample1 и sample2:

- Python: статистика = 105.123, p-значение = 2.95e-23
- R: статистика = 102.582, p-значение = 0.0

Гипотезы отвергаются/подтверждаются в соответствии с ожиданиями на обоих языках.

8.3.3 Критерий Бартлетта

Критерий Бартлетта — это статистический тест, используемый для проверки гипотезы о равенстве дисперсий (вариаций) двух или более выборок. Он основан на предположении, что данные в каждой выборке следуют нормальному распределению. Критерий Бартлетта чувствителен к отклонениям от нормальности, поэтому его использование оправдано только в случае, когда данные действительно следуют нормальному распределению. Применим его на двух разных наборах:

Sample2 и sample3:

- Python: статистика = 1.2350904430034635, p-значение = 0.26641934738473266
- R: K-squared = 0.031384, df = 1, p-value = 0.8594

Sample1, sample2 и sample3:

- Python: статистика = 192.979, p-значение = 1.244e-42
- R: K-squared = 52.032, df = 2, p-value = 5.027e-12

На обоих языках гипотезы, как и ожидалось, отвергаются.

8.3.4 Критерий Флингера-Килина

Критерий Флингера-Килина — это статистический тест, используемый для проверки гипотезы о равенстве дисперсий (вариаций) двух или более выборок. Он является альтернативой тесту Бартлетта и тесту Левена, особенно когда данные не следуют нормальному распределению. Критерий Флингера-Килина менее чувствителен к отклонениям от

нормальности и поэтому часто предпочтителен в практических приложениях. Сравним на моих данных дисперсию для пробега автомобилей марок nissan и honda.

- Python: статистика = 4.144687534875316, p-значение = 0.041765362840962716
- R: med chi-squared = 4.1447, df = 1, p-value = 0.04177

Результаты на разных языках получили одинаковые.

Если брать уровень доверия 0.99, то гипотеза не отвергается, при остальных уровнях - отвергается.

9. Исследовать корреляционные взаимосвязи в данных с помощью коэффициентов корреляции Пирсона, Спирмена и Кендалла.

Для этого пункта я искала зависимости между пробегом и другими характеристиками для автомобилей, продающихся в городе Мумбаи.

9.1 Коэффициент корреляции Пирсона

Коэффициент корреляции Пирсона измеряет линейную зависимость между двумя количественными переменными. Он основывается на ковариации между переменными, делённой на произведение их стандартных отклонений. Он предполагает нормальное распределение данных. Я сгенерировала выборку из нормального распределения и столбец y = x + eps, где eps - шум, по модулю не превышает 0.3

- Python: 0.98
- R: 0.97

Несмотря на шум в данных, коэффициент Пирсона все равно выявляет линейную зависимость

9.2 Коэффициент корреляции Спирмена

Коэффициент корреляции Спирмена является ранговым коэффициентом корреляции. Он оценивает силу монотонной (не обязательно линейной) зависимости между двумя переменными и не требует нормальности распределения. Для расчёта Спирмена данные сначала преобразуются в ранги, и затем используется обычная корреляция Пирсона между этими рангами.

Исследую зависимость между пробегом и количеством мест

- Python: 0.26
- R: 0.27

Низкое значение корреляции.

9.3 Коэффициент корреляции Кендалла

Коэффициент корреляции Кенделла также является ранговым коэффициентом. Он измеряет степень согласованности рангов между двумя переменными, используя концепцию "параллельных" и "перекрещивающихся" пар

Исследую зависимость между пробегом и возрастом автомобиля

- Python: 0.47
- R: 0.47

Значение корреляции не очень высокое, вывода о линейной зависимости сделать нельзя.

Замечу, что для зависимости пробега от числа сидений коэффициент Кенделла ниже, чем Спирмена. Вероятно, это связано с тем, что на коэффициент Кенделла влияет порядок данных.

10. Продемонстрировать использование методов хи-квадрат, точного теста Фишера, теста МакНемара, Кохрана-Мантеля-Хензеля.

10.1 Хи-квадрат

Критерий хи-квадрат — это статистический тест, который используется для проверки гипотезы о том, что распределение наблюдаемых данных соответствует теоретическому распределению или что две выборки независимы друг от друга. Критерий хи-квадрат применяется для категориальных данных, а не для непрерывных переменных, и используется для оценки соответствия наблюдаемого распределения с ожидаемым или для сравнения двух выборок, чтобы проверить, насколько они схожи в терминах категориальных переменных.

Применим метод хи-квадрат для сравнения рынка в Мумбае и Хайдарабаде. Для этого предварительно посчитаем квантили цены для всего датасета и сопоставлять будем по числу цен на автомобили, попавших в соответствующие квантили

Хайдарабад - город на юге Индии, административный центр одного из штатов. Население - около 10 миллион человек (В Мумбае 13 млн). В Хайдарабаде находится много промышленных предприятий, а также научно-исследовательских предприятий ОПК и много IT-компаний, что делает Хайдарабад, как и Мумбаи, одним из наиболее богатых городов Индии, поэтому и можем предположить, что рынок там будет схожим.

Python:

```
            Q1   Q2   Q3   Q4
mumbai     211  156  131  291
hyderabad  220  128  126  265
Chi-Square Statistic: 2.628290639364584
p-value: 0.4525514049416288
Degrees of Freedom: 3
Expected Frequencies:
[[222.55170157 146.64659686 132.70484293 287.09685864]
 [208.44829843 137.35340314 124.29515707 268.90314136]]
```

R:

```
           Q1  Q2  Q3  Q4
mumbai    211 156 131 291
hyderabad 220 128 126 265
Chi-Square Statistic: 2.63
p-value: 0.45255
Degrees of Freedom: 3
Expected Frequencies:
                Q1       Q2       Q3       Q4
mumbai    222.5517 146.6466 132.7048 287.0969
hyderabad 208.4483 137.3534 124.2952 268.9031
```

Результаты схожие. Оба теста принимают гипотезу об отсутсвии значительных различий в распределений цен на автомобили в двух городах.

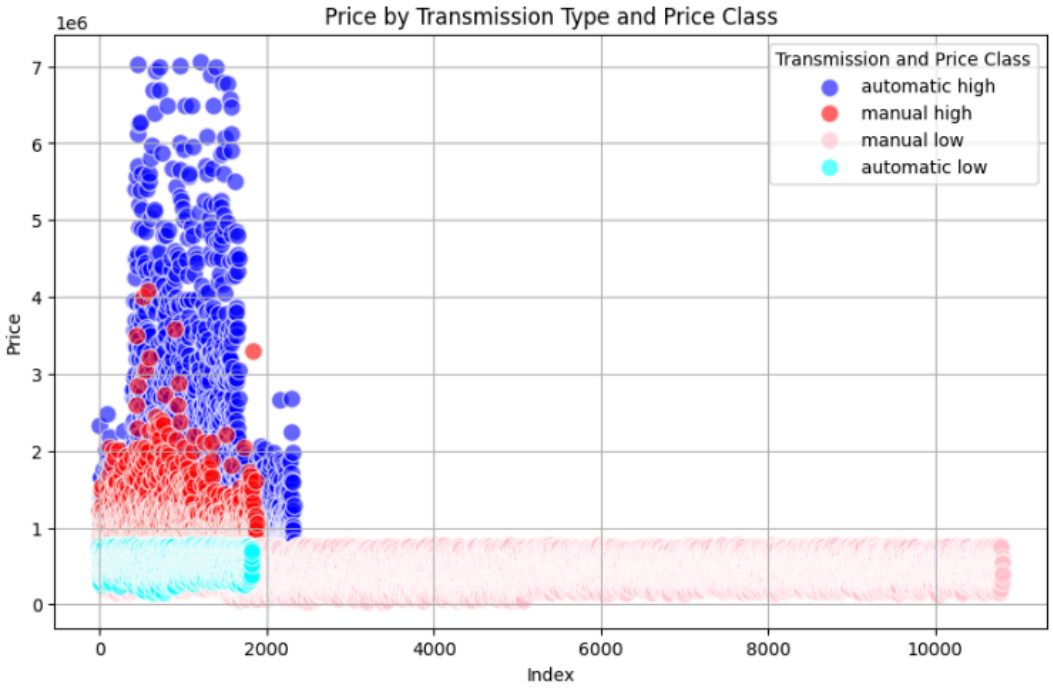
10.2 Точный тест Фишера

Точный тест Фишера — это статистический тест, используемый для анализа зависимости между двумя категориальными переменными в маленьких выборках. Этот тест является точным, так как не использует приближений (например, нормального распределения), а базируется на точных гипергеометрических вероятностях. Осуществим проверку корреляции между ценой и типом коробки передач (механика/автомат). Цены поделим на две колонки: больше и меньше 0.75-квантиля.

transmission	price high	low
automatic	2319	1818
manual	1873	10801

Тест Фишера показывает Odds Ratio: 7.3558 и

p-value: 0.0 и для Python, и для R. Вывод: по тесту Фишера данные не коррелируют. Визуализируем данные, потому что результат кажется странным:



Тут видно, что корреляция есть, то есть машины с автоматической коробкой передач почти всегда дороже машин с механической, но тест Фишера эту зависимость не уловил.

10.3 Тест МакНемара

Тест Макнемара — это статистический тест, используемый для проверки гипотезы о том, что две связанные дихотомические переменные имеют одинаковые распределения. Применяю его, чтобы исследовать взаимосвязь между возрастом автомобиля для машин с разным типом коробки передач. В категорию “newer” отношу машины, возраст которых не больше 4 лет.

transmission	age newer	older
automatic	656	2038
manual	1433	7393

- Python: McNemar Statistic: 105.1, p-value: 1.15e-24
- R: McNemar's chi-squared = 105.1, df = 1, p-value < 2.2e-16

Гипотеза не подтверждается.

10.4 Тест Кохрана-Мантеля-Хензеля

Тест Кохрана-Мантеля-Хензеля — это статистический тест, используемый для анализа ассоциации между двумя бинарными переменными в стратифицированных данных. Он позволяет проверять гипотезу о независимости двух переменных, учитывая при этом наличие третьей переменной (страты), которая может влиять на результаты. За две изначальные переменные беру опять коробку передач и возраст, за третью - марку автомобиля. Для уменьшения вычислений взяты только 5 самых частых марок.

Python: CMH Odds Ratio: 1.73, p-value: 0.0

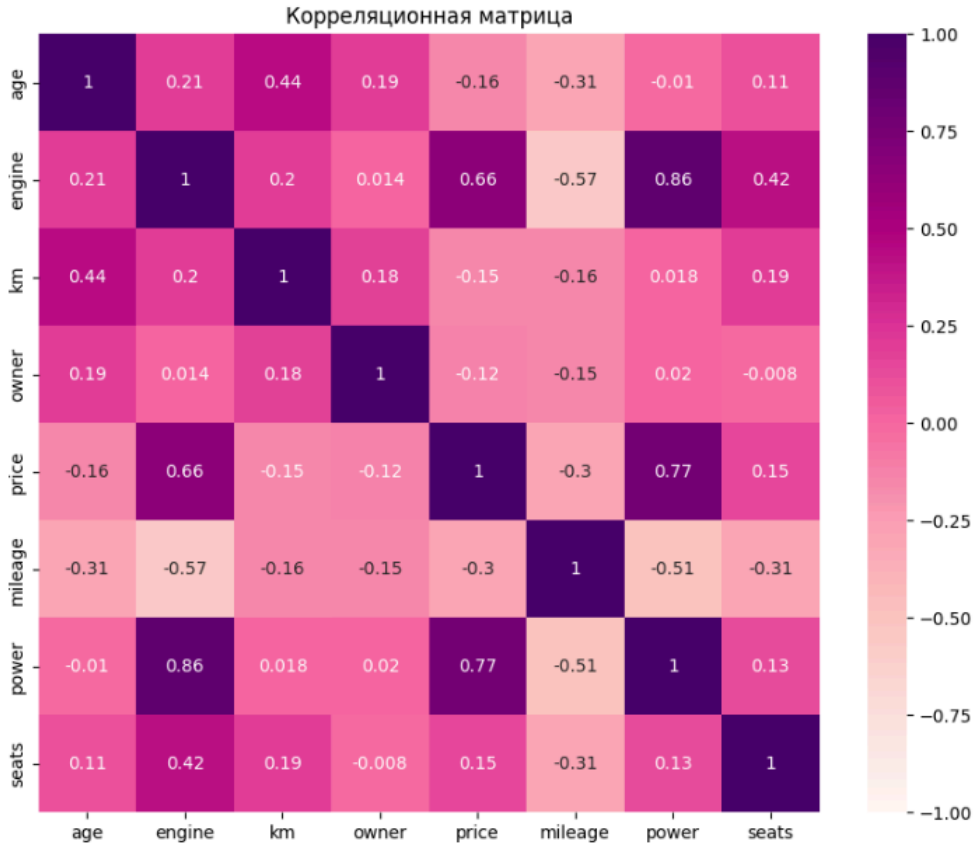
R: CMH Odds Ratio: 1.73, p-value: 2.547903e-20

Результаты одинаковые, корреляции снова нет.

11. Проверить наличие мультиколлинеарности в данных с помощью корреляционной матрицы и фактора инфляции дисперсии.

Построим корреляционную матрицу для всех числовых столбцов

Python:


Корреляционная матрица
age 1 0.21 0.44 0.19 -0.16 -0.31 -0.01 0.11
engine 0.21 1 0.2 0.014 0.66 -0.57 0.86 0.42
km 0.44 0.2 1 0.18 -0.15 -0.16 0.018 0.19
owner 0.19 0.014 0.18 1 -0.12 -0.15 0.02 -0.008
price -0.16 0.66 -0.15 -0.12 1 -0.3 0.77 0.15
mileage -0.31 -0.57 -0.16 -0.15 -0.3 1 -0.51 -0.31
power -0.01 0.86 0.018 0.02 0.77 -0.51 1 0.13
seats 0.11 0.42 0.19 -0.008 0.15 -0.31 0.13 1
age engine km owner price mileage power seats
1.00 0.75 0.50 0.25 0.00 −0.25 −0.50 −0.75 −1.00

R:


Корреляционная матрица
age engine km owner price mileage power seats
age 1.00 0.08 0.50 0.39 -0.31 -0.27 0.00 -0.01
engine 0.08 1.00 0.20 0.06 0.66 -0.62 0.86 0.43
km 0.50 0.20 1.00 0.24 -0.17 -0.15 0.02 0.24
owner 0.39 0.06 0.24 1.00 -0.10 -0.14 0.02 0.02
price -0.31 0.66 -0.17 -0.10 1.00 -0.34 0.77 0.09
mileage -0.27 -0.62 -0.15 -0.14 -0.34 1.00 -0.53 -0.33
power 0.00 0.86 0.02 0.02 0.77 -0.53 1.00 0.13
seats -0.01 0.43 0.24 0.02 0.09 -0.33 0.13 1.00
1 0.8 0.6 0.4 0.2 0 -0.2 -0.4 -0.6 -0.8 -1

Некоторые значения незначительно отличаются, но наиболее коррелированные значения совпадают. Наибольшие значения корреляции:

engine~power: 0.86

engine~price: 0.66

price~power: 0.77

mileage~engine: -0.54

mileage~power: -0.51

Из-за того, что между объемом двигателя и лошадиными силами очень сильная корреляция(что неудивительно), с другими столбцами эти данные коррелируют “в паре”. Число миль за литр горючего имеет с этими значениями обратную корреляцию, потому что чем больше объем двигателя, тем больше топлива он потребляет. Также оказалось, что объем двигателя и лошадиные силы значительно влияют на цену автомобиля, намного больше, чем пробег или возраст.

12. Исследовать зависимости в данных с помощью дисперсионного анализа

Исследование зависимостей в данных с помощью дисперсионного анализа (ANOVA, Analysis of Variance) — это статистический метод, используемый для оценки различий между средними значениями двух или более групп. Дисперсионный анализ позволяет определить, насколько значимы различия между группами по сравнению с различиями внутри групп. Основная цель дисперсионного анализа — проверка гипотезы о том, что средние значения групп равны. Он предполагает нормальность данных, поэтому генерирую три выборки, со средними 5.0, 5.1, 4.9

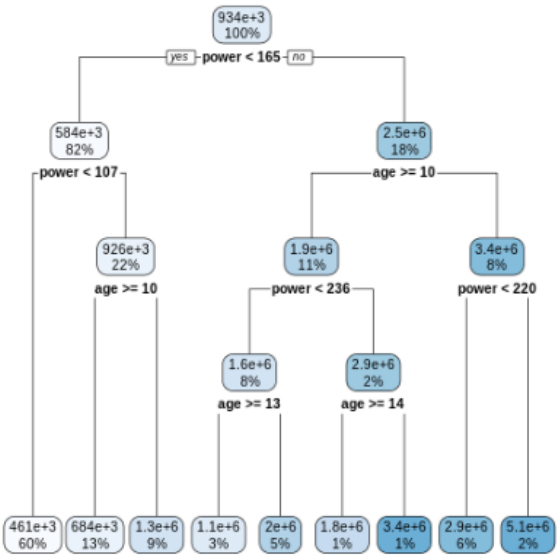
- Python: F-статистика: 3.1062, P-значение: 0.0477
- R: F-статистика: 2.52, P-значение: 0.0839

получили разные результаты - с уровнем доверия 0.95 Python гипотезу отвергает, R - принимает. Скорее всего, повлияло то, что выборки генерируются разными.

13. Подогнать регрессионные модели (в том числе, нелинейные) к данным, а также оценить качество подобной аппроксимации.

Будем с помощью всех числовых столбцов предсказывать цену на автомобиль. В качестве метрики качества использовалась r squared.

- **Линейная регрессия:** И на Python, и на R получаем r squared 0.69
- **Полиномиальная регрессия:** Используя вторую степень, на Python получаем v 0.81, на R 0.807 (примерно одинаково).
- Деревья решений: На Python удалось добиться r squared 0.88, на R из-за невозможности подобрать параметры вышло 0.76



Вывод: на моих данных лучшую точность демонстрируют деревья решений с правильно подобранными параметрами.

14.Выводы

По итогам работы был приобретен опыт анализа данных на Python и R, отмечены различия между статистическими критериями. Были сделаны выводы:

- О зависимостях между пробегом и типом топлива, ценой и типом топлива, пробегом и числом владельцев (графики);
- Применены критерий Граббса, тест Диксона, методы заполнения пропусков и различные статистические критерии к данным из нормальной и своей выборки, принята гипотеза о схожести распределений цен на марки премиум-сегмента Audi и BMW;
- Применены коэффициенты для поиска корреляционных взаимосвязей, обнаружена почти линейная зависимость между объемом двигателя и лошадиными силами на выборке автомобилей из города Мумбаи;
- Применены методы хи-квадрат, точного теста Фишера, теста МакНемара, Кохрана-Мантеля-Хензеля, принята гипотеза о схожести рынков в городах Мумбаи и Хайдарабад, для отношения цен по графику показано, что один из тестов неверно отработал на данных
- Построена таблица для поиска мультиколлинеарности данных, замечена высокая зависимость между ценой, объемом двигателя и лошадиными силами;
- Применен дисперсионный анализ;
- Обучены модели машинного обучения, получен высокий результат метрики точности.

В сравнении языки Python и R показывали в основном одинаковые результаты, но в нескольких случаях критерии, реализованные на Python выдавали заведомо неверный результат, что означает, что возможно выборка, сгенерированная на Python, была дальше от нормального распределения, чем для R. В целом Python из-за обилия библиотек легче для применения и, в особенности, построения графиков.

Литература

1. Брюс П., Брюс Э., Гедек П. Практическая статистика для специалистов Data Science. – 2-е изд., перераб. и доп. — СПб.: БХВ-Петербург, 2021. – 352 с.

2. Мастицкий С.Э., Шитиков В.К. Статистический анализ и визуализация данных с помощью R. – М.: ДМК Пресс, 2015. – 496 с.

3. Нильсен Э. Практический анализ временных рядов: прогнозирование со статистикой и машинное обучение. – СПб.: ООО «Диалектика», 2021. – 544 с.