Универзитет у Београду

Електротехнички факултет



**Модел за предикцију цена некретнина**

Извештај о домаћем задатку

Студент: Вања Обрадовић

Предмет: Софтверско инжењерство великих база података

Београд, фебруар 2023.

**Садржај**

[1. Увод 1](#_Toc127184641)

[2. Анализа података 2](#_Toc127184642)

[2.1 Скуп за тренирање 2](#_Toc127184643)

[2.2 Преглед улазних атрибута 3](#_Toc127184644)

[3. Нормализација података 5](#_Toc127184645)

[3.1 Нормализација података 5](#_Toc127184646)

[3.2 Резултати нормализације 6](#_Toc127184647)

[4. Избор модела 8](#_Toc127184648)

[4.1 Линеарна регресија 8](#_Toc127184649)

[4.2 Random Forest Regression 8](#_Toc127184650)

[4.3 Histogram Gradient Boosting Regression 8](#_Toc127184651)

[4.4 Multi-layer Perceptron Regression 9](#_Toc127184652)

[5. Закључак 10](#_Toc127184653)

# Увод

Циљ овог задатка је генерисање модела за предвиђање цена некретнина у Москви, кроз анализу одређених параметара саме некретнине и њене околине.

Скуп података који је дат у изазовз чини велики број некретнина (10.000). Свака некретнина је описана низом следећих параметара:

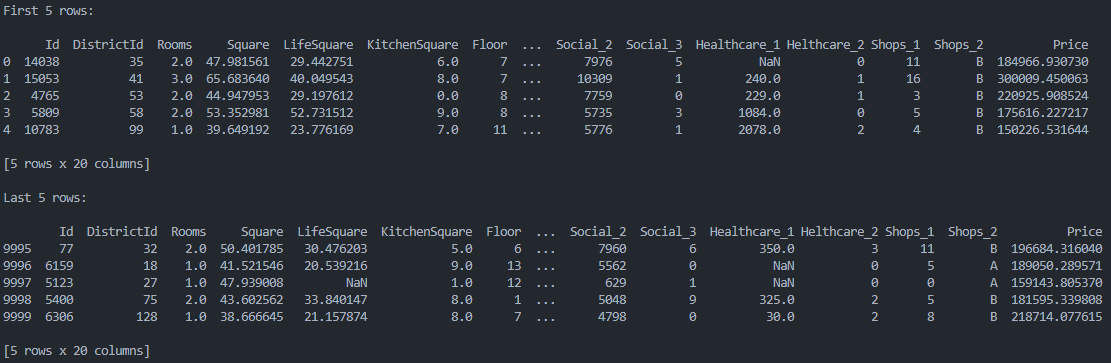
* **Id** – јединствени идентификатор некретнине
* **DistrictId** – јединствени идентификатор области
* **Rooms** – број соба
* **Square** – бруто површина
* **LifeSquare** – нето површина
* **KitchenSquare** – површина кухиње
* **Floor** – број спрата
* **HouseFloor** – број спратова у објекту
* **HouseYear** – година изграње објекта
* **Ecology\_1, Ecology\_2, Ecology\_3** – еколошки параметри околине
* **Social\_1, Social\_2, Social\_3** – друштвени параметри околине
* **Healthcare\_1, Heаlthcare\_2** – параметри околине везани за здравствену заштиту
* **Shops\_1, Shops\_2** - параметри околине везани за куповину и тржне центре
* **Price** – цена некретнине

За имплементацију модела коришћен је програмски језик *Python.*

# Анализа података

У овом поглављу ће бити представљена детаљна анализа улазних параметара датих у фајлу *train.csv* и како исти излаз тј. цену некретнине.

## Скуп за тренирање

На слици испод може се видети првих и последњих 5 редова из датог скупа за тренирање модела.

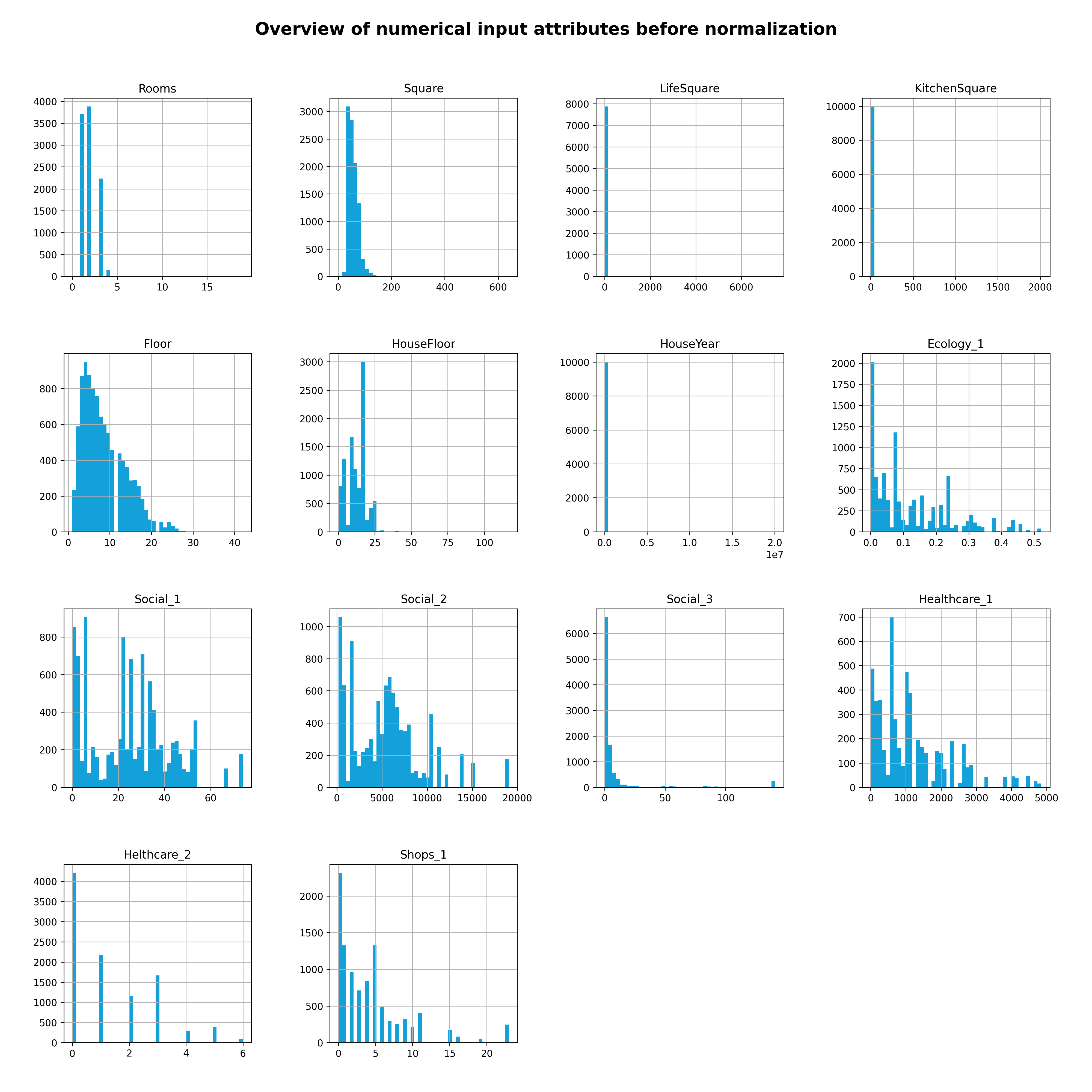
A picture containing text, plaque

Description automatically generatedText

Description automatically generatedНа сликама у наставку могу се видети опште информације о колонама, број недостајућих редова по колонама и генералне статистичке информације о колонама.

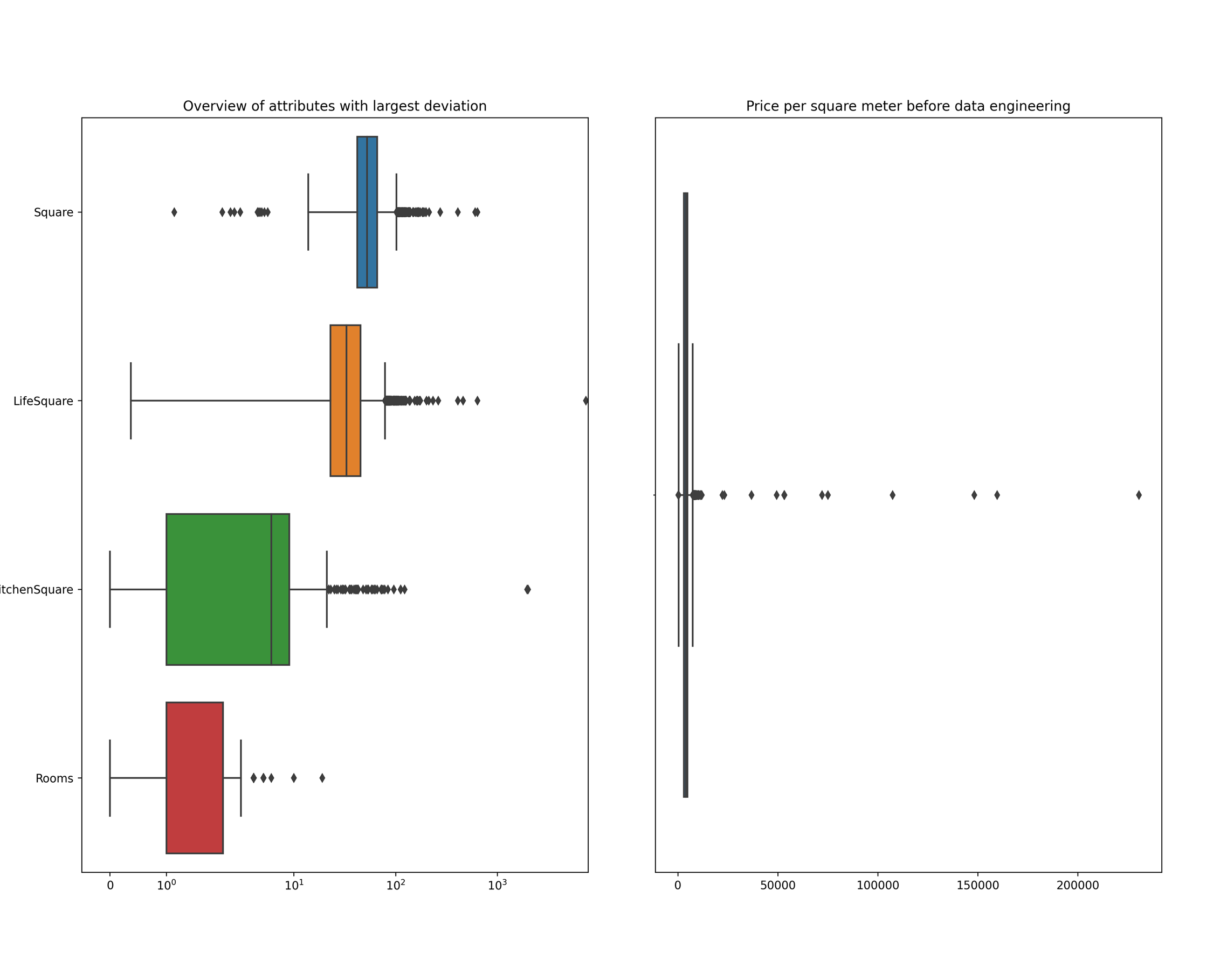
Graphical user interface, text

Description automatically generatedСкуп за тренирање има 19 колона и 10000 редова. Може се приметити да колоне ***LifeSquare*** и ***Healthcare\_1*** имају прилично велике празнине. Стога се ове колоне морају попунити на одговарајући начин, али тако да немају негативан утицај на исход предвиђања.

Обзиром да се овде ради о регресивном проблему, јер модел треба да предвиђа реалан број (цену некретнине), у наставку ће бити детаљно размотрене улазне колоне од значаја.

## Преглед улазних атрибута

На графицима изнад се може приметити да колоне ***LifeSquare***, ***KitchenSquare*** и ***HouseYear*** садрже вредности које значајно одступају од нормалних и вероватно су последица грешке. Додатно, колоне ***Rooms*** и ***Square*** имају релативно високе вредности, које ће као и горе поменуте грешке морати да се адекватно скалирају или уколне.

На графицима у наставку је приказана расподела вредности одређених горе поменутих колона и једне фиктивне колоне ***Price per square meter***. На овим графицима се могу још јасније видети екстремуми по колонама.

На основу датих графика може се закључити да подаци нису савршени, садрже одређни број вредности које су екстремне и немају смисла у контексту у којем их разматрамо. Те вредности ће, као што је већ речено морати да се санирају на адекватан начин. Поред екстремума, вредности колона које се односе на друштвене параметре и параметре везане за куповину је тешко интерпретирати, стога ће исте остати у изворном облику.

# Нормализација података

У овом поглављу ће бити представљени поступак нормализације података и резултати истог.

## Text Description automatically generatedНормализација података

Text

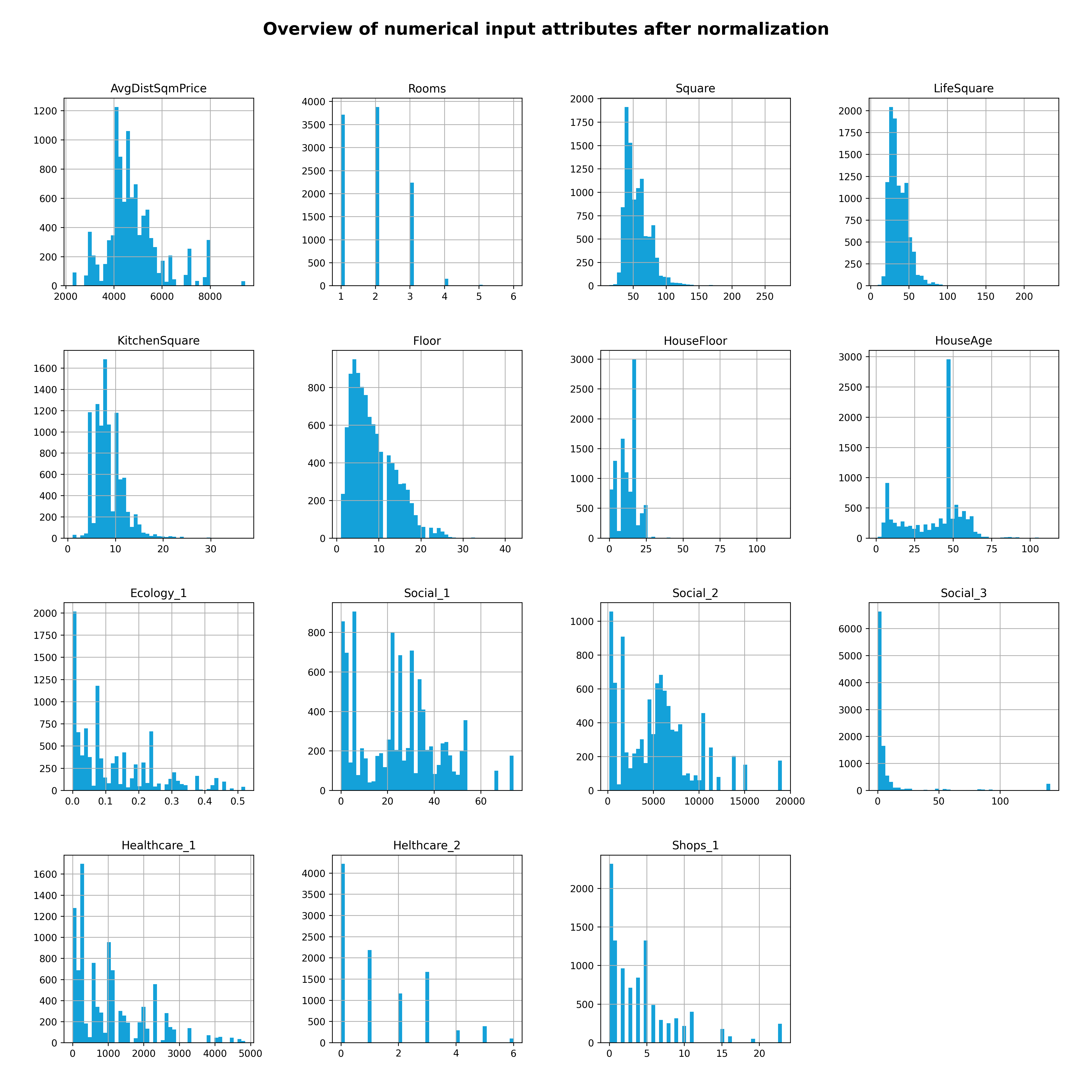
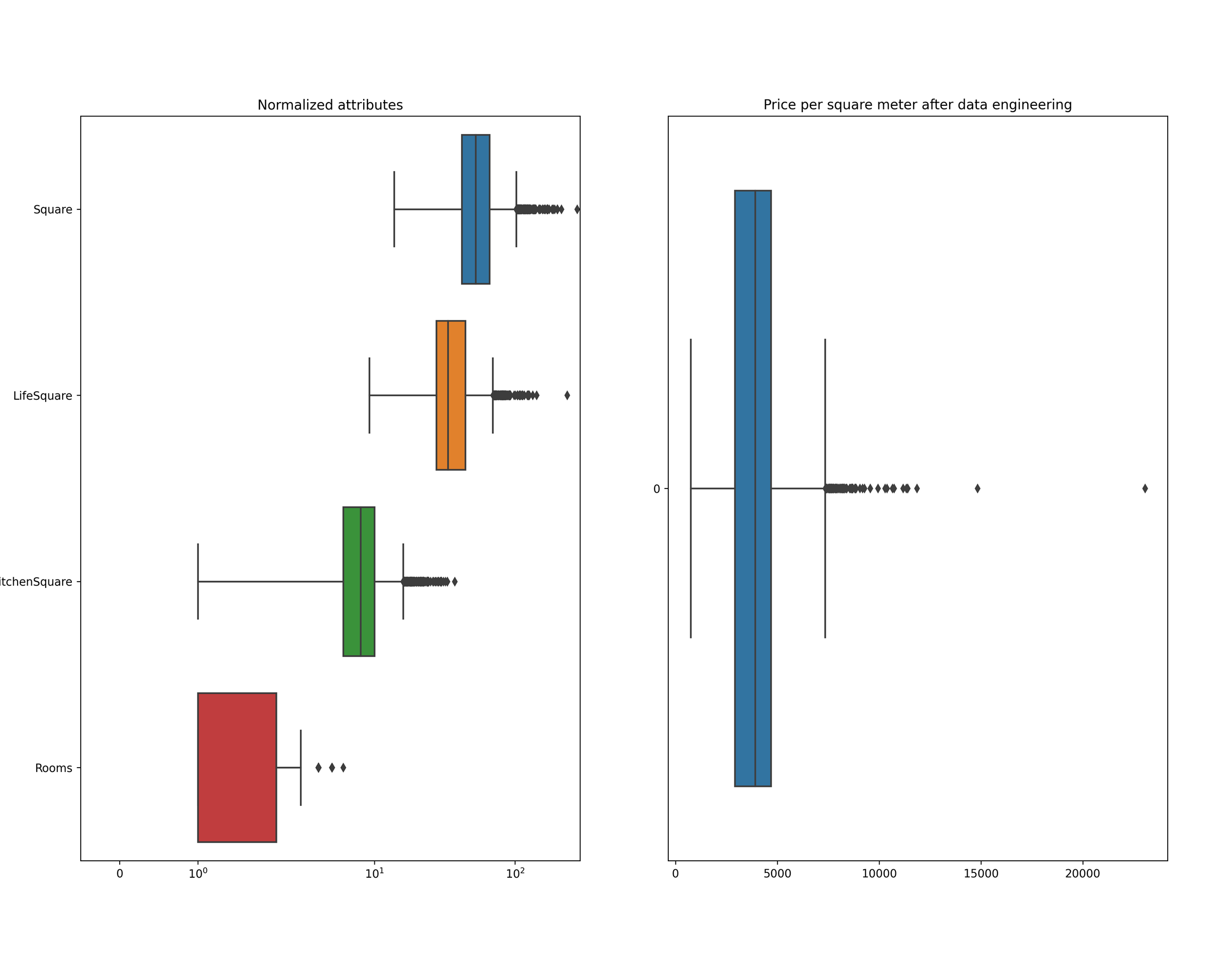
Description automatically generatedОбзиром на нелогичности у бруто квадратури, прво је извршено то скалирање на горе приказан начин. Након тога постављен је минималан број соба, док су редови са превише соба у односу на нето квадратуру постављени на просечан број соба. Затим је дата колона ***HouseYear***, замењена са колоном ***HouseAge***, да би се боље представила старост објекта.

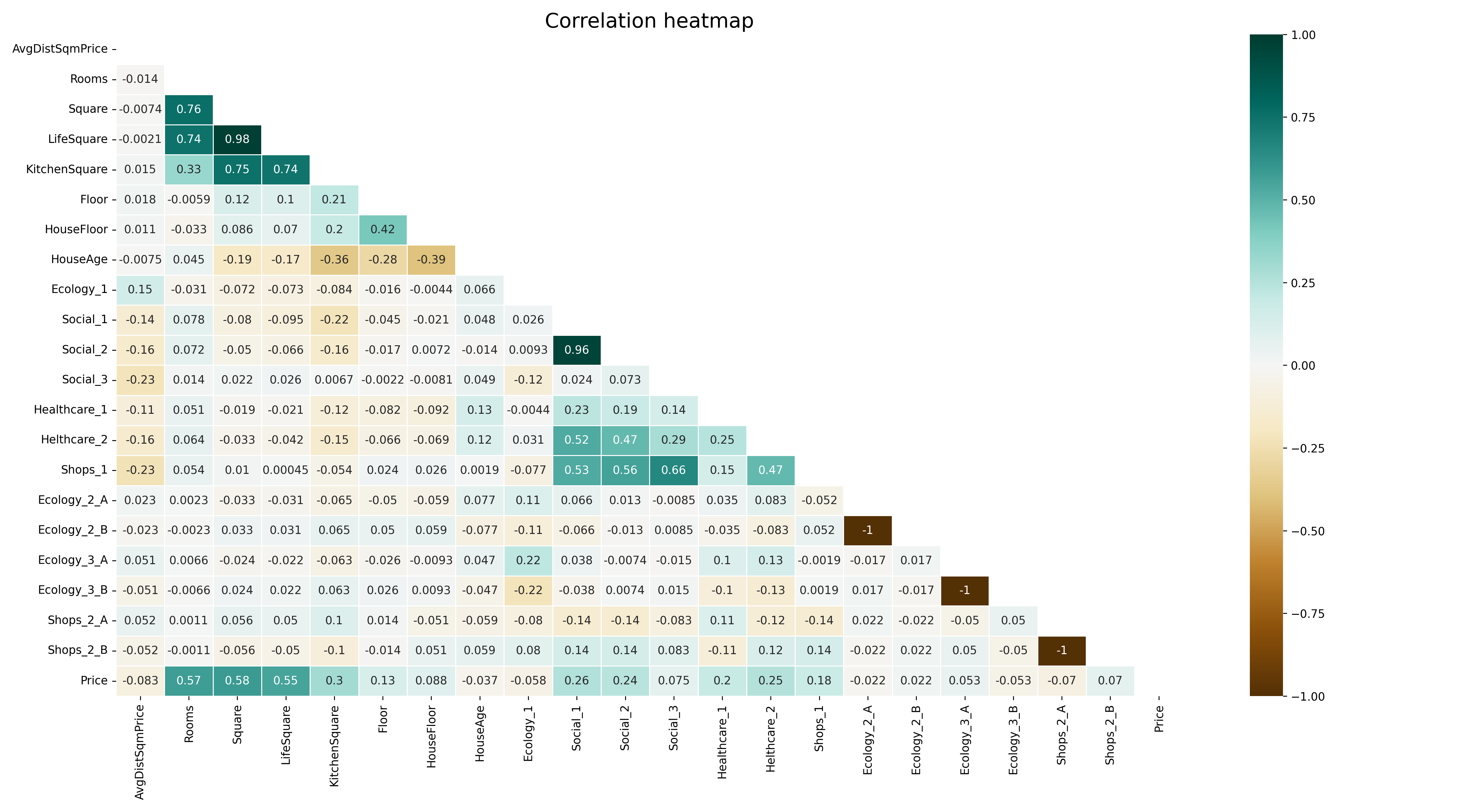
Text

Description automatically generatedНа горњој слици је приказан начин на који је извршено попуњавање недостајућих вредности у колонама ***LifeSquare*** и ***Healthcare\_1***,а затим и скалирање екстремума колона ***LifeSquare*** и ***KitchenSquare*** просечним уделом у бруто тј. нето квадратури респективно.

На крају је додата колона ***AvgDistSqmPrice*** која представља просечну цену некретнина по области.

## Резултати нормализације

Са графика испод можемо видети да се ситуација поправила и да су сад подаци довољно добри да се започне процес тренирања модела.

На крају, од користи је и анализирати топлотну мапу корелација између колона. Таква мапа је приказана испод:

Топлотна мапа нам даје могућност прегледа зависности између колона. Када је у питању дати, сређени скуп података можемо приметити да цена некретнине највише зависи од броја соба и квадратуре, док је нешто слабија зависност од друштвених параметара и параметара везнаих за здравствену заштиту.

# Избор модела

У овом поглављу ће бити разматрани различити регресивни модели и њихова прецизност када је у питању дати проблем.

За поређење су одабрани Линеарна Регресија, *Random Forest Regression,* *Histogram Gradient Boosting Regression* и *Multi-layer Perceptron regression* као једни од најзаступљенијих регресивних алгоритама. За поделу података на део за тренирање и део за тестирање се користила *train\_test\_split()* метода из *sklearn* библиотеке подешена тако да се 80% података користи за тренирање, а остатак за тестирање. Како би регресивни модели функционисали, било је потребно претворити све текстуалне вредности у јединствене идентификаторе. Ово је постигнуто уз помоћ *get\_dummies()* библиотечке функције.

## Линеарна регресија

Линеарна регресија је једна од најједноставнијих техника статистичког моделирања односа између зависне променљиве и једне или више независних промељивих. Циљ линеарне регресије је да се пронађе линија која најбоље описује зависност промељивих тј. која може предвидети вредност зависне промељиве на основу вредности независних променљивих. Једноставна је за имплементацију, али није погодна за функције зависности које су сложене и нелинеарне. Стога, треба напоменути да се линеарна регресија базира на идеји да постоји линеарна зависност између промнљивих и да су грешке нормално распоређене.

У случају линеарне регресије за две или више независних промељивих, једначина праве гласи:

y = β0 + β1x1 + β2x2 + ... + βnxn

где су x1, x2, ..., xn независне промељиве, β0 пресек са y осом, а β1, β2, ..., βn одговарајући коефицијенти. Модел проналази линију која најбоље одговара (вредности коефицијената βi ) тако што минимизира збир квадрата разлика између предвиђених и стварних вредности.

## Random Forest Regression

*Random Forest* је “ансамбл” метода регресије која генерише предвиђања комбинацијом излаза више стабала одлучивања. Свако стабло које чини “ансамбл” тј. *Random Forest* се креира на основу другачијег скупа података и за свако се бира другачији поскуп карактеристика за раздвајање на чворовима. Процес раздвајања се рекурзивно понавља док стабло не достигне максималну дубину или постављени лимит. Свако од стабала из шуме потом даје своје индивидуално предвиђање. Коначни резултат се добија тако што се сва ова предвиђања затим усредњују, што овај алгоритам чини прецизнијим и бољим од једног стабла одлучивања.

Овај регресивни модел је флексибилан и може радити са нелинеарним односима између улазних карактеристика и излазне променљиве. Принцип рада стабла одлучивања се најбоље описује као гомила if-else условa. Слабо је осетљив на екстремуме (*енг. outliers*) у подацима и може се користити у случају великог број улазних карактеристика.

## Histogram Gradient Boosting Regression

*Histogram Gradient Boosting Regression* је регресивни *gradient boosting* алгоритам. Представља варијацију основног *gradient boosting* алгоритма, с циљем да се убрза процес тренирања модела и побољша прецизност. У следећих пар тачака биће описано функционисање овог алгоритма:

1. **“*Binning*” континуалних вредности** – овај алгоритам врши “*binning*” тј. дискретизацију континуалних вредности у одређени број фиксних вредности, како би се смањила комплексност модела
2. **Креирање стабла одлучивања** – овај алгоритам користи стабла одлучивања као “слабе ученике” за предвиђање
3. ***“Gradient boosting”***– прави се ансамбл стабала одлучивања корак по корак, где се свако стабко уклапа у негативни градијент функције губитка*.* Идеја је да се алгоритам фокусира на грешке претходних стабала, које ће потом исправити нова стабла. Нова стабла се генеришу све док се не достигне задати критеријум (максималан број стабала или минимално побољшање функције губитка).
4. **Предвиђање** – коначно предвиђање је пондерисана просек предвиђања свих стабала

Histogram Gradient Boosting Regression је алгоритам који је ефикасан, има добре перформансе и може да подржи рад са великим бројем улазних карактеристика.

## Multi-layer Perceptron Regression

*Multi-layer Perceptron Regression* је неурална мрежа која се користи за регресивне моделе машинског учења. Представља тип неуралне мреже која се састоји од више слојева неурона (чворова).

Принцип рада:

1. **Улазни слој** – прихвата улазне карактеристике података, тако што сваки чвор представља једну карактеристику
2. **Скривени слојеви** – састоје се од више неурона тј. Чворова. Сваки чвор узима пондерисану суму излаза претходног слоја, на основу које помоћу нелинеарне активационе функције израчунава резултат који се прослеђује следећем слоју
3. **Излазни слој** – излазни слој се састоји од једног чвора који представља резултат предвиђања
4. **Оптимизација тежина** – тежине веза између чворова се током тренирања оптимизују тако да се минимизује средња квадратна грешка (стандардна девијација) између стварне и предвиђене вредности.

*Multi-layer Perceptron Regression* је флексибилан и јак регресиван модел који може да подржи нелинеарне односе између улазних карактеристика и излазне променљиве. Међутим, захтева велику количину података за тренирање и јако је осетљив на избор хиперпараметара, као што су:

* Број скривених слојева
* Број неурона у слојевима
* Активациона функција

# Закључак

Text

Description automatically generatedНа следећој слици се може видети поређење резултата тестираних алгоритама и њихових тјунираних варијанти по параметрима средње апсолутне грешке, корена средње квадратне грешке као и коефицијент детерминације R2:

Ови резултати минимално варирају са сваком итерацијом тренирања модела. Посматрајући резултате може се закључити да најбољи резултат даје *Histogram Gradient Boosting Regressor*, а одмах иза њега и *Random Forest Regressоr*. Коефицијент детерминације од ~0.76 представља задовољавајући резултат обзиром на присуство колона које представљају параметре околине, а чије вредности нису могле да се интерпретирају, како би се адекватно искористиле за тренирање модела и имајући у виду да је тренутно најбољи резултат на изазову 0.77.

*Multi-layer Perceptron* *Regressor* у овом случају није остварио добре резултате због ограничене процесорске моћи, обзиром да проналажење најбољих хиперпараметара за овај модел захтева велику процесорску моћ и време.