Учебен център Нет Ит

Курс по програмиране на Python

Курсов проект

на

Пенка Ковачева

Тема: „Проектиране и разработване на интерфейс за предвиждане на цени на имоти на база основни характеристики на имота„

София 2020

Съдържание

[Увод 2](#_Toc30494086)

[1 Въведение 2](#_Toc30494087)

[2 Проектиране 2](#_Toc30494088)

[3 Разработване 2](#_Toc30494089)

[4 Експерименти и тестове 2](#_Toc30494090)

[Заключение 2](#_Toc30494091)

[Източници 2](#_Toc30494092)

[Приложение 2](#_Toc30494093)

# Увод

Целта на проекта е да предостави възможност на потребители на уеб приложение да получават прогнозна цена на къща, след като въведат нейни основни характеристики. Уеб приложението, което се цели да се създаде е базирано на характериски и Machine Learning модел, който да ги използва. Стремежа интерфейса да е максимално опростен спрямо крайния потребител и да дава бързо информация за цената на имота.

Важно нещо при създаването на този проект е, че той трябва да положи основи за по-сложно структуриране на изчисленията, които дават финалната прогноза на цената на имота. Тук се има впредвид да се използват няколко модела за изчисление, които да се съпоставят като ефикасност, за да бъде крайната информация към потребителя максимално вероятна.

# Въведение

За въвеждане на модела ще използваме данни за имоти в окръг Кинг, щата Вашингтон. Те съдържат дузина на брой характериски какво и крайна цена за всяка една къща. Избираме тои тип данни, защото са лесно достъпни и съдържат всички което ни е необходимо на този етап.

Ще разгледаме стойностите и връзките на отделните характеристики за да изберем коя/кои ще ни бъдат необходими като входни данни за прогноза на цената. Към момента не целим най-точна прогноза на база даден модел, а създаване на цялостна услуга.

Предстоящото изследване се провежда спрямо обособен регион и това трябва да е максимално ясно на крайния потребител, за да не му се подава двусмислена информация. За целта това ще е явно видимо и в самото приложение.

# Проектиране

В тази разработна крайният продук е опростено уеб приложение, което има за въвеждане само конкретна/и характеристика/и. Всичко останало е визуализация на прогнозата и графично изобразяване на примерните данни.

Основните стъпки са: зареждане на данни, изследване на масива от данни, изчистване на данните за да се обработват по-лесно, създаване на модел, създаване на APIs към модела, създаване на интерфейса, тестване на финалния продукт.

## Зареждане на данни

Данните които ползваме са свободно достъпни, но ние ще използваме учебни, които са за годините 2014 и 2015. Има готови такива във .csv формат ([линк към файла](https://s3-api.us-geo.objectstorage.softlayer.net/cf-courses-data/CognitiveClass/DA0101EN/coursera/project/kc_house_data_NaN.csv)), но трябва да се отбележи, че не са изчистени и подредени за анализ.

## Изследване на масива от данни

След като имаме зареден файл, започваме да разглеждаме неговото съдържание в табличен вид. Необходимо е да се провери какви колони съдържа таблицата, каква е дистрибуцията в тях, има ли нулеви стойности, както и типа на данните във всяка една колона. Ако има разминаване на типа данни е хубаво да се коригира, също така да се съобразят мерните единици които са използвани. Често в такива масиви има и много повече информация от колкото ни е необходима за нашите цели, хубаво е тя да бъде изчистена, за да улесни работата след това. Трябва да видим и линейната връзка на всяка една характеристика от масива спрямо цената, но това е добре да се направи след като сме сигурни, че данните ни са в правилен формат за обработка.

## Изчистване на данните

Като за начало е необходимо да се изтрият всички колони които нямат отношение към работата ни по-нататък.

На база наблюденията ни трябва да вземем решение как да се справим с нулевите стойности във файла, като тук опциите отново са няколко: да изтрием редовете с нулеви стойности, да заменим стойностите със средна, минимална или максимална, за конкретната колона. Тъй като масива ни от данни не е много голям е удачно да използваме замяна на Null стойност със средната стойност за колоната.

Важен аспект от изчистването на данните е правилния тип данни на всяка една колона. Ако забележим несъответсвие трябва да коригираме типа на записите в тази колона.

За удобство можем да сменим мерните иденици и да конвертираме стойностите на колоните спрямо тях. Това не е задължителна стъпка, но много полезна от гледна точна на използване от крайния потребител.

## Създаване на модел

След като имаме всичко необходимо в табличен вид започваме и с „обучаване“ на модел, който да предвижда цената на къща. Модела, който ще използваме е линейна регресия с един параметър. Това означава, че е необходимо да открием коя характеристика на имот има най-висока стойност на корелация спрямо цената. Още на етап изследване на данни ние сме получили информация за тези стойности и лесно можем да изберем.

## Създаване на API

За да можем лесно да достъпваме модела ни за предвиждане на цената е необходимо да създадем API. Това ще ни осигури лесна имплементация на модела в последвалото приложение. Важно е също да изполваме вариант за сериализиране и десериализиране на модела, с оглед на това, че е неефективно модела да се „обучава“ при всяка една справка на потребителя.

## Създаване на интерфейс:

Най-лесно достъпно за голям брой потребители е уеб приложението. В случая създаваме приложение с една страница в което потребителя въвежда стойност и на базата на тази стойност получава съобщение за вероятна цена на къща. При създаването на основната страница ще добавим и интерактивна графика обобщаваща информация от масива данни на базата на който предвиждаме цената.

## Тестване

Тестването на функционалността се прави през целия етап на разработка. Започваме с тест след обучаване на модела, за да видим какви стойности предвижда. След създаването на интерфейса отново пробваме с различни стойности за да видим дали програмата се държи по правилния начин.

# Разработване

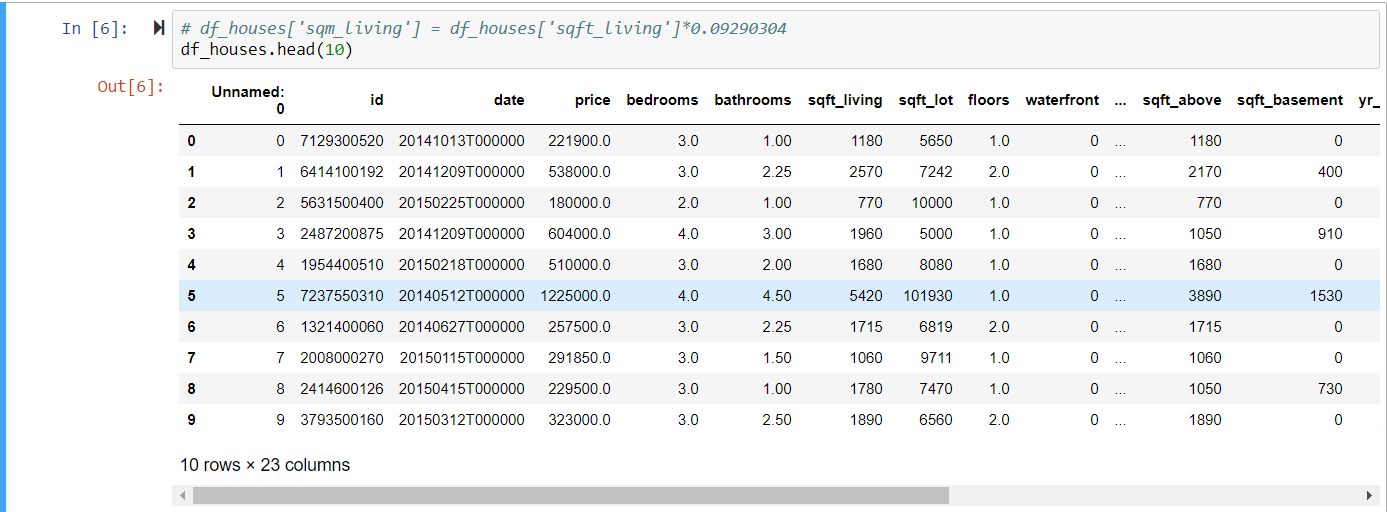
Разработка - от 2 до 10 страници

Конкретизират се технологиите. Класове, методи и др.

Използвани софтуерни продукти, библиотеки, frameworks и API-та, инсталиране, конфигуриране, функционалности, взаимодействие.

Първоначално започваме работа по проекта в Jupyter Notebook, където удобно можем да документираме стъпките при изледване на данните и тяхната промяна, както и създаване и оценяване на модела. Библиотеките които ще ползваме при изследване на данните и модела са pandas, sklearn, numpy, seaborn.

Масива ще бъде преобразуван в т.нар. Pandas DataFrame, защото е много удобен вариант за осъществяване на анализи. Това става чрез метода read.csv(). В резултат получаваме следната таблица:



След като имаме таблицата в началния й вариант използваме отново няколко метода от библиотеката pandas с опознавателна цел. Това са : describe(), columns, info(), isnull().

На базата на тези методи виждаме, че e необходимо да изтрием няколко колони, да променим типа на данните в една колона и да заменим Null стойностите в още две колони. Както по-горе описах, ще заменим със средни стойности за колоната. На този етап методите които ползваме са: drop(), to\_datetime(), replace(), mean().

Има изчистени данни, както и информация за линейната връзка между различните характеристики и цената. На базата на това се вижда че най-свързан параметър е площта на имота. Създаваме нова колона, която да съдържа дании за общата площ в квадратни метри.

След това използваме sklearn.LinearRagression() за да обучим модела ни.

С помощта на seaborn изобразяваме графичното съпоставяне на цената, предвидена от модела и цената, която е реално в масива от данни.

За да се продължа с работата по проекта използвам средата PyCharm, на която ще реализираме кода, за създаване на уеб приложението.

Стартираме със пресмянате на модела в файла “LinearRegression.py”, като се използват гореспоменатите библиотеки заедно с joblib, който е част от sklearn.externals. Последното служи за сериализиране на модела и създаване на файловете „LR\_ser.pkl” и “training\_data.pkl”.

Следващата библиотека която е необходима е Flask и тя се използва във съдаването на API на модела.

За финал при създаването на приложението ще използваме библиотеката Dash, за създаване на приложението и библиотеката plotly.graph\_objs, за създаване на интерактивната графика.

# Експерименти и тестове

Тестовете по време на разработката бяха с различни стойности, които да се въведат на мястото на стойноста за площ. Приложението е рестриктирано до въвеждането само на числа, което също е проверено.

Направени са тестове с още два модела, като там са използвани повече от една характеристика(Multiple Linear Regression, Polynomial Regression), като дават по висока точност спрямо използвания в проекта.

# Заключение

Приложението дава бърза справка са предполагаема цена на имот само след въвеждането на площ, което е много удобно, но и не и напълно точно. В бъдеща разработка би могло да се заложи на обновяващи се данни за обучение и въвеждането на няколко модела. Увеличаването с още 1 или 2 параметъра също учвеличават точността на продукта

Тези стъпки биха дали надеждност, но и да усложнят крайния потребител. За целта е удачна разработка на функционално ниво, без съществени промени в интерфейса.

# Източници

* <https://dash.plotly.com/>
* <https://pandas.pydata.org/>
* <https://joblib.readthedocs.io/en/latest/>
* <https://seaborn.pydata.org/>