Федеральное государственное образовательное бюджетное
учреждение высшего образования

«Финансовый университет при Правительстве Российской Федерации»

Московский филиал Финуниверситета

Кафедра «Информационные технологии и анализ больших данных»

ДОМАШНЕЕ ТВОРЧЕСКОЕ ЗАДАНИЕ

по дисциплине «_____Эконометрика_____»

на тему: «__Эконометрический_анализ_рынка_жилой_недвижимости__»

Выполнила студентка __2__ курса,

группы ____ПМ22-3_____,

формы обучения__очная_____

___Перминова_Мария_Александровна__

Проверил преподаватель:

___Михайлолва_Светлана_Сергеевна_____

____профессор ДАДиМО_____

Москва 2023 г.

Содержание:

1) Актуальность, цель, задачи, гипотезы работы

Актуальность: анализ рынка жилой недвижимости и дальнейшее построение моделей для прогноза стоимости важная задача, в качественной реализации которой заинтересованы многие группы лиц: риэлторы, люди, планирующие приобретение жилья, инвесторы, экономисты и пр.

Цель работы: Создать программу, которая автоматизирует процесс построения спецификаций эконометрических моделей. С помощью полученного алгоритма на основе данных по регионам России за 2018-2021 года создать для каждого из регионов качественную модель.

Задачи:

1) Найти готовый датасет, в котором будут представлены данные о российском рынке недвижимости (мы будем рассматривать только квартиры).
2) Предобработать данные, посмотреть статистики, сделать выводы
3) Написать алгоритм автоматизированного создания спецификации модели. Данный алгоритм должен включать в себя: отбор факторных переменных, проверку на значимость коэффициентов и уравнения в целом, проверку на наличие гетероскедастичности, автокорреляции и мультиколлинеарности.
4) Выбрать из датафрейма несколько наблюдений, сделать для них предсказание, рассчитать ошибку, сделать вывод о качестве моделей.

Гипотеза: у каждого региона в силу экономических, географических и прочих особенностей цены на квартиры формируются по-разному. Определенные значения факторов могут сильно увеличивать стоимость квартиры.

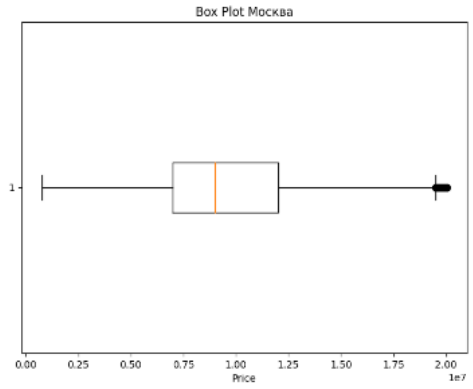
2) Данные

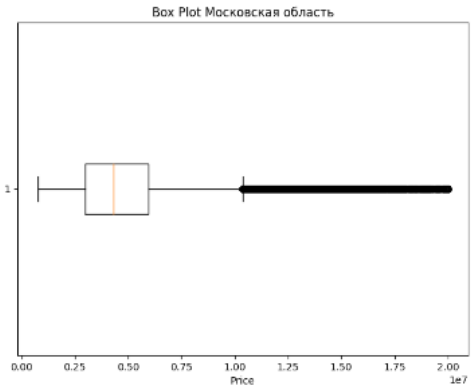
В качестве данных для работы были выбраны данные с платформы kaggle о российской недвижимости по 84 регионам за 2014-2021 года.
https://www.kaggle.com/datasets/mrdaniilak/russia-real-estate-20182021
Данные имеют следующие поля:

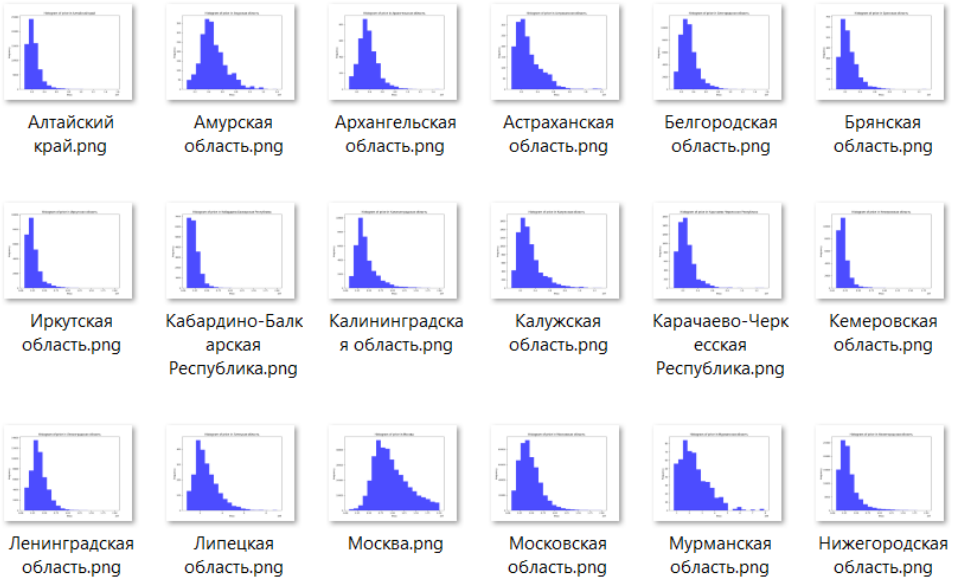
- `date` - дата публикации объявления;
- `time` - время публикаци объявления;
- `geo_lat` - значение координаты (широта);
- `geo_lo` - значение координаты (долгота);
- `region` - код региона РФ;
- `building_type` - Тип здания. 0 - Прочее. 1 - Панельный дом. 2 - Монолит. 3 - Кирпичный. 4 - Блочный. 5 - Деревянный;
- `object_type` - Тип квартиры. 1 - Вторичное жилье; 11 - Новая квартира в новостройке;
- `level` - Этаж, на котором находится квартира;
- `levels` - Количество этажей;
- `rooms` - Количество жилых комнат. Если значение -1 - это значит, что квартира является "студией";
- `area` - Совокупная площадь квартиры;
- `kitchen_area` - Площадь кухни;
- `price` - Цена в рублях РФ.

Была проведена предобработка данных. Были убраны выбросы, добавлен столбец region_name, путём получения с помощью сервиса (https://www.geonames.org/) по значениям широты и высоты названий регионов России. Среди регионов было решено оставить те, по которым есть хотя бы 500 наблюдений, т.к. например для Сахалинской области в датасете было только 8 наблюдений, что крайне мало.

Также данные были проверены на наличие пустых значений, таковых не оказалось. После предобработки для каждого из регионов были построены гистограммы распределения цен и графики box plot. Данные графики уже на этапе анализа данных могут многое сказать о рынке недвижимости того или иного региона.
Например для Москвы средняя стоимость квартиры примерно 8 млн. руб. В то время как для Московской области средняя цена приблизительно 5 млн. руб.

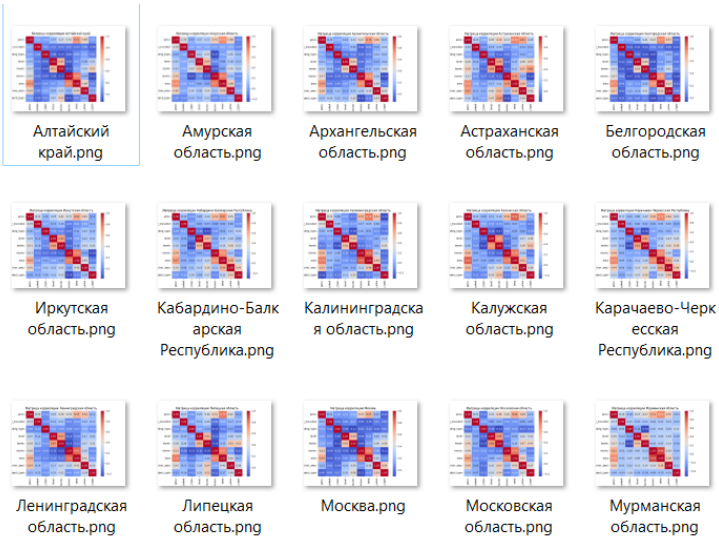






А если посмотреть на графики распределения цен на квартиры, будет видно, что многие из них имеют правостороннюю асимметрию.

Также были построены матрицы корреляции и матрицы частных коэффициентов корреляции для удобства ручного анализа.



3) Алгоритм автоматизированного создания спецификации модели

a) Автоматизированный отбор значимых факторов с помощью метода включения

В качестве подхода к отбору факторов для модели я использовала пошаговый метод.
1) На первом шаге мы строим модель парной регрессии с фактором, коэффициент корреляции у которого с целевой переменной максимальный.
2.1) Далее на каждом шаге цикла выбирается тот фактор, добавление которого в модель увеличивает коэффициент детерминации наибольшим образом.
2.2) Важно, что если рассматриваемый фактор имеет сильную корреляционную связь, (более 0.8) с другими, уже отобранными факторами, такой регрессор в модель не включаем.
3) В случае малого количества факторных переменных, мы можем менять значение переменной level, которая отвечает за минимальный прирост коэффициента детерминации при добавлении нового фактора в модель
Как результат мы получаем датафрейм, где каждому региону соответствует список факторных переменных.

	region_name	factors
0	Санкт-Петербург	['area', 'date_encoded', 'kitchen_area', 'room...
1	Московская область	['area', 'levels', 'date_encoded', 'kitchen_ar...
2	Нижегородская область	['area', 'rooms', 'date_encoded', 'levels', 'o...
3	Краснодарский край	['area', 'date_encoded', 'building_type', 'roo...
4	Москва	['area', 'date_encoded', 'building_type', 'obj...

b) Подсчёт основных статистических показателей для всех моделей: статистики Стьюдента для коэффициентов регрессии, тест Фишера для всей модели

Далее для каждого региона строим модели множественной линейной регрессии с помощью пакета statsmodels, рассчитываем t статистики Стьюдента для каждого из факторов. Сохраняем списки статистически значимых факторов для каждого из регионов. Далее опять строим модели множественной линейной регрессии, но уже беря только статистически значимые факторы, рассчитываем F статистику Фишера для всего уравнения в целом. Результат данного этапа выглядит следующим образом:

	region_name	factors	t_test	factors_significant	F_test
0	Санкт-Петербург	[area, date_encoded, kitchen_area, rooms, leve...	[True, True, True, True, True, True, True, Tru...	[area, date_encoded, kitchen_area, rooms, leve...	True
1	Московская область	[area, levels, date_encoded, kitchen_area, obj...	[True, True, True, True, True, True]	[area, levels, date_encoded, kitchen_area, obj...	True
2	Нижегородская область	[area, rooms, date_encoded, levels, object_typ...	[True, True, True, True, True, True, True]	[area, rooms, date_encoded, levels, object_typ...	True

c) Проверка на наличие гетероскедастичности

Для проверки на наличие гетероскедастичности, было решено использовать графический метод (Q-Q plot), а также тест Шапиро-Уилка. Данный тест хорошо себя показывает как на малых, так и на больших выборках, в случае наших данных. Тест не выполняется ни для одного из регионов. Это показалось подозрительным, поэтому я провела ещё тест Хетера-Бройша-Пагана, результат тот же. Наличие гетероскедастичности я могу объяснить тем, что в наших данных присутствуют наблюдения для разных квартир, и, условно, нашу выборку можно разбить на более мелкие подвыборки. Посмотрим как это скажется на предсказательной способности модели.

d) Проверка на наличие автокорреляции

В качестве теста на автокорреляцию был выбран тест Дарбина-Уотсона, все предпосылки для его применения выполняются: гетероскедастичность отсутствует, нет ошибок в спецификации моделей. Все регионы прошли тест, автокорреляция отсутствует.

e) Проверка на наличие мультиколлинеарности

Учитывая, что при отборе факторов для каждой модели при прямом проходе мы добавляли регрессор в модель только в том случае, если он не коррелировал с остальными, мы сильно снизили вероятность возникновения мультиколлинеарности. Будем использовать VIF тест. Высокий VIF для одного из регрессоров означает, что этот регрессор сильно коррелирован с другими переменными в модели. Факторы с высоким значением VIF мы убираем.

Также в самом конце добавим в нашу единую таблицу столбец с коэффициентами регрессии. В результате мы получили таблицу, в которой содержится вся необходимая информация для построения моделей.

4) Оценка качества модели

Выберем для каждого региона рандомные 300 строк, сделаем предсказания и оценим качество модели с помощью метрики MAPE.

Средняя ошибка модели 0.25, что является хорошим результатом.

```
np.mean(mape_list)
```

```
0.25070693056413246
```

Вот значения MAPE для первых 5 регионов:

```
mape_list[:5]
```

```
[0.286043550959676,
 0.3326830012276756,
 0.3776038953490468,
 0.3256199459970478,
 0.22828061419768422]
```

Минимальные и максимальные значения ошибки. В наилучшем случае модель ошибается в 10% наблюдений, в наихудшем в 40%

```
min(mape_list), max(mape_list)
```

```
(0.12572382926590125, 0.4027298009514719)
```

5) Заключение

Вывод: как результат мы получили таблицу с помощью которой можно делать предсказания о стоимости квартиры на основе её характеристик. (см файл df_factors2.csv в папке решения)

	region_name	factors	t_test	factors_significant	F_test	shapiro	het_breus	durbin_w	vif	add_const	coefs	mape
2	Санкт-Петербург	['area', 'date_encoded',	[True, True, True, True, T	['area', 'date_encoded', 'kit	True	False	False	True	[False, True, True, True, True, True	False	[120949.48228507 5 -73987.74700564 -23	0.286043550959676
3	Московская обла	['area', 'levels', 'date_en	[True, True, True, True, T	['area', 'levels', 'date_enco	True	False	False	True	[False, True, True, True, True, True	False	[55639.81406445 982 -37121.56648809]	0.3326830012276756
4	Нижегородская г	['area', 'rooms', 'date_en	[True, True, True, True, T	['area', 'rooms', 'date_enco	True	False	False	True	[False, True, True, True, True, True	False	[94058.83850506 -87 -49856.55266906 -28	0.3776038953490468
5	Краснодарский к	['area', 'date_encoded',	[True, True, True, True, T	['area', 'date_encoded', 'bu	True	False	False	True	[False, True, True, True, True, True	False	[72855.05936403 27 -50385.39941305 11	0.3256199459970478
6	Москва	['area', 'date_encoded',	[True, True, True, True, T	['area', 'date_encoded', 'bu	True	False	False	True	[False, True, True, True, True, True	False	[129367.95666714 8 51024.99739393 19	0.22828061419768422
7	Самарская обла	['area', 'levels', 'object_	[True, True, True, True, T	['area', 'levels', 'object_typ	True	False	False	True	[False, True, True, True, True, True	False	[63097.19444804 44 2357.54331551 13	0.24763887753204378
8	Республика Тат	['area', 'levels', 'rooms',	[True, True, True, True, T	['area', 'levels', 'rooms', 'd	True	False	False	True	[False, True, True, True, True, True	False	[72394.74901543 40 28454.10529208 -62	0.27424820083367135
9	Ставропольский	['area', 'date_encoded',	[True, True, True, True, T	['area', 'date_encoded', 'kit	True	False	False	True	[False, True, True, True, True, True	False	[46782.17075452 18 2332.45041344 -12	0.213770799953368
10	Республика Баш	['area', 'levels', 'date_en	[True, True, True, True, T	['area', 'levels', 'date_enco	True	False	False	True	[False, True, True, True, True, True	False	[55011.98131985 79 -40011.78193451 -1	0.3190631669622959
11	Свердловская о	['area', 'levels', 'rooms',	[True, True, True, True, T	['area', 'levels', 'rooms', 'd	True	False	False	True	[False, True, True, True, True, True	False	[72997.97473136 40 18382.32329797 -31	0.34760135428039907
12	Ростовская обла	['area', 'levels', 'object_	[True, True, True, True, T	['area', 'levels', 'object_typ	True	False	False	True	[False, True, True, True, True, True	False	[56585.80558464 34 240.18998271 125	0.2158458430925294
13	Республика Ком	['area', 'levels', 'kitchen_	[True, True, True, True, T	['area', 'levels', 'kitchen_ar	True	False	False	True	[False, True, True, True, True, True	False	[35171.90526069 850 -2863.03255546 900	0.3078299023684677
14	Челябинская об	['area', 'levels', 'date_en	[True, True, True, True, T	['area', 'levels', 'date_enco	True	False	False	True	[False, True, True, True, True, True	False	[35316.24314837 394 -17864.69022234 -51	0.1876221106650494
15	Иркутская облас	['area', 'levels', 'date_en	[True, True, True, True, T	['area', 'levels', 'date_enco	True	False	False	True	[False, True, True, True, True, True	False	[52074.81268527 78 -2891.06853989 -24	0.3334857045660632
16	Ленинградская с	['area', 'levels', 'date_en	[True, True, True, True, T	['area', 'levels', 'date_enco	True	False	False	True	[False, True, True, True, True, True	False	[41677.62366837 703 -31552.31916112 -10	0.3382393328008357
17	Пермский край	['area', 'levels', 'date_en	[True, True, True, True, T	['area', 'levels', 'date_enco	True	False	False	True	[False, True, True, True, True, True	False	[43404.85489455 35 -129891.23931278 5	0.2378103322330944
18	Алтайский край	['area', 'levels', 'date_en	[True, True, True, True, T	['area', 'levels', 'date_enco	True	False	False	True	[False, True, True, True, True, True	False	[47358.09269256 55 -30473.95103928 -10	0.2377523488772200
19	Республика Бур	['area', 'date_encoded',	[True, True, True, True, T	['area', 'date_encoded', 'le	True	False	False	True	[False, True, True, True, True]	False	[39459.89494309 31	0.2529767436577236
20	Ярославская об	['area', 'rooms', 'date_en	[True, True, True, True, T	['area', 'rooms', 'date_enco	True	False	False	True	[False, True, True, True, True, True	False	[60869.41732618 -40 2926.297247 -602	0.2390956664943492

Способы применения работы:

1) Если к имеющимся данным добавить различные экономические факторы с росстата, например рождаемость в регионе, уровень развития строительного бизнеса и пр., можно будет выявить нетривиальные закономерности между регионом и факторами, которые влияют на стоимость квартир.
2) Если спарсить данные с 2021 по 2023 года и добавить к нашим данным, то можно использовать алгоритм для предсказания стоимости квартир в регионах на следующий год. Это может быть полезно риелторам, а также людям, планирующим покупку жилья в будущем.
3) Несмотря на то, что наличие гетероскедастичности не сильно испортило качество модели, в будущем будет полезно научиться её смягчать. Для этого можно использовать как классические подходы, так и пытаться строить модели на данных в разрезе региона и других факторов.

Что пробовала, но не получилось:

1) Парсить данные. За основу было взято готовое решение (https://github.com/lenarsaitov/cianparser), однако данный парсер извлекал из данных не все доступные поля (метро, например), а также использование сторонних прокси, не решало проблему появления капчи. В будущем планируется написание собственного парсера, способного извлекать информацию о расстоянии до метро, название района, где находится квартира и пр.

Полезные ресурсы:

1) https://www.kaggle.com/datasets/mrdaniilak/russia-real-estate-20182021
2) https://studfile.net/preview/2829140/page:8/
3) https://github.com/romanakentev/real-estate/blob/main/Real_Estate__ENG.ipynb
4) https://www.hse.ru/mirror/pubs/share/423684138.pdf
5) https://github.com/lenarsaitov/cianparser