САНКТ-ПЕТЕРБУРГСКИЙ
ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ

Прикладная математика и информатика

Исследование операций и принятие решений в задачах
оптимизации, управления и экономики

Выпускная квалификационная работа

Головчанская Юлия Игоревна

Линейные модели в статистике

Научный руководитель:
канд. физ.-мат. наук, доцент Бухвалова В. В.

Рецензент:
ПАО Сбербанк, руководитель направления
ПЦП Центра развития технологий Ковальчук А. В.

Санкт-Петербург
2019 г.

Saint Petersburg State University

Applied Mathematics and Computer Science

Operation Research and Decision Making in Optimisation,
Control and Economy

Graduation Thesis

Golovchanskaia Iuliia Igorevna

Linear Models in Statistics

Scientific Supervisor:
Associate Professor Bukhvalova V. V.

Reviewer:
Sberbank, Center of Technology development,
Head of Department Kovalchuk A. V.

Saint Petersburg
2019

Оглавление

Введение

Линейная регрессия — метод восстановления зависимости одной переменной y (зависимой) от другой или нескольких других переменных (независимых переменных) x. Одной из целей регрессионного анализа является предсказание значения зависимой переменной с помощью независимой(-ых). На практике линия регрессии чаще всего ищется с помощью метода наименьших квадратов. Так как прогнозирование в целом и, в особенности, финансовых показателей сопряжено с рядом трудностей, то возникла потребность альтернативы методу наименьших квадратов, которая была бы менее чувствительна к выбросам.

В 1978 году была опубликована статья «Квантильная регрессия» (Regression Quantiles) Роджера Коенкера и Гильберта Бассета [9], в которой впервые была введена квантильная регрессия. Было установлено: если ошибки не подчинены нормальному закону распределения, квантильная регрессия может быть более эффективна, чем метод наименьших квадратов. Данная работа делиться на две части:

- Линейная регрессия: способы нахождения коэффициентов прямой;
- Квантильная регрессия.

В работе показано как находить коэффициенты регрессии с помощью линейного программирования.

1. Линейная регрессия

В данном разделе рассматриваться модель парной регрессии, а также способы нахождения соответствующей линии.

1.1. Постановка задачи

В реальной жизни не следует ожидать получения точного соотношения между какими-либо двумя экономическими показателями. В статистическом анализе факт неточности соотношения выражается путем явного включения в него случайного фактора, описываемого случайным остаточным членом.

Начнем с рассмотрения простейшей модели. Пусть имеется выборка $(x_i, y_i),\ i = 1, \ldots, n$. Значения y_i предположительно находятся под влиянием значений x_i в следующей линейной зависимости:

$$y_i = kx_i + b + e_i, \tag{1}$$

где k – коэффициент регрессии, отражает наклон линии, вдоль которой рассеяны данные наблюдений, b – постоянная, (0, b) – точка пересечения прямой с осью y, e_i – ошибка или значение помехи, также называемая остатком.

Задачу можно сформулировать следующим образом: подобрать функцию $f(x)$ из семейства линейных функций
$\{f(x, k, b) = kx + b \mid k \in \mathbb{R}, b \in \mathbb{R}\}$, которая наилучшим образом описывает зависимость y от x.

Отобразим пары (x_i, y_i) точками на плоскости.

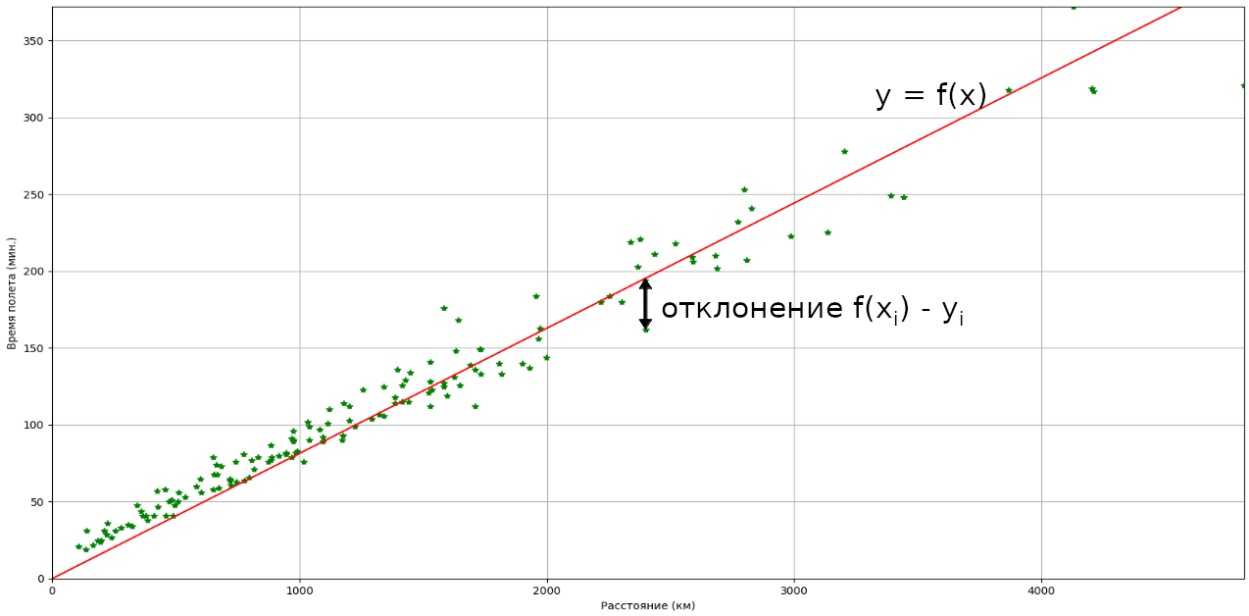

Рис. 1. Пример регрессии: зависимость времени полета от расстояния

В [1] перечислены причины, объясняющие существование остатка. Главные из них:

1. Соотношение между y и x почти наверняка являются очень большим упрощением. В действительности, существуют другие факторы, влияющие на y. Влияние этих факторов приводит к тому, что наблюдаемые точки лежат вне прямой.

2. Во многих случаях рассматриваемая зависимость – это попытка объединить вместе некоторое число экономических соотношений. Например, функция суммарного потребления – это попытка общего выражения совокупности решений отдельных индивидов о расходах. Разные соотношения могут иметь разные параметры.

3. Если в измерении одной или более взаимосвязанных переменных имеются ошибки, то существующее расхождение будет вносить вклад в остаточный член.

Остаток является суммарным проявлением всех этих факторов.

1.2. Способы нахождения линии регрессии

В качестве меры отклонения функции $f(x)$ от набора наблюдений в этой главе будут рассмотрены:

1. сумма квадратов отклонений: $\sum\limits_{i=1}^{n}(y_i - f(x_i))^2$,
2. сумма модулей отклонений: $\sum\limits_{i=1}^{n}|y_i - f(x_i)|$.
3. максимальный модуль отклонения: $\max\limits_{i=1,\ldots,n}|y_i - f(x_i)|$.

Соответствующие способы нахождения линии регрессии получили следующие названия:

1. Метод наименьших квадратов (МНК):

$$\min_{k,b}\left(\sum_{i=1}^{n}(y_i - f(x_i,k,b))^2\right),$$

2. Метод наименьшей суммы модулей (МНСМ):

$$\min_{k,b}\left(\sum_{i=1}^{n}|y_i - f(x_i,k,b)|\right),$$

3. Метод наименьшего максимального модуля (МНММ):

$$\min_{k,b}\left(\max_{i=1,\ldots,n}|y_i - f(x_i,k,b)|\right).$$

Достоинством МНК является наличие прямых формул для вычисления коэффициентов k и b:

$$k = \sum_{j=1}^{n}\frac{x_j - \overline{x}}{\sum_{i=1}^{n}(x_i - \overline{x})^2}y_j,$$

$$b = \overline{y} - k\overline{x},$$

где $\overline{x} = \frac{\sum_{i=1}^{n}x_i}{n}$, $\overline{y} = \frac{\sum_{i=1}^{n}y_i}{n}$.

2. Линейное программирование в регрессионном анализе

В данном разделе приведен обзор статьи Харви Вагнера, в котором показаны способы сведения множественной линейной регрессии к задачам ЛП. Далее разобраны МНСМ и МНММ на примере парной регрессии и выписаны явные матричные формы этих задач.

2.1. Обзор статьи Харви Вагнера

Статья [12] была опубликована в 1959 году в Journal of the American Statistical Association. Эта статья состоит из 5 частей:

- Введение;
- Двойственная задача линейного программирования (ЛП);
- Минимизация суммы абсолютных отклонений;
- Минимизация максимума абсолютного отклонения;
- Численные примеры.

Далее рассмотрим подробнее каждую часть.

Введение

Карст О. [8] предложил итеративную процедуру для нахождения прямой линии, наилучшим образом проходящей через множество точек на плоскости: сумма абсолютных значений вертикальных отклонений этих точек от линии является минимальной. Во времена написания данного материала ЛП было относительно новым инструментом, применяемым в статистике, тем самым главной целью статьи являлось элементарное представление применения данной техники в многомерной версии задачи Карста.

Двойственная задача линейного программирования

Прямая и двойственная задачи ЛП рассматриваются в следующей постановке:

$$\begin{gathered}\max_x(c_1x_1+\cdots+c_nx_n),\\ a_{1h}x_1+\cdots+a_{nh}x_n\le b_h,\quad \forall h\in M_1,\\ a_{1h}x_1+\cdots+a_{nh}x_n= b_h,\quad \forall h\in M_2,\\ M=M_1\cup M_2,\\ x_l\ge 0,\quad \forall l\in N_1.\end{gathered}\tag{2}$$

Соответствующая двойственная задача ЛП:

$$\begin{gathered}\min_x(u_1b_1+\cdots+u_mb_m),\\ u_1a_{l1}+\cdots+u_ma_{lm}\ge c_l,\quad \forall l\in N_1,\\ u_1a_{l1}+\cdots+u_ma_{lm}= c_l,\quad \forall l\in N_2,\\ N=N_1\cup N_2,\\ u_h\ge 0,\quad \forall h\in M_1.\end{gathered}\tag{3}$$

В частных случаях одно или более из множеств M_1, M_2, N_1, N_2 могут быть пустыми. В теореме двойственности утверждается, что x^* из множества допустимых решений задачи (2) оптимален тогда и только тогда, когда существует u^* и

$$c_1x_1^*+\cdots+c_nx_n^*=u_1^*b_1+\cdots+u_m^*b_m.\tag{4}$$

Минимизация суммы модулей отклонений

Пусть $M=\{1,\ldots,p\}$, $N=\{1,\ldots,k\}, k>p$. Рассмотрим набор $x_{ij}, i\in N, j\in M$ как k значений p независимых переменных. $y_i, i\in N$ — соответствующие значения зависимой переменной. Необходимо определить коэффициенты линейной регрессии b_j такие, что

$$\min_{b_j}\sum_i\left|\sum_j x_{ij}b_j-y_i\right|.\tag{5}$$

Данная задача эквивалентна следующей задаче ЛП:

$$\begin{gathered}\min\left(\sum_i \varepsilon_{1i} + \sum_i \varepsilon_{2i}\right),\\ \sum_j x_{ij}b_j + \varepsilon_{1i} - \varepsilon_{2i} = y_i, \quad i = 1,\dots,k,\\ \varepsilon_{1i} \geq 0, \quad i = 1,\dots,k,\\ \varepsilon_{2i} \geq 0, \quad i = 1,\dots,k.\end{gathered} \tag{6}$$

Данная модель включает $p + 2k$ неизвестных, k ограничений, $2k$ знаковых ограничений.

Автор статьи отмечает, что данная модель имеет очевидный недостаток. Если количество наблюдений k существенно, то задача становится вычислительно громоздкой. Тогда следует воспользоваться двойственной задачей ЛП (d_i - двойственные переменные), при помощи которой можно найти b_j как побочный продукт.

$$\begin{gathered}\max\sum_i y_i d_i,\\ \sum_i x_{ij}d_i = 0, \quad j = 1,\dots,p,\\ d_i \leq 1, \quad i = 1,\dots,k,\\ -d_i \leq 1, \quad i = 1,\dots,k.\end{gathered} \tag{7}$$

Данная модель состоит из k неизвестных, $p + 2k$ ограничений. Чтобы сократить задачу до модели, содержащей p ограничений и k знаковых ограничений, положим

$$f_i := d_i + 1, i = 1,\dots,k.$$

Тогда задача примет вид:

$$\begin{gathered}\max\left(\sum_i y_i f_i - \sum_i y_i\right),\\ \sum_i x_{ij}f_i = \sum_i x_{ij}, \quad j = 1,\dots,p,\\ 0 \leq f_i \leq 2, \quad i = 1,\dots,k.\end{gathered} \tag{8}$$

Теперь модель содержит p линейных ограничений и k неотрицательных ограниченных переменных, и может быть решена вполне быстро при помощи специальных вариантов симплекс-метода [6, 7].

Минимизация максимального модуля

Рассмотрим следующую задачу

$$\min_{b_j} \left(\max_i \left| \sum_j x_{ij} b_j - y_i \right| \right),$$

и преобразуем ее к задаче ЛП

$$\begin{gathered} \min \varepsilon, \\ -\sum_j x_{ij} b_j + \varepsilon \geq -y_i, \quad i = 1, \ldots, k, \\ \sum_j x_{ij} b_j + \varepsilon \geq y_i, \quad i = 1, \ldots, k, \\ b_j \geq 0, \quad \forall j \in M_1, \\ \varepsilon \geq 0. \end{gathered} \tag{9}$$

Тогда двойственная задача имеет вид:

$$\begin{gathered} \max \left(-\sum_i y_i d_{1i} + \sum_i y_i d_{2i} \right), \\ -\sum_i x_{ij} d_{1i} + \sum_i x_{ij} d_{2i} \leq 0, \quad \forall j \in M_1, \\ -\sum_i x_{ij} d_{1i} + \sum_i x_{ij} d_{2i} = 0, \quad \forall j \in M_2, \\ \sum_t d_{1i} + \sum_t d_{2i} \leq 1, \\ d_{1i} \geq 0, \quad \forall i = 1, \ldots, k, \\ d_{2i} \geq 0, \quad \forall i = 1, \ldots, k. \end{gathered} \tag{10}$$

Добавим дополнительные переменные и приведем последнюю задачу к канонической форме.

$$\begin{gathered}
\max\left(-\sum_i y_i d_{1i} + \sum_i y_i d_{2i}\right), \\
-\sum_i x_{ij} d_{1i} + \sum_i x_{ij} d_{2i} + s_j = 0, \quad \forall j \in M_1, \\
-\sum_i x_{ij} d_{1i} + \sum_i x_{ij} d_{2i} = 0, \quad \forall j \in M_2, \\
\sum_i d_{1i} + \sum_i d_{2i} + t = 1, \\
d_{1i} \geq 0, \quad i = 1, \ldots, k, \\
d_{2i} \geq 0, \quad i = 1, \ldots, k, \\
s_j \geq 0, \quad \forall j \in M_1, \\
t \geq 0.
\end{gathered} \tag{11}$$

Численный пример

В статье [12] был следующий численный пример и вычисляются коэффициенты, полученные разными способами.

x_i	-12,5	-8,5	-6,5	-3,5	-2,5	-1,5	-0,5	2,5	4,5	8,5	8,5	11,5
y_i	-8,4	-5,4	3,6	-2,4	-4,4	1,6	-0,4	-0,4	-2,4	3,6	5,6	9,6

Метод наименьших квадратов:

$$y = 0.539x.$$

Метод наименьших сумм модулей:

$$y = 0.659x.$$

Метод наименьшего максимума модуля:

$$y = 0.333x.$$

Данный пример был решен с использованием программы из приложений (1) и (2). Полученные результаты совпали с приведенными выше.

2.2. Сведение к задаче ЛП

Пусть (x_i, y_i), $i = 1, \ldots, n$ — выборка, $x_i, y_i \in \mathbb{R}$. Значения y_i предположительно находятся под влиянием значений x_i. Необходимо найти коэффициенты регрессии k и b.

МНСМ

$$\min_{k,b} \left(\sum_{i}^{n} |y_i - x_i k - b| \right). \tag{12}$$

С помощью введения дополнительных переменных задача трансформируется в следующую:

$$\min \left(\sum_{i}^{n} \varepsilon_{1i} + \sum_{i}^{n} \varepsilon_{2i} \right), \tag{13}$$

при ограничениях

$$\begin{aligned} x_i k + b + \varepsilon_{1i} - \varepsilon_{2i} &= y_i, \qquad i = 1, \ldots, n, \\ \varepsilon_{1i} &\geq 0, \qquad i = 1, \ldots, n, \\ \varepsilon_{2i} &\geq 0, \qquad i = 1, \ldots, n. \end{aligned} \tag{14}$$

Можно интерпретировать ε_{1i} и ε_{2i} как отклонение «над» и «под» соответсвенно от i-го наблюдения, то есть $\varepsilon_{1i} + \varepsilon_{2i}$ абсолютное отклонение $kx_i + b$ от y_i.

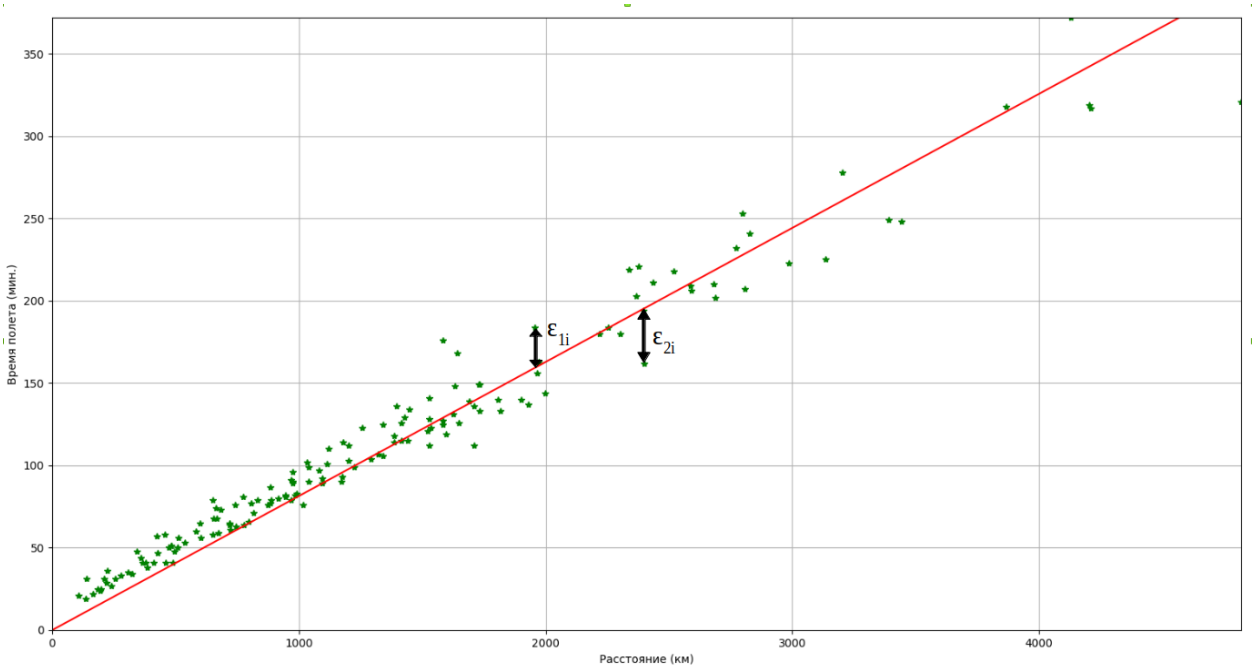

Рис. 2. Иллюстрация к дополнительным переменным

Трансформированная задача является задачей ЛП и имеет вид:

$$\begin{aligned} &\min_{z}(c[N] \times z[N]), \\ &A[M,N] \times z[N] = d[M], \\ &z[N_1] \geq 0, \qquad N_1 \subset N. \end{aligned} \tag{15}$$

Теперь приведем явный вид матричной формы этой задачи.

$$A = \begin{pmatrix} x_1 & 1 & 1 & -1 & 0 & 0 & \dots & 0 \\ x_2 & 1 & 0 & 0 & 1 & -1 & 0 \dots & 0 \\ \vdots & \vdots & & & & \ddots & & \\ x_n & 1 & 0 & & \dots\dots & & 0 & 1 & -1 \end{pmatrix}, \; c = \begin{pmatrix} 0 \\ 0 \\ 1 \\ 1 \\ \vdots \\ 1 \end{pmatrix}, \; d = \begin{pmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{pmatrix}.$$

Вектор переменных имеет вид:

$$z = \begin{pmatrix} k \\ b \\ \varepsilon_{11} \\ \varepsilon_{21} \\ \vdots \\ \varepsilon_{1n} \\ \varepsilon_{2n} \end{pmatrix}.$$

МНММ

$$\min_{k,b} \left(\max_i |y_i - x_i k - b| \right).$$

С помощью введения дополнительной переменной задача трансформируется в следующую:

$$\min \varepsilon, \tag{16}$$

при ограничениях

$$\begin{aligned} -x_i k - b + \varepsilon \geq -y_i, \qquad & i = 1, \ldots, n, \\ x_i k + b + \varepsilon \geq y_i, \qquad & i = 1, \ldots, n, \\ \varepsilon \geq 0. & \end{aligned} \tag{17}$$

Трансформированная задача является задачей ЛП и имеет вид (15). Теперь приведем явный вид матричной формы этой задачи.

$$A = \begin{pmatrix} -x_1 & -1 & 1 \\ \vdots & \vdots & \vdots \\ -x_n & -1 & 1 \\ x_1 & 1 & 1 \\ \vdots & \vdots & \vdots \\ x_n & 1 & 1 \end{pmatrix}, \; c = \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix}, \; d = \begin{pmatrix} -y_1 \\ \vdots \\ -y_n \\ y_1 \\ \vdots \\ y_n \end{pmatrix}.$$

Вектор переменных имеет вид:

$$z = \begin{pmatrix} k \\ b \\ \varepsilon \end{pmatrix}.$$

У задач (13 – 14) и (16 - 17) оптимальные планы существуют, так как множества планов этих задач непусты, и на этих множествах целевые функции ограничены снизу. Таким образом, найдя решение задачи ЛП, соответсвующей выбранному методу, мы найдем функцию $f(x)$.

2.3. Вычислительные эксперименты

Эксперимент 1

Fortune 500 — это ранжированный список крупнейших компаний США по общей выручке за соответсвующий финансовый год. На сайте [14] был найден последний набор данных Fortune 500 (2018 год).

		Company Info		KEY FINANCIALS					
Rank	Company Name	Number of Employees	Previous Rank	Revenues ($millions)	Revenue Change	Profits ($millions)	Profit Change	Assets ($millions)	Market Value As of 3/29/18 ($m)
1	Walmart	2,300,000	1	$500,343	3.0%	$9,862.0	-27.7%	$204,522	$263,563
2	Exxon Mobil	71,200	4	$244,363	17.4%	$19,710.0	151.4%	$348,691	$316,157
3	Berkshire Hathaway	377,000	2	$242,137	8.3%	$44,940.0	86.7%	$702,095	$492,008
4	Apple	123,000	3	$229,234	6.3%	$48,351.0	5.8%	$375,319	$851,318
5	UnitedHealth Group	260,000	6	$201,159	8.8%	$10,558.0	50.5%	$139,058	$207,080
6	McKesson	64,500	5	$198,533	3.1%	$5,070.0	124.5%	$60,969	$29,067
7	CVS Health	203,000	7	$184,765	4.1%	$6,622.0	24.5%	$95,131	$63,114
8	Amazon.com	566,000	12	$177,866	30.8%	$3,033.0	27.9%	$131,310	$700,668

Рис. 3. Фрагмент списка Fortune 500

Пусть в качестве независимой переменной выступает выручка (revenue), а в качестве зависимой – рыночная капитализация (assets). Вычислим регрессию, используя методы МНСМ, МНММ и МНК.

Сначала осуществим предварительную обработку данных. В рейтинге присутствуют компании, у которых пропущены значения рыночной стоимости. Удаляя эти записи из данных, получим список, содержащий 472 компании. Для анализа будем рассматривать компании только из сектора «Технологии»: осталось 38 компаний.

В данном эксперименте полагаем $b = 0$. Коэффициенты наклона прямой:

- МНСМ: $k = 1{,}78$;
- МНММ: $k = 1{,}929$;
- МНК: $k = 1{,}807$.

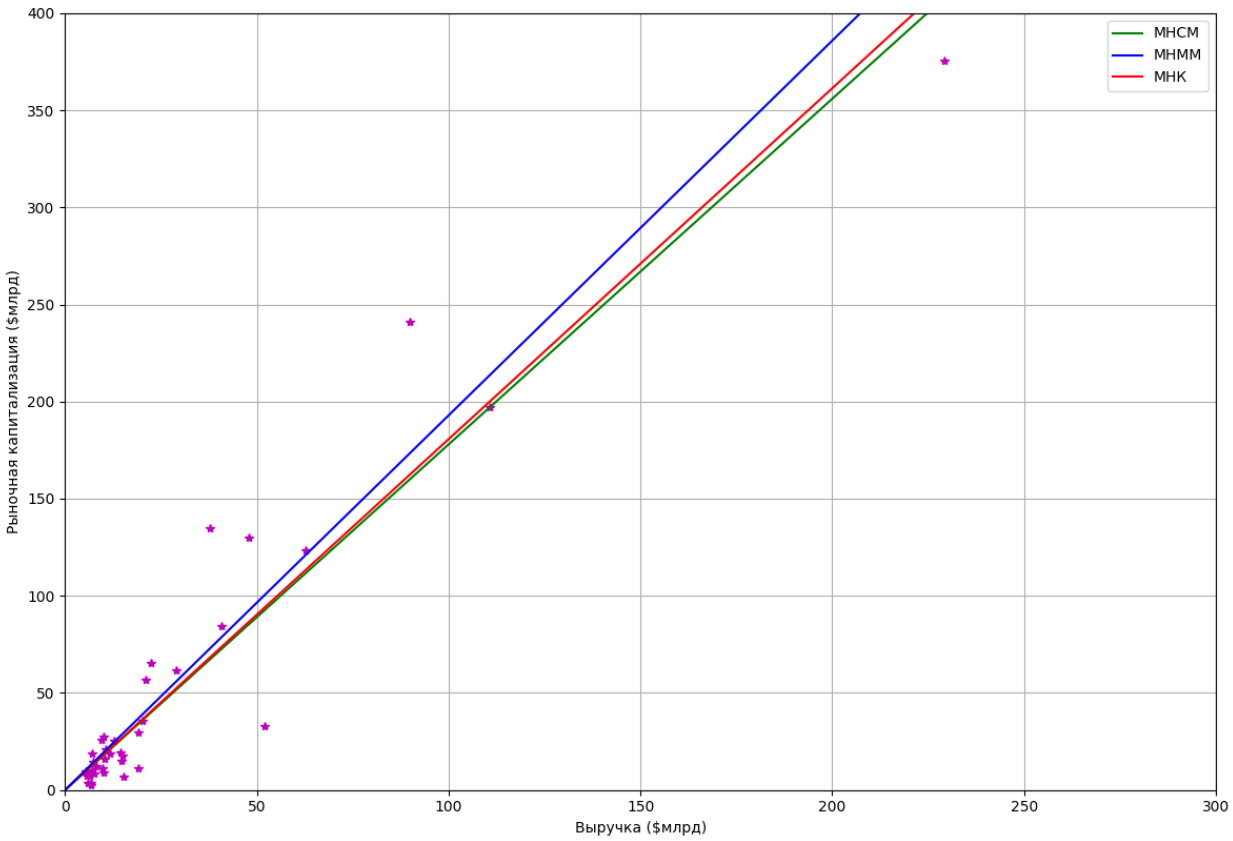

Рис. 4. Fortune 500: сектор «Технологии» (2018 год)

Посчитаем коэффициенты детерминации для моделей:

- МНСМ: $R^2 = 0{,}9047$;
- МНММ: $R^2 = 0{,}8989$;
- МНК: $R^2 = 0{,}905$.

R^2 принимает значения от 0 до 1. Чем ближе значение коэффициента к 1, тем сильнее зависимость. При оценке регрессионных моделей это интерпретируется как соответствие модели данным.

Данный эксперимент показывает, что если в данных не наблюдаются значительных выбросов, то линии, построенные при помощи МНК и МНСМ, почти совпадают.

В статье [11] описано, как использовать МНММ для обнаружения выбросов.

Эксперимент 2

Рассмотрим применение линейной регрессии для прогнозирования движения автомобиля. Сперва введем некоторые определения.

Полоса движения — продольный участок проезжей части, шириной достаточной для движения автомобилей в один ряд.

Ширина полосы движения по ГОСТ: 3,5 м – 3,75 м.

Область интересов — область в зоне наблюдения, определенная для конкретной цели.

Пусть дан путь автомобиля (трек) в виде набора $(x_i, y_i), i = 1, \ldots, n$:

x_i	-15,1	-13,49	-12,29	-11,5	-10,1	...	1,7	2,1	2,5	3,5	4,69	9,5
y_i	2,1	2,1	2,1	2,2	2,0	...	1,69	0,3	1,74	1,89	2,2	2,1

Точка $(2.1, 0.3)$ является «выбросом». Данный выброс может возникнуть из-за неточности измерительной системы, при помощи которой был получен путь автомобиля. В данном эксперименте полоса движения является областью интересов, а задача заключается в следующем: определить будет ли автомобиль находиться в области интересов при $x = 25$ м. Строить прогноз на отрезке $[9.5, 25]$ по оси x будем, анализируя точки на отрезке $[0, 10]$.

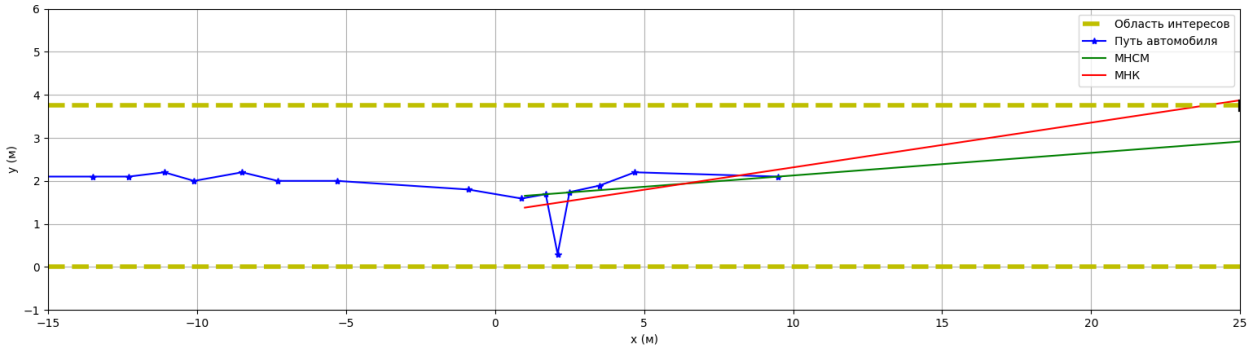

Рис. 5. График пути автомобиля

Как видно на рис. 5, используя МНК, мы получаем ответ: при $x = 25$ м автомобиль не будет находиться в области интересов. Используя МНСМ, мы получаем противоположный ответ, который является верным.

Данный эксперимент был проведен на 54 треках и показал следующие результаты:

	МНК	**МНСМ**
Доля верных ответов	72.2%	92,6%
Общее время обработки данных	0,076 сек.	0,568 сек.

3. Квантильная регрессия

В данном разделе рассматриваться модель множественной квантильной регрессии и ее сведение к задаче ЛП.

3.1. Основные определения и модель

Определение. *Квантилью порядка* θ, $\theta \in (0,1)$, *случайной велечины* y *называется число* y_θ *такое, что выполнены неравенства*

$$P(y < y_\theta) \leq \theta \leq P(y \leq y_\theta).$$

После преобразования мы можем записать эти неравенства в виде:

$$P(y \leq y_\theta) \geq \theta, \quad P(y \geq y_\theta) \geq 1 - \theta.$$

Рассмотрим дискретный случай. Пусть y_i, $i = 1, \ldots, n$ — выборка. Соответствующая выборочная квантиль $Quant_\theta$ порядка θ будет определяться по следующей формуле:

$$Quant_\theta = \underset{y \in \mathbb{R}}{argmin} \left(\sum_{i:y_i \geq y} \theta |y_i - y| + \sum_{i:y_i < y} (1-\theta)|y_i - y| \right) \tag{18}$$

Пусть (x_i, y_i), $i = 1, \ldots, n$ — выборка, где x_i — k-мерный вектор-строка. В квантильной регрессии предполагается, что значения y_i находятся под влиянием значений x_i в следующей зависимости:

$$\begin{gathered} y_i = x_i \beta_1^\theta + \beta_2^\theta + e_{\theta,i}, \quad \beta_1^\theta \in \mathbb{R}^k, \beta_2^\theta \in \mathbb{R}, i = 1, \ldots, n, \\ Quant_\theta(y_i | x_i) = x_i \beta_1^\theta + \beta_2^\theta, \quad i = 1, \ldots, n. \end{gathered} \tag{19}$$

В данной модели не рассматриваются такие проблемы, как ошибки измерения и пропущенные переменные. Из (19) следует, что $e_{\theta,i}$ удовлетворяют ограничению на квантиль:

$$Quant_\theta(e_{\theta,i} | x_i) = 0, \quad i = 1, \ldots, n.$$

Квантильная регрессия расширяет данную задачу нахождения θ-ой простой квантили ($0 < \theta < 1$), позволяя учитывать независимые значения. Коэффициенты квантильной регрессии – решение следующей задачи

минимизации:

$$\underset{\beta^\theta \in \mathbb{R}^{k+1}}{argmin} \left(\sum_{i:y_i \geq x_i\beta_1^\theta + \beta_2^\theta} \theta|y_i - x_i\beta_1^\theta - \beta_2^\theta| + \sum_{i:y_i < x_i\beta_1^\theta + \beta_2^\theta} (1-\theta)|y_i - x_i\beta_1^\theta - \beta_2^\theta| \right), \tag{20}$$

где

$$\beta^\theta = \begin{pmatrix} \beta_1^\theta \\ \beta_2^\theta \end{pmatrix}.$$

3.2. Модель Коенкера-Бассета

Статья [9], в которой впервые была введена квантильная регрессия, была опубликована в 1978 году в журнале «Econometrica». Далее в этом разделе приведено краткое изложение этой статьи.

Введение и мотивация

В статистике термин робастность означает свойство статистического метода, характеризующее независимость влияния на результат исследования различного рода выбросов. Одним из важных свойств квантильной регрессии является то, что данная модель устойчива к выбросам, которые часто встречаются на практике, в особенности в случае показателей финансового рынка.

Среди прикладных методов наиболее распространенным является МНК, позволяющий получить глубокие статистические результаты при предположении, что случайные ошибки распределены по нормальному закону. Так как математические предположения на практике могут не выполняться, существовала потребность появления альтернативы МНК для линейной модели. Аномальные наблюдения, или выбросы, как их назвали в последствии, было сложно выделить в моделях. Многие знаменитые ученые (Гаусс, Лаплас, Лежандр, и другие) полагали, что, если некоторые наблюдения ненадежны, то МНСМ мог бы быть более предпочтительным, чем МНК. В 1818 году Лаплас доказал, что в простой модели парной регрессии без смещения формула оценки по МНСМ имеет меньшую асимптотическую

дисперсию, чем формула оценки по МНК. Этот результат положил начало исследованиям большой части теории статистики, основанной на обычных квантилях. Поэтому при прогнозе финансовых показателей использование модели квантильной регрессии более предпочтительно, чем использование МНК.

Фундаментальные свойства

Отправной точкой в определении квантильной регрессии авторы выбрали определение выборочной квантили. Это определение можно сформулировать не только через вариационный ряд, но и как решение задачи минимизации (18).

Квантильная регрессия является прямым обобщением задачи минимизации, упомянутой выше.

Пусть (x_i, y_i), $i = 1, \ldots, n$ — выборка, где x_i — k-мерный вектор-строка. Квантильная регрессия в статье и все последующие теоремы рассмотрены для случая без смещения:

$$y_i = x_i b + e_{\theta,i}, \quad i = 1, \ldots, n,$$
$$\min_{b \in \mathbb{R}^k} \left(\sum_{i: y_i \geq x_i b} \theta |y_i - x_i b| + \sum_{i: y_i < x_i b} (1-\theta) |y_i - x_i b| \right). \tag{21}$$

Метод МНСМ является частным случаем квантильной регрессии: $\theta = 0,5$.

Введем некоторые обозначения. Пусть $B^*(\theta)$ — множество решений задачи (21) при некотором фиксированном θ, $\mathcal{N} = \{1, \ldots, n\}$, $\mathcal{H}$ — набор k-элементых подмножеств $\mathcal{N}$. Для каждого $h \in \mathcal{H}$: $\overline{h} = \mathcal{N} \backslash h$. Пусть $y(h)$ — k-мерный вектор, состоящий из элементов $\{y_i : i \in h\}$, а $X(\overline{h})$ — это матрица размерности $(n-k) \times k$, состоящая из строк $\{x_i : i \in \overline{h}\}$. Наконец, пусть $H = \{h \in \mathcal{H} | rank(X(h)) = k\}$.

Теорема. *Если rank(X) = k, то множество $B^*(\theta)$ имеет хотя бы один элемент $\beta^*(\theta)$:*

$$\beta^*(\theta) = X(h)^{-1} y(h)$$

для некоторого $h \in H$. Кроме того, $B^*(\theta)$ является выпуклой оболочкой всех решений, имеющих такой вид.

Теорема. *Если $\beta^*(\theta, y, X) \in B^*(\theta, y, X)$, тогда верны следующие утверждения:*
(i) $\beta^*(\theta, \lambda y, X) = \lambda\beta^*(\theta, y, X)$, $\lambda \in [0, +\infty)$,
(ii) $\beta^*(1-\theta, \lambda y, X) = \lambda\beta^*(\theta, y, X)$, $\lambda \in (-\infty, 0]$,
(iii) $\beta^*(\theta, y + X\gamma, X) = \beta^*(\theta, y, X) + \gamma$, $\gamma \in \mathbb{R}^k$,
(iv) $\beta^*(\theta, y, XA) = A^{-1}\beta^*(\theta, y, X)$, $A_{k\times k}$ — невырожденная матрица.

Теорема. *Если $\beta^*(\theta) \in B^*(\theta, y, X)$, тогда $\beta^*(\theta) \in B^*(\theta, X\beta^* + Du^*, X)$, где $u^* = y - X\beta^*$, D — любая $n \times n$ диагональная матрица с неотрицательными элементами.*

Использование данных теорем может существенно ускорить процесс поиска коэффициентов регрессии при использовании симплекс-метода. Их доказательства приведены в [9]. Кроме того, авторы ссылаются на статью Х. Вагнера [12], обзор которой находится в разделе «Линейное программирование в регрессионном анализе» данной работы. Важно отметить, что методы по усовершенствованию решения задачи ЛП, описанные в [12] также распространяются на решение задачи ЛП для поиска коэффициентов квантильной регрессии.

3.3. Представление квантильной регрессии в виде задачи ЛП

Пусть (x_i, y_i), $i = 1, \ldots, n$ — выборка, $x_i, y_i \in \mathbb{R}$. С помощью введения дополнительных переменных задача (20) трансформируется в следующую:

$$\min \left(\theta \sum_{i=1}^{n} u_i^+ + (1-\theta) \sum_{i=1}^{n} u_i^- \right), \tag{22}$$

при ограничениях

$$\begin{aligned} x_i\beta_1^\theta + \beta_2^\theta + u_i^+ - u_i^- &= y_i, \qquad i = 1, \ldots, n, \\ u_i^+ &\geq 0, \qquad i = 1, \ldots, n, \\ u_i^- &\geq 0, \qquad i = 1, \ldots, n. \end{aligned} \tag{23}$$

Трансформированная задача является задачей ЛП и имеет вид (15). Теперь приведем явный вид матричной формы этой задачи.

$$A = \begin{pmatrix} x_1 & 1 & 1 & -1 & 0 & 0 & \dots & 0 \\ x_2 & 1 & 0 & 0 & 1 & -1 & 0 \dots & 0 \\ \vdots & \vdots & & & & & \ddots & \\ x_n & 1 & 0 & & \dots\dots & & 0 \; 1 \; -1 & \end{pmatrix}, \; c = \begin{pmatrix} 0 \\ 0 \\ \theta \\ 1-\theta \\ \vdots \\ \theta \\ 1-\theta \end{pmatrix}, \; d = \begin{pmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{pmatrix}.$$

Вектор переменных имеет вид:

$$z = \begin{pmatrix} \beta_1^\theta \\ \beta_2^\theta \\ u_1^+ \\ u_1^- \\ \vdots \\ u_n^+ \\ u_n^- \end{pmatrix}.$$

Оптимальный план задачи (22 — 23) существует, так как множество планов задачи непусто, и целевая функция на нем ограничена снизу. Таким образом, найдя решение задачи ЛП, мы найдем коэффициенты квантильной регрессии β_1^θ и β_2^θ.

3.4. Вычислительный эксперимент

На сайте Kaggle [13] имеется набор данных о продажах домов в США в 2014 году.

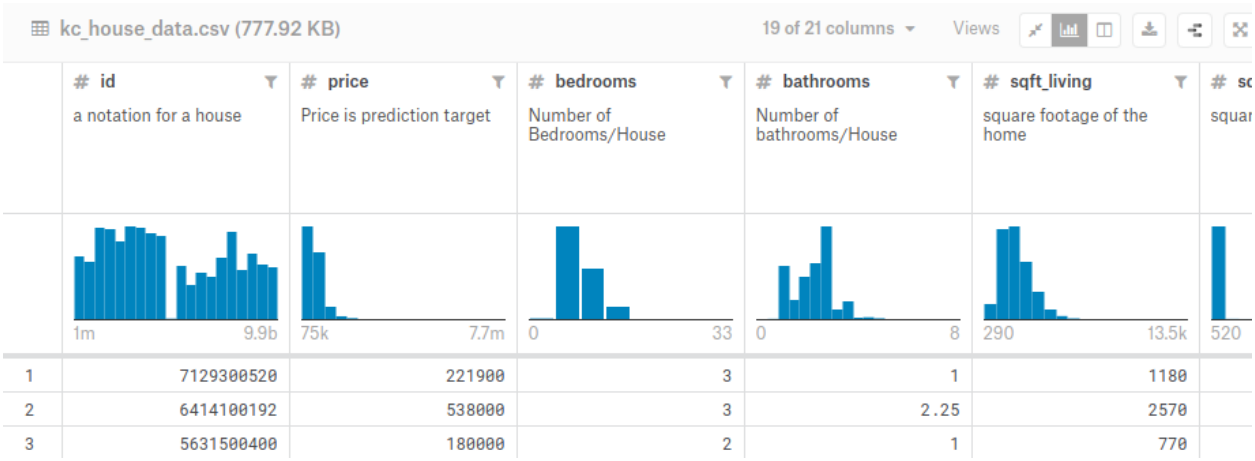

Рис. 6. Фрагмент списка

Пусть в качестве независимой переменной выступает площадь дома (sqft living), а в качестве зависимой — цена (price). Вычислим квантильную регрессию при $\theta = 0,75$; $\theta = 0,5$; $\theta = 0,25$.

Сначала осуществим предварительную обработку данных. Площадь дома представлена в квадратных футах. Для конвертации в квадратные метры делим каждое значение в столбце «sqft living» на $10,764$. В наборе представлены данные о 21613 домах (объектах). Для эксперимента рассматривается выборка из 300 объектов.

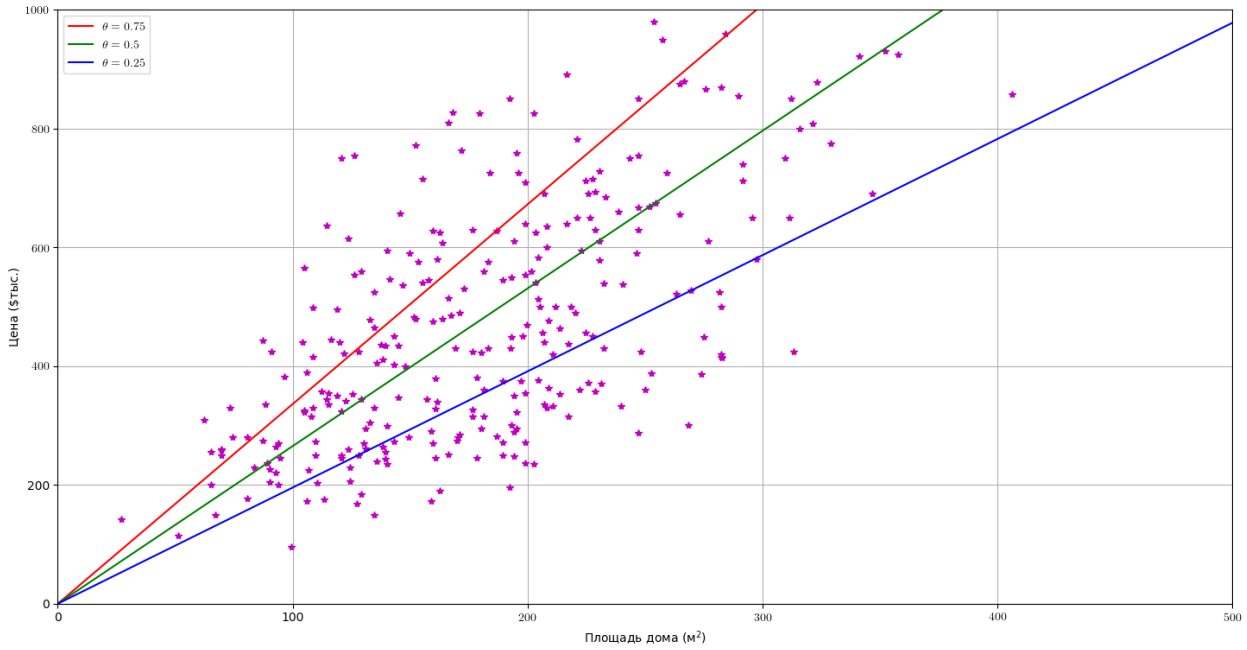

Рис. 7. График зависимости цены квартиры от площади

Коэффициенты наклона прямой:

- для $\theta = 0,75$: $\beta_1^\theta = 3{,}362$;
- для $\theta = 0,5$: $\beta_1^\theta = 2{,}654$;
- для $\theta = 0,25$: $\beta_1^\theta = 1{,}956$.

Квантильная регрессия позволяет взглянуть на данные по-другому. Например, под линией регрессии, соответствующей $\theta = 0,75$ располагаются 75% рассматриваемых наблюдений, то есть 225 объектов. Таким образом, получается, что для «дорогих» домов в данном списке при увеличении площади на единицу стоимость дома возрастает примерно на $3,362$ денежные единицы. Аналогичным образом для «дешевых» домов при увеличении площади на единицу стоимость дома возрастает примерно на $1,956$ денежные единицы.

Заключение

В процессе выполнения выпускной квалификационной работы были разобраны метод наименьших квадратов, метод наименьших сумм модулей, метод наименьшего максимального модуля, квантильная регрессия и их сведение к задачам линейного программирования. Был проведен сравнительный анализ данных методов, из которых были выбраны самые подходящие в условиях конкретных задач.

На языке Python были написаны классы, реализующие вышеперечисленные методы. Классы полностью подготовлены к встраиванию в программы. Проведено несколько вычислительных экспериментов, позволивших оценить работу классов. Время работы программ также может зависеть от архитектуры компьютера, на котором проводится вычислительный эксперимент, и от операционной системы. Была разработана необходимая документация для использования классов.

Список литературы

1. Доугерти К. Введение в эконометрику. — М.: ИНФРА-М, 1999.
2. Магнус Я. Р., Катышев П. К., Пересецкий А. А. Эконометрика. – М.: Дело, 2004.
3. Постникова Е. Квантильная регрессия. — Новосибирск: НГУ, 2006
4. Фролов А. Н. Краткий курс теории вероятностей и математической статистики: Учебное пособие. — СПб.: Лань, 2017.
5. Anderson David R., Sweeney Dennis J., Williams Thomas A. Statistics for business and economics.South-Western, Cengage Learning, 2011.
6. Charnes A., Lemke C. E. Computational Theory of Linear Programming: The Bounded Variables Problem. Graduate School of Industrial Administration, Carnegie Institute of Technology, 1954.
7. Dantzig, G. B. Upper Bounds, Secondary Constraints, and Block Triangularity in Linear Programming // Econometrica, Apr., 1955. Vol. 23, No. 2. P. 174–183.
8. Karst O. J. Linear Curve Fitting Using Least Deviations // Journal of the American Statistical Association, 1958. Vol. 53, No. 281.
9. Koenker R., Basset G. Regression Quantiles // Econometrica, Jan., 1978, Vol. 46, No. 1. P. 33–50.
10. Koenker R., Hallock K. F. Quantile Regression // Journal of Economic Perspectives, Vol. 15, No. 4 (Fall., 2001), P. 143–156.
11. Sposito V. A. Minimizing the maximum absolute deviation // ACM SIGMAP Bulletin, Feb., 1976. Issue 20. P. 51–53.
12. Wagner Harvey M. Linear Programming Techniques for Regression Analysis // Journal of the American Statistical Association, Mar., 1959. Vol. 54, No. 285. P. 206–212.
13. Kaggle: https://www.kaggle.com/harlfoxem/housesalesprediction (Дата обращения 17.05.19)
14. Someka: https://www.someka.net/excel-template/fortune-500-excel-list (Дата обращения 17.05.19)

Приложения

В качества языка программирования был выбран язык Python 3.6.7. Для решения задач ЛП используется библиотека scipy.optimize, функция linprog(method='simplex').

1. Код класса для решения квантильной регрессии

Листинг 1: Quantile regression

```
import numpy as np
from scipy.optimize import linprog

class QuantileRegression(object):
    def __init__(self, theta, use_bias=False):
        self.use_bias = use_bias
        self.theta = theta
        self.X = None
        self.Y = None
        self.beta_1 = 0
        self.beta_2 = 0
        self.success = False

    def fit(self, X, Y):
        self.X = X
        self.Y = Y
        size = len(self.X)

        # Определение матрицы и векторов задачи ЛП
        A = np.zeros((size, 2*size + 2), dtype=np.float)
        shift = 2
        for i in range(size):
```

```
23              A[i][0] = X[i]
24              A[i][1] = 1
25              A[i][shift] = 1
26              A[i][shift + 1] = -1
27              shift += 2
28
29          d = Y
30          c = []
31          c.append(0)
32          c.append(0)
33          for i in range(len(X)):
34              c.append(self.theta)
35              c.append(1 - self.theta)
36          c = np.array(c, dtype=np.float)
37
38          # Знаковые ограничения переменных задачи ЛП
39          x_bounds = [(0, None)]*(2*len(X) + 2)
40          x_bounds[0] = (None, None)
41          x_bounds[1] = (None, None)
42
43          # Для того чтобы искать линию регрессии,
44          # выходящую из начала координат,
45          # необходимо из матрицы и векторов
46          # удалить элементы, отвечающие за смещение.
47          if not self.use_bias:
48              # Удаление второго столбца матрицы A
49              A = np.delete(A, 1, 1)
50              # Удаление второго элемента вектора c
51              c = np.delete(c, 1)
52              # Удаление ограничения на смещение
53              del x_bounds[1]
```

```
54
55          # Решение задачи линейного программирования
56          optimize_res = linprog(c, A_eq = A, b_eq = d,
57                                 bounds = x_bounds,
58                                 method='simplex')
59          if optimize_res.success:
60              self.beta_1 = optimize_res.x[0]
61              if self.use_bias:
62                  self.beta_2 = optimize_res.x[1]
63              self.success = True
64          else:
65              print(optimize_res)
```

2. Код класса для решения MHMM

Листинг 2: Minimizing the maximum absolute deviation

```
1  import numpy as np
2  from scipy.optimize import linprog
3
4  class MinMaxAbsoluteDeviation(object):
5      def __init__(self,use_bias=False):
6          self.use_bias = use_bias
7          self.X = None
8          self.Y = None
9          self.k = 0
10         self.b = 0
11         self.success = False
12
13     def fit(self, X, Y):
14         self.X = X
15         self.Y = Y
```

```
16          size = len(self.X)
17
18          # Определение матрицы и векторов задачи ЛП
19          A = np.zeros((2*size, 3), dtype=float)
20          for i in range(size):
21              A[i][0] = X[i]
22              A[i][1] = 1
23              A[i][2] = -1
24
25          for i in range(size, 2*size):
26              A[i][0] = -X[i - size]
27              A[i][1] = -1
28              A[i][2] = -1
29
30          d = np.concatenate((Y, -Y), axis=0)
31          c = np.array([0,0,1])
32          # Знаковые ограничения переменных задачи ЛП
33          x_bounds = [(None, None)]*3
34          x_bounds[2] = (0, None)
35
36          # Для того чтобы искать линию регрессии,
37          # выходящую из начала координат,
38          # необходимо из матрицы и векторов
39          # удалить элементы, отвечающие за смещение.
40          if not self.use_bias:
41              # Удаление второго столбца матрицы A
42              A = np.delete(A, 1, 1)
43              # Удаление второго элемента вектора c
44              c = np.delete(c, 1)
45              # Удаление ограничения на смещение
46              del x_bounds[1]
```

```
47
48          # Решение задачи линейного программирования
49          optimize_res = linprog(c, A_ub = A,b_ub = d,
50                                 bounds = x_bounds,
51                                 method='simplex')
52
53          if optimize_res.success:
54              self.k = optimize_res.x[0]
55              if self.use_bias:
56                  self.b = optimize_res.x[1]
57              self.success = True
58          else:
59              print(optimize_res)
```

3. Инструкции к классам

Пусть имеется выборка (x_i, y_i), $i = 1, \ldots, n$.

Использование класса QuantileRegression {1}:

При помощи данного класса осуществляется поиск коэффициентов квантильной регрессии. Файл с классом кладется в папку проекта с именем «quantiles.py».

- Импорт класса:
  import quantiles

- Подготовка данных:
  X — вектор, состоящий из значений x_i, $i = 1, \ldots, n$;
  X — объект типа 'numpy.ndarray';
  форма объекта X: (n,).

  Y — вектор, состоящий из значений y_i, $i = 1, \ldots, n$;
  Y — объект типа 'numpy.ndarray';
  форма объекта Y: (n,).

theta — порядок квантили
theta — объект типа 'int'.

use_bias — объект типа 'bool';
если use_bias = False, то смещение равно 0;
иначе считаем со смещением.

- Работа с классом:
  Сначала необходимо инициализировать переменные theta и use_bias:
  theta = 0.75
  use_bias = True

  Создаем экземпляр класса:
  clf = QuantileRegression(use_bias, theta)

  Обучаем модель:
  clf.fit(X,Y)
  print(clf.beta_1)
  print(clf.beta_2)

- Пример рабочей программы:

Листинг 3: Пример использования класса QuantileRegression

```
import numpy as np
import quantiles

X = np.array([1,2,3,4])
Y = np.array([4,3,2,1])
theta = 0.75
use_bias = True
clf = quantiles.QuantileRegression(use_bias, theta)
clf.fit(X,Y)
print(clf.beta_1)
print(clf.beta_2)
```

Использование класса MinMaxAbsoluteDeviation {2}:
При помощи данного класса осуществляется поиск коэффициентов линейной регрессии методом наименьшего максимального модуля. Файл с классом кладется в папку проекта с именем «mnmm.py».

- Импорт класса:
  import mnmm

- Подготовка данных:
  Аналогично предыдущему классу.

- Работа с классом:
  Сначала необходимо инициализировать переменную use_bias:
  use_bias = False

  Создаем экземпляр класса:
  clf = MinMaxAbsoluteDeviation(use_bias)

Обучаем модель:

clf.fit(X,Y)

print(clf.k)

print(clf.b)

- Пример рабочей программы:

Листинг 4: Пример использования класса MinMaxAbsoluteDeviation

```
import numpy as np
import mnmm

X = np.array([1,2,3,4])
Y = np.array([4,3,2,1])
use_bias = False
clf = mnmm.MinMaxAbsoluteDeviation(use_bias)
clf.fit(X,Y)
print(clf.k)
print(clf.b)
```