

Федеральное государственное автономное образовательное  
учреждение высшего образования  
Национальный исследовательский университет  
«Высшая школа экономики»

Факультет экономических наук  
Образовательная программа "Экономика"

Курсовая работа  
на тему

# Байесовские модели для прогнозирования многомерных временных рядов

Выполнила студентка группы БЭК 151, 3 курса,  
Кожевина Анна Владимировна

Научный руководитель:

Демешев Борис Борисович,

Старший преподаватель, факультет экономических наук, департамент прикладной экономики

Москва, 2018

# Содержание

<b>1</b>	<b>Введение</b>	<b>3</b>
1.1	План работы . . . . .	3
<b>2</b>	<b>Теоретическая часть</b>	<b>4</b>
2.1	Векторные авторегрессии . . . . .	4
2.2	Байесовский подход . . . . .	4
2.3	Основные априорные распределения . . . . .	5
2.4	Получение выборки из распределения . . . . .	8
2.5	Построение прогноза . . . . .	9
2.6	Качество прогноза . . . . .	9
<b>3</b>	<b>Практическая часть</b>	<b>10</b>
3.1	Оценка эффективности прогнозов . . . . .	13
<b>4</b>	<b>Результаты и выводы</b>	<b>16</b>
<b>5</b>	<b>Список литературы</b>	<b>18</b>

# 1 Введение

Каждый день люди сталкиваются с необходимостью принимать решения, последствия от которых в момент принятия решения неизвестны. Примеров таких задач множество: от необходимости решить, сколько килограммов сахара купить в магазине, до определения планов макроэкономической политики государства на ближайшие десять лет. Как принимаются решения с условиях неопределенности? Чаще люди опираются на свою интуицию. Намного реже люди пытаются построить “модель”, предсказывающую будущее. Логично, что перед покупкой сахара нет необходимости тщательно просчитывать все аспекты и вероятности, влияющие на будущие цены и спрос на сахар. Но когда речь идет, например, о решении о покупке машины или определении будущего целого государства, имеет смысл приложить больше усилий и учесть при принятии решения некоторое множество факторов, которые могут влиять на результат.

Конечно, не каждая такая “модель” представляет собой непосредственно расчеты с учетом всех рисков или программу, которая принимает решения. В повседневной жизни иллюстрацией такого метода принятия решений является то, что люди опираются на предыдущий опыт подобных решений, опыт своих знакомых или свои представления о том, как должно определяться будущее состояние некоторых явлений. Данная работа является математизированной и формализованной реализацией такой модели принятия решения: будет показано как, опираясь на предположение о правилах поведения некоторой величины, можно строить прогнозы о ее будущих значениях.

## 1.1 План работы

В первой, теоретической, части работы будет рассмотрен байесовский подход к решению задач прогнозирования. Сначала будет показана суть векторных авторегрессий и их отличие от байесовских векторных авторегрессий. Далее будут рассмотрены самые популярные априорные распределения и их классификация. После определения всех необходимых “пререквизитов”, можно переходить непосредственно к построению прогнозов и оценке результатов.

Вторая, практическая, часть будет посвящена непосредственно применению полученных знаний на реальных данных, а также оценку точности построенных прогнозов. Прогнозирующий код будет реализован на языке программирования Python с использованием библиотеки `ruflux`.

В заключительной части проводится анализ результатов проделанной работы, выявляются направления для дальнейшего развития темы и последующих работ.

## 2 Теоретическая часть

### 2.1 Векторные авторегрессии

Итак, пусть имеется несколько векторов-значений целевой многомерной переменной:  $y_i = (y_{i1}, y_{i2}, \dots, y_{im})'$ , весь массив информации об этой переменной:  $Y = [y_1, y_2, \dots, y_T]'$ .

Пусть  $\varepsilon \sim N(0, \Sigma)$  — вектор случайных ошибок той же размерности, что и  $y_i$ ,  $\Phi_{const} = (c_1, \dots, c_m)$  — вектор констант той же размерности, а  $\Phi_j$  — авторегрессионные матрицы размера  $m \times m$ .

Векторной авторегрессией (VAR) назовем:

$$y_i = \Phi_{const} + \sum_{k=1}^p \Phi_k y_{i-k} + \varepsilon \quad (1)$$

Введя дополнительные обозначения  $x_i = [y'_{i-1}, \dots, y'_{i-k}, 1]'$  и  $\Phi = [\Phi_1, \dots, \Phi_k, \Phi_{const}]'$ , получим сокращенную форму VAR:

$$y_t = \begin{bmatrix} \Phi_1 \\ \Phi_2 \\ \dots \\ \Phi_{t-p} \\ \Phi_{const} \end{bmatrix} (y_{t-1} \quad y_{t-2} \quad \dots \quad y_{t-p} \quad 1) = \Phi' x_t + \varepsilon_t \quad (2)$$

Далее, можно ввести более общие обозначения,  $Y = [y_1, y_2, \dots, y_T]'$ ,  $X = (x_1, x_2 \dots x_T)'$  и  $E = [\varepsilon_1, \varepsilon_2, \dots, \varepsilon_T]'$ , чтобы получить наиболее компактную запись VAR:

$$Y = X\Phi + E \quad (3)$$

Классическая (частотная) оценка VAR может быть проведена последовательным применением метода наименьших квадратов к каждому из уравнений системы.

### 2.2 Байесовский подход

В отличие от эконометрических моделей, в которых параметры моделей считаются истинными и константными, в основе байесовского подхода лежит предположение о случайности параметров и их принадлежности некоторому распределению. Таким образом, задача байесовского подхода — в поиске  $p(\Phi, \Sigma|Y)$  с помощью предположений об априорном распределении  $p(\Phi, \Sigma)$  и функции максимального правдоподобия  $p(y|\Phi, \Sigma)$ . Это можно сделать с помощью обычной формулы условной вероятности:

$$p(\Phi, \Sigma|Y) = \frac{p(\Phi, \Sigma)p(Y|\Phi, \Sigma)}{p(Y)} \quad (4)$$

$p(Y)$  — функция, которая не зависит от наших переменных, поэтому для любого конкретного  $Y$  принимает некоторое константное значение. Поскольку это значение не влияет на меняющиеся переменные, его можно не учитывать в анализе:

$$p(\Phi, \Sigma|Y) \propto p(\Phi, \Sigma)p(Y|\Phi, \Sigma) \quad (5)$$

Так как  $\varepsilon_i \sim N(0, \Sigma)$ , то функцию правдоподобия для оценки максимального правдоподобия коэффициентов можно представить как

$$p(Y|\Phi, \Sigma) \propto |\Sigma|^{-\frac{T}{2}} \exp \left\{ -\frac{1}{2} [\Sigma^{-1}(Y - X\Phi)'(Y - X\Phi)] \right\} \quad (6)$$

Задача байесовского прогнозирования будет решаться по следующему плану:

- сгенерировать параметры апостериорного распределения с помощью параметров априорного распределения (в работе — алгоритмом Гиббса или Монте-Карло);
- сгенерировать вектор ошибок на текущей итерации;
- посчитать с помощью сгенерированных значений целевую переменную;
- перейти к следующей итерации.

Далее рассмотрим все теоретические аспекты, необходимые для построения прогнозов.

## 2.3 Основные априорные распределения

Если апостериорное распределение того же семейства, что и априорное, то речь идет о сопряженном априорном распределении, в ином случае априорное распределение будет независимым. Знание таких распределений позволяет значительно упростить расчеты в формуле Байеса и найти решение аналитически сведением к операциям над параметрами распределений. В этой части работы будут рассмотрены наиболее известные априорные распределения: независимое нормальное-обратное Уишарта и сопряженное нормальное-обратное Уишарта. Также будет приведено более подробное описание частных случаев: распределения Миннесоты и Джеффриса для обоих случаев.

### Независимое нормальное-обратное Уишарта априорное распределение

Особенность этого распределения в том, что матрица параметров и ковариационная матрица ошибок независимы. Также здесь нет ограничений на вид ковариационной матрицы параметров.

Задается такое распределение следующим образом:

$$\begin{cases} \phi \sim N(\underline{\phi}; \Xi) \\ \Sigma \sim IW(\underline{S}, \underline{\nu}) \\ \phi \text{ и } \Sigma \text{ независимы} \end{cases} \quad (7)$$

Для этого распределения используют сэмплирование по Гиббсу, чтобы избавиться от условности вычисленного аналитически апостериорного распределения. Эта процедура будет более подробно описана в следующем разделе о выводе апостериорного распределения. В работе Karlsson (2012) показано, что условные апостериорные распределения имеют вид:

$$\begin{cases} \phi | \Sigma, Y \sim N(\bar{\phi}; \bar{\Xi}) \\ \Sigma | \phi, Y \sim IW(\bar{S}, \bar{\nu}) \end{cases} \quad (8)$$

Где гиперпараметры:

$$\begin{aligned} \bar{\nu} &= \underline{\nu} + T \\ \bar{S} &= \underline{S} + E'E, \text{ где } E = Y - X\Phi \\ \bar{\Xi} &= (\Xi^{-1}\underline{\phi} + \text{vec}(X'Y\Sigma^{-1})) \end{aligned}$$

Распределения Миннесоты и нормальное-Джеффриса можно представить как две стороны распределения Уишарта: в первом случае матрица  $\Sigma$  считается известной, во втором она имеет неинформативное распределение. Рассмотрим их более подробно. являются

**Априорное распределение Миннесоты** Распределение Миннесоты — частный случай многомерного нормального распределения Уишарта с независимыми гиперпараметрами, диагональной ковариационной матрицей ошибок. Его создание связано с желанием ученых отразить нестационарность макроэкономических рядов, а также соответствие их изменения и закона случайного блуждания.

$$\begin{cases} \phi \sim N(\underline{\phi}; \Xi) \\ \Sigma = \text{const} \end{cases} \quad (9)$$

Это частный случай при  $\underline{S} = (\underline{\nu} - m - 1) \cdot \Sigma$  и  $\underline{\nu} \rightarrow \infty$

Апостериорное распределение будет также относиться к классу нормальных. Его параметры будут зависеть от исходных данных модели, поэтому в литературе предлагается воспользоваться численным методом Монте-Карло, предполагающим извлечение значений из нормального распределения с указанными характеристиками. Более подробно этот метод также будет описан в следующей главе.

Особенностями априорного распределения Миннесоты являются относительная простота задания и реализации, а также слабое для реальных данных предположение о неизменности ковариационной матрицы.

### Независимое нормальное-Джеффриса априорное распределение

$$\begin{cases} \phi \sim N(\underline{\phi}; \Xi) \\ \Sigma \sim |\Sigma|^{-\frac{m+1}{2}} \\ \phi \text{ и } \Sigma \text{ независимы} \end{cases} \quad (10)$$

Это частный случай при  $\underline{S} = \underline{\nu}^{\frac{1}{m}} \cdot I$  и  $\underline{\nu} \rightarrow 0$ . Это распределение не является информативным. Такие распределения используются из-за того, что они не зависят от набора параметров, в котором описано параметрическое пространство. Однако использование этого распределения нарушает сильный принцип максимального правдоподобия, потому что итоговая оценка параметров зависит не только от вероятности выборки, но и от множества возможных исходов процесса, в ходе которого была получена выборка. Соответственно, априорные вероятности Джеффриса, а, следовательно, и использующие их выводы, могут быть разными для двух экспериментов, использующих один и тот же набор параметров, и даже одну и ту же функцию правдоподобия, а это нарушение сильной формулировки принципа максимального правдоподобия.

### Сопряженное нормальное-обратное Уишарта распределение:

Распределение задается следующим образом:

$$\begin{cases} \phi|\Sigma \sim N(\phi, \Sigma \otimes \underline{\Omega}) \\ \Sigma \sim IW(\underline{S}, \underline{\nu}) \end{cases} \quad (11)$$

Негативно на результатах исследования может отразиться особенность этого распределения: между дисперсиями различных коэффициентов модели для разных уравнений могут возникать зависимость, то есть, объявляя ограничения на некоторые параметры, исследователь может ограничить и остальные параметры, а это нарушает целостность и ценность построенной модели.

Для этого случая существуют выведенные формулы искомого распределения параметров, поэтому предлагается использовать метод Мотне-Карло, вместо более трудоемкого алгоритма Гиббса.

### Распределение Миннесоты при $\Xi = \Sigma \otimes \Omega$

Интересно, что распределение Миннесоты — частный случай независимого нормально-обратного распределения Уишарта, а частный случай распределения Миннесоты — частный

случай сопряженного нормально-обратного распределения Уишарта. В этом случае для получения выборки рекомендуют использовать как алгоритм Гиббса, так и алгоритм Монте-Карло.

### Сопряженное нормальное-Джеффриса априорное распределение

$$\begin{cases} \phi|\Sigma \sim N(\underline{\phi}, \Sigma \otimes \underline{\Omega}) \\ \Sigma \sim |\Sigma|^{-\frac{m+1}{2}} \end{cases} \quad (12)$$

## 2.4 Получение выборки из распределения

### Алгоритм Гиббса

Алгоритм на каждом шаге берет одну случайную величину и выбирает её значение при условии фиксированных остальных, поэтому для него не нужен явный вид совместного распределения необходимых величин — нужно знать только условные вероятности для каждой переменной, входящей в распределение. Этот метод удобно применять в случаях, когда совместное распределение случайных величин неизвестно явно и проще оказывается вычислить условные вероятности.

- в начале каждой итерации для некой случайной величины задаются исходные значения, возможно, на усмотрение исследователя;
- генерируется случайная величина из условного распределения для одной из переменных (индекс обычно выбирается последовательно), остальные остаются теми же, что в предыдущей итерации;
- поскольку начальные значения неких случайных величин были неизвестны и задавались на усмотрение исследователя, первоначальные итерации относятся к «периоду прожига» и не учитываются;
- в итоговую выборку из апостериорного распределения входят только последние итерации.

### Метод Монте-Карло

В этом методе происходит моделирование целевого процесса с помощью генерирования случайных величин. Такая процедура повторяется несколько раз, затем эмпирически определяются вероятностные характеристики исследуемого процесса.

- определяется функция распределения каждой переменной, которая оказывает влияние на целевую переменную;
- с помощью генерирования случайных значений исследуемой величины происходит моделирование целевого процесса;



- повторение предыдущего шага большое количество раз (например, больше 1000);
- эмпирически определяются вероятностные характеристики целевой переменной.

## 2.5 Построение прогноза

В этой работе рассматривается основная цель байесовских авторегрессий — построение прогнозов: точечных, интервальных или прогнозов плотности. В прикладных работах, использующих BVAR, строятся точечные прогнозы и прогнозы плотности. В практической части будет показана реализация точечных и интервальных прогнозов.

Важное понятие в процессе прогнозирования — апостериорная прогнозная плотность  $p(y_{T+1}, \dots, y_{T+N})$ , или прогноз на  $N$  периодов после периода  $T$ , до которого значения целевой переменной известны. Функция может быть выведена с помощью формулы вероятности произведения событий):

$$p(y_{T+1}, \dots, y_{T+N}|y_T) = \int p(y_{T+1}, \dots, y_{T+N}|y_T, \phi) p(\phi|y_T) d\phi \quad (13)$$

Первый множитель под интегралом — функция плотности будущих наблюдений при условии фиксированных параметров  $\phi$  и данных вплоть до периода  $T$ , второй — апостериорная плотность параметров. При этом для простоты можно считать каждое наблюдение из  $\{y_{T+1}, \dots, y_{T+N}\}$  независимой реализацией прогнозного распределения.

Повторим порядок действий для решения основной задачи работы — построения прогноза:

- сгенерировать параметры  $\Phi, \Sigma$  апостериорного распределения с помощью параметров априорного распределения (чаще всего — алгоритмом Гиббса или Монте-Карло);
- сгенерировать вектор ошибок  $E$  для текущей итерации;
- посчитать с помощью сгенерированных значений целевую переменную  $Y = X\Phi + E$ ;
- перейти к следующей итерации.

## 2.6 Качество прогноза

Когда исследователю необходимо не только построить прогноз, но и оценить его качество, проверку можно осуществить на последнем отрезке временного ряда, который уже имеется в распоряжении. Есть два метода провести такую оценку: по сдвигающейся или по

растущей выборке. В первом случае фиксируется количество наблюдений, на каждом шаге начало временного отрезка сдвигается, во втором случае фиксируется начало временного отрезка, на каждой итерации отрезок увеличивается.

Для оценки качества точечного прогноза чаще всего используют среднеквадратичную ошибку прогноза (MSFE, mean squared forecasting error) или квадратный корень из нее. Для периода прогноза на период  $T + h$  переменной  $y_i$ , сделанного в периоде  $T$ :

$$MSFE_{i,h} = \frac{1}{N_h} \sum_T (y_{i,T+h|T}^{predicted} - y_{i,T+h|T})^2$$

На этом введение в теоретическую базу, необходимую для погружения в тему байесовского прогнозирования временных рядов, закончен. Далее, в практической части, такое прогнозирование будет реализовано и визуализировано, а также будет сделана оценка качества модели.

### 3 Практическая часть

В этой части будет показано, как с помощью библиотеки `pyflux` можно построить прогноз и оценить его качество. Сначала импортируем необходимые библиотеки:

```
import pyflux as pf #работа с временными рядами
import pandas as pd #работа с датасетами
import numpy as np #математический пакет
import pylab #визуализация данных
import matplotlib.pyplot as plt #визуализация данных
%matplotlib inline
from sklearn.metrics import mean_squared_error as mse
#расчет метрик качества
import quandl #загрузка данных
```

Данные будем брать на ресурсе `quandl.com`. Использовать будем данные Федерального Резерва США о состоянии экономики: квартальные показатели с 1975 года по ВВП, M1 Money Stock, ставка по федеральным фондам (Effective Federal Funds Rate), уровень безработицы (Civilian Unemployment Rate) и коэффициент личных сбережений (Personal Saving Rate). Импорт данных происходит с помощью API, сразу формируем из отдельных рядов общий датасет:

```
data_1 = quandl.get('FRED/GDP', start_date="1975-01-01", end_date="2018-03-31",
collapse = "quarterly")
```

```

data_2 = quandl.get('FRED/M1', start_date="1975-01-01", end_date="2018-03-31",
collapse = "quarterly")
data_3 = quandl.get('FRED/DFF', start_date="1975-01-01", end_date="2018-03-31",
collapse = "quarterly")
data_4 = quandl.get('FRED/UNRATE', start_date="1975-01-01", end_date="2018-03-31",
collapse = "quarterly")
data_5 = quandl.get('FRED/PSAVERT', start_date="1975-01-01", end_date="2018-03-31",
collapse = "quarterly")

dates = data_4.index
the_data = pd.DataFrame(index = dates,
columns = ['data_1', 'data_2', 'data_3', 'data_4', 'data_5'])
the_data['data_1'] = data_1
the_data['data_2'] = data_2
the_data['data_3'] = data_3
the_data['data_4'] = data_4
the_data['data_5'] = data_5

```

Масштабы всех рядов разные, как видно из первого изображения, поэтому для наглядности визуализируем ряды по отдельности:

```
the_data.head()
```

	data_1	data_2	data_3	data_4	data_5
Date					
1975-03-31	1619.553	273.4	5.39	8.6	12.3
1975-06-30	1656.448	283.3	5.97	8.8	13.9
1975-09-30	1713.812	286.1	6.22	8.4	12.6
1975-12-31	1765.867	287.2	5.37	8.2	11.5
1976-03-31	1824.509	294.2	5.07	7.6	11.8

Рис. 1: Данные модели



Рис. 2: Ряды модели

Векторные авторегрессии уже встроены в пакет `rflux`, поэтому создать нужную модель достаточно просто: нужно лишь указать количество лагов, априорное распределение, значение его параметров и количество периодов, на которые будет происходить прогнозирование. Очень удобно то, что в библиотеке уже реализовано обрано-нормальное распределение Уишарта, а значит можно простыми методами реализовать задачу работы. Также библиотека предлагает 3 самых популярных метода решения задачи: метод максимального правдоподобия (MLE), метод наименьших квадратов (OLS) и метод максимального правдоподобия со штрафами (PML). В функции `pf.InverseWishart` первый аргумент обозначает параметр  $\nu$ , а второй обозначает  $\Xi$ . Итак, зададим две модели, которые будут отличаться  $\nu$ , чтобы потом выяснить, на каком значении предсказание оказывается лучше. Выведем предсказания двух моделей на последние 30 периодов и, для начала, сравним результаты визуально:

```
model_1 = pf.VAR(data=the_data, lags = 20)
my_covariance_prior = np.eye(50)
model_1.adjust_prior(0, pf.InverseWishart(500, my_covariance_prior))
model_1.plot_predict_is(h = 30, fit_method = 'MLE')
```

```

model_2 = pf.VAR(data=the_data, lags = 20)
my_covariance_prior = np.eye(50)
model_2.adjust_prior(200, pf.InverseWishart(500, my_covariance_prior))
model_2.plot_predict_is(h = 30, fit_method = 'MLE')

```

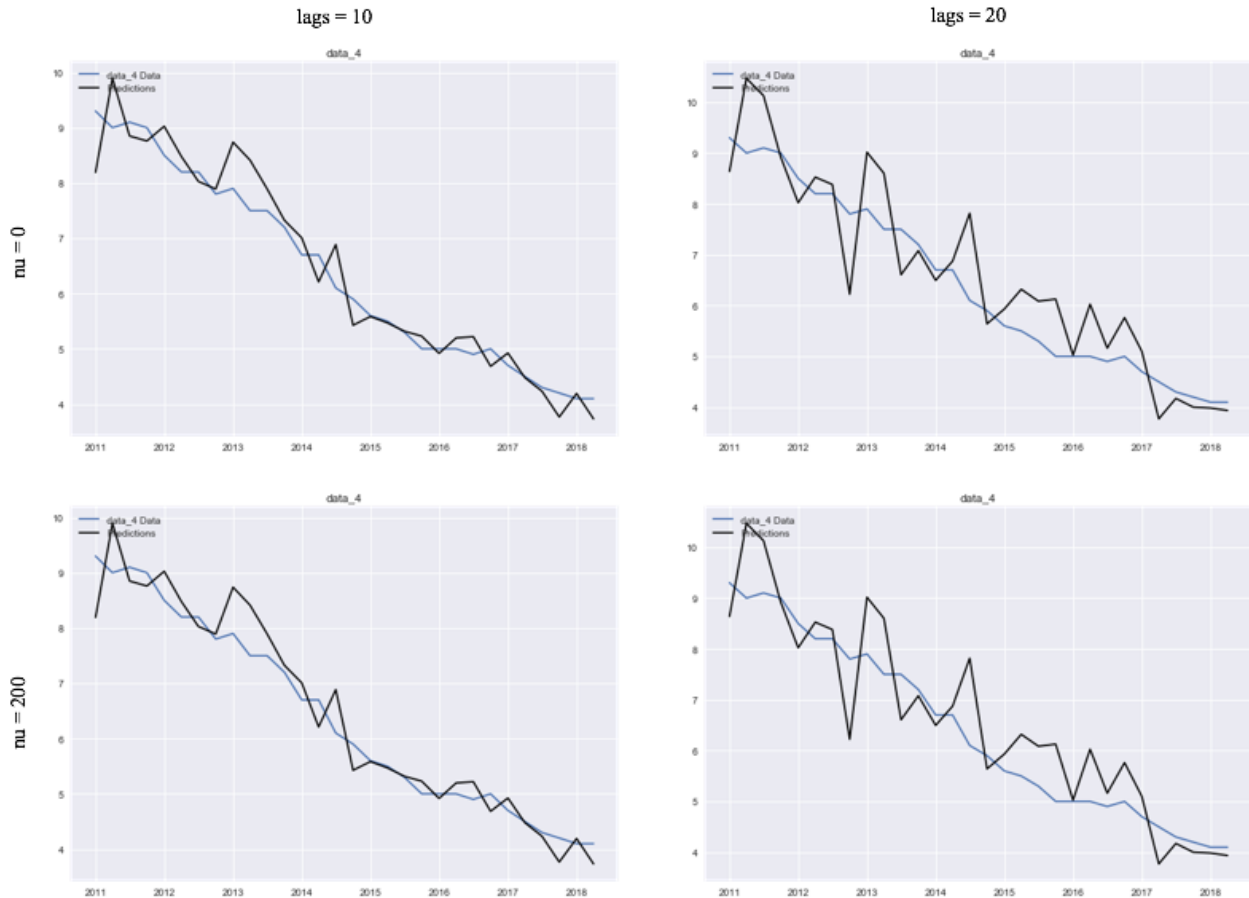


Рис. 3: Параметр  $\nu$  и количество лагов

Как видно, от изменения параметра  $\nu$  предсказание не меняется, тогда как при изменении количества лагов качество может значительно меняться. В данном случае при уменьшении числа лагов качество ухудшается.

### 3.1 Оценка эффективности прогнозов

Далее я реализую для каждого из рядов метрику качества, описанную в теоретической части — корень из среднеквадратичной ошибки прогноза. Отметим, что значительным минусом квадратичной ошибки является ее чувствительность к масштабам значений ряда.

Поэтому я ввожу дополнение к этой метрике — отношение корня среднеквадратичной к среднему значению ряда. Это поможет в каком-то смысле "нормировать" результат. Для оценки общего качества предлагается брать среднее значение итоговой "нормированной" метрики. Рассчитаем сначала общую метрику:

```
def mean_norm_msfe(h):  
    prediction = model.predict_is(h=h)  
    true_data = the_data[-h:]  
    quality = []  
    for i in true_data.columns:  
        quality.append((mse(true_data[i],  
                             prediction[i]))**0.5/np.mean(true_data[i]))  
    return sum(quality)/len(quality)
```

Далее эмпирически оценим, на каком количестве периодов предсказания метрика качества оказывается лучше всего. Для этого изобразим график качества в зависимости от периода прогнозирования:

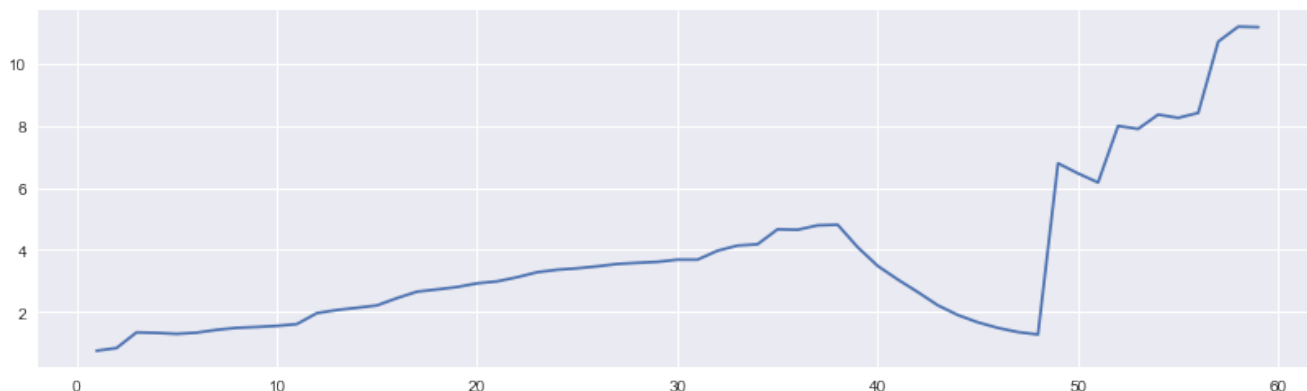


Рис. 4: Общее качество модели от периода прогнозирования

Итак, график показывает понятный вывод — чем дальше мы хотим прогнозировать, тем это сложнее и, соответственно, тем хуже качество прогноза. Однако при прогнозировании на 50 периодов наблюдается низкое снижение доли ошибки в среднем значении ряда, что, скорее всего, является ошибочным значением.

Теперь визуализируем качество индивидуальных прогнозов для каждого ряда в зависимости от количества периодов прогнозирования.

```
def one_var_msfe(var):  
    lst = []
```

```

for h in range(1, 50):
    prediction = model.predict_is(h=h)[var]
    true_data = the_data[-h:][var]
    lst.append((mse(true_data, prediction)**0.5/np.mean(true_data)))
df = pd.DataFrame(index = range(1, 50), data = lst)
plt.figure(figsize=(14, 4))
pylab.plot(df, label = var)
pylab.legend()

```

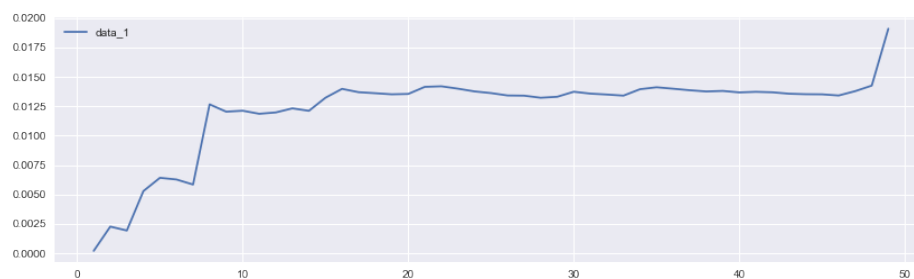


Рис. 5: Качество для 1 ряда от периода прогнозирования

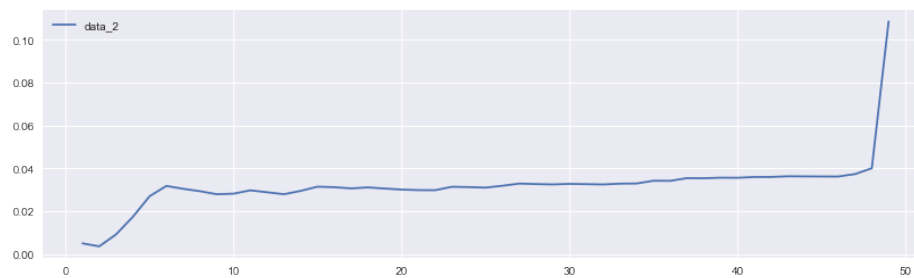


Рис. 6: Качество для 2 ряда от периода прогнозирования

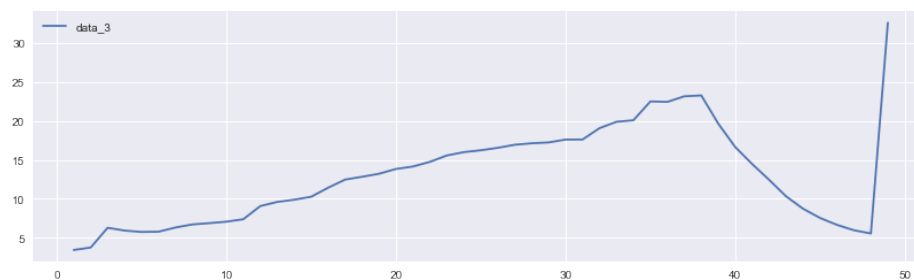


Рис. 7: Качество для 3 ряда от периода прогнозирования

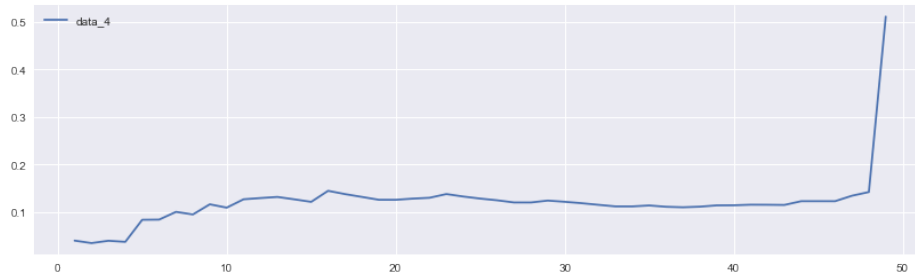


Рис. 8: Качество для 4 ряда от периода прогнозирования

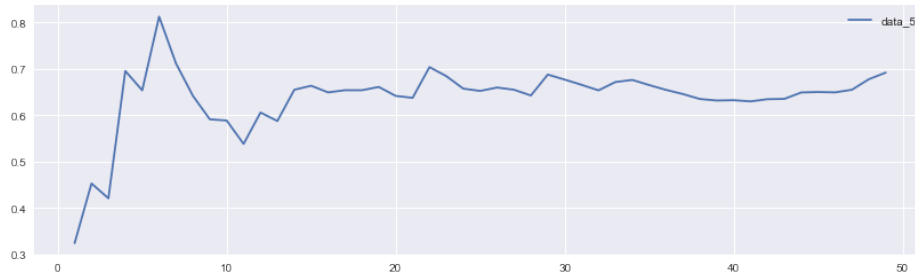


Рис. 9: Качество для 5 ряда от периода прогнозирования

Как видно, для каждого ряда результат получился индивидуальным, и выявить общие закономерности качества прогноза от периода прогнозирования не получится.

## 4 Результаты и выводы

Данная работа на этом этапе по большей части имеет ознакомительный характер, которого достаточно для первого погружения в тему. В ней рассмотрена суть векторных авторегрессий и идея байесовского подхода, теория, необходимая для понимания сути такого прогнозирования, а также приведена реализация этих подходов.

В качестве продолжения задачи данной работы можно выделить следующие задачи:

- более глубокий математический анализ модели
- самостоятельный вывод апостериорных распределений для популярных априорных распределений
- полная реализация алгоритмов на языке программирования python

Также хочется выделить задачи, который могут быть расширением этой работы в качестве планов для выпускной работы :

- анализ других популярных методов работы с временными рядами



- сравнение байесовских авторегрессий с другими моделями по эффективности прогнозирования и сложности реализации
- решение практических задач для бизнеса с помощью байесовских методов прогнозирования

## 5 Список литературы

- [1] Б.Б.Демешев, О.А.Малаховская: Картографирование BVAR // Прикладная эконометрика, 2016, т.43, с.118–141
- [2] М.Г.Тиунова: Влияние монетарной политики на динамику реального сектора экономики в России // ВЕСТН. МОСК. УН-ТА. СЕР. 6. ЭКОНОМИКА. 2017. № 3
- [3] Karlsson, S. (2012). Forecasting with Bayesian Vector Autoregressions. Working Papers 2012: 12. Orebo Univercity, School of Business