

## **Курсовая работа по дисциплине "Эконометрика"**

**Цель работы:** Закрепление теоретических знаний и приобретение практических навыков построения, анализа и верификации эконометрических моделей.

**Общие требования**

**Объем работы:** 25-35 страниц

**Варианты заданий (Исходный код для генерации данных для расчета приведен в конце МУ к курсовой работе)**

Чтобы получить значения исходных данных для выполнения задания необходимо скопировать и запустить исходный код (в соответствии с вашим вариантом), который приведен в ПРИЛОЖЕНИИ, в среду выполнения Python со всеми предустановленными библиотеками, список которых можно найти в разделе «Рекомендации по использованию программного обеспечения».

### **Общие рекомендации по выполнению курсовой работы**

После получения исходных данных необходимо выполнить их анализ в соответствии с пунктом «Задачи», указанным в каждом варианте задания. Выполнять задание целесообразно в среде Python со всеми предустановленными библиотеками, список которых можно найти в разделе «Рекомендации по использованию программного обеспечения» или в Excel. Выбор одной из этих двух сред для анализа данных остается на усмотрение обучающегося и не влияет на итоговую оценку за курсовую работу.

В разделе «Методические указания по выполнению» приводится **ориентировочный** пример содержания курсовой работы с демонстрацией примеров решения отдельных пунктов задания на языке python. Обучающийся для выполнения своего варианта курсовой работы должен самостоятельно найти способ решения задач, поставленных в конкретном варианте курсовой работы. Для этих целей обучающийся может использовать теоретический материал, изложенный в лекциях к данному курсу, литературу,

приведенную в списке рекомендованной литературы, а также любую другую информацию, находящуюся в свободном доступе, включая образцы исходного кода и примеры решения сходных задач в Excel. Ключевым моментом, влияющим на оценивание курсовой работы является способность обучающегося правильно найти способ решения поставленной задачи, понимание сути эконометрических параметров, которые он анализирует и умение правильно анализировать полученные результаты.

**Выбор варианта:** Последние две цифры пароля определяют номер варианта.

### **Вариант 01 (последние цифры 01-05)**

**Тема:** "Анализ факторов, влияющих на заработную плату"

**Исходные данные:** Микроданные по 500 работникам:

- $Y$  - заработная плата (тыс. руб.)
- $X_1$  - образование (лет)
- $X_2$  - опыт работы (лет)
- $X_3$  - пол (0-женщины, 1-мужчины)
- $X_4$  - отрасль (коды 1-5)

**Задачи:**

1. Построить описательные статистики по всем переменным
2. Оценить линейную модель множественной регрессии
3. Проверить наличие мультиколлинеарности
4. Проанализировать остатки на гетероскедастичность
5. Построить прогноз для типичного работника

### **Вариант 02 (последние цифры 06-10)**

**Тема:** "Моделирование спроса на недвижимость"

**Исходные данные:** 100 наблюдений по продажам квартир:

- $Y$  - цена квартиры (млн руб.)
- $X_1$  - площадь ( $m^2$ )

- $X_1$  - этаж
- $X_2$  - расстояние до центра (км)
- $X_3$  - год постройки

**Задачи:**

1. Подобрать оптимальную функциональную форму модели
2. Оценить параметры выбранной модели
3. Проверить значимость коэффициентов
4. Оценить качество модели
5. Построить интервальный прогноз
- 6.

**Вариант 03 (последние цифры 11-15)**

**Тема:** "Анализ временных рядов макроэкономических показателей"

**Исходные данные:** Ежеквартальные данные за 10 лет:

- $Y$  - ВВП (трлн руб.)
- $X_1$  - инфляция (%)
- $X_2$  - курс доллара (руб.)
- $X_3$  - инвестиции (трлн руб.)

**Задачи:**

1. Проверить ряды на стационарность
2. Построить модель ARIMA для ВВП
3. Оценить модель регрессии с лагами
4. Проверить остатки на автокорреляцию
5. Построить прогноз на 4 квартала

**Вариант 04 (последние цифры 16-20)**

**Тема:** "Исследование эффективности рекламных кампаний"

**Исходные данные:** Ежемесячные данные за 3 года:

- $Y$  - объем продаж (тыс. ед.)
- $X_1$  - расходы на интернет-рекламу (млн руб.)

- $X_1$  - расходы на ТВ-рекламу (млн руб.)
- $X_2$  - сезонность (1-12)

**Задачи:**

1. Построить модель с фиктивными переменными
2. Оценить эластичность спроса по рекламе
3. Проверить наличие структурных сдвигов
4. Построить адаптивную модель Хольта-Винтерса
5. Сравнить точность прогнозов разных моделей

**Вариант 05 (последние цифры 21-25)**

**Тема:** "Анализ производственной функции предприятия"

**Исходные данные:** Данные по 50 предприятиям отрасли:

- $Y$  - объем выпуска (тыс. ед.)
- $X_1$  - численность работников (чел.)
- $X_2$  - стоимость основных фондов (млн руб.)
- $X_3$  - затраты на материалы (млн руб.)

**Задачи:**

1. Оценить линейную модель
2. Оценить функцию Кобба-Дугласа
3. Проверить гипотезу о постоянной отдаче от масштаба
4. Сравнить качество моделей
5. Рассчитать предельные продукты факторов

**Вариант 06 (последние цифры 26-30)**

**Тема:** "Исследование факторов успеваемости студентов"

**Исходные данные:** Данные по 300 студентам:

- $Y$  - средний балл за сессию
- $X_1$  - время подготовки к занятиям (час/нед)

- $X_1$  - посещаемость (%)
- $X_2$  - форма обучения (0-очная, 1-заочная)
- $X_3$  - наличие работы (0-нет, 1-да)

**Задачи:**

1. Построить модель с бинарными переменными
2. Проверить гипотезы о коэффициентах
3. Проанализировать взаимодействие факторов
4. Оценить вероятность получения высокого балла
5. Дать рекомендации по улучшению успеваемости

**Вариант 07 (последние цифры 31-35)**

**Тема:** "Моделирование потребительских расходов домохозяйств"

**Исходные данные:** Данные обследования 200 домохозяйств:

- $Y$  - расходы на питание (тыс. руб./мес)
- $X_1$  - доход домохозяйства (тыс. руб./мес)
- $X_2$  - размер семьи (чел.)
- $X_3$  - регион (1-5)
- $X_4$  - тип населенного пункта (1-город, 2-село)

**Задачи:**

1. Построить модель с группировкой данных
2. Оценить параметры для разных групп
3. Проверить однородность коэффициентов
4. Построить доверительные интервалы
5. Проанализировать различия в поведении групп

**Вариант 08 (последние цифры 36-40)**

**Тема:** "Анализ финансовых временных рядов"

**Исходные данные:** Ежедневные данные за 2 года:

- $Y$  - цена акции компании (руб.)
- $X_1$  - объем торгов (тыс. акций)

- $X_1$  - индекс РТС
- $X_2$  - цена нефти (\$)

**Задачи:**

1. Проверить ряды на единичные корни
2. Построить модель GARCH для волатильности
3. Оценить модель коррекции ошибок
4. Проверить наличие коинтеграции
5. Построить прогноз на 10 дней

**Вариант 09 (последние цифры 41-45)**

**Тема:** "Исследование детерминант безработицы в регионах"

**Исходные данные:** Данные по 80 регионам за последний год:

- $Y$  - уровень безработицы (%)
- $X_1$  - ВРП на душу населения (тыс. руб.)
- $X_2$  - инвестиции в основной капитал (% ВРП)
- $X_3$  - доля сельского населения (%)
- $X_4$  - средняя заработная плата (тыс. руб.)

**Задачи:**

1. Построить пространственную регрессию
2. Проверить на гетероскедастичность
3. Оценить модель с робастными ошибками
4. Построить карты распределения показателей
5. Дать рекомендации по региональной политике

**Вариант 10 (последние цифры 46-50)**

**Тема:** "Сравнительный анализ методов прогнозирования"

**Исходные данные:** Ежеквартальные данные за 8 лет:

- $Y$  - объем промышленного производства (индекс)
- $X_1$  - реальный эффективный курс рубля

- $X_1$  - мировая цена на нефть (\$)
- $X_2$  - индекс деловой активности

#### **Задачи:**

1. Построить регрессионную модель
2. Построить модель ARIMA
3. Построить модель VAR
4. Сравнить точность прогнозов
5. Построить комбинированный прогноз

### **Методические указания по выполнению**

#### **1. Теоретическая часть**

**Объем:** 5-8 страниц

#### **Содержание:**

- Описание экономической проблемы
- Обзор литературы по теме
- Теоретическое обоснование выбранных методов
- Математическая постановка задачи

**Пример:** для варианта 1 (заработная плата) следует описать:

- Теорию человеческого капитала
- Модель Минцера
- Методы оценки параметров
- Подходы к проверке гипотез

#### **2. Практическая часть**

**Объем:** 15-25 страниц

#### **Этапы выполнения:**

##### **2.1. Предварительный анализ данных**

```
# Пример кода для описательной статистики
import pandas as pd
import numpy as np

data = pd.read_csv('data.csv')
print(data.describe())
print(data.info())

# Построение графиков
import matplotlib.pyplot as plt
data.hist(bins=20, figsize=(15,10))
plt.show()
```

## **2.2. Построение и оценка модели**

```
# Пример оценки линейной регрессии
import statsmodels.api as sm

X = data[['X1', 'X2', 'X3']]
X = sm.add_constant(X)
y = data['Y']

model = sm.OLS(y, X).fit()
print(model.summary())
```

## **2.3. Проверка предпосылок МНК**

- Тест на нормальность остатков (Шапиро-Уилк)
- Тест на гетероскедастичность (Уайт, Бреуша-Пагана)
- Тест на автокорреляцию (Дарбина-Уотсона)
- Проверка мультиколлинеарности (VIF)



## 2.4. Анализ результатов

- Интерпретация коэффициентов
- Оценка качества модели
- Построение прогнозов
- Анализ устойчивости результатов

## 3. Пример выполнения пункта (Вариант 1)

**Задача:** проверить наличие гетероскедастичности

**Теоретическое обоснование:**

Гетероскедастичность нарушает предположение о постоянстве дисперсии ошибок, что приводит к неэффективности МНК-оценок (Лекция 6).

**Практическая реализация:**

```
# Тест Уайта
from statsmodels.stats.diagnostic import het_white

white_test = het_white(model.resid, model.model.exog)
labels = ['Test Statistic', 'Test Statistic p-value',
          'F-Statistic', 'F-Test p-value']
print(dict(zip(labels, white_test)))

# Визуальный анализ
plt.scatter(model.fittedvalues, model.resid)
plt.axhline(y=0, color='red')
plt.xlabel('Предсказанные значения')
plt.ylabel('Остатки')
plt.title('График остатков')
plt.show()
```

**Интерпретация результатов:**

"При проведении теста Уайта получено  $p\text{-value} = 0.023$ , что меньше 0.05. Следовательно, гипотеза о гомоскедастичности отвергается. На графике остатков наблюдается эффект "веера", что подтверждает наличие гетероскедастичности."

#### **4. Рекомендации по использованию программного обеспечения**

##### **Python:**

- Библиотеки: pandas, numpy, statsmodels, scikit-learn, matplotlib
- Среда: Jupyter Notebook или Google Colab

##### **Excel:**

- Надстройка "Анализ данных"
- Функции регрессионного анализа

#### **Требования к оформлению работы**

##### **1. Структура работы**

1. Титульный лист
2. Содержание
3. Введение (1-2 стр.)
4. Глава 1. Теоретические основы (3-5 стр.)
5. Глава 2. Эмпирический анализ (15-20 стр.)
6. Заключение (1-2 стр.)
7. Список литературы (10-15 источников)
8. Приложения

##### **2. Технические требования**

- Объем: 25-35 страниц
- Шрифт: Times New Roman, 14pt (основной текст)
- Интервал: 1,5
- Поля: слева - 3 см, остальные - 2 см
- Нумерация: сквозная, внизу страницы

##### **3. Оформление таблиц и графиков**

- Все таблицы и рисунки должны иметь названия
- Ссылки на них в тексте
- Единицы измерения в названиях

- Источник данных под таблицей/рисунком

#### **4. Оформление формул**

- Нумерация формул справа в круглых скобках
- Пояснение всех символов

**Пример:**

$$Y = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \varepsilon \quad (1)$$

где:

$Y$  - зависимая переменная,

$X_1, X_2$  - объясняющие переменные,

$\beta_0, \beta_1, \beta_2$  - параметры модели,

$\varepsilon$  - случайная ошибка.

#### **5. Требования к списку литературы**

- Не менее 10 источников
- Включать научные статьи, учебники, статистические сборники
- Оформление по ГОСТ Р 7.0.5-2008

#### **Критерии оценки**

**Отлично:**

- Полное соответствие требованиям
- Глубокий теоретический анализ
- Корректное применение методов
- Творческий подход к решению
- Качественное оформление

**Хорошо:**

- Незначительные недочеты в теоретической части
- Минорные ошибки в расчетах
- Удовлетворительное оформление

**Удовлетворительно:**

- Существенные пробелы в теории

- Серьезные ошибки в расчетах
- Неполное выполнение заданий

**Неудовлетворительно:**

- Работа не соответствует требованиям
- Грубые методологические ошибки
- Невыполнение основных заданий

**Рекомендуемая литература**

1. Магнус Я.Р., Катышев П.К., Пересецкий А.А. Эконометрика. Начальный курс. - М.: Дело, 2007.
2. Доугерти К. Введение в эконометрику. - М.: ИНФРА-М, 2009.
3. Вербик М. Путеводитель по современной эконометрике. - М.: Научная книга, 2008.
4. Greene W.H. Econometric Analysis. - Prentice Hall, 2018.
5. Wooldridge J.M. Introductory Econometrics: A Modern Approach. - Cengage Learning, 2019.

**ПРИЛОЖЕНИЕ**

**Исходные данные по вариантам контрольной работы**

**Вариант 01: "Анализ факторов, влияющих на заработную плату"**

## Полный набор данных (500 наблюдений):

```
import pandas as pd
import numpy as np

# Генерация реалистичных данных
np.random.seed(42)
n = 500

data = {
    'salary': np.random.normal(45, 15, n).clip(25, 120),      #
    зарплата
    'education': np.random.randint(10, 20, n),                #
    образование
    'experience': np.random.randint(0, 40, n),                 # опыт
    'gender': np.random.choice([0, 1], n, p=[0.45, 0.55]),    # пол
    'industry': np.random.choice([1, 2, 3, 4, 5], n),          #
    отрасль
    'age': np.random.randint(22, 65, n),                       #
    возраст
    'city_size': np.random.choice([1, 2, 3], n, p=[0.3, 0.5,
0.2]) # размер города
}

# Добавление зависимостей
data['salary'] = (data['salary'] +
                  data['education'] * 1.5 +
                  data['experience'] * 0.8 +
                  data['gender'] * 8 +
                  data['industry'] * 2 +
                  np.random.normal(0, 5, n))

df = pd.DataFrame(data)
print(df.head(10).to_string())
```

## Вариант 02: "Моделирование спроса на недвижимость"

### Полный набор данных (100 наблюдений):

```
# Генерация данных по недвижимости
np.random.seed(123)
n = 100

data = {
    'price': np.random.normal(6, 2, n).clip(3, 12),          # цена
    'area': np.random.randint(30, 150, n),                    #
    площадь
    'floor': np.random.randint(1, 25, n),                     # этаж
```

```

        'distance_center': np.random.uniform(0.5, 15, n),          #
расстояние до центра
        'year_built': np.random.randint(1960, 2020, n),          # год
постройки
        'rooms': np.random.randint(1, 5, n),                      #
комнаты
        'metro_distance': np.random.uniform(0.1, 3, n)           # до
метро
    }

# Формирование цены с зависимостями
data['price'] = (data['price'] +
                 data['area'] * 0.05 +
                 (25 - data['floor']) * 0.02 +
                 (1/data['distance_center']) * 10 +
                 (2020 - data['year_built']) * (-0.01) +
                 np.random.normal(0, 0.5, n))

real_estate_df = pd.DataFrame(data)
print(real_estate_df.head(10).round(2).to_string())

```

### **Вариант 03: "Анализ временных рядов макроэкономических показателей"**

**Данные за 10 лет (40 кварталов):**

```

# Создание временного ряда
dates = pd.date_range('2013-01-01', '2022-12-31', freq='Q')
n = len(dates)

# Базовые тренды
trend = np.linspace(20, 45, n)
seasonal = 2 * np.sin(2 * np.pi * np.arange(n) / 4)
cycle = 3 * np.sin(2 * np.pi * np.arange(n) / 16)

# Генерация показателей
macro_data = {
    'date': dates,
    'gdp': trend + seasonal + cycle + np.random.normal(0, 0.5,
n),
    'inflation': 4 + 0.5 * np.sin(2 * np.pi * np.arange(n) / 8)
+ np.random.normal(0, 0.3, n),
    'exchange_rate': 30 + 0.1 * np.arange(n) + 2 * np.sin(2 *
np.pi * np.arange(n) / 6) + np.random.normal(0, 1, n),
    'investments': trend * 0.3 + seasonal * 0.5 +
np.random.normal(0, 0.3, n),
    'oil_price': 60 + 10 * np.sin(2 * np.pi * np.arange(n) / 12)
+ np.random.normal(0, 5, n)
}

```

```
macro_df = pd.DataFrame(macro_data)
print(macro_df.head(10).round(2).to_string())
```

#### **Вариант 04: "Исследование эффективности рекламных кампаний"** **Данные за 3 года (36 месяцев):**

```
# Создание месячных данных
dates = pd.date_range('2019-01-01', '2021-12-31', freq='M')
n = len(dates)

sales_data = {
    'date': dates,
    'sales': np.zeros(n),
    'digital_ads': np.random.uniform(1, 5, n),
    'tv_ads': np.random.uniform(0.5, 3, n),
    'month': [d.month for d in dates],
    'year': [d.year for d in dates]
}

# Сезонность и тренд
trend = np.linspace(100, 200, n)
seasonality = 20 * np.sin(2 * np.pi * (np.arange(n) + 6) / 12)

# Формирование продаж
sales_data['sales'] = (trend + seasonality +
                      sales_data['digital_ads'] * 15 +
                      sales_data['tv_ads'] * 25 +
                      np.random.normal(0, 10, n))

sales_df = pd.DataFrame(sales_data)
print(sales_df.head(12).round(2).to_string())
```

#### **Вариант 05: "Анализ производственной функции предприятия"** **Данные по 50 предприятиям:**

```
np.random.seed(456)
n = 50

production_data = {
    'output': np.zeros(n),
    'labor': np.random.randint(20, 200, n),
    'capital': np.random.randint(50, 500, n),
    'materials': np.random.randint(30, 300, n),
    'technology_level': np.random.uniform(0.8, 1.5, n),
    'region': np.random.choice([1, 2, 3, 4], n)
}
```

```

# Производственная функция Кобба-Дугласа
A = 2.5 # технологический коэффициент
alpha = 0.6 # эластичность по труду
beta = 0.3 # эластичность по капиталу
gamma = 0.1 # эластичность по материалам

production_data['output'] = (A *
                             production_data['labor']**alpha *
                             production_data['capital']**beta *
                             production_data['materials']**gamma
                             *
                             production_data['technology_level']
                             *
                             np.random.lognormal(0, 0.1, n))

production_df = pd.DataFrame(production_data)
print(production_df.head(10).round(2).to_string())

```

## **Вариант 06: "Исследование факторов успеваемости студентов"** **Данные по 300 студентам:**

```

np.random.seed(789)
n = 300

students_data = {
    'gpa': np.zeros(n),
    'study_hours': np.random.randint(5, 35, n),
    'attendance': np.random.uniform(60, 100, n),
    'study_type': np.random.choice([0, 1], n, p=[0.7, 0.3]),
    'working': np.random.choice([0, 1], n, p=[0.6, 0.4]),
    'previous_gpa': np.random.uniform(3.0, 4.5, n),
    'motivation': np.random.uniform(1, 10, n)
}

# Формирование GPA
students_data['gpa'] = (2.0 +
                       students_data['study_hours'] * 0.05 +
                       students_data['attendance'] * 0.02 +
                       students_data['previous_gpa'] * 0.3 +
                       students_data['motivation'] * 0.1 -
                       students_data['working'] * 0.3 -
                       students_data['study_type'] * 0.2 +
                       np.random.normal(0, 0.2, n))

students_data['gpa'] = students_data['gpa'].clip(2.0, 4.5)

students_df = pd.DataFrame(students_data)
print(students_df.head(15).round(2).to_string())

```



## **Вариант 07: "Моделирование потребительских расходов домохозяйств"**

### **Данные по 200 домохозяйствам:**

```
np.random.seed(321)
n = 200

household_data = {
    'food_expenses': np.zeros(n),
    'income': np.random.normal(60, 20, n).clip(25, 150),
    'family_size': np.random.randint(1, 6, n),
    'region': np.random.choice([1, 2, 3, 4, 5], n),
    'settlement_type': np.random.choice([1, 2], n, p=[0.7,
0.3]),
    'age_head': np.random.randint(25, 70, n),
    'education_head': np.random.randint(1, 6, n)
}

# Формирование расходов на питание
household_data['food_expenses'] = (15 +
                                household_data['income'] *
0.15 +
                                household_data['family_size']
* 3 +
    (household_data['settlement_type'] - 1) * (-2) +
    household_data['education_head'] * (-1) +
                                np.random.normal(0, 3, n))

household_data['food_expenses'] =
household_data['food_expenses'].clip(5, 50)

household_df = pd.DataFrame(household_data)
print(household_df.head(12).round(2).to_string())
```

## **Вариант 08: "Анализ финансовых временных рядов"**

### **Ежедневные данные за 2 года (504 наблюдения):**

```
# Создание финансового временного ряда
dates = pd.date_range('2020-01-02', '2021-12-31', freq='B')
n = len(dates)
```

```

# Базовый тренд и волатильность
trend = np.linspace(1500, 1800, n)
volatility = np.zeros(n)
returns = np.zeros(n)

# GARCH процесс для волатильности
omega, alpha, beta = 0.1, 0.1, 0.85
volatility[0] = np.sqrt(omega / (1 - alpha - beta))
returns[0] = np.random.normal(0, volatility[0])

for t in range(1, n):
    volatility[t] = np.sqrt(omega + alpha * returns[t-1]**2 +
    beta * volatility[t-1]**2)
    returns[t] = np.random.normal(0, volatility[t])

financial_data = {
    'date': dates,
    'price': trend + np.cumsum(returns) * 10,
    'volume': np.random.lognormal(8, 1, n),
    'rts_index': 1200 + 200 * np.sin(2 * np.pi * np.arange(n) /
63) + np.random.normal(0, 50, n),
    'oil_price': 60 + 15 * np.sin(2 * np.pi * np.arange(n) /
126) + np.random.normal(0, 5, n),
    'usd_rub': 70 + 5 * np.sin(2 * np.pi * np.arange(n) / 84) +
np.random.normal(0, 2, n)
}

financial_df = pd.DataFrame(financial_data)
print(financial_df.head(10).round(2).to_string())

```

## **Вариант 09: "Исследование детерминант безработицы в регионах"**

### **Данные по 80 регионам:**

```

np.random.seed(654)
n = 80

regions_data = {
    'region': [f'Region_{i}' for i in range(1, n+1)],
    'unemployment': np.zeros(n),
    'gdp_per_capita': np.random.normal(450, 100, n).clip(250,
800),
    'investment_share': np.random.uniform(15, 35, n),
    'rural_share': np.random.uniform(10, 60, n),
    'avg_salary': np.random.normal(35, 8, n).clip(20, 60),
    'education_level': np.random.uniform(20, 80, n),
    'migration_balance': np.random.normal(0, 500, n)
}

# Формирование уровня безработицы
regions_data['unemployment'] = (8 -

```

```

regions_data['gdp_per_capita'] *
0.01 +
regions_data['rural_share'] *
0.05 -
regions_data['avg_salary'] * 0.1
-
regions_data['education_level'] *
0.03 +
regions_data['investment_share']
* (-0.1) +

np.abs(regions_data['migration_balance']) * 0.0001 +
np.random.normal(0, 1, n))

regions_data['unemployment'] =
regions_data['unemployment'].clip(2, 15)

regions_df = pd.DataFrame(regions_data)
print(regions_df.head(12).round(2).to_string())

```

### **Вариант 10: "Сравнительный анализ методов прогнозирования" Квартальные данные за 8 лет (32 наблюдения):**

```

# Создание макроэкономического временного ряда
dates = pd.date_range('2015-01-01', '2022-12-31', freq='Q')
n = len(dates)

# Тренды и циклы
trend = np.linspace(100, 130, n)
business_cycle = 5 * np.sin(2 * np.pi * np.arange(n) / 8)
seasonal = 2 * np.sin(2 * np.pi * np.arange(n) / 4)

macro_forecast_data = {
    'date': dates,
    'industrial_production': trend + business_cycle + seasonal +
np.random.normal(0, 1, n),
    'real_exchange_rate': 85 + 2 * np.sin(2 * np.pi *
np.arange(n) / 6) + 0.1 * np.arange(n) + np.random.normal(0,
0.5, n),
    'oil_price': 60 + 8 * np.sin(2 * np.pi * np.arange(n) / 12)
+ np.random.normal(0, 3, n),
    'business_activity': 50 + 10 * np.sin(2 * np.pi *
np.arange(n) / 8) + np.random.normal(0, 2, n),
    'inflation': 4 + 1 * np.sin(2 * np.pi * np.arange(n) / 16) +
np.random.normal(0, 0.3, n),
    'interest_rate': 7 + 2 * np.sin(2 * np.pi * np.arange(n) /
10) + np.random.normal(0, 0.2, n)
}

macro_forecast_df = pd.DataFrame(macro_forecast_data)
print(macro_forecast_df.head(8).round(2).to_string())

```