**Заданиe**

1) Ситать данные из training.csv. Ответы на тестовой выборке testing.csv не следует использовать ни в каких экспериментах, кроме финального. Проверить является ли ряд стационарным в широком смысле. Это можно сделать двумя способами:

а) Провести визуальную оценку, отрисовав ряд и скользящую статистику(среднее, стандартное отклонение). Постройте график на котором будет отображен сам ряд и различные скользящие статистики.

б) Провести тест Дики - Фуллера.

Сделать выводы из полученных результатов. Оценить достоверность статистики.

2)Разложить временной ряд на тренд, сезональность, остаток в соответствии с аддитивной, мультипликативной моделями. Визуализировать их, оценить стационарность получившихся рядов, сделать выводы. (15 баллов)

3)Проверить является ли временной ряд интегрированным порядка k. Если является, применить к нему модель ARIMA, подобрав необходимые параметры с помощью функции автокорреляции и функции частичной автокорреляции. Выбор параметров обосновать. Отобрать несколько моделей. Предсказать значения для тестовой выборки. Визуализировать их, посчитать r2 score для каждой из моделей. Произвести отбор наилучшей модели с помощью информационного критерия Акаике. Провести анализ получившихся результатов. (50 баллов)

За все правильно выполненные пункты можно получить 90 баллов.

**Подход к решению**

**Временной ряд** - совокупность значений какого-либо показателя за определенное время.  
  
Параметры временного ряда:  
1. Период времени  
2. Уровни ряда - значения показателя  
  
**Временной ряд** y называется **стационарным**, если E[y] = Const, D[y] = Const, Corr[yk, y(k-1)] = Const \* k - т.е если эти параметры не зависят от времени. Другое определение стационарности - это отсутствие тренда.  
  
**Трендом** временного ряда называется изменение, определяющее общее направление развития, т.е. грубо говоря общее направление графика ряда (возрастает/убывает/не изменяется)  
  
**Скользящая средняя** (Moving Average - MA) - среднее арифметическое значений исходной функции за установленный период.  
  
**Стандартное отклонение** - показывает, на сколько в среднем отклонился ряд от средней вариации ряда (от среднего арифметического, в нашем случае).  
  
Скользящая средняя вместе со стандартным отклонением составляют \*скользящие статистики\*.  
  
**Тест Дики-Фуллера** используется для проверки ряда на стационарность. Он проверяет ряд на наличие так называемых единичных корней.  
  
Временной ряд имеет **единичный корень** (хотя бы один), если его первые разности образуют стационарный ряд.  
  
y(t) ~ I(1), т.е. Δy(t)=y(t)-y(t-1) ~ I(0), где Δ - разностный оператор, I(j) - означает, что ряд является интегрированным порядка j, I(0) - ряд стационарен. Определение интегрированности порядка k дается далее.

**Реализация**

1) Cчитываем данные из файла “training.csv”

*pd.read\_csv('training.csv', parse\_dates = ['Date'], index\_col = 'Date')*

Вычисляем среднее значение ряда

*ts.rolling(window=15).mean()*

Вычисляем стандартное отклонение

*ts.rolling(window=15).std()*

Строим в одних координатных осях график исходного временного ряда, среднего значения и стандартного отклонения. Визуально оцениваем стационарность ряда.

Ряд является стационарным, если E[y] = Const, D[y] = Const, Corr[yk, y(k-1)] = Const \* k - т.е если эти параметры не зависят от времени.

Тест Дики – Фуллера. Вызывается с помощью функции DF() , которая возвращает булевскре значение

test = st.adfuller(data.Value)

test[0] > test[4]['5%']

С помощью теста Дики – Фуллера проверяем наличие единичных корней. Для того, чтобы знать есть ли у нас единичные корни сравниваем достоверность статистики p-value с критическими значениями значимости 5% - critical\_values[‘5%’]. То есть, если:

* test[0]> test[4][‘5%’], то есть единичные корни, а следовательно, ряд не стационарен
* test[0] <= test[4][‘5%’], то единичные корни отсутствуют и ряд стационарен

2) Раскладываем временной ряд на на тренд, сезональность и остаток в соответсвии с аддитивной и мультипликативной моделями

seas.seasonal\_decompose(ts, model = dtype)

Строим общий график. Строим графики сезонной составляющей, тренда и шума по отдельности и проводим для них тест Дики-Фуллера DF(data). Для этого вводим функцию.

Chart(title, data)

3) Находим порядок интегрированности ряда, т.е. дифференциируем пока результат будет являться стационраным рядом.

tmp.diff().dropna()

Проверку на стационарность делаем с помощью теста Дики-Фуллера

DF(tmp)

Ряд является интегрированным 1 порядка, нашли параметр d для модели ARIMA.

Находим функции автокорелляции и частичной автокорелляции для исходного ряда и отрисовываем их

acf(ts, nlags=20)  
pacf(ts, nlags=20, method='ols')

По диаграммам можно найти параметры для ARIMA модели p и q. Q – номера коэффициентов сильно отличных от нуля (ACF график). P – наибольший номер коэффициента сильно отличного от нуля (PACF график).

Строим ARIMA модель с найденными параметрами p, d ,q

ARIMA(ts, order = (p, d ,q))

Тренируем модель

model.fit()

Делаем прогноз согласно модели

model.predict(start='1989-01-01', end='1993-12-01', typ='levels', dynamic=True)

Отрисовываем график и сравниваем значения прогноза и реальных данных

test\_ts = pd.read\_csv('testing.csv', parse\_dates = ['Date'], index\_col = 'Date')

Считаем R2 score и критерий Акаике

r2\_score(test\_ts, forecast) model.aic

Лучшая модель имела параметры

p = 1, q = 1, d = 1

И имела следующие результаты тестов

R2 score: -3

AIC: 248

**Используемые библиотеки**

* pandas – библиотека Python для анализа данных
  + - pandas.read\_csv(filepath\_or\_buffer,  index\_col, parse\_dates) – необходим для считывания данных из csv – файла и преобразования этих в DatаFrame
    - pandas.DateFrame.plot(ax) – строит график DataFrame с помощью matplotlib или pylab
    - pandas.DateFrame.rolling(window).mean – считает скользящее среднее
    - pandas.DateFrame.rolling(window).std – считает стандартное отклонение
* statsmodels – модуль, предоставляющий широкий набор средств и методов для проведения статистического анализа и эконометрики
  + - statsmodels.tsa.stattools.adfuller(x) – тест Дики – Фуллера, проверяющий наличие единичных корней у ряда
    - statsmodels.tsa.seasonal.seasonal\_decompose(x, model='additive') — разложение ряда на составляющие
    - statsmodels.tsa.stattools.acf(x, nlags=40) — функция автокорелляции
    - statsmodels.tsa.stattools.pacf(x, nlags=40, method='ywunbiased') — функция частичной автокорелляции
    - statsmodels.tsa.arima\_model.**ARIMA**(endog, order) — создание ARIMA модели
* scikit-learn — библиотека машинного обучения
  + - sklearn.metrics.r2\_score(y\_true, y\_pred) — считает r2 score
* Numpy - математика
  + - numpy.diff(a, n=1, axis=-1) - дифференцирует ряд
* matplotlib – для построения диаграммы
  + - matplotlib.pyplot.bar(x, height) – рисует диаграмму
    - matplotlib.pyplot.xlabel(s) – записывает строку s на оси х
    - matplotlib.pyplot.ylabel(s) - записывает строку s на оси y
    - matplotlib.pyplot.title(s) – записывает строку s в оглавлении
    - matplotlib.pyplot.show() – выводит на дисплей все рисунки

**Инструкции по запуску**

**Необходимое ПО**

* Jupyter
* Python 3.6
* Установленные библиотеки

**Запуск(из окна терминала)**

Из директории проекта:

jupyter-notebook task1.ipynb

В jupyter:

Cell > Run All

**Участники**

Абдыкалик Чингиз — 1 часть задания + readme

Донской Иван — 2 часть задания + readme

Кононов Сергей — 3 часть + readme