Прогноз качества воздуха
Лучшая производительность набора данных Abalone с NN

Ильда Алушай
Данил Кириленко

Методы искусственного интеллекта в анализе данных

16 декабря 2020

- Dataset 1 : https://archive.ics.uci.edu/ml/datasets/Air+Quality
- Dataset 2 : https://archive.ics.uci.edu/ml/datasets/Abalone
- Данные получены из репозитория машинного обучения UCI. Он был зарегистрирован 5 химическими датчиками оксидов металлов, расположенными в сильно загрязненном районе итальянского города, и мы решили проанализировать один из них, CO (поскольку это задача многомерного временного ряда). Набор данных содержит 9538 объектов с марта 2004 г. по февраль 2005 г.

Предварительная обработка данных

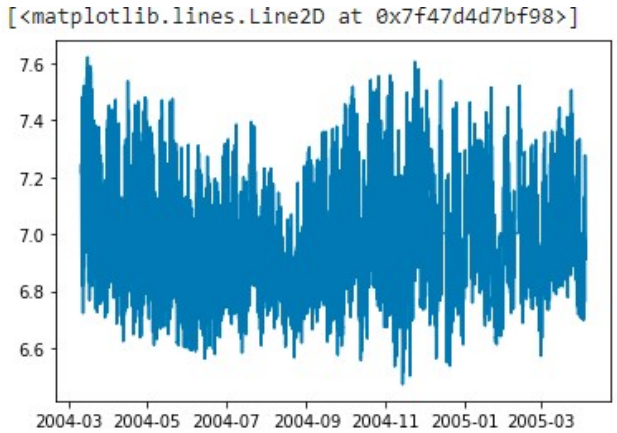
- В этих наблюдениях отсутствуют значения, помеченные как «-200», которые мы преобразовали как NaN для последующей предварительной обработки.
- Мы создали новые переменные из целевых переменных и заменили значение NaN их средним значением.
- Чтобы получить индекс datetime, мы объединили столбцы «Date» и «Time», преобразовав тип данных из строки в datetime и сохранили новый «Datetime» в списке

Анализ данных

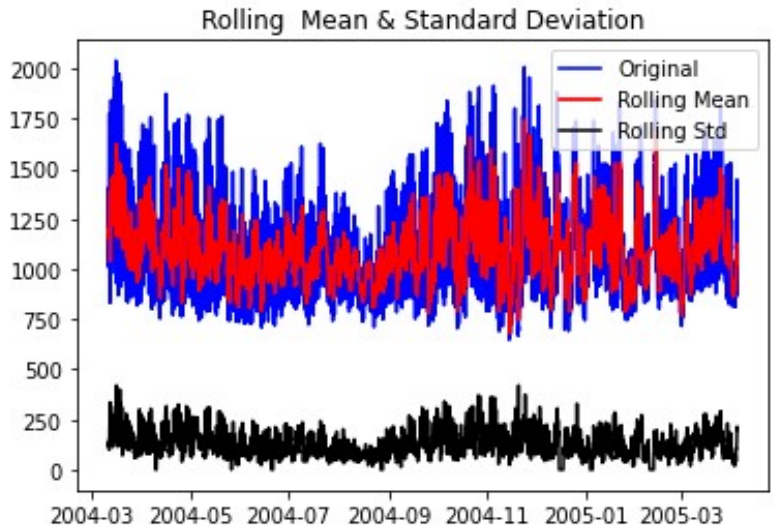
- Графики графиков временных рядов уровней S1 (CO).
- Мы посмотрели на графики за более короткий период: «2004-10-05» - «2004-10-08», S1-S5 и заметили, что шаблон постоянно повторяется в течение определенного периода времени.

Проверка стационарности временных рядов

- Временной ряд является стационарным, если его статистические свойства, такие как среднее значение, дисперсия, остаются постоянными во времени и автоковариация, которая не зависит от времени.
- Мы использовали график скользящей статистики вместе с тестом Дики-Фуллера
- Сборка test stationary также для S2, S3, S4, S5.
- Мы попытались сделать серию стационарной, используя логарифмическое преобразование.



```
test_stationarity(S1)
```



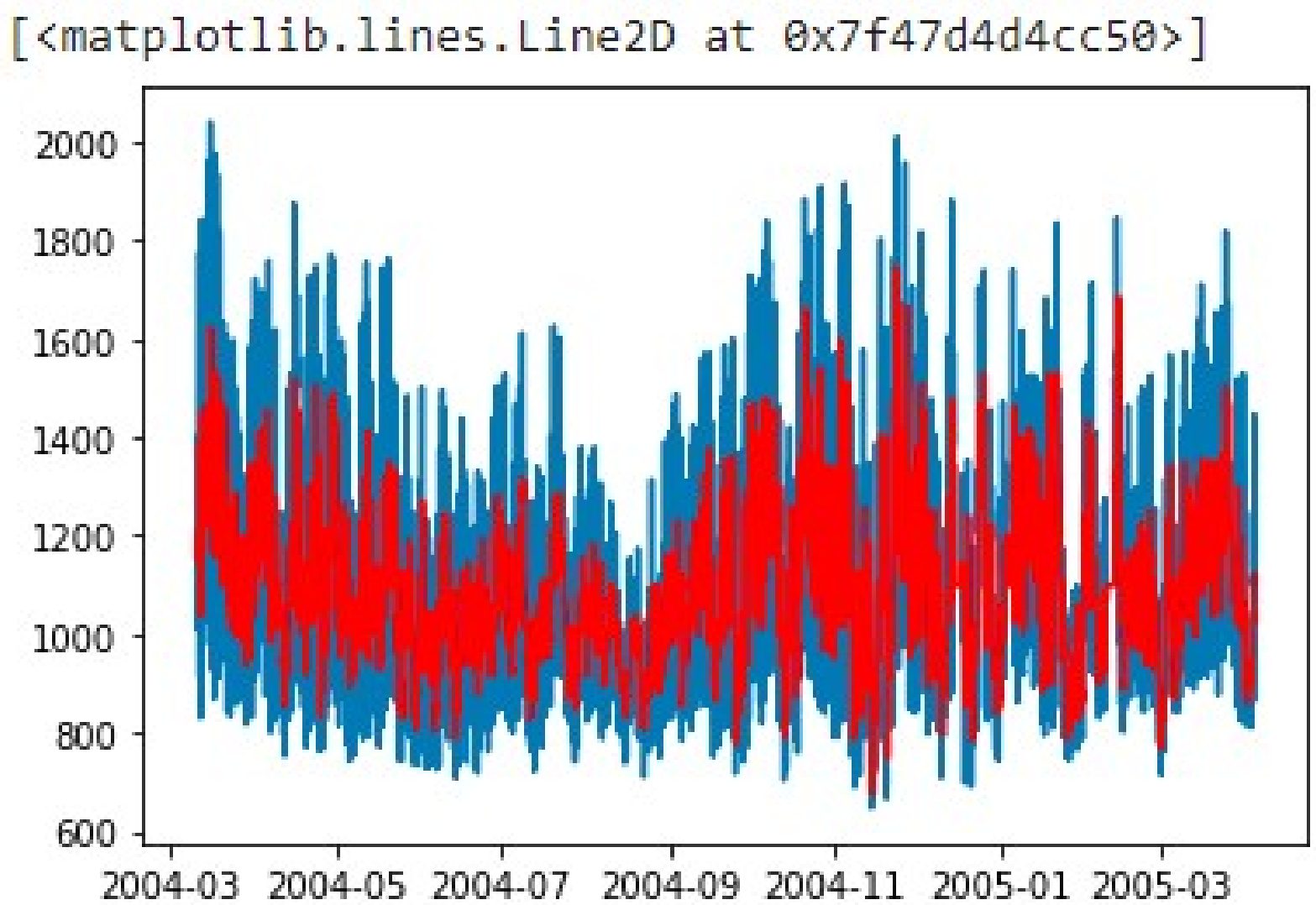

```
Results of Dickey-Fuller Test:
Test Statistic                  -9.732748e+00
p-value                          8.914162e-17
#Lags Used                       3.800000e+01
Number of Observations Used      9.318000e+03
Critical Value (1%)             -3.431052e+00
Critical Value (5%)             -2.861850e+00
Critical Value (10%)            -2.566935e+00
dtype: float64
```

С постоянным скользящим средним и скользящей дисперсией, а также с p-значением теста Дики-Фуллера, близким к 0, мы можем сказать, что S1 является слабым и стационарным.

Скользящая средняя

- В этом подходе мы берем среднее значение «k» последовательных значений в зависимости от частоты временных рядов. Здесь мы можем взять среднее значение за последний год, т.е. за последние 12 значений. В Pandas есть определенные функции для определения скользящей статистики.
- Мы берем «взвешенное скользящее среднее», где более поздним значениям присваивается больший вес. Эта TS имеет еще меньшие вариации в среднем и стандартном отклонении по величине. Кроме того, статистика теста меньше критического значения 1%, что лучше, чем в предыдущем случае..

Скользящая средняя S1 за последний год.

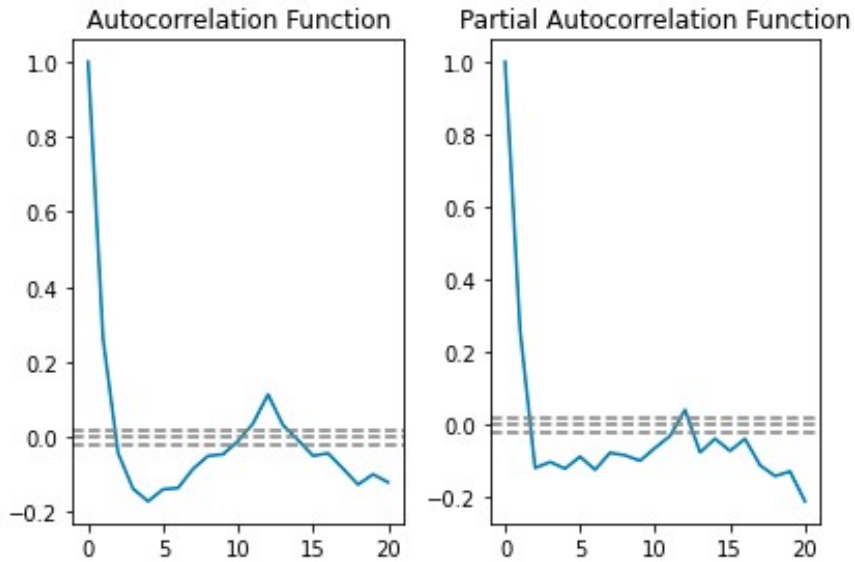


Устранение тренда и сезонности

- Два метода:
  1. Разница (снятие разницы с определенным временным лагом)
  2. 2. Декомпозиция (моделирование тренда и сезонности и удаление их из модели)
- Мы использовали оба этих метода для нашего ряда S1, и наш временной ряд, наконец, очень близок к стационарному.

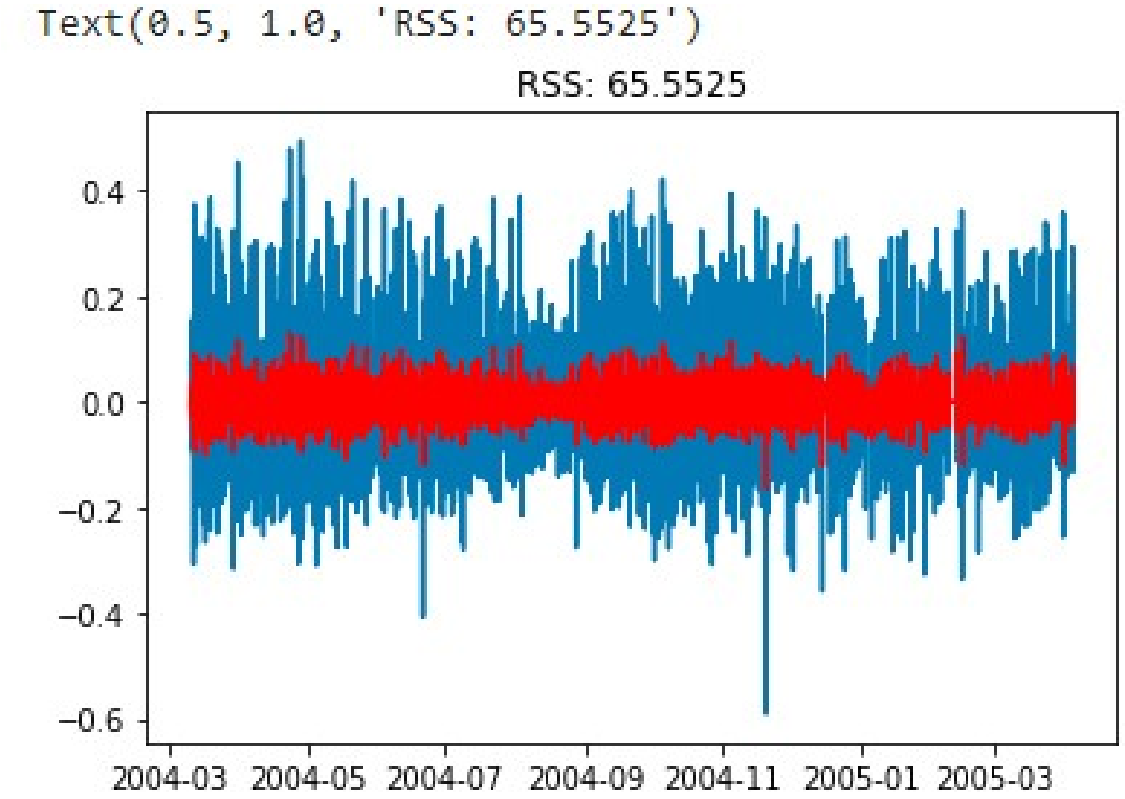
Прогнозирование временного ряда

- Использование модели ARIMA / SARIMAX для прогнозирования будущих значений временного ряда S1
- Мы создали графики ACF и PACF. На этом графике две пунктирные линии по обе стороны от 0 представляют собой доверительные интервалы. Их можно использовать для определения значений «p» и «q».



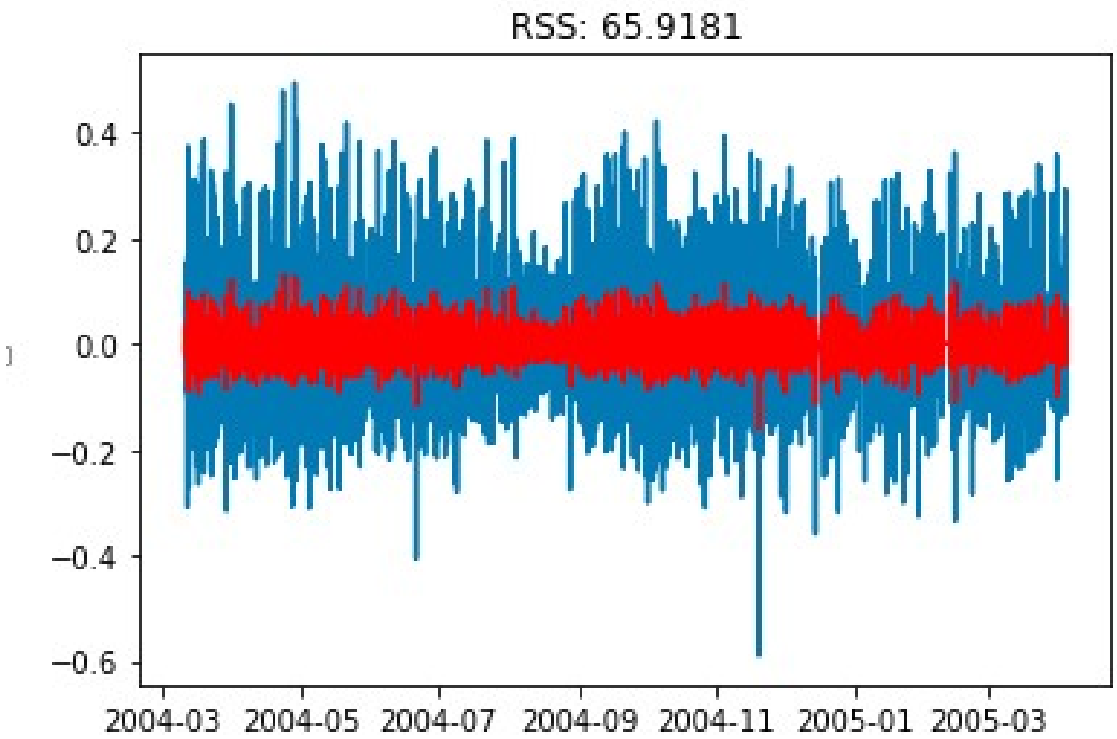
Модель AR и ее RSS (Residual Sum of Squares)

После этого мы создали модель ARIMA, сочетающую модели AR и MA. Здесь мы видим, что модели AR и MA имеют почти одинаковый RSS, но их сочетание значительно лучше



Модель MA и ее RSS (Residual Sum of squares)

```
Text(0.5, 1.0, 'RSS: 65.9181')
```



Комбинированная модель ARIMA:

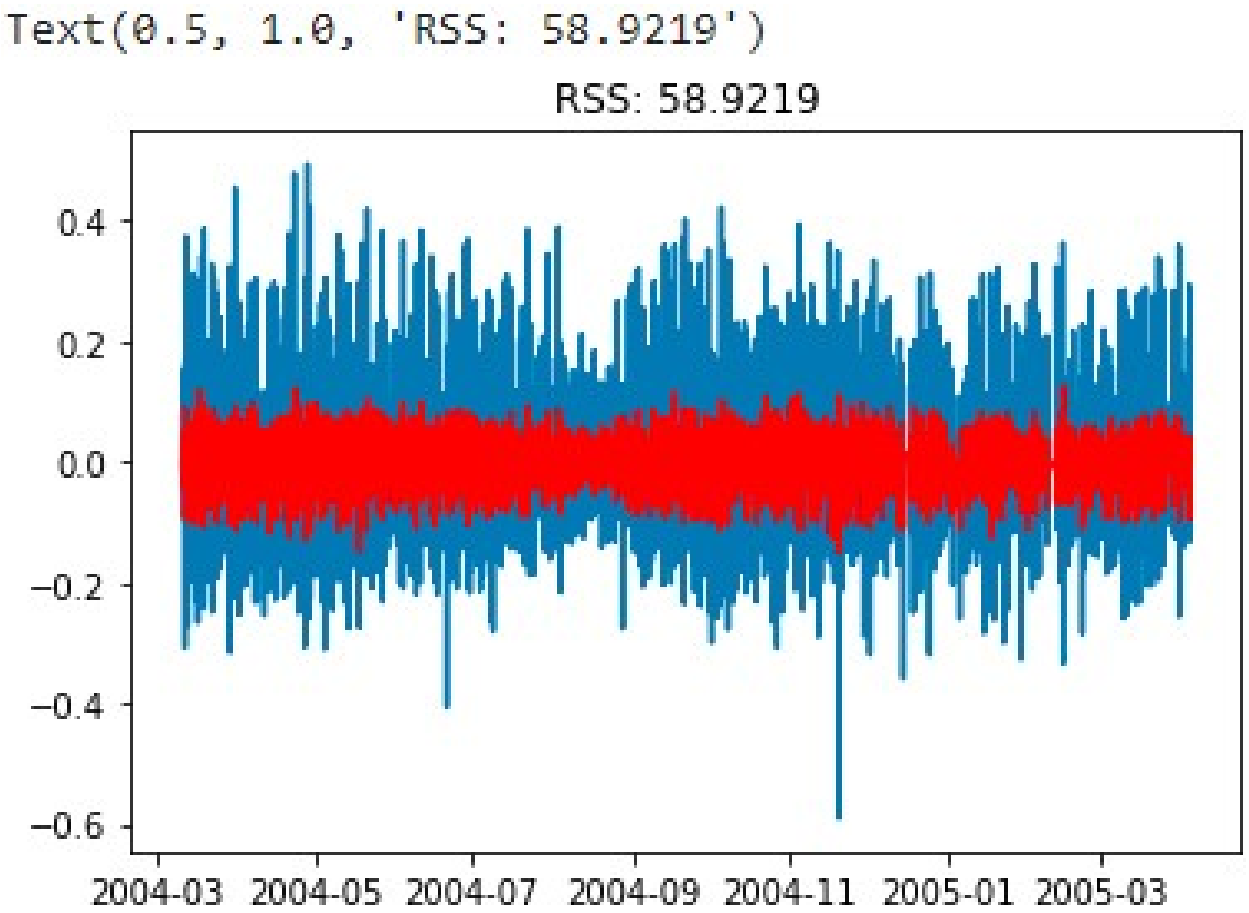


Рис.: Здесь мы видим, что модели AR и MA имеют почти одинаковый RSS, но их сочетание значительно лучше.

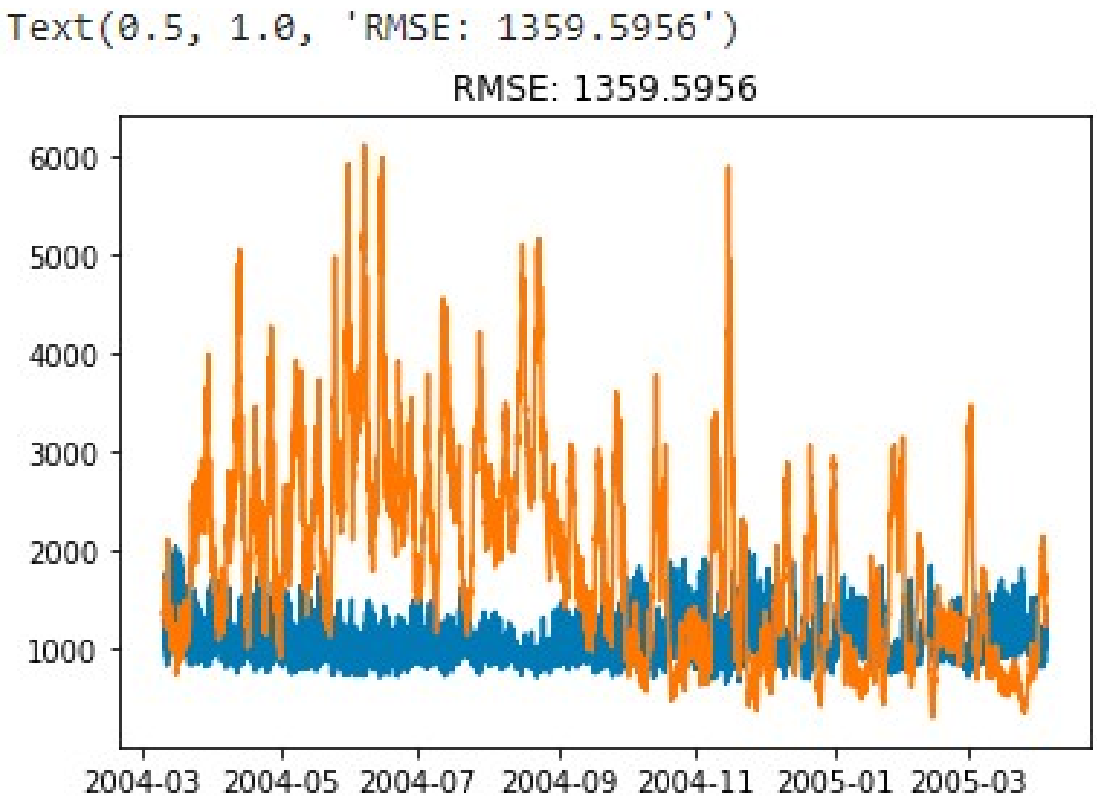


Рис.: С помощью прогнозов мы видим, что эта модель не так хороша, как должна быть, и среднеквадратичная ошибка также очень высока..

Другой метод -SARIMAX

	coef	std err	z	P>|z|	[0.025	0.975]
ar.L1	0.8702	0.006	152.905	0.000	0.859	0.881
ma.L1	0.2064	0.010	21.554	0.000	0.188	0.225
ar.S.L12	-0.1714	0.010	-17.111	0.000	-0.191	-0.152
ma.S.L12	-0.9370	0.003	-275.473	0.000	-0.944	-0.930
sigma2	0.0051	5.75e-05	89.050	0.000	0.005	0.005

Рис.: Столбец coef показывает вес (то есть важность) каждой функции и то, как каждая из них влияет на временной ряд. P> | z | Столбец информирует нас о важности каждого веса функции. Здесь каждый вес имеет p-значение ниже 0,05, поэтому разумно сохранить их все в нашей модели.

Выбор параметров для модели временных рядов SARIMAX

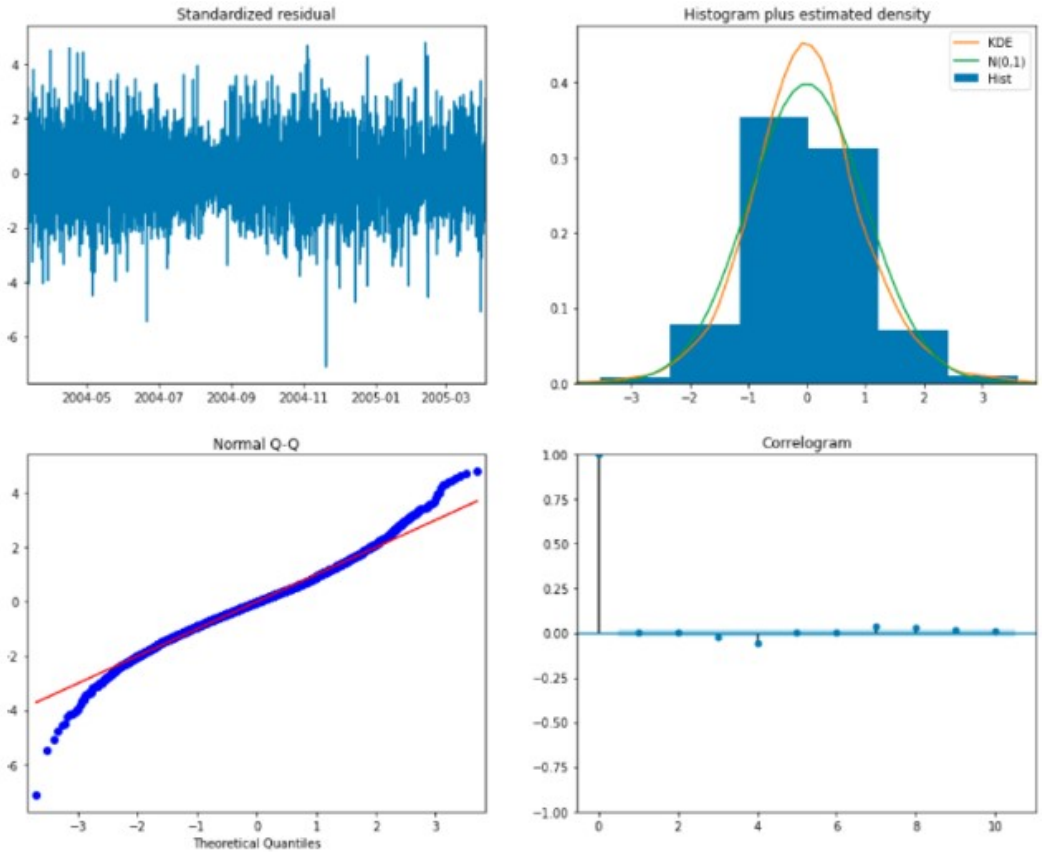
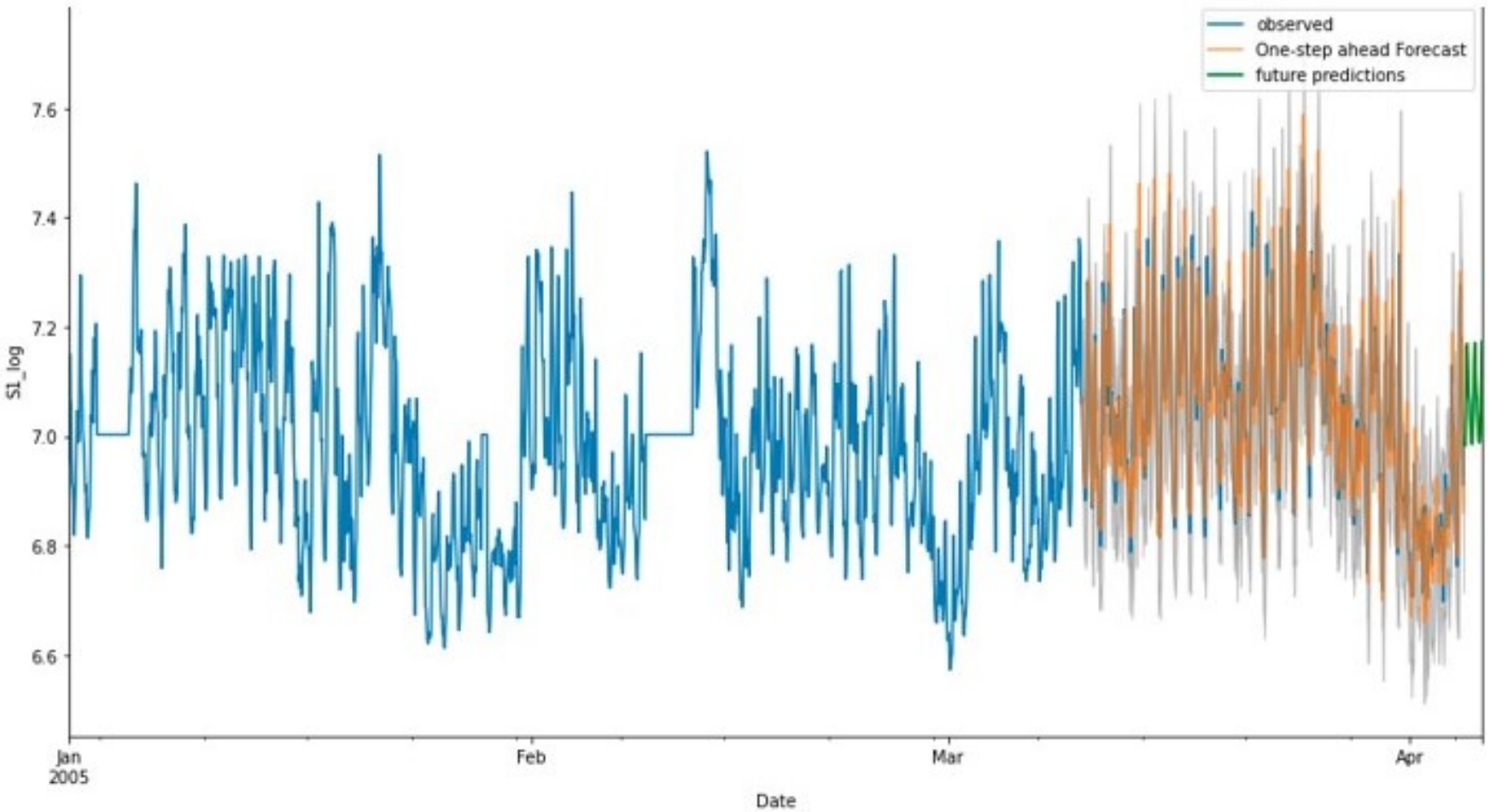


Рис.: Генерация диагностики модели и поиск необычного поведения

Комментарий к рисунку

- 1.На правом верхнем графике мы видим, что красная линия KDE близко следует за линией N (0,1) (где N (0,1) - стандартное обозначение для нормального распределения со средним 0 и стандартным отклонением 1). Это хороший показатель того, что остатки распределены нормально.
- 2. Q-Q график в нижнем левом углу показывает, что упорядоченное распределение остатков (синие точки) следует линейному тренду выборок, взятых из стандартного нормального распределения с N (0, 1). Опять же, это явный признак того, что остатки распределены нормально.
- 3. Невязки во времени (верхний левый график) не показывают явной сезонности и выглядят как белый шум. Это подтверждается графиком автокорреляции (то есть коррелограммой) в правом нижнем углу, который показывает, что остатки временных рядов имеют низкую корреляцию с запаздывающими версиями самих себя.

Проверка прогнозов



Полученные результаты

```
Mean Squared Error of forecast : 0.005
Mean Absolute Percentage Error: 0.76%
The Root Mean Squared Error of our prediction is 0.07
```

Имея низкий уровень ошибки, мы заключаем, что этот прогноз точен.

Предсказание возраста морских ушек

- Из набора данных из предыдущей задачи были выделены два самых часто встречающихся класса (9 и 10), которые между собой сбалансированы (689 и 634 образца). В качестве baseline была использована простая логистическая регрессия (sklearn.linearmodel.LogisticRegression), эта модель показала следующие результаты:

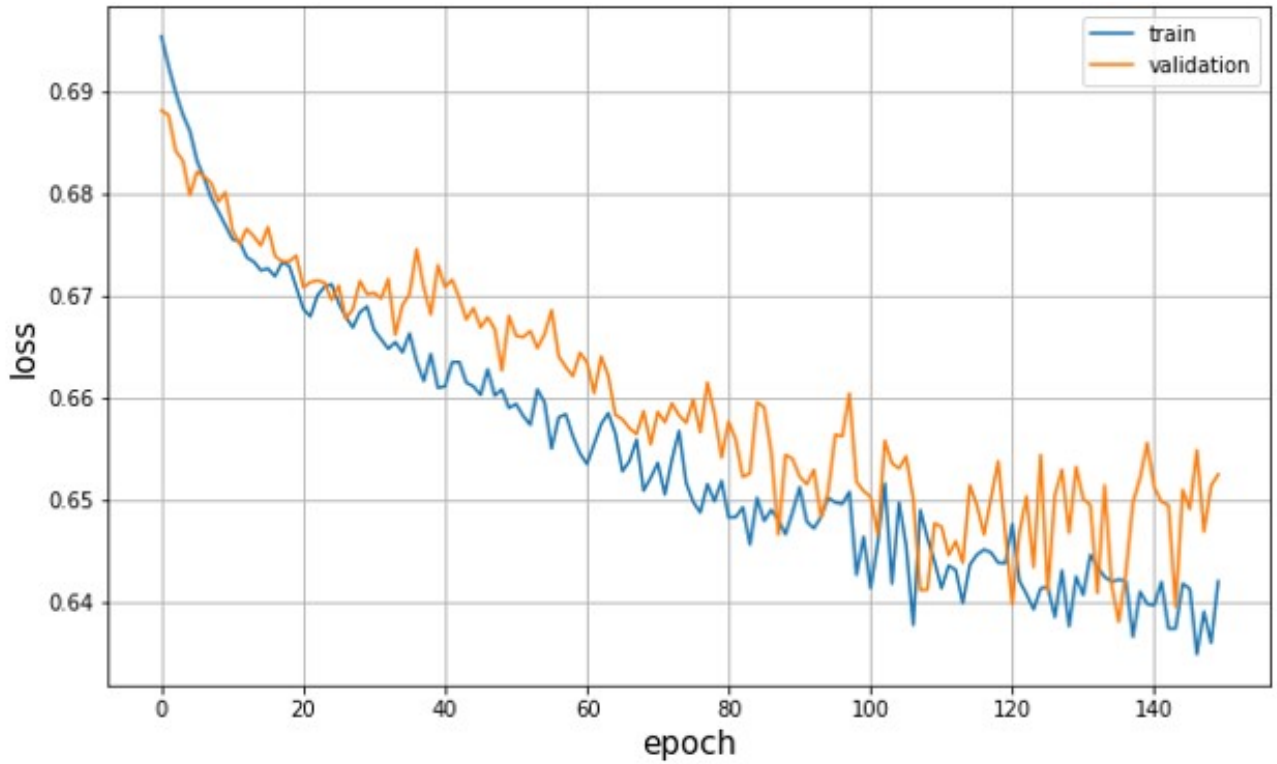
	precision	recall	f1-score	support
9.0	0.57	0.76	0.65	153
10.0	0.71	0.49	0.58	178
accuracy			0.62	331
macro avg	0.64	0.63	0.62	331
weighted avg	0.64	0.62	0.61	331

```
Roc-auc score: 0.6295439524124256
Wall time: 9 ms
```

- Для улучшения полученных результатов искалась подходящая архитектура полносвязной нейронной сети, среди различных вариантов лучше всего себя показала следующая модель:

```
----------------------------------------------------------------
        Layer (type)               Output Shape         Param #
================================================================
            Linear-1                [-1, 10]             110
            Linear-2                [-1, 80]             880
            Linear-3                [-1, 50]           4,050
            Linear-4                 [-1, 1]              51
================================================================
Total params: 5,091
Trainable params: 5,091
Non-trainable params: 0
----------------------------------------------------------------
Input size (MB): 0.00
Forward/backward pass size (MB): 0.00
Params size (MB): 0.02
Estimated Total Size (MB): 0.02
----------------------------------------------------------------
```

- Кроме этого использовался Dropout с вероятностью 0,1. Во время обучения при получении лучшего результата на валидации веса модели сохранялись, в итоге, при оценке использовалась та модель, которая лучше все себя показала на валидационном множестве за все время обучения, ниже графики обучения и итоговые результаты обученной модели.



	precision	recall	f1-score	support
0.0	0.57	0.78	0.66	153
1.0	0.72	0.49	0.58	178
accuracy			0.63	331
macro avg	0.65	0.64	0.62	331
weighted avg	0.65	0.63	0.62	331

Roc-auc score: 0.6867885731071455

Спасибо за внимание!

alushaj.i@phystech.edu
kirilenko.de@phystech.edu