ПРАВИТЕЛЬСТВО РОССИЙСКОЙ ФЕДЕРАЦИИ
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ
НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ
«ВЫСШАЯ ШКОЛА ЭКОНОМИКИ»

Факультет компьютерных наук
Образовательная программа бакалавриата «Программная инженерия»

УДК 004.8

СОГЛАСОВАНО
профессор департамента анализа данных и искусственного интеллекта факультета компьютерных наук НИУ ВШЭ, д-р физ.-мат. наук

_____ В. А. Громов
«___» _____ 2022 г.

УТВЕРЖДАЮ
Академический руководитель образовательной программы «Программная инженерия»
профессор департамента программной инженерии, канд. техн. наук

_____ В. В. Шилов
«___» _____ 2022 г.

Выпускная квалификационная работа
(академическая)
на тему: **Прогнозирование хаотических временных рядов: алгоритм self-healing для прогнозирования с помощью кластеризации**
по направлению подготовки 09.03.04 «Программная инженерия»

Приложения

Выполнила студентка
образовательной программы
09.03.04 «Программная инженерия»
группы БПИ182

_____ А.Т. Антонова

«___» _____ 2022 г.

Москва – 2022

Оглавление

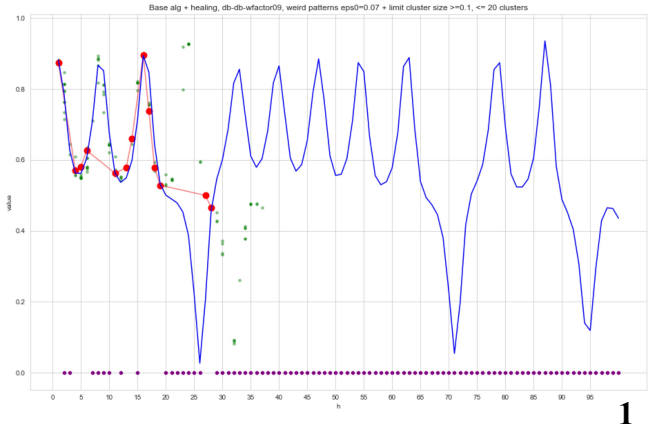
Приложение А. Условные обозначения алгоритмов

Таблица 1. Условные обозначения алгоритмов, в последнем столбце указан номер страницы подробного описания в основном отчете

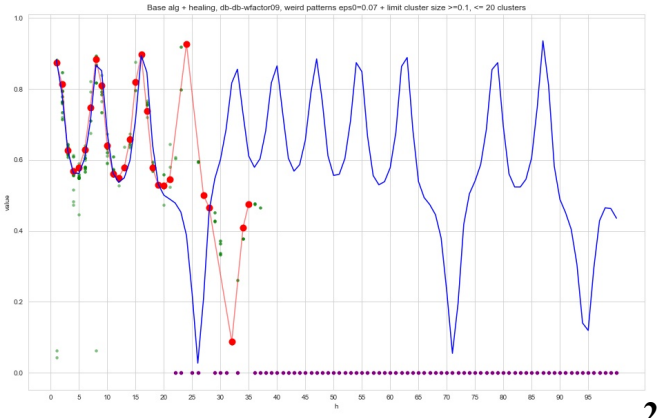
Обозначение	**Описание**	**Стр.**
Тип алгоритма прогноизрования		
s	Поточенное прогнозирование	16
tp	Траекторное прогнозирование	16
sh	Алгоритм self-healing	20
Алгоритм кластеризации обобщенных z-векторов		
cl_db	Мотивы — это центры кластеров, алгоритм кластеризации — DBSCAN с параметрами $\epsilon = 0.01$, min_samples $= 5$, если не указано иное.	15
cl_wi	Мотивы — это центры кластеров, алгоритм кластеризации — Wishart с параметрами significance $= 0.01$, min_samples $= 5$, если не указано иное.	15
Алгоритмы определения непрогнозируемых точек		
fp	Вынужденное прогнозирование (forced prediction)	17
ls	Большой разброс (large spread)	17
rg	Быстрый рост разброса (rapid growth)	18
rd	Быстрый рост количества кластеров DBSCAN (rapid growth DBSCAN)	18
rw	Быстрый рост количества кластеров Wishart (rapid growth Wishart)	18
lcs	Ограничение на размер максимального кластера и на количество кластеров. Например, lcs_0.1_2 — доля точек в максимальном кластере должна быть хотя бы 0.1, всего кластеров не более 2 (без учета выбросов)	18
bl	Большой скачок (big leap)	23
blbi	Большой скачок между итерациями (big leap between iterations)	23
wp	Странные паттерны (weird patterns)	24
Алгоритмы вычисления единого прогнозного значения		
db	Центр максимального кластера DBSCAN с параметрами $\epsilon = 0.01$, min_samples $= 5$, если не указано иное.	19
wi	Центр максимального кластера Wishart с параметрами significance $= 0.01$, min_samples $= 5$, если не указано иное.	19
wa	Взвешенное среднее	22
dc	Двойная кластеризация	22
factor	Центр кластера DBSCAN, максимального с учетом весов по итерации	22-23
dist	Центр кластера DBSCAN, максимального с учетом весов по расстоянию до мотива	22-23
pl	Центр кластера DBSCAN, максимального с учетом весов по длине паттерна	22-23

Приложение Б. Дополнительные графики

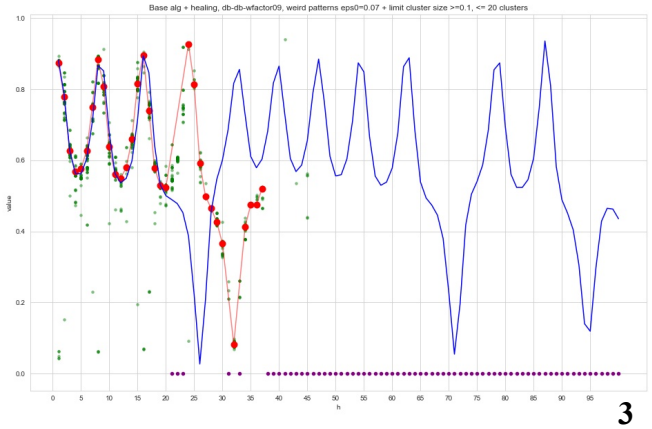
Рисунки 1-13. Алгоритм self-healing. Кластеризация мотивов — db, 20%. Алгоритм вычисления единого прогнозного значения — db, db + factor = 0.9 для self-healing. Алгоритм определения непрогнозируемых точек — lcs_0.1_20, lcs_0.1_20 + wp для self-healing. Синим обозначен настоящий временной ряд, зеленые точки — возможные прогнозные значения, красные точки и линии — единые прогнозные значения, фиолетовые точки на прямой $y = 0$ — непрогнозируемые точки. Порядок итераций: слева направо, сверху вниз.



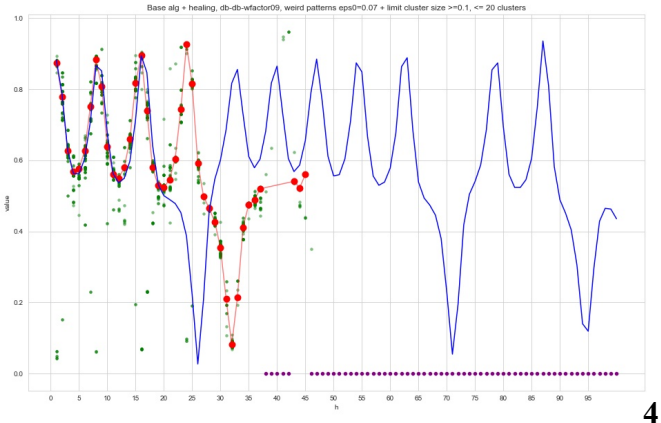
1



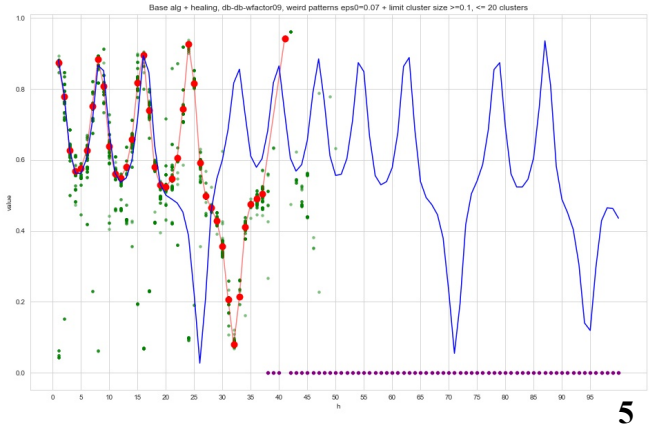
2



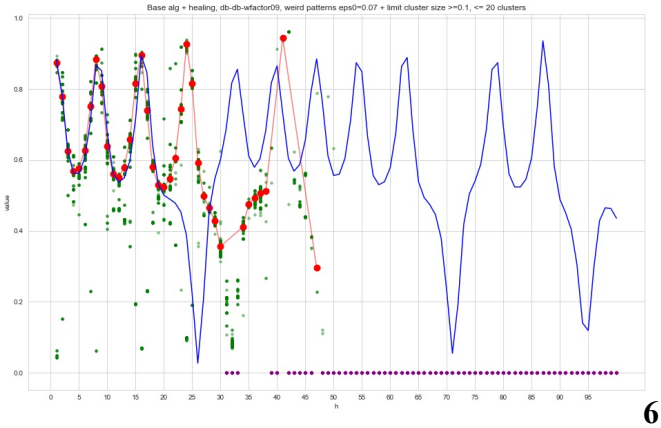
3



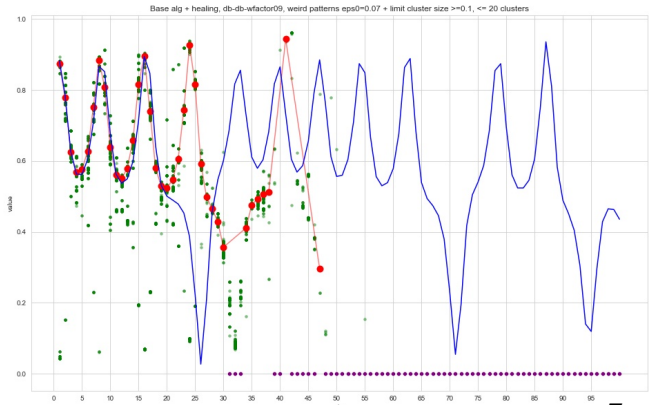
4



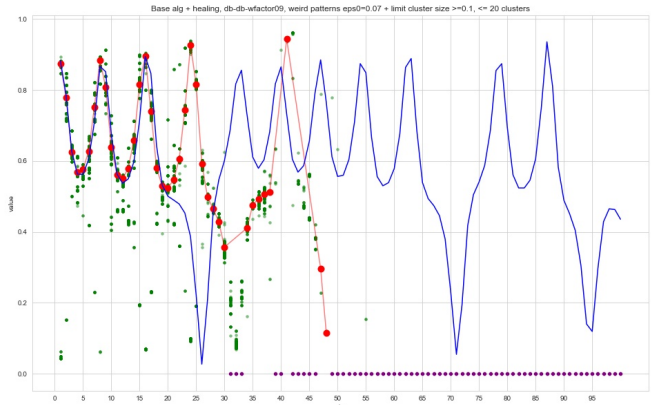
5



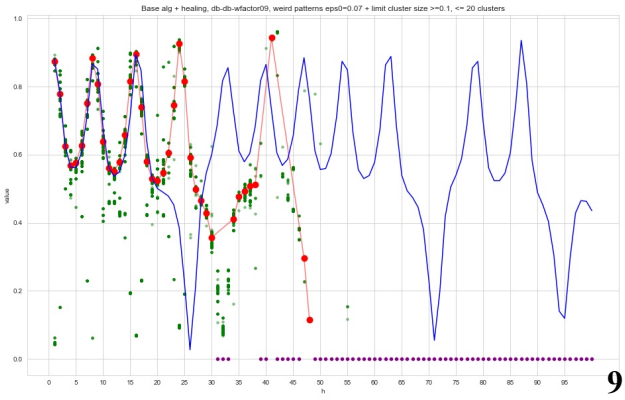
6



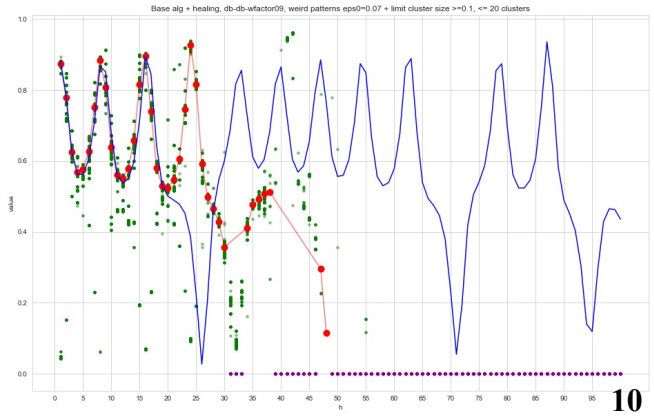
7



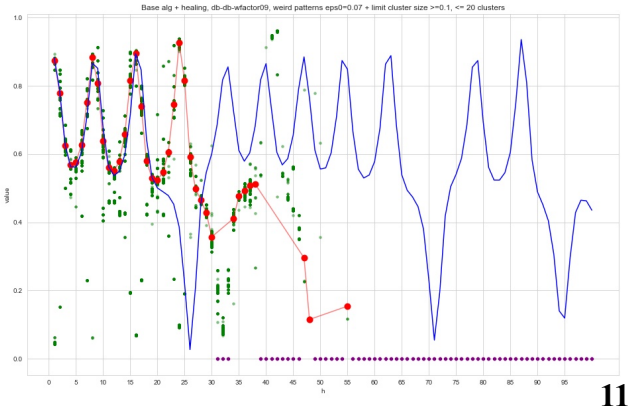
8



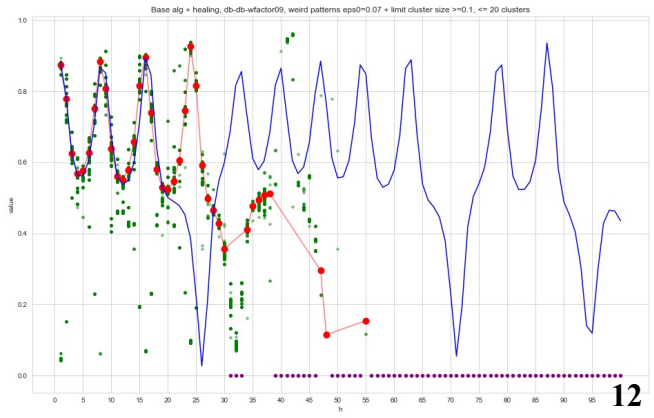
9



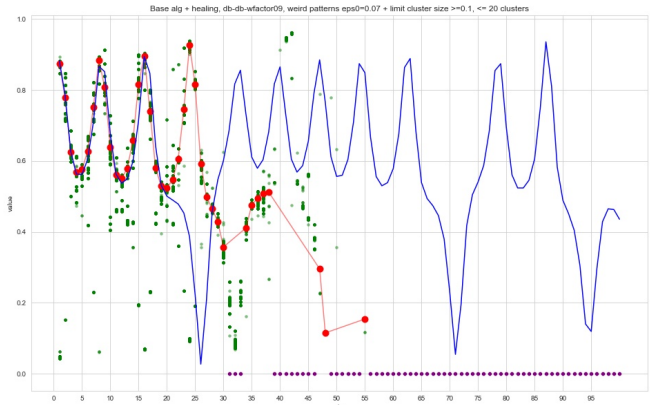
10



11



12



13

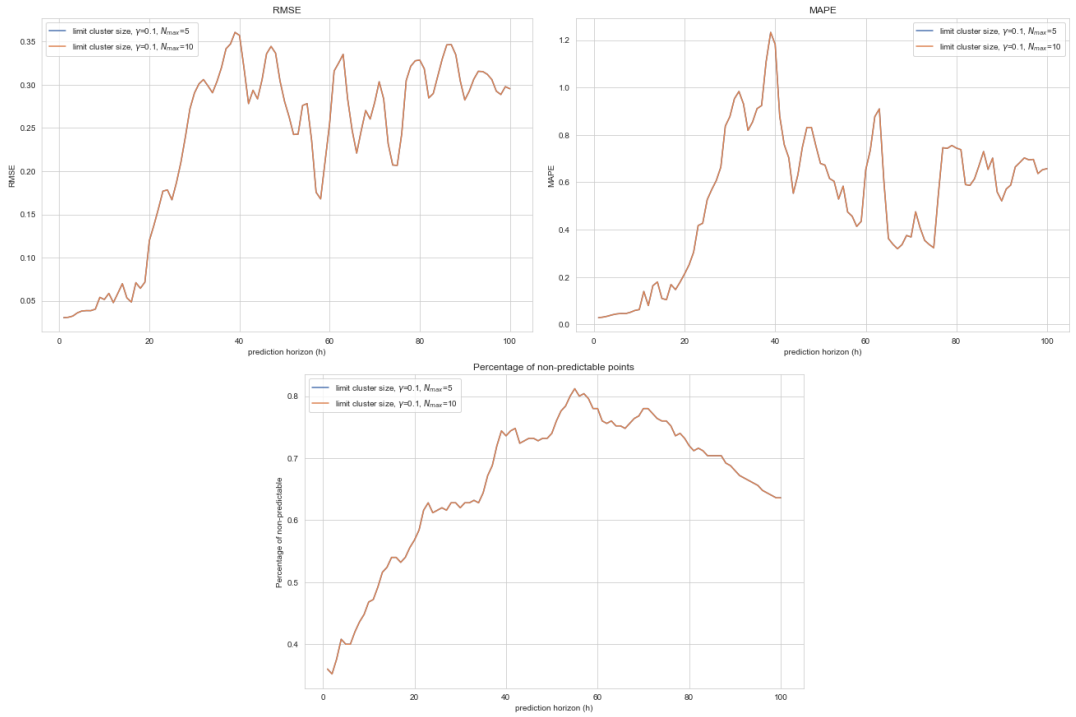


Рисунок 14. Графики зависимости RMSE, MAPE и количества непрогнозируемых точек от горизонта прогнозирования. Горизонт прогнозирования h=100. Тестовая выборка 250. Кластеризация мотивов — db, 20%. Алгоритм вычисления единого прогнозного значения — db. Алгоритмы определения непрогнозируемых точек: lcs_0.1_5 (синий), lcs_0.1_10 (оранжевый).

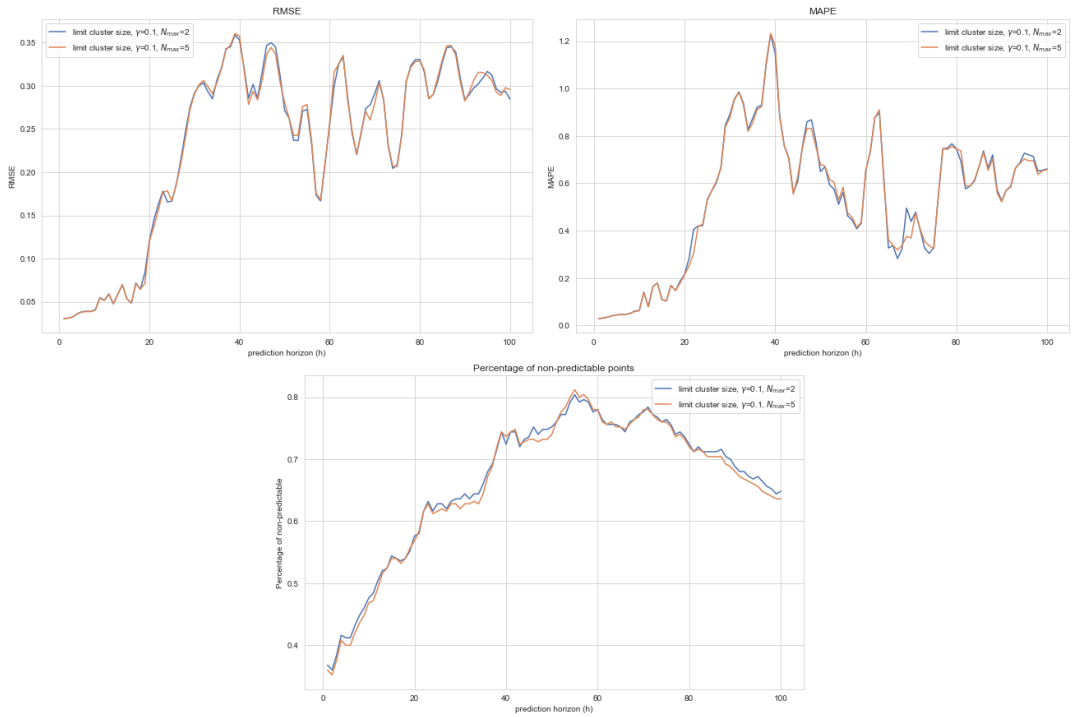


Рисунок 15. Графики зависимости RMSE, MAPE и количества непрогнозируемых точек от горизонта прогнозирования. Горизонт прогнозирования h=100. Тестовая выборка 250. Кластеризация мотивов — db, 20%. Алгоритм вычисления единого прогнозного значения — db. Алгоритмы определения непрогнозируемых точек: lcs_0.1_2 (синий), lcs_0.1_5 (оранжевый).

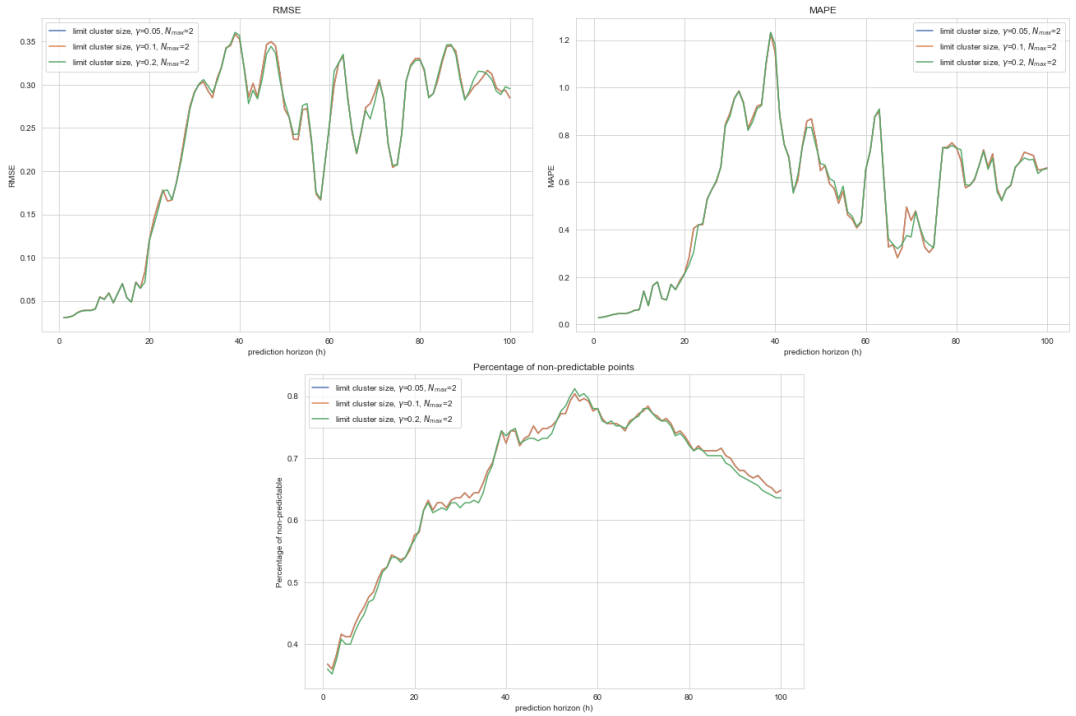


Рисунок 16. Графики зависимости RMSE, MAPE и количества непрогнозируемых точек от горизонта прогнозирования. Горизонт прогнозирования h=100. Тестовая выборка 250. Кластеризация мотивов — db, 20%. Алгоритм вычисления единого прогнозного значения — db. Алгоритмы определения непрогнозируемых точек: lcs_0.05_2 (синий), lcs_0.1_2 (оранжевый), lcs_0.2_2 (зеленый).

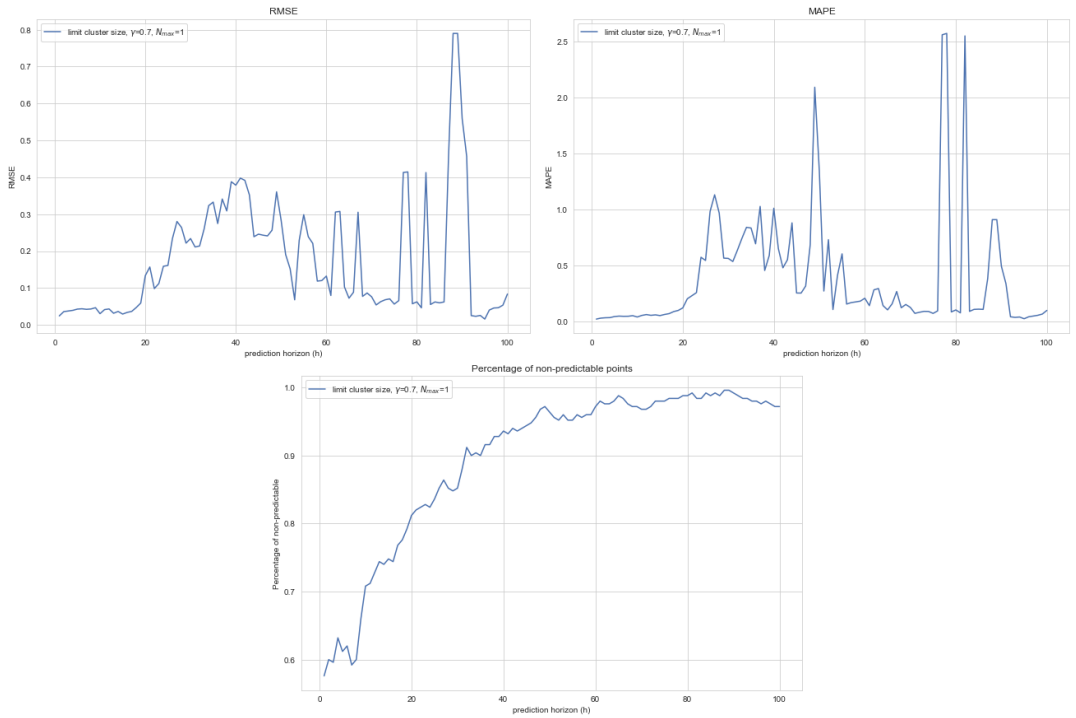
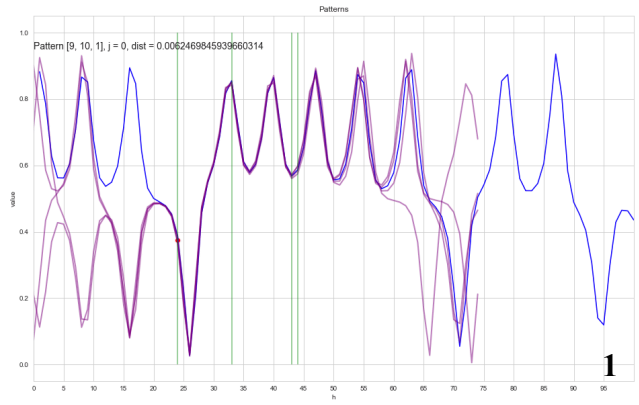
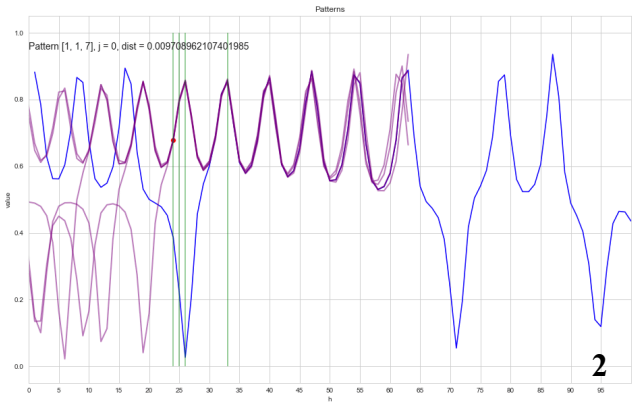
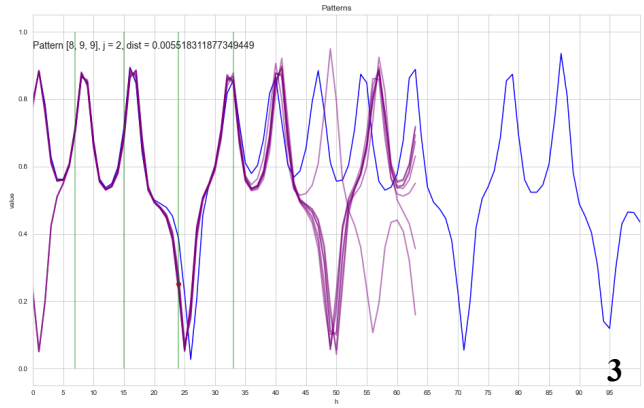


Рисунок 17. Графики зависимости RMSE, MAPE и количества непрогнозируемых точек от горизонта прогнозирования. Горизонт прогнозирования h=100. Тестовая выборка 250. Кластеризация мотивов — db, 20%. Алгоритм вычисления единого прогнозного значения — db. Алгоритмы определения непрогнозируемых точек: lcs_0.7_1 (синий).







Рисунки 18-20. Иллюстрация к исследованию конкретных отрезков настоящего временного ряда, которые составляют кластеры, центры которых близки к наблюдаемому ряду, для точки 24. На рисунках представлено по одному мотиву из каждого кластера множества возможных прогнозных значений, не являющегося выбросами. В исследовании были проанализированы все мотивы. Синим обозначен настоящий временной ряд, зеленые вертикальные линии — паттерн, по которому взят близкий мотив, фиолетовым — отрезки ряда, отвечающие кластеру обобщенных z-векторов, красным — центр кластера множества возможных прогнозных значений.

Приложение В. Описание классов и функций библиотеки *time_series_prediction*

В ходе работы была использована среда Jupyter Notebook, а также реализована библиотека *time_series_prediction*. Она содержит код алгоритма прогнозирования, в том числе алгоритма self-healing, код вычислительных экспериментов и код для постарения графиков, некоторые из которых представлены в отчете.

1. Модуль *predictor*

Модуль *predictor* содержит класс *TimeSeriesPredictor*, который реализует базовый алгоритм прогнозирования хаотических временных рядов с помощью кластеризации и алгоритм self-healing.

Таблица 2. Атрибуты и методы класса TimeSeriesPredictor

Класс TimeSeriesPredictor			
Атрибуты			
Имя	Тип	Описание	
clustered_motifs	list of size (n_patterns)	Список кортежей кластеризованных мотивов: [(pattern, clusters)] Кластеры (clusters) имеют формат: list of length (n_patterns) of np.array of shape (n_clusters, pattern_length + 1)	
non_pred_model	NonPredModel	Модель для определение непрогнозируемых точек	
k_max	int, default=10	Максимальное расстояние внутри паттерна	
pattern_length	int, default=3	Длина паттерна	
Методы			
Имя	Возвращаемое значение	Параметры	Описание
__init__	—	• clustered_motifs : list of size (n_patterns) — список кортежей кластеризованных мотивов • non_pred_model : NonPredModel — модель для определение непрогнозируемых точек • k_max : int — максимальное расстояние внутри паттерна • pattern_length : int — длина паттерна	Конструктор

cluster_motifs	• clustered_motifs : list of size (n_patterns) — список кортежей кластеризованных мотивов	• Y1 : list or 1d np.ndarray • — обучающая часть временного ряда • beta : float from 0 to 1, default=0.1 — процент использованных паттернов • mc_method : {'wi', 'db'}, default='db' — Метод кластеризации мотивов: 'wi' - Wishart, 'db' - DBSCAN • k_max : int, default=10 — максимальное расстояние внутри паттерна • pattern_length : int, default=3 — длина паттерна **kwargs : dict • eps : float from 0 to 1, default=0.01 — максимальное расстояние в одном кластере для кластеризации DBSCAN и Wishart • min_samples : int > 1 or float from 0 to 1, default=5 — минимальное число объектов в одном кластере для кластеризации DBSCAN и Wishart	Кластеризация мотивов, результат также сохраняется в атрибуты класса
set_motifs	—	• clustered_motifs : list of size (n_patterns) — список кортежей кластеризованных мотивов • k_max : int — максимальное расстояние внутри паттерна • pattern_length : int — длина паттерна	Установка кластеризованных мотивов в атрибуты класса
set_non_pred_model	—	• non_pred_model : NonPredModel — модель для определение непрогнозируемых точек	Установка модели определения непрогнозируемых точек в атрибуты класса
predict	• Y_pred : np.ndarray — прогнозные значения, 'N' для непрогнозируемых точек • possible_predictions_list : list — список множеств возможных прогнозных значений • trajectories : list — прогнозные траектории для alg_type='tp'	• Y_preceding : list or 1D np.array — сегмент временного ряда, который предшествует прогнозируемуму сегменту • h : int — горизонт прогнозирования • up_method : str from {'a', 'wi', 'db', 'op'} — алгоритм вычисления единого прогнозного значения • alg_type : str from {'s', 'tp'} — тип алгоритма: 's' - поточенное, 'tp' - траекторное • match_thershold : float, default=0.01 — порог для близкого мотива **kwargs • n_trajectories : int, default=20 — кол-во траекторий для alg_type='tp'	Базовый алгоритм прогнозирования на h шагов вперед

predict_one_step	• possible_predictions : np.ndarray — множество возможных прогнозных значений • distances : np.ndarray — расстояния до мотива, используются в некоторых алгоритмах вычисления единых прогнозных значений	• Y_preceding : list or 1D np.array — сегмент временного ряда, который предшествует прогнозируемуму сегменту • Y_pred : list or 1D np.ndarray — спрогнозированные на предыдущих шагах значения • match_thershold : float, default=0.01 — порог для близкого мотива	Один шаг базового алгоритма прогнозирования
unified_prediction	• avg : float or ’N' — единое прогнозное значение, ’N' для непрогнозируемых точек	• possible_predictions : np.ndarray — множество возможных прогнозных значений • up_method : str from {'a', 'wi', 'db', ‘op'} — алгоритм вычисления единого прогнозного значения (см. Приложение А)	Алгоритм вычисления единого прогнозного значения
unified_prediction_weighted	• avg : float or ’N' — единое прогнозное значение, ’N' для непрогнозируемых точек	• possible_predictions : np.ndarray — множество возможных прогнозных значений • sep_indices : list — список индексов, которые разделяют итерации в списке возможных прогнозных значений • up_method : str from {'a', 'wi', 'db', ‘op'} — алгоритм вычисления единого прогнозного значения (см. Приложение А) • weight_method : str from {'double_clustering', 'weighred_average', 'factor', 'pattern_length', 'pattern_length_dist', 'dist', ‘dist_factor'} — метод вычисления единого прогнозного значения с использованием весов (см. Приложение А)	Алгоритм вычисления единого прогнозного значения с использованием весов

self_healing_one_iteration	• new_up : np.ndarray — список новых прогнозных значений • possible_predictions : list of np.ndarrays — спиок множеств возможных прогнозных значений • sep_indices : list — список индексов, которые разделяют итерации в списке возможных прогнозных значений	• Y_preceding : list or 1D np.array — сегмент временного ряда, который предшествует прогнозируемуму сегменту • sep_indices : list — список индексов, которые разделяют итерации в списке возможных прогнозных значений • unified_predictions : list or 1D np.ndarray of length h — список единых прогнозных значений с предыдущей итерации • possible_predictions : list of h np.ndarrays — спиок множеств возможных прогнозных значений с предыдущей итераций • up_method : str from {'a', 'wi', 'db', ‘op'} — алгоритм вычисления единого прогнозного значения (см. Приложение А) • fixed_points_idx : list — спиок точек, которые не изменяют статус прогнозируемых • weight_method : str from {'double_clustering', 'weighred_average', 'factor', 'pattern_length', 'pattern_length_dist', 'dist', ‘dist_factor'} — метод вычисления единого прогнозного значения с использованием весов (см. Приложение А)	Одна итерация алгоритма self-healing
self_healing	• unified_predictions : list or 1D np.ndarray of length h — список единых прогнозных значений • possible_predictions : list of np.ndarrays — спиок множеств возможных прогнозных значений	• Y_preceding : list or 1D np.array — сегмент временного ряда, который предшествует прогнозируемуму сегменту • h : int — горизонт прогнозирования • unified_predictions : list or 1D np.ndarray of length h — список единых прогнозных значений из базового алгоритма • possible_predictions : list of h np.ndarrays — спиок множеств возможных прогнозных значений из базового алгоритма • healing_up_method : str from {'a', 'wi', 'db', ‘op'} — алгоритм вычисления единого прогнозного значения (см. Приложение А) • fixed_points_idx : list — спиок точек, которые не изменяют статус прогнозируемых • weight_method : str from {'double_clustering', 'weighred_average', 'factor', 'pattern_length', 'pattern_length_dist', 'dist', ‘dist_factor'} — метод вычисления единого прогнозного значения с использованием весов (см. Приложение А)	Алгоритм self-healing

2. Модуль *wishart*

Класс *Wishart* содержит реализации алгоритма кластеризации Wishart

Таблица 3. Атрибуты и методы класса Wishart

Класс TimeSeriesPredictor			
Атрибуты			
Имя	Тип	Описание	
wishart_neighbors	int	Минимальное число объектов в одном кластере	
significance_level	int	Максимальный уровень значимости кластера	
Методы			
Имя	Возвращаемое значение	Параметры	Описание
__init__	—	• wishart_neighbors : int — минимальное число объектов в одном кластере • significance_level : int — максимальный уровень значимости кластера	Конструктор
fit	• result : list — список меток кластеров	• X : list — список объектов для кластеризации	Алгоритм кластеризации Wishart

3. Модуль *experiment*

Таблица 4. Функции модуля experiment

Модуль experiment.py			
Функции			
Имя	Возвращаемое значение	Параметры	Описание
experiment_no_pm	—	• Y2: list or 1D np.ndarray — тестовая часть ряда • h_max: int — максимальный горизонт прогнозирования • n_iterations: int — размер тестовой выборки эксперимента • iterations_range: tuple — ограничения тестовой выборки • motif_clustering_params: dict — параметры кластеризации мотивов • prediction_params: dict — параметры базового алгоритма • healing_params: dict — параменты алгоритма self-healing • non_pred_model_prediction : NonPredModel — модель определения непрогнозируемых точек для базового алгоритма • non_pred_model_healing : NonPredModel — модель определения непрогнозируемых точек для алгоритма self-healing	Эксперимент оценки качества прогнозирования, не использует матрицу прогнозов, запись результатов в файл. Есть возможность параллельного выполнения задач.
experiment	—	Те же, что и в функции *experiment_no_pm*	Эксперимент оценки качества прогнозирования, использует матрицу прогнозов
thrown_points_experiment	—	• Y1: list or 1D np.ndarray — обучающая часть ряда • Y2: list or 1D np.ndarray — тестовая часть ряда • h_max: int — максимальный горизонт прогнозирования • healing_params: dict — параменты алгоритма self-healing • logs_filepath : str — путь к файлу записи результатов • step : int — шаг	Эксперимент для ряда с выкинутыми точками

4. Модуль *graph*

В таблице перечислены основные функции для постарения графиков для анализа результатов исследования. Остальные функции модуля являются служебными.

Таблица 5. Функции модуля graph.py

Модуль graph.py			
Функции			
Имя	Возвращаемое значение	Параметры	Описание
plot_stats	—	• stats_ds: pd.DataFrame columns: points_left, n_iterations, non_predictable, rmse, mape — таблица статистики по исследованию либо • stats_filename: str — путь к файлу с таблицей	Графики зависимости RMSE, MAPE и количества непрогнозируемых точек от горизонта прогнозирования из готового файла со статистикой
plot_unified_and_possible_preds	—	• up: list or 1D np.ndarray — список единых прогнзных значений • pp: list of lists or list of np.ndarrays — список множеств возможных прогнозных значений • Y2: list or 1D np.ndarray — настоящие значения временного ряда	График единых прогнозных значений и множеств возможных прогнозных значений, а также настоящие значения временного ряда
plot_healing_animation	—	• working_path : str — путь к файлу с логами по алгоритму self-healing • Y2: list or 1D np.ndarray — настоящие значения временного ряда	Анимация алгоритма self-healing
plot_experiment_results	—	• working_directory : str — путь к директории с матрицами прогнозов • exp_short_names : list — список коротких наименований алгоритмов, матрицы прогнозов которых используются в исследовании • h_max : int — максимальный горизонт прогнозирования • n_iterations : int —размер тестовой выборки	Графики зависимости RMSE, MAPE и количества непрогнозируемых точек от горизонта прогнозирования из матриц прогнозов
plot_thrown_points_exp_results	—	• working_directory : str — путь к директории с логами алгоритма self-healing • exp_short_names : list — список коротких наименований алгоритмов, логи которых используются в исследовании • h_max : int — максимальный горизонт прогнозирования • n_iterations : int —размер тестовой выборки	Графики зависимости количества итераций, количества непрогнозируемых точек, RMSE и MAPE от количества выкинутых точек

5. Модуль *non_pred_model*

Таблица 6. Методы корневого класса NonPredModel, наследниками которого являются все классы моделей определения непрогнозируемых точек

Класс NonPredModel			
Методы			
Имя	Возвращаемое значение	Параметры	Описание
is_predictable	is_predictable : boolean — прогнозируемая ли точка	possible_predictions : list — множество возможных прогнозных значений **kwargs — другие параметры, необходимые в классах-наследниках	Функция, которая определяет прогнозируемая ли точка по множеству возможных прогнозных значений
reset	—	—	Сброс атрибутов класса
is_predictable_by_up	is_predictable : boolean — прогнозируемая ли точка	unified_predictions : list — единые прогнозные значения **kwargs — другие параметры, необходимые в классах-наследниках	Функция, которая определяет прогнозируемая ли точка по списку единых прогнозных значений
is_predictable_by_up_log	is_pred : list of boolean — прогнозируемы ли точки	up_log : list — лог единых прогнозных значений **kwargs — другие параметры, необходимые в классах-наследниках	Функция, которая определяет прогнозируемая ли точка по логу единых прогнозных значений

Таблица 7. Классы моделей определения непрогнозируемых точек

Модуль non_pred_model.py	
Классы	
Имя	Алгоритм, который реализуется в классе
ForcedPredictionNPM	Принудительное прогнозирование (fp)
LargeSpreadNPM	Большой разброс (ls)
RapidGrowthNPM	Быстрый рост разброса (rg)
RapidGrowthDBSCANNPM	Быстрый рост разброса кластеров DBSCAN (rd)
RapidGrowthWishartNPM	Быстрый рост разброса кластеров Wishart (rw)
LimitClusterSizeNPM	Ограничение на размер максимального кластера и на количество кластеров (lcs)
BigLeapNPM	Большой скачок (big leap)
BigLeapBtwIterationsNPM	Большой скачок между итерациями (big leap between iterations)
WeirdPatternsNPM	Странные паттерны (weird patterns)