

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ
ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ
«БЕЛГОРОДСКИЙ ГОСУДАРСТВЕННЫЙ НАЦИОНАЛЬНЫЙ
ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ» (Н И У « Б е л Г У »)

ИНСТИТУТ ИНЖЕНЕРНЫХ И ЦИФРОВЫХ ТЕХНОЛОГИЙ
Кафедра информационных и робототехнических систем

**АВТОМАТИЗИРОВАННАЯ СИСТЕМА ПРОГНОЗИРОВАНИЯ
ОБЪЕМОВ ПРОДАЖ ОПТОВОЙ ТОРГОВОЙ КОМПАНИИ С
УЧЕТОМ СЕЗОННОСТИ**

Магистерская диссертация
обучающегося по направлению подготовки
09.04.02 Информационные системы и технологии
заочной формы обучения
группы 12001673
Литвиновой Алины Александровны

Научный руководитель
к.т.н. Щербина Н.В.

Рецензент
Коммерческий директор
ООО «ИКРА.РУ ФУДСЕРВИС»
Васильченко А.В

БЕЛГОРОД 2019

СОДЕРЖАНИЕ

ВВЕДЕНИЕ.....	5
1 Исследование процесса прогнозирования объемов продаж.....	8
1.1 Процесс прогнозирования объемов продаж	8
1.2 Обзор существующих методов и моделей прогнозирования	13
1.3 Оценка эффективности различных методов и моделей прогнозирования	20
1.4 Актуальность разрабатываемой модели	21
2 Моделирование и проектирование информационной системы	24
2.1 Разработка комбинированного способа прогнозирования.....	24
2.2 Описание используемых алгоритмов	28
2.3 Выбор программных средств реализации.....	33
2.4 Проектирование информационной системы	36
3 Описание разработанной автоматизированной системы прогнозирования продаж	40
3.1 Описание разработанного пользовательского интерфейса.....	40
3.2 Оценка эффективности системы	54
ЗАКЛЮЧЕНИЕ	57
СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ	59
ПРИЛОЖЕНИЕ А	61

ВВЕДЕНИЕ

Постановка целей, планирование и анализ ошибок – это три ключевых момента, без которых сегодня невозможно добиться успеха ни в одном деле, будь то повышение квалификации, ведение семейного бюджета или управление государством.

Все это безусловно применимо и к управлению бизнесом. В настоящее время каждое даже самое малое предприятие имеет не только вполне конкретные цели, но и план продаж, позволяющий этих целей достичь. Для того, чтобы планы были выполнимы и реалистичны, необходимо учитывать множество факторов, которые могут повлиять на объем продаж, а также анализировать предыдущие результаты. На основе этих данных компания и составляет прогноз объема продаж.

Воеводин В.В. в книге «Вычислительная математика и структура алгоритмов» дает такую рекомендацию: «к процессу прогнозирования объема продаж предприятия следует подходить реалистично, понимая, что абсолютно надежного метода не существует».

Существующие методы прогнозирования можно представить в виде статистических и оценочных прогнозов. Для статистических прогнозов используют методы: наименьших квадратов, регрессионного и корреляционного анализа, скользящей средней, экспоненциального сглаживания (взвешенной скользящей средней), следящего сигнала и др.

Оценочные прогнозы основаны на мнении экспертов. В качестве экспертов выступают специалисты, знающие рынок, новые тенденции движения цен, продукцию конкурентов, а также любую другую информацию о возможных изменениях потребности товара.

В настоящее время существуют статистические методы, позволяющие спрогнозировать объем продаж с учетом не только внутренних факторов, но и внешних, количественных и даже качественных.

При прогнозировании лучше всего совмещать статистические прогнозы и оценочные, конкретные числа и деловую интуицию, основанную на опыте экспертов».

Для автоматизации процесса прогнозирования на практике чаще всего используют MS Office Excel, используя встроенные пакеты и внося данные вручную. И редко используются специализированные программы, такие как Statistica и Autobox, ForeCAST Pro, KORUS Forecast.

Объектом исследования выступает процесс прогнозирования объема продаж с сезонной компонентой.

Предметом исследования магистерской работы является прогнозирование объема продаж предприятия с учетом сезонной компоненты.

Целью магистерской диссертации является оптимизация прогнозирования объема продаж с сезонной компонентой.

Для выполнения поставленной задачи необходимо создание информационной системы прогнозирования объема продаж с сезонной компонентой, которая должна облегчить процесс прогнозирования объема продаж, обеспечить возможность варьирования сезона.

Научная новизна состоит в разработке оригинальной модели прогнозирования объема продаж с сезонной компонентой.

В ходе выполнения магистерской диссертации необходимо решить следующие задачи:

- проанализировать предметную область и существующие подходы в сфере прогнозирования объемов продаж;
- выявить и формализовать факторы, учитываемые в процессе прогнозирования;
- построить функциональные модели бизнес-процесса прогнозирования объема продаж;
- адаптировать метод поиска похожих подпоследовательностей временного ряда прогнозирования объема продаж с учетом сезонной

компоненты под особенности прогнозирования объема продаж конкретного предприятия;

- разработать автоматизированную систему прогнозирования объемов продаж оптовой торговой компании с учетом сезонности;

- оценить эффективность адаптации метода поиска похожих подпоследовательностей временного ряда для прогнозирования объемов продаж на предприятии.

Магистерская диссертация включает введение, три раздела, заключение, список использованных источников, приложение.

В первом разделе рассматривается существующее состояние предметной области: описываются характеристики и особенности прогнозирования, разновидности способов прогнозирования, методы оценки результатов прогнозирования, анализируются недостатки существующих информационных систем для прогнозирования объема продаж, и обосновывается предложение по устранению найденных недостатков и, внедрению новых подходов.

Второй раздел содержит описание используемых алгоритмов и особенностей их применения в конкретной ситуации. Выполнено обоснование проектных решений по информационному, программному и технологическому обеспечению задачи, а также проектирование информационной системы.

Третий раздел содержит описание разработанной автоматизированной системы, в том числе описание разработанного пользовательского интерфейса, а также оценку эффективности метода. Выполнен прогноз объема продаж с учетом сезонности.

В заключении подведены итоги диссертационного исследования: оценена практическая значимость результатов, задачи проверены на выполнение, отмечено достижение заданной цели исследования.

Магистерская диссертация написана на 65 листах, содержит 35 рисунков, 2 таблицы и приложение.

1 Исследование процесса прогнозирования объемов продаж

1.1 Процесс прогнозирования объемов продаж

Независимо от отрасли и сферы хозяйственной деятельности фирмы, ее руководству постоянно приходится принимать решения, последствия которых проявятся в будущем. Практически любое решение в компании основывается на аналитике имеющихся данных. Главным из инструментов в помощи принятия решения на основе имеющихся данных является прогнозирование.

Одним из путей повышения эффективности деятельности компании является качественное прогнозирование продаж. Прогноз продаж представляет собой величину объема продаж, которую возможно достигнуть при выполнении неких условий или при реализации некоторых событий. Данная величина может использоваться для уточнения плана продаж, который является одной из ключевых точек планирования в компании.

Точно рассчитанный прогноз позволяет:

- более эффективно проводить контроль и оптимизацию расходов.
- конкретизировать цель. Менеджеры работают на достижение численных показателей: количество холодных звонков, общая сумма сделок за месяц, число подписанных контрактов и т.д.
- вести бизнес более гибко. Сезонные изменения или трудности, вызванные влиянием внешних факторов, негативно сказываются на объеме продаж. Прогнозирование помогает вовремя подстроиться под изменения, а постоянный мониторинг показателей помогает быстрее принимать решения [1].

На практике существует огромное количество подходов к прогнозированию, но программного обеспечения, позволяющего выполнить автоматизированный прогноз продаж не так много. В основном эти решения предназначены для крупного и среднего бизнеса и стоят достаточно дорого

для предпринимателей, чей бизнес невелик. Среди таких решений находятся Autobox, ForeCAST Pro, KORUS | Forecast и другие, предоставляющие широкий инструментарий, который учитывает большое количество товарных позиций, сезонность, содержат большое количество реализованных статистических методов. Все они поддерживают импорт данных в самых используемых табличных форматах.

Существуют так же программные решение STATISTICA, которое предоставляет широкие функции по прогнозированию, в том числе при помощи таких современных инструментов как нейронные сети (Рисунок 1).

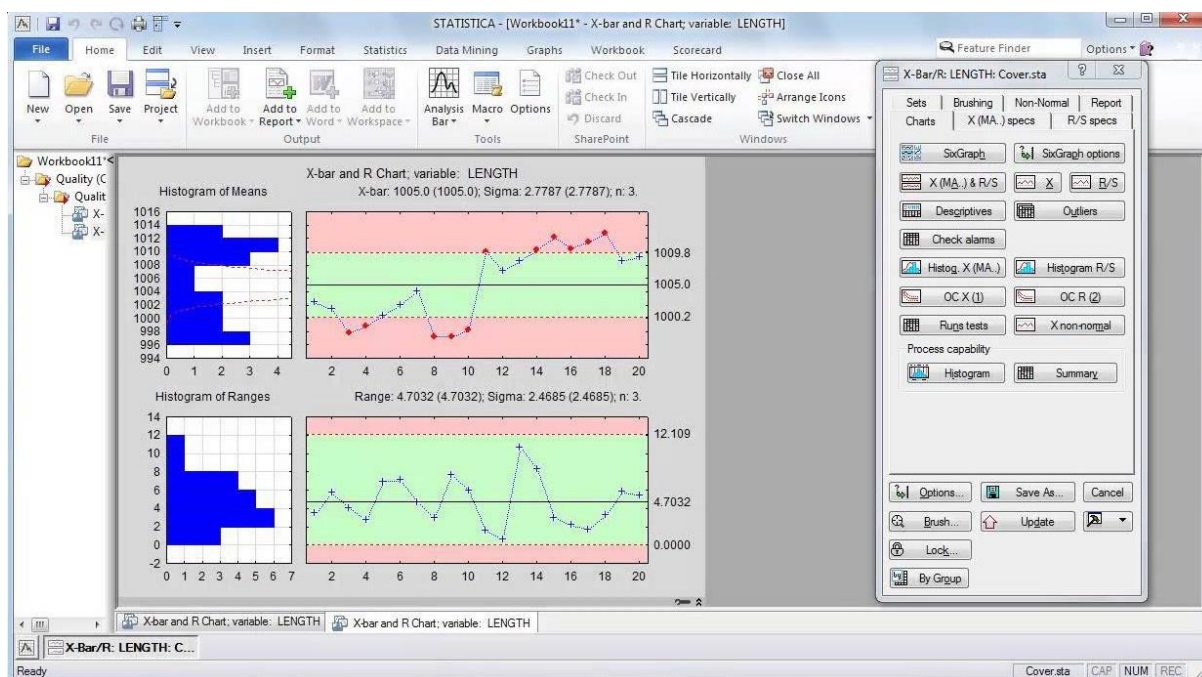


Рисунок 1 - Интерфейс программного продукта STATISTICA

STATISTICA является продуктом из высокого ценового сегмента, а потому может стать неоправданно дорогим продуктом для предприятий малого бизнеса. Кроме того, интерфейс на английском языке с большим количеством надстроек труден для понимания.

Отдельная категория систем – это системы управления цепочками поставок. Это категории экспертных систем для огромного ассортимента,

который подвержен активным изменениям и позволяет строить прогнозные модели.

Одна из популярных систем «1С: Предприятие 8. Управление производственным предприятием» тоже подходит для малого, среднего и крупного бизнеса своим функционалом и позволяет строить прогнозные отчеты по предприятию и подразделениям, содержит большое количество аналитических функций.

В системе «1С:Предприятие» существует небольшой, но достаточно современный пакет аналитических функций, позволяющих построить собственный прогноз. Так как эта система очень распространена, то небольшим предприятиям хотя бы с одним программистом штате может быть вполне целесообразно использовать эти функции. Рядовому же пользователю доступна только линия тренда, которую можем видеть на рисунке 2.

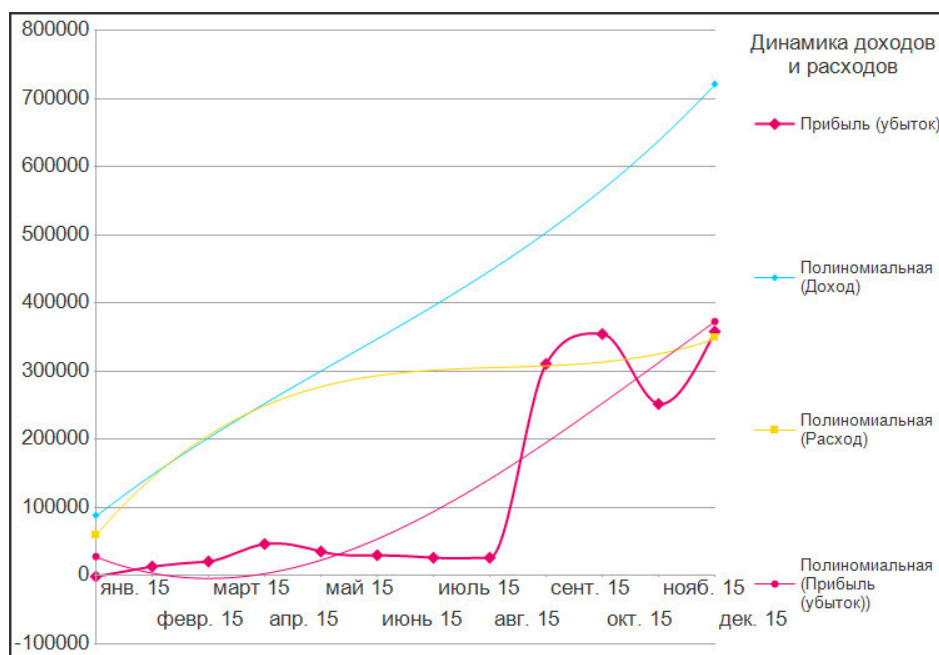


Рисунок 2 - Прогнозирование объема продаж при помощи линии тренда

К продуктам высокого класса прогнозирования так же относятся KORUS Forecast и ForeCAST Pro. Интерфейс первого из которых можно рассмотреть ниже на рисунке 3.

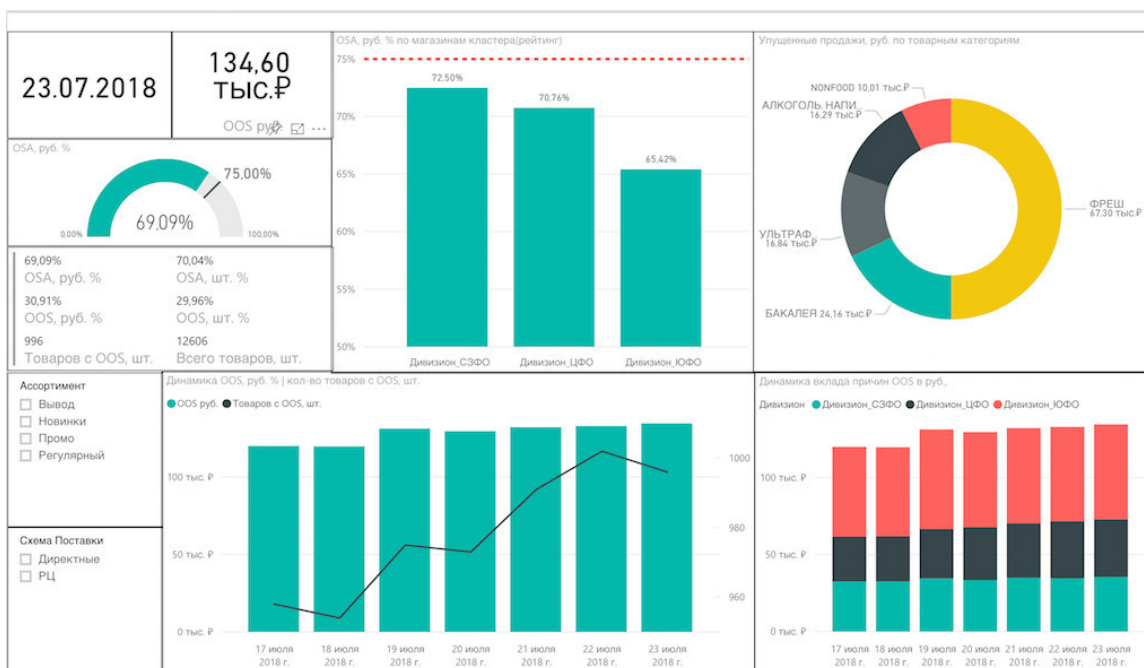


Рисунок 3 - Интерфейс программного продукта KORUS FORECAST

KORUS FORECAST является облачным сервисом для прогнозирования спроса. Он разработан специалистами «КОРУС Консалтинг» с учетом отраслевой специфики ритейла.

На официальном сайте заявлены следующие возможности этого продукта, который предоставляет пакет настроенных аналитических отчетов, позволяющих в режиме реального времени отслеживать состояние наиболее важных показателей по прогнозу и логистике:

- отчет по оценке прогноза на любом уровне детализации географической и товарной иерархии;
- отчет, показывающий ошибки прогнозирования и причины их возникновения;
- отчет об уровне сервиса логистики по всей сети с возможностью детализации до конкретного магазина (может быть доработан под ваши требования);
- отчет по периоду оборачиваемости товаров с агрегацией до недели, позволяющий получить информацию в динамике;
- отчет по списанию продукции;

- отчет по эффективности промо-акций;
 - отчет по неликвидным запасам;
 - диаграмма рассеивания по отсутствию товара на полке в разрезе категорий, магазинов;
 - диаграмма рассеивания по оборачиваемости и товарному запасу.
- На рисунке 4 представлен интерфейс программы ForeCAST Pro.

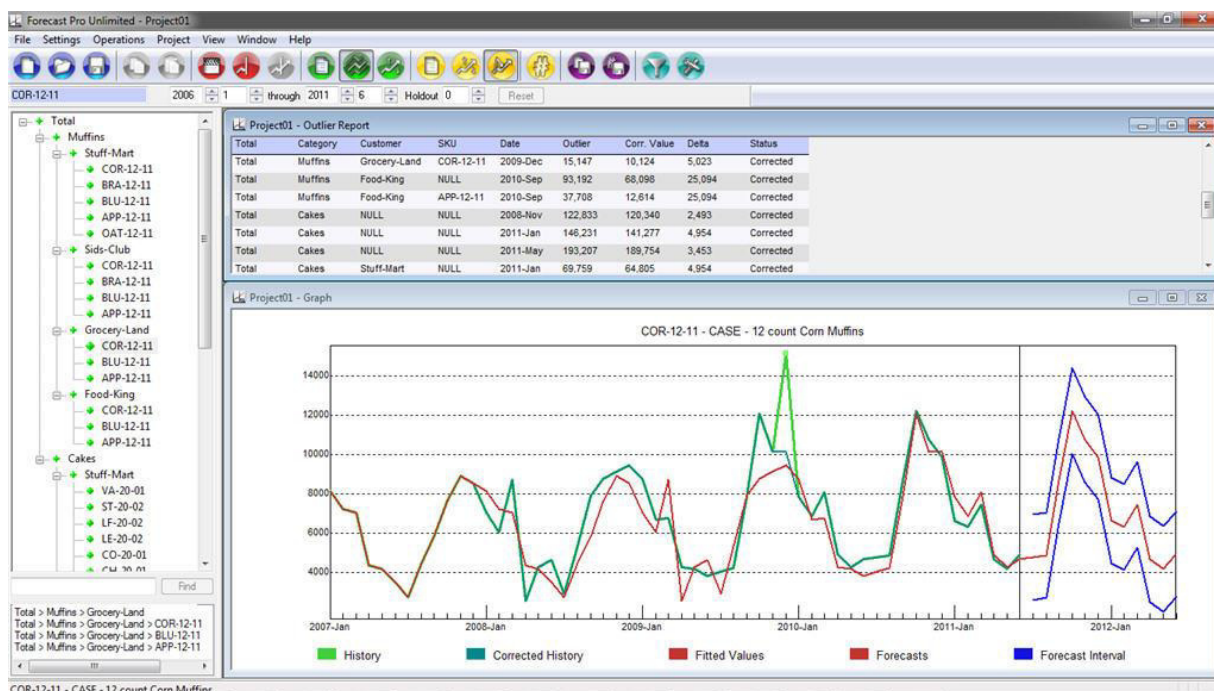


Рисунок 4 - Интерфейс программного продукта FORECAST PRO

Novo Forecast, что программа автоматически анализирует временные ряды, подбирает модель прогнозирования и рассчитывает прогноз на любом уровне детализации - по географическим локациям, клиентам, товарным категориям, товарам, месяцам, дням. Для подбора моделей используются алгоритмы машинного обучения и библиотека из 3000+ комбинаций прогнозных моделей. Подбор моделей происходит в каждом расчетном цикле. Алгоритмы машинного обучения адаптируются к специфике данных и поддерживают максимально высокое качество прогнозов.

Официальный сайт утверждает, что прогноз регулярно, автоматически корректируется факторами. Корректировка прогноза происходит по расписанию и с частотой, установленной клиентом. Для корректировки используются согласованные факторы из интерфейса Novo Forecast Enterprise или из других источников.

Большая же часть компаний малой и средней величины пользуются пакетом MS Office Excel. Этот пакет содержит огромные аналитические возможности, содержит основные статистические методы, позволяет реализовать вручную любую аналитическую модель, визуализировать данные, построить отчет. Этот инструмент очень популярен и опыт работы в нем имеет уже большое количество людей. Однако для построения сложных методов необходимо иметь опыт использования математических инструментов и большое количество времени.

1.2 Обзор существующих методов и моделей прогнозирования

Что такое продажа? Это сделка в определенный момент времени, которая приносит прибыль. У нас есть последовательность сделок в прошлом и предполагаемые сделки в будущем. Соответственно, мы можем использовать для прогнозирования на основании прошлых продаж или предположить вероятности сделок в будущем.

Власов Д.В., Божко В.П. в пособии «Информационные технологии в экономике и управлении» описывают методы прогнозирования следующим образом: «существуют, например, самые простейшие методы прогнозирования, основанные на присвоении каждой сделке строго определенного процента вероятности. А затем произведения вероятности сделки на её сумму суммируются.

В этом подходе очевидно, что большие суммы сделок в районе вероятных 20–60 % могут сделать ожидаемый результат (взвешенный прогноз) нереально большим.

Можно сделать некоторые градации и распределить прогнозы по категориям: пессимистичный, реалистичный и оптимистичный прогноз, а также добавить им веса-коэффициенты. Продавец пишет, какую сумму он ожидает от всех сделок в худшем случае (пессимистичный вариант), затем – что, скорее всего, произойдет (реалистичный вариант), и, наконец, какой суммы стоит ожидать, если все будет развиваться в пользу компании (оптимистичный вариант). Этот вариант самый простой, со всеми плюсами и минусами, вытекающими из этого факта.

Отрицательной чертой этого метода является очевидный недостаток информации о каждой конкретной сделке. Вероятность того, что факт-результат совпадет с реалистичным вариантом прогноза, очень мала. Естественно, в этой системе взвешенный прогноз вычисляется как «реалистичный вариант».

Е.Н. Барикаев в статье «Методы экспертных оценок» группирует предлагает следующую классификацию методов:

- методы экспертных оценок;
- экономико-статистические, математические методы.

Большой экономический словарь гласит, что «метод экспертных оценок – это способ прогнозирования и оценки будущих результатов действий на основе прогнозов специалистов. При применении метода экспертных оценок проводится опрос специальной группы экспертов (5-7 человек) с целью определения определенных переменных величин, необходимых для оценки исследуемого вопроса. В состав экспертов следует включать людей с разными типами мышления - образное и словесно-логическое, что способствует успешному решению проблемы. Методы бывают индивидуальные и коллективные. Для уменьшения риска ошибки общий результат прогноза обычно складывается из взвешенных оценок нескольких экспертов. К ним

относят метод «комиссий», «метод Дельфи», метод «коллективной генерации идей» («мозговая атака»), метод морфологического анализа и другие».

Завьялов П.С. в книге «Маркетинг в схемах, рисунках и таблицах» приводит следующую классификацию экспертных методов, которая представлена в таблице 1.

Таблица 1 - Классификация экспертных методов прогнозирования

Метод	Определение	Формы, диапазон	Экономический пример	Преимущества и проблемы
Метод Дельфи	Форма опроса экспертов, при которой их анонимные ответы собирают в течение нескольких туров и через ознакомление с промежуточными результатами участниками получают групповую оценку интересующего процесса	Долгосрочные	Производство, сбыт отрасли при различных величинах влияющих факторов, изменения общественных норм поведения	Наглядность результатов, привлечение экспертов по интересующей проблеме, анонимность участников. Негибкая методика, высокая потребность во времени, тенденция к консервативным оценкам, непредсказуемость технических изменений
Сценарии	Предсказание развития и будущего состояния факторов, влияющих на предприятие и определение возможных действий предпринимателя	Долгосрочные	Исследования типа: «Вертолет начала следующего столетия — будущее коммуникационной техники — будущее международных космических исследований»	Подходит для сложных комплексных проблем типа генерации идей новых продуктов или структурирования стратегического планирования диверсификации. Высокая субъективность оценок.

Эти методы предпочтительно применяются в случаях затрудненности качественного прогноза автоматизированными средствами [2].

Экономико-статистические и математические методы - группа методов, включающая в себя обширный набор различных количественных средств.

Здесь для начала необходимо ввести некоторые понятия. Ю. Г. Чучуева в своей докторской диссертации «Прогнозирование методом наибольшего подобия» выделяет указанные ниже различия метода и модели.

Метод прогнозирования - представляет собой последовательность действий, которые нужно совершить для получения модели прогнозирования.

Модель прогнозирования — это функциональное представление, адекватно описывающее исследуемый процесс и являющееся основой для получения его будущих значений [3].

Общую классификацию методов и моделей можно видеть на рисунке 5 ниже.

К достоинствам регрессионных моделей и методов моделей относят простоту, гибкость, а также единообразие их анализа и проектирования [4]. При использовании линейных регрессионных моделей результат прогнозирования может быть получен быстрее, чем при использовании остальных моделей. Кроме того, достоинством является прозрачность моделирования, т. е. доступность для анализа всех промежуточных вычислений.

Ю. Г. Чучуева в своей докторской диссертации «Прогнозирование методом наибольшего подобия» пишет, что «основным недостатком нелинейных регрессионных моделей является сложность определения вида функциональной зависимости, а также трудоемкость определения параметров модели. Недостатками линейных регрессионных моделей являются низкая адаптивность и отсутствие способности моделирования нелинейных процессов.



Рисунок 5 - Классификация методов и моделей прогнозирования

Важными достоинствами авторегрессионных моделей являются их простота и прозрачность моделирования. Еще одним достоинством является единообразие анализа и проектирования, заложенное в работе [5]. На сегодняшний день данный класс моделей является одним из наиболее популярных, а потому в открытом доступе легко найти примеры применения авторегрессионных моделей для решения задач прогнозирования временных рядов различных предметных областей. Недостатками данного класса моделей являются: большое число параметров модели, идентификация которых неоднозначна и ресурсоемка [6]; низкая адаптивность моделей, а также линейность и, как следствие, отсутствие способности моделирования нелинейных процессов, часто встречающихся на практике.

Достоинствами моделей и методов экспоненциального сглаживания являются простота и единообразие их анализа и проектирования. Данный

класс моделей чаще других используется для долгосрочного прогнозирования [7]. Недостатком данного класса моделей является отсутствие гибкости. Преимуществом нейросетевых моделей является нелинейность, т. е. способность устанавливать нелинейные зависимости между будущими и фактическими значениями процессов. Другими важными достоинствами являются: адаптивность, масштабируемость (параллельная структура ANN ускоряет вычисления) и единообразие их анализа и проектирования [8]. При этом недостатками ANN являются отсутствие прозрачности моделирования; сложность выбора архитектуры, высокие требования к непротиворечивости обучающей выборки; сложность выбора алгоритма обучения и ресурсоемкость процесса их обучения.

Простота и единообразие анализа и проектирования являются достоинствами моделей на базе цепей Маркова. Недостатком данных моделей является отсутствие возможности моделирования процессов с длинной памятью [9].

Модели на базе классификационно-регрессионных деревьев. Достоинствами данного класса моделей являются: масштабируемость, за счет которой возможна быстрая обработка сверхбольших объемов данных; быстрота и однозначность процесса обучения дерева (в отличие от ANN) [9], а также возможность использовать категориальные внешние факторы. Недостатками данных моделей являются неоднозначность алгоритма построения структуры дерева; сложность вопроса останова т. е. вопроса о том, когда стоит прекратить дальнейшие ветвления; отсутствие единообразия их анализа и проектирования [10]».

Среди такого многообразия моделей и методов сложно не заблудиться. Одним из подходов к выбору метода прогнозирования является определение степени структурированности решаемой задачи.

Существуют так называемые стандартные задачи. В этом случае изменение фактора приводит к однозначному результату. Зависимость между фактом и результатом может быть выражена математической формулой.

Следующий класс - структурированные задачи. Для них характерно следующее свойство: при изменении фактора результат может меняться в достаточно узком интервале значений. В слабоструктурированных задачах изменение фактора в свою очередь приводит к изменению результата достаточно широком интервале значений. И последний класс — это неструктурированные задачи. Здесь изменение фактора может приводить к совершенно непредсказуемому результату. Определение класса зависит от особенностей объекта прогнозирования, уровня управления и срока упреждения. Общие тенденции таковы: интервал прогнозирования обратно пропорционален уровню структурированности задачи прогнозирования, как и уровень управления. Для стандартных задач рекомендуется применять методы экстраполяции и трендовые модели. Для структурированных – трендовые и экономические. Для слабоструктурированных – эконометрические модели и методы экспертных оценок. Для неструктурированных – методы экспертных оценок.

Необходимо помнить так же, выбор метода определяется так же достаточным объемом «чистых» данных, особенностями бизнеса, располагаемым бюджетом, скоростью окупаемости и перспективами развития компании.

Каждая из рассмотренных групп методов обладает определенными достоинствами и недостатками. Но для наилучших результатов следует обеспечивать одновременное использование количественных и качественных методов прогнозирования.

Нужно дополнительно отметить, что ни для одной из рассмотренных групп моделей (и методов) в достоинствах не указана точность прогнозирования. Это сделано в связи с тем, что точность прогнозирования того или иного процесса зависит не только от модели, но и от опыта исследователя, от доступности данных, от располагаемой аппаратной мощности и многих других факторов.

1.3 Оценка эффективности различных методов и моделей прогнозирования

Следующие распространенные показатели ошибки относятся к показателям средних ошибок прогнозирования.

Для оценки эффективности, которая достигается за счет повышения точности прогноза используется MAPE – средняя абсолютная ошибка в процентах:

$$\text{MAPE} = \frac{1}{N} \sum_{t=1}^N \frac{|z(t) - \hat{z}(t)|}{z(t)} \cdot 100\%, \quad (1)$$

где $z(t)$ – фактическое значение временного ряда, а $\hat{z}(t)$ – прогнозное.

Данная оценка показывает на сколько велики ошибки в сравнении со значениями ряда, хорошо подходит для сравнения одной модели для разных рядов, часто используется для сравнения разных моделей для одного ряда.

Она также применяется для временных рядов, фактические значения которых значительно больше 1. Например, оценки ошибки прогнозирования энергопотребления почти во всех статьях приводятся как значения MAPE.

Если же фактические значения временного ряда близки к 0, то в знаменателе окажется очень маленькое число, что сделает значение MAPE близким к бесконечности – это не совсем корректно. Для рядов, содержащих значения близкие к нулю, применяют следующую оценку ошибки прогноза.

Если ряды содержат значения близкие к нулю, тогда мы применяли оценку ошибки прогноза MAE – средняя абсолютная ошибка:

$$\text{MAE} = \frac{1}{N} \sum_{t=1}^N \frac{|z(t) - \hat{z}(t)|}{z(t)}, \quad (2)$$

где $z(t)$ – фактическое значение временного ряда, а $\hat{z}(t)$ – прогнозное.

Кроме указанных методов иногда используют другие оценки ошибки.

Формула средней ошибки:

$$ME = \frac{1}{N} \sum_{t=1}^N (z(t) - \hat{z}(t)) \quad (3)$$

Встречается еще другое название этого показателя — Bias (англ. – смещение) демонстрирует величину отклонения, а также — в какую сторону прогноз продаж отклоняется от фактической потребности. Этот индикатор показывает, был ли прогноз оптимистичным или пессимистичным. То есть, отрицательное значение Bias говорит о том, что прогноз был завышен (реальная потребность оказалась ниже), и, наоборот, положительное значение о том, что прогноз был занижен [12].

Дополнительным и более практически применимым способом оценки точности прогноза является простое сравнение значения, спрогнозированного тем или иным методом, с фактическим показателем за этот месяц. Однако это невозможно сделать заранее, очевидно.

1.4 Актуальность разрабатываемой модели

Прогнозирование начинается с корректной формулировки проблемы. Выбор конкретного метода прогнозирования зависит от множества моментов: достаточно ли объективной информации о продажах (как долго существует данный товар или его аналоги), ожидаются ли значительные изменения в ассортименте, закупках, уровне спроса, существуют ли внутренние зависимости между изучаемыми показателями и/или внутри массивов данных, являются ли данные временным рядом, существуют ли сезонные колебания и т.д.

Часто краеугольным ограничением при выборе метода прогнозирования является исходная информация: ее тип, доступность, возможность обработки, однородность, объем, чистота.

Описанные методы могут учитывать огромное количество факторов, но не все они доступны для малого бизнеса, будучи реализованными в продуктах высокого ценового сегмента. Реализация же их самостоятельно менеджером может занять длительное время, либо же станет совершенно невозможной в зависимости от математической сложности метода. Но даже будучи реализованной вручную, например, в MS Office Excel, она по-прежнему будет занимать большое количество времени, когда прогноз необходимо будет повторить для других данных, либо с другими параметрами.

Подбор метода прогнозирования затрудняется в случае, когда временной ряд трудно аппроксимировать даже полиномиальной функцией. Примеры таких временных рядов можно встретить в компании, продающей так называемый монопродукт даже в b2b сегменте. Главная особенность таких продаж состоит в том, что большую роль играет человеческий фактор: как хорошо работают менеджеры по продажам, как часто болеют, как часто меняются, как часто происходят перебои с поставками продукта, сбои на производстве и т.д. При этом сама цена на продукт меняется крайне медленно и незначительно.

Все это говорит нам о том, что учесть такие параметры математически крайне сложно, а потому наиболее точный прогноз менеджер может выполнить, опираясь на свое внутреннее профессиональное чутье. Основных путей расширения объема продаж два: расширение клиентской базы или увеличение объема продаж конкретному клиенту. Как рассчитать объем продаж, если мы знаем, что в ближайшее время клиент собирается организовать акцию? Такое событие можно только знать, предсказать его по изменению курса валют и политики крайне затруднительно. Дополнительно продукт может иметь свою специфичную сезонность.

Особенностью такой компании в свою очередь также является наличие двух категорий клиентов: постоянной и хаотичной. Поэтому фактором, влияющим на объем продаж является в первую очередь потеря постоянного крупного клиента.

С математической точки зрения существующий временной ряд содержит резкие всплески или падения, часть из которых являются признаками хаотичных покупок клиентов, восстановление и прерывание поставок и прочие «случайные» события. Другие же всплески говорят о сезонности и являются циклическими, закономерными.

Таким образом, необходимо создать автоматизированную систему, позволяющую пользователю дополнительно указывать цикличность, сезонность - параметры, позволяющие оптимизировать процесс прогнозирования. При этом она должна быть простой в использовании,

Выводы по первому разделу: в данном разделе были исследованы процесс прогнозирования, рассмотрены современные методы и модели прогнозирования, описаны принципы оценки их эффективности. Проведен анализ существующих аналогичных информационных систем и выявлены их недостатки. Обоснована необходимость в разработке нового подхода прогнозирования.

2 Моделирование и проектирование информационной системы

2.1 Разработка комбинированного способа прогнозирования

Одной из популярных современных тенденций в области создания моделей прогнозирования является создание комбинированных моделей и методов. Подобный подход дает возможность компенсировать недостатки одних моделей при помощи других, и направлен на повышение точности прогнозирования, как одного из главных критериев эффективности модели. Одной из первых работ в этой области является статья [13]. В ней предлагается подход, в котором прогнозирование временного ряда осуществляется в два этапа. На первом этапе на основании моделей распознавания образов (pattern recognition) выделяются гомогенные группы (patterns) временного ряда. На следующем этапе для каждой группы строится отдельная модель прогнозирования. В статье указывается, что при комбинированном подходе удастся повысить точность прогнозирования временных рядов.

Формализуем нашу задачу прогнозирования объема продаж представив данные об объеме продаж как временной ряд из пар – момент времени и значение выручки. Пусть имеется временной ряд $X(t)$, где $t = 1, \dots, T$ – моменты времени, а X – значение объема продаж в эти периоды. Цель метода – получение значений функции X в моменты времени $T+1, T+2, \dots, T+n$.

Зная, что временной ряд может обладать повторением событий причин всплесков, мы можем найти интервалы, наиболее похожие на текущий, а затем предположить, что их характер обоснован той же комбинацией внешних факторов, что и сейчас. Таким образом можно спрогнозировать объем продаж на основе исторических данных.

Кроме того, так как ряд может обладать схожими некоторым образом всплесками в силу цикличности и сезонности, мы можем попросить менеджера по продажам указать, в какие временные интервалы спрос увеличивается именно в силу внешнего сезона.

Теперь, имея дополнительные параметры, в нашем случае можно оптимизировать работу последовательного метода. Поиск похожих отрезков можно ускорить при помощи кластеризации, производимой на основе введенных в систему показателей цикличности и сезонности.

Смысл процесса кластеризации состоит в том, что все исходные объекты, а в нашем случае объемы продаж, разбиваются на несколько не пересекающихся множеств таким образом, чтобы объекты в одной конкретной группе имели сходные характеристики, в то время как у объектов из разных групп эти характеристики должны значительно отличаться. Полученные группы и называются кластерами.

Таким образом на первом этапе алгоритма происходит поиск похожих закономерностей в предыдущих статистических наборах значений методом последовательного поиска. Затем выстраиваются аналогичные тренды, но в соответствии с текущими значениями. Если прогнозируемый интервал сезонный, имеет цикличность, то поиск идет в первую очередь в аналогичных интервалах и в соседних с ними и им отдается предпочтение.

В этом случае при совпадении отрезков этот интервал получит наибольший приоритет, а система избежит ошибок в прогнозировании.

Кроме того, в случае, если прогнозируемый интервал не имеет подобных себе в истории, то можно использовать статистические данные, чтобы выдать вероятностный прогноз в трех интервалах. Это даст наглядное представление пользователю о том, что скорее всего будет происходить в будущем с точки зрения статистики.

Интервалы вероятностей можно сформировать из точек, собранных за все аналогичные периоды, а затем разделить их на 3 кластера. Для этого метод k-средних также подходит. Это простой в реализации метод кластеризации, которого будет достаточно для решения данной задачи.

Для поиска похожих отрезков используется алгоритм динамической трансформации шкалы времени. С помощью него при сравнении отрезков как наборов точек строится таблица из значений расстояний между каждой парой точек. Затем по этой таблице находится наиболее короткий путь, который и означает расстояние между этими отрезками. Затем необходимо выбрать наиболее короткие из всех этих отрезков. Недостатки и достоинства этого алгоритма будут рассмотрены ниже. Наиболее важным фактором при выборе данного алгоритма является тот факт, что он не требует данных ни о каких внутренних закономерностях в данных. Именно поэтому он так хорошо подходит для временных рядов любого вида, особенно для нестационарных временных рядов.

Итоговая блок-схема программы изображена на рисунке 6.

Сначала идет статистический сбор данных, затем анализ в соответствии с введенными значениями параметров. Затем идет поиск реализованным и адаптированным методом поиска подпоследовательности во временном ряде, сравнение отрезков производится при помощи алгоритма динамической трансформации временной шкалы.

На следующем этапе проводится анализ результатов и вычисление прогноза, соответствующего ситуации в настоящем времени на основе найденных статистических результатов.

После чего алгоритм строит графики, выводит результаты на экран и завершает свое действие.

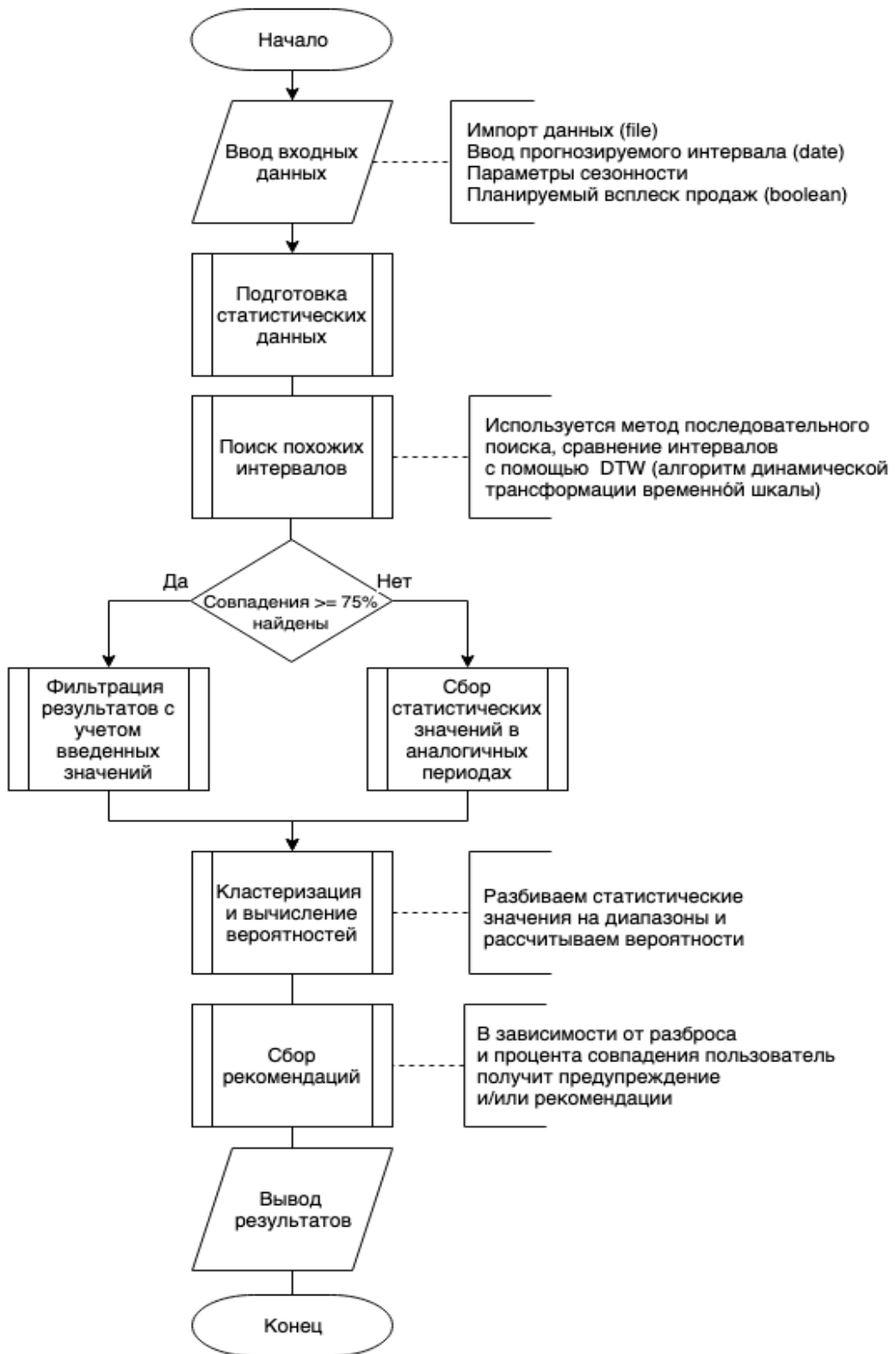


Рисунок 6 - Общая блок-схема алгоритма

2.2 Описание используемых алгоритмов

Для поиска схожих интервалов ряда в истории обратимся к алгоритму последовательного поиска, недавно реализованному крайне оптимальным с точки зрения производительности образом [25]. Его реализация находится в открытом доступе. Этот алгоритм заключается в переборе всех возможных подпоследовательностей. Пусть данные представляют собой временной ряд $T = t_1, t_2, \dots, t_M$. А запрос представлен временным рядом $Q = q_1, q_2, \dots, q_M$. Необходимо найти K подпоследовательностей $T_{ij} = t_i, \dots, t_j$ длины N , для которых минимально значение функции схожести $D(T_{ij}, Q)$.

В качестве метрики расстояния используется динамическая трансформация шкалы времени (DTW). Часто бывает так, что характерные части временных рядов не только отстоят друг от друга по оси времени, но ещё и растянуты или сжаты. Тогда ни евклидова метрика, ни кросс-корреляция не дают показательных результатов. Алгоритм DTW не имеет такого недостатка. Впервые он был применен в распознавании речи, где использован для определения того, как два речевых сигнала представляют одну и ту же исходную произнесённую фразу.

Мовчан А.А в статье «Нахождение похожих подпоследовательностей временного ряда с помощью многоядерного сопроцессора Intel Xeon Phi» наглядно описывает действие алгоритма DTW. «DTW-алгоритм строит матрицу возможных отображений одного ряда на другой, с учётом того, что отсчёты рядов могут как сдвигаться, так и изменять уровень. Путь в этой матрице с минимальным значением стоимости — и есть DTW-расстояние. Похожий принцип используется в алгоритме Вагнера-Фишера нахождения минимального редакционного расстояния между строками.

Классический DTW алгоритм по этим последовательностям строит путь наименьшей стоимости. Определим матрицу $\Omega_{n \times m}$ так, чтобы её элемент (i, j) соответствовал расстоянию между i -ым и j -ым элементами последовательностей Q и S , то есть соответствовал выравниванию между q_i и

c_j . Будем брать евклидово расстояние: $d(q_i, c_j) = (q_i - c_j)^2$. В качестве метрики можно взять и другие функции, например: $d(q_i, c_j) = |q_i - c_j|$.

По матрице Ω построим некоторый путь W . Этот путь выражает соответствие между Q и C . k -ый элемент W определяется как $w_k = (i, j)$. Далее под $d(w_k)$, где $w_k = (i, j)$, будем понимать $d(q_i, c_j)$, т. е. $d(w_k) = d(q_i, c_j) = (q_i - c_j)^2$.

Графически это можно изобразить следующим образом на рисунке 7 ниже:

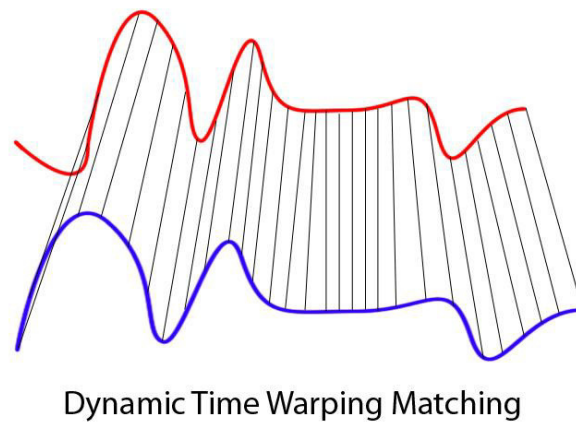
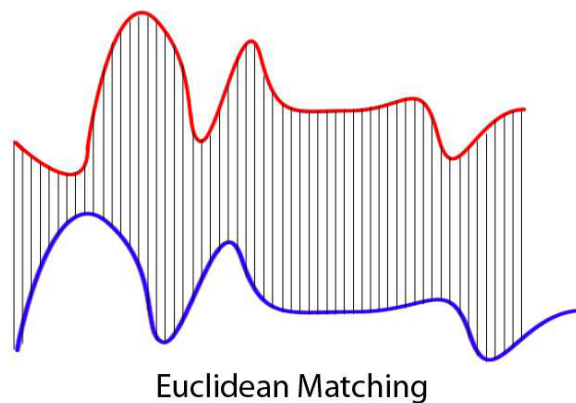


Рисунок 7 - Сравнение метрик расстояния DTW (снизу) и расстояния Евклида (сверху)

Итак, мы имеем $W = w_1, w_2, \dots, w_k, \dots, w_K$, где K — длина пути. K очевидно удовлетворяет следующему условию: $\min(m, n) \leq K < m + n - 1$. Пусть путь W удовлетворяет следующим условиям, перечисленным ниже.

Граничные условия. Обычно предполагают, что $w_1 = (1, 1)$ и $w_K = (n, m)$, т. е. начало и конец W находятся на диагонали в противоположных углах Ω .

Непрерывность. Пусть $w_k = (a, b)$ и $w_{k-1} = (p, q)$. Тогда $a - p \leq 1$, $b - q \leq 1$. Это ограничение нужно, чтобы в шаге пути W участвовали только соседние элементы матрицы (включая соседние по диагонали).

Монотонность. Пусть $w_k = (a, b)$ и $w_{k-1} = (p, q)$. Тогда $a - p > 1$, $b - q > 1$. Это ограничение нужно, чтобы точки W монотонно перемещались во времени. Путей, удовлетворяющих этим трем условиям, может быть очень много. Однако нам нужен путь, на котором достигается минимум стоимости пути:

$$DTW(Q, C) = \frac{\sum_{k=1}^K d(w_k)}{K} \rightarrow \min, \quad (4)$$

Знаменатель K нужен для того, чтобы учесть различную длину W . Выравнивание временных рядов: прогнозирование с использованием DTW».

Таким образом, путь наименьшей стоимости (выравнивающий путь) для последовательностей Q и C это путь W , на котором достигается минимум стоимости пути $DTW(Q, C)$.

Перед сравнением к временным рядам применяется модифицированная Z нормализация, которая более устойчива к аномалиям, чем обычная.

Z -нормализация вычисляется по формуле:

$$x'_i = \frac{x_i - \mu}{\sigma}, \quad \sigma = \frac{1}{N} \sum_{i=1}^N |x_i - \mu|, \quad (5)$$

где μ – среднее арифметическое; σ – среднее абсолютное отклонение.

Метод k -средних (или k -means) является одним из наиболее популярных методов кластеризации. Ему часто отдается предпочтение из-за его простоты реализации и большой скорости.

Кластер – группа объектов со схожими свойствами, которые сугубо отличаются от других групп. По сути, это итеративный алгоритм, который

делит данное множество точек значений на k кластеров, а затем точки, которые являются максимально приближенными к их центрам, а сама кластеризация происходит за счет смещения этих же центров.

Алгоритм принципа простейшей кластеризации методом k -средних:

- выбрать из множества k точек те точки, которые будут центрами соответствующих k кластеров. Начальные центры выбираются случайно или при помощи специальной метрики. Теперь начинается цикл процесса подбора центров кластеров. Он продолжается до тех пор, пока центры не перестанут изменяться.

- обходим каждую точку и определяем, к какому из центров эта точка находится ближе всего.

- если такой центроид найден, то теперь точка принадлежит к его кластеру.

- если все точки имеют центроид, высчитываем новые координаты центроидов k кластеров.

- проверяем координаты новых центроидов. Если они соответственно равны предыдущим центроидам — выходим из цикла, если нет возвращаемся к пункту два.

Основной принцип данного алгоритма заключается в стремлении минимизировать суммарное квадратичное отклонение точек кластеров от центров этих кластеров.

В случае использования иерархических алгоритмов возникает вопрос о том, как объединять группы, как рассчитывать «расстояния» между ними.

На этот случай существует несколько метрик:

a) одиночное соединение (расстояние от ближайшего соседа).

b) полное общение (расстояние от самых дальних соседей).

c) невзвешенное среднее по паре.

d) средневзвешенное пары.

В методе одиночного соединения расстояние между двумя группами определяется расстоянием между двумя ближайшими объектами

(ближайшими соседями) в разных группах. Получающиеся группы имеют тенденцию быть соединенными.

В методе полного общения между группами определяются наибольшим расстоянием между двумя объектами в разных группах (то есть, более отдаленные соседи). Этот метод обычно работает очень хорошо, когда объекты приходят из отдельных групп. Если группы вытянуты или их естественный тип «скован», то этот метод не подходит.

В методе невзвешенного среднего расстояние между двумя различными группами рассчитывается как среднее расстояние между всеми парами объектов в них. Этот метод эффективен, когда объекты образуют разные группы, но он работает так же хорошо в случае расширенных группировок (тип «цепочка»).

Метод средневзвешенное пары идентичен методу невзвешенного парного среднего, за исключением того, что размер соответствующих групп (то есть количество объектов, которые они содержат) используется в качестве весового коэффициента в расчетах. Поэтому этот метод следует использовать, когда предполагаются неравные размеры групп [14].

К проблемам алгоритма k-средних можно отнести то, что необходимо знать заранее количество группировок. В нашем случае их число равно трем, так как такой диапазон вероятностей целесообразен для прогнозирования. Кроме всего прочего, алгоритм очень чувствителен к выбору начальных кластерных центров. Классическая версия предполагает случайный выбор групп, что очень часто является источником ошибок. В качестве решения необходимо провести объектные исследования, чтобы более точно определить центры исходных групп. В нашей ситуации эта проблема решается диапазоном шкалы значений, а отклонение в одну-две точки можно считать несущественным.

2.3 Выбор программных средств реализации

Методологии, технологии и инструментальные средства проектирования (CASE - средства) составляют основу проекта любой ИС. Методология реализуется через конкретные технологии и поддерживает ее стандарты, методики и инструментальные средства, которые обеспечивают выполнение процессов жизненного цикла.

Р.П. Гахов в учебном пособии «Компьютерное моделирование экономических процессов» отмечает, что «процесс бизнес-моделирования может быть реализован в рамках различных методологий, отличающихся прежде всего своим подходом к тому, что представляет собой моделируемая организация. В соответствии с различными представлениями об организации методологии принято делить на объектные и функциональные (структурные).

Объектные методологии рассматривают моделируемую организацию как набор взаимодействующих объектов – производственных единиц. Объект определяется как осязаемая реальность – предмет или явление, имеющие четко определяемое поведение. Целью применения данной методики является выделение объектов, составляющих организацию, и распределение между ними ответственностей за выполняемые действия.

Функциональные методологии, наиболее известной из которых является методика IDEF, рассматривают организацию как набор функций, преобразующий поступающий поток информации в выходной поток. Процесс преобразования информации потребляет определенные ресурсы. Основное отличие от объектной методики заключается в четком отделении функций (методов обработки данных) от самих данных.

Гахов Р.П. добавляет там же, что «методология SADT (Structured Analysis and Design Technique - технология AllFusion Process Modeler 7 (ранее BPwin) – инструмент для моделирования, анализа, документирования и оптимизации бизнес-процессов. AllFusion Process Modeler 7 можно использовать для графического представления бизнес-процессов. Графически

представленная схема выполнения работ, обмена информацией, документооборота визуализирует модель бизнес-процесса. Графическое изложение этой информации позволяет перевести задачи управления организацией из области сложного ремесла в сферу инженерных технологий».

АвтомONOва А. Г. в книге «Методологии в проектировании информационных систем» пишет, что методология SADT (Structured Analysis and Design Technique - технология структурного анализа и проектирования) разработана Дугласом Т. Россом в 1969-1973 годах. Технология изначально создавалась для проектирования систем более общего назначения по сравнению с другими структурными методами, выросшими из проектирования программного обеспечения. SADT - одна из самых известных и широко используемых методик проектирования.

Новое название методики, принятое в качестве стандарта - IDEF0 (IcamDEFinition).

В IDEF0 система представляется как совокупность взаимодействующих работ (или функций). Связи между работами определяют технологический процесс или структуру взаимосвязи внутри организации. Модель SADT представляет собой серию диаграмм, разбивающих сложный объект на составные части.

Каждый блок IDEF0-диаграммы может быть представлен несколькими блоками, соединенными интерфейсными дугами, на диаграмме следующего уровня. Эти блоки представляют подфункции (подмодули) исходной функции. Каждый из подмодулей может быть декомпозирован аналогичным образом. Дуги могут разветвляться и соединяться. Ветвление означает множественность (идентичные копии одного объекта) или расщепление (различные части одного объекта). Соединение означает объединение или слияние объектов. Используются несколько видов стрелок.

Вход (Input) - материал или информация, которые используются работой для получения результата (стрелка, входящая в левую грань).

Управление (Control) - правила, стратегии, стандарты, которыми руководствуется работа (стрелка, входящая в верхнюю грань). В отличии от входной информации управление не подлежит изменению.

Выход (Output) - материал или информация, которые производятся работой (стрелка, исходящая из правой грани). Каждая работа должна иметь хотя бы одну стрелку выхода, т.к. работа без результата не имеет смысла и не должна моделироваться.

Механизм (Mechanism) - ресурсы, которые выполняют работу (персонал, станки, устройства - стрелка, входящая в нижнюю грань).

Вызов (Call) - стрелка, указывающая на другую модель работы (стрелка, исходящая из нижней грани)».

AllFusion ERwin Data Modeler (ранее ERwin) – CASE-средство для проектирования и документирования баз данных, которое позволяет создавать, документировать и сопровождать базы данных и хранилища. Модели данных помогают визуализировать структуру данных, обеспечивая эффективный процесс организации, управления и администрирования таких аспектов деятельности предприятия, как уровень сложности данных, технологий баз данных и среды развертывания [15].

Для проектирования диаграммы был выбран Microsoft Visio – векторный графический редактор, редактор диаграмм и блок-схем для Windows. Данный редактор предоставляет возможность создавать диаграммы различных схем, в том числе нотации UML.

Компания Microsoft в настоящий момент продвигает несколько языков программирования для быстрой и эффективной разработки для их операционной системы, наиболее полной справкой и эффективной производительностью итогового исполняемой программы, является язык C#. Для разработки на этом языке наиболее популярной средой интегрированной разработки является «Microsoft Visual Studio», в наиболее актуальных версиях этой среды имеется множество средств улучшения качества и скорости разработки приложений на языке C#, таких как удобный текстовый редактор

с авто дополнением используемых лексем (функция «IntelliSense»), инструменты для отладки кода, графический “конструктор” для создания оконных приложений и многое другое [16].

Выбранные средства позволяют выполнить необходимые этапы для проектирования автоматизированной системы поиска оптимальной торговой стратегии для рынка ценных бумаг, а также разработки необходимых модулей и графического интерфейса пользователя.

2.4 Проектирование информационной системы

Контекстная диаграмма показывает взаимодействие системы с окружающей среды. Стандарт IDEF0 представляет организацию как набор модулей, здесь существует правило — наиболее важная функция находится в верхнем левом углу, кроме того, есть правило стороны: стрелка входа всегда приходит в левую кромку активности, стрелка управления — в верхнюю кромку, стрелка механизма — нижняя кромка, стрелка выхода — правая кромка.

На рисунке 8 представлена контекстная диаграмма IDEF0, которая отражает общее описание информационной системы и её взаимодействие с внешней средой. Это контекстная диаграмма процесса прогнозирования объемов продаж «Как есть». Как видно, в данном случае входными данными являются только заказы покупателей на продажу. В качестве управляющего воздействия выступает менеджер по продажам, который и осуществляет прогноз простым методом умножения объема продаж на цену и суммой по всем клиентам.

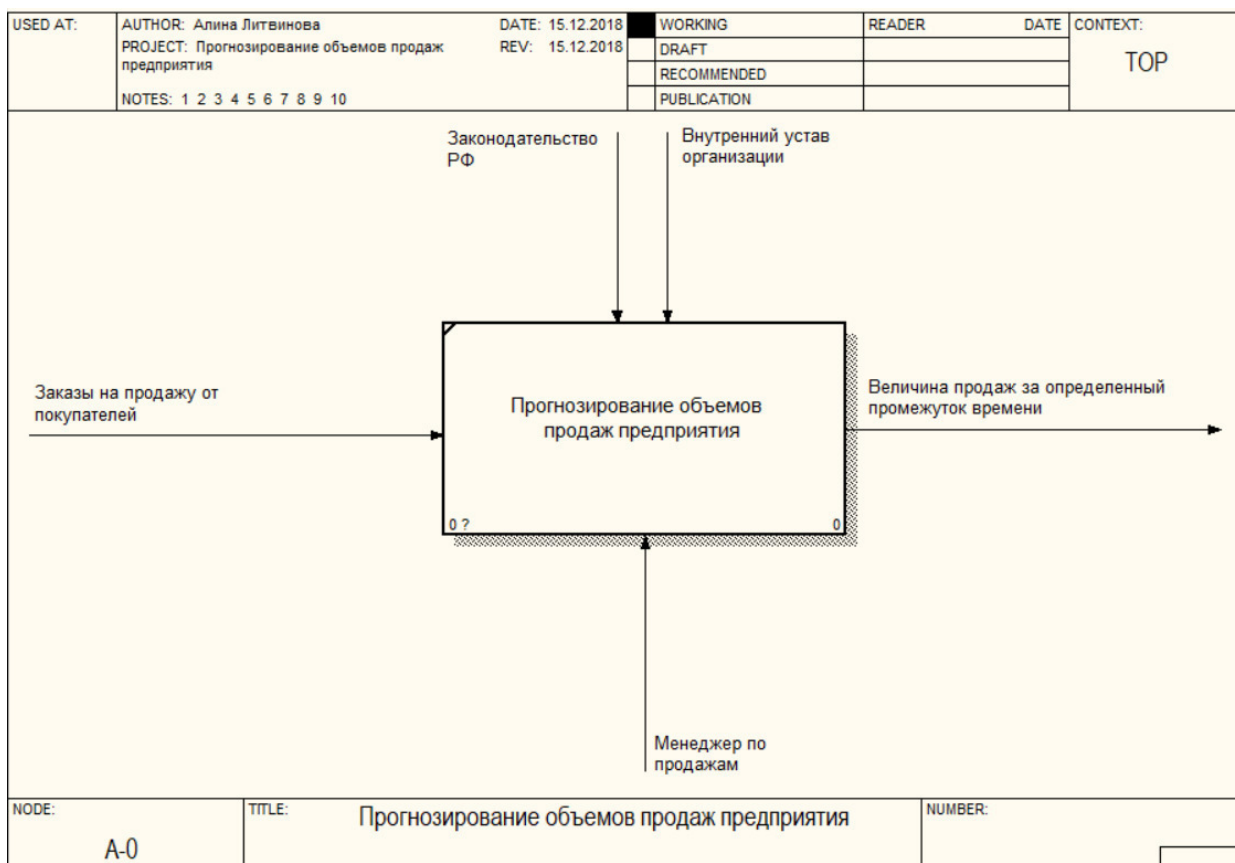


Рисунок 8 - Контекстная диаграмма процесса прогнозирования объемов продаж, модель «Как есть»

Модуль прогнозирования продаж здесь полностью выполняется менеджером по продажам и зависит от того, каким методом прогнозирует менеджер.

Такой процесс является малоэффективным. Хранящиеся в базе статистические данные никак не учитываются, хотя данные имеют цикличность и сезонность.

Добавление этих параметров в систему поможет повысить точность прогноза, увеличить эффективность планирования закупок.

Ниже на рисунке 9 представлена контекстная диаграмма процесса прогнозирования объемов продаж, модель «Как будет».

Входными данными являются данные о фактах продаж, о клиентах, а также параметры прогнозирования, например, период прогнозирования.

Пользователь будет иметь возможность настраивать отображаемые данные в удобном для себя формате.

Выходными данными являются результаты прогнозирования: спрогнозированная величина объема продаж и спрогнозированная величина объема продаж среднестатистическая по периоду.

В качестве управляющего воздействия выступает менеджер по продажам и сама система. К регламентирующей части относятся внутренние устав организации и законодательство Российской Федерации.

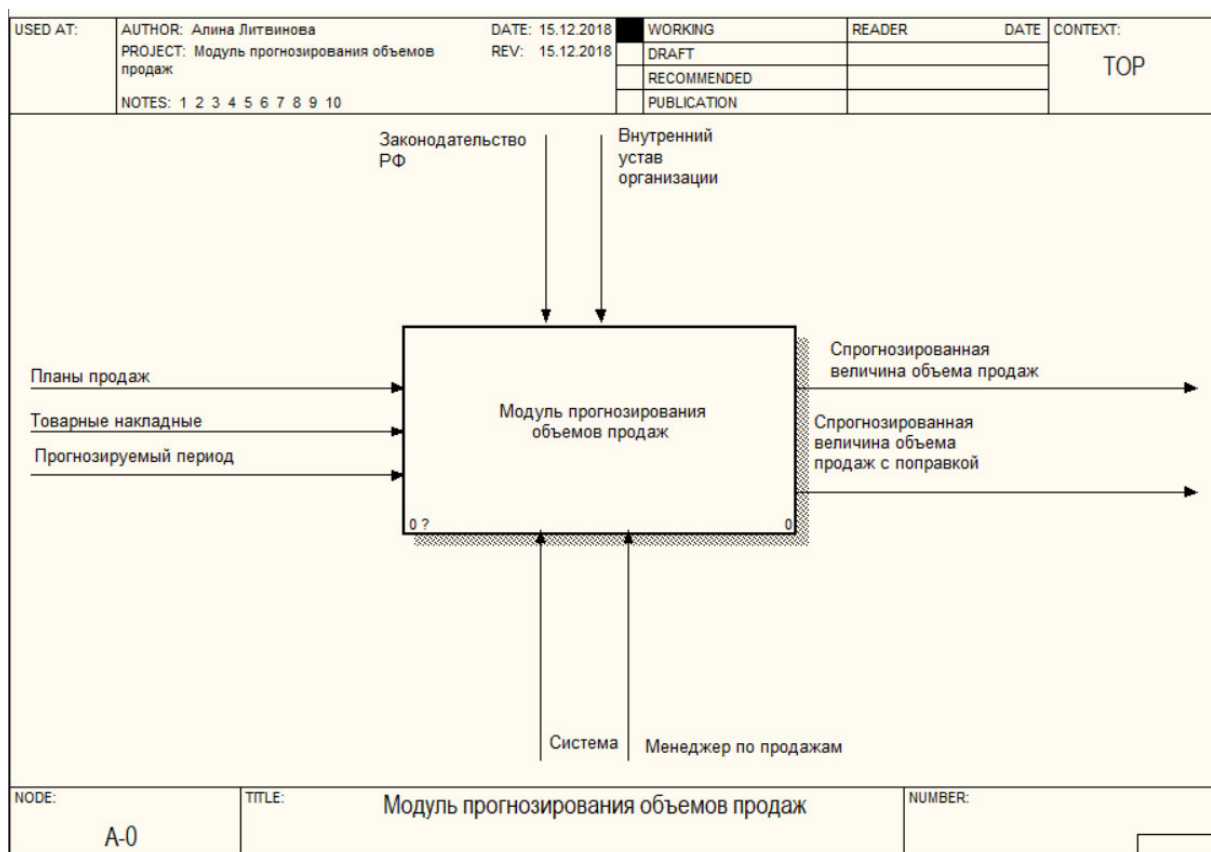


Рисунок 9 - Контекстная диаграмма процесса прогнозирования объемов продаж, модель «Как будет»

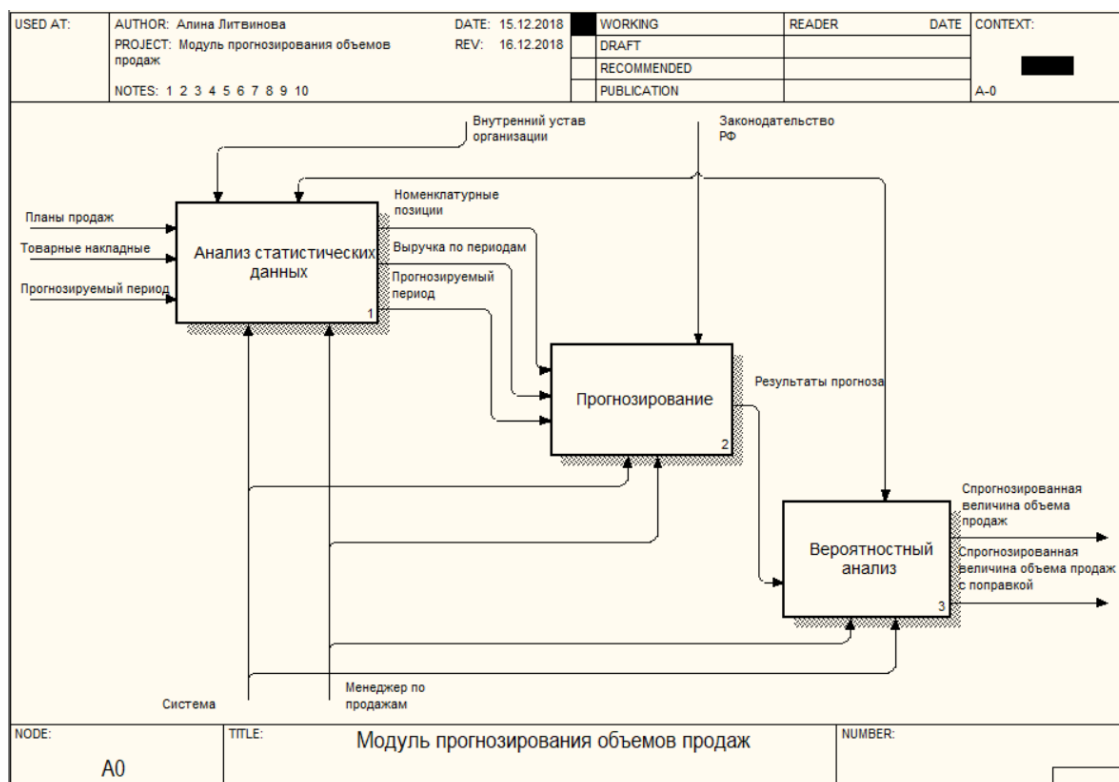


Рисунок 10 - Диаграмма декомпозиции первого уровня «Прогнозирование объемов продаж»

На рисунке 10 представлена диаграмма декомпозиции первого уровня для центрального блока контекстной диаграммы.

В качестве управляющего воздействия так же выступает менеджер по продажам и сама система. К регламентирующей правила части относятся внутренний устав организации и законодательство Российской Федерации.

Основной блок декомпозируется на три составляющих: настройку параметров, прогнозирование и вывод результатов на экран. Результатом прогнозирования является сформированный прогноз. Управляющим воздействием является менеджер по продажам, который определяет значения входящих параметров системы. Данные о клиентах и продажах поступают в виде файла Excel.

3 Описание разработанной автоматизированной системы прогнозирования продаж

3.1 Описание разработанного пользовательского интерфейса

Для ввода данных и отображений результатов было создано графическое приложение с помощью интерфейса программирования приложений Windows Form. Главное окно изображено на рисунке 11.

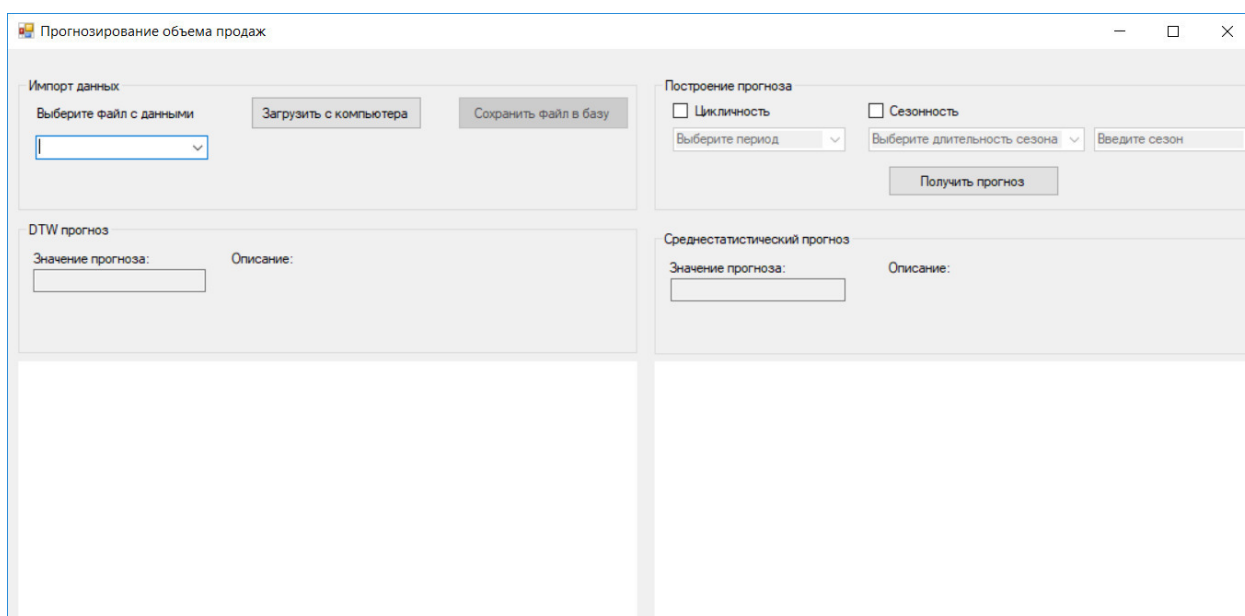


Рисунок 11 – Главное окно программы

Основное окно состоит из следующих частей: загрузки файла и сохранения его в историю, выпадающего списка с заранее сохраненными файлами, полосы меню, двумя разделами в соответствии с типами прогноза и описанием этого прогноза, раздел с параметрами прогноза, две панели с графиками, на которых изображены прогнозы с результатами после оптимизации с учетом параметров. Для удобства пользователя программа позволяет изменять размер графиков и таблиц.

Первый раздел «Импорт данных» представлен на рисунке 12.

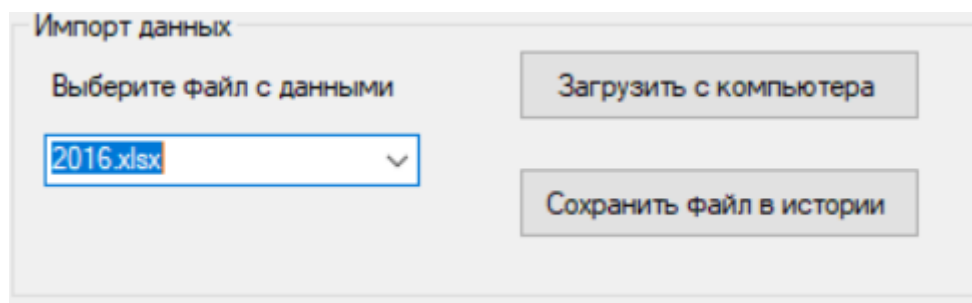


Рисунок 12 – Раздел «Импорт данных»

Чтобы добавить временной ряд для прогнозирования, пользователю необходимо выбрать его название из списка доступных загруженных ранее файлов, либо загрузить новый с компьютера. Это окно можно видеть на рисунке 13 ниже.

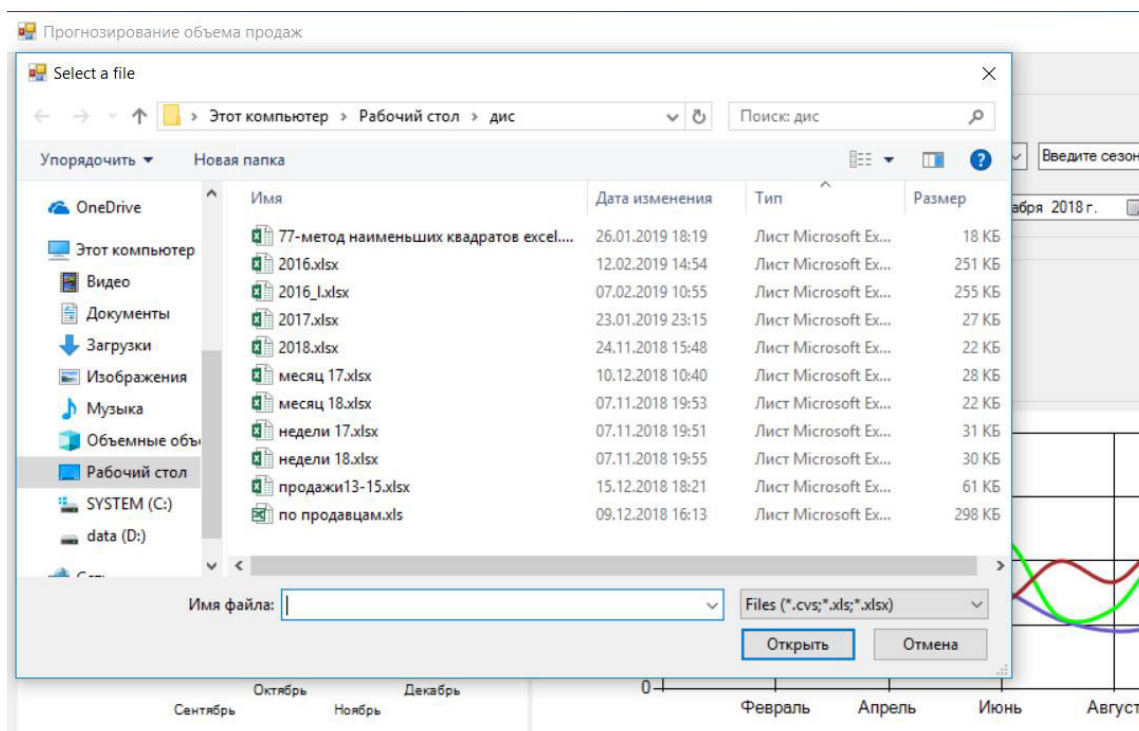


Рисунок 13 - Выбор документа на компьютере

Заранее установлены фильтры файлов, которые пользователь может загружать – это только файлы табличных форматов.

Пользователь должен выбрать файл, нажать кнопку «Открыть» и файл загрузится. После загрузки файла он может сохранить его в список быстро доступных файлов.

Оповещение об успешно загруженном файле представлено на рисунке 14.

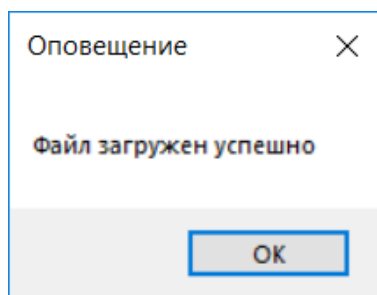


Рисунок 14 – Оповещение об успешно загруженном файле

Чтобы после загрузки файла его название отобразилось в списке, необходимо нажать кнопку «ОК» на окне оповещения.

Справа от данного раздела расположена секция параметров прогнозирования.

Для добавления параметров прогнозирования необходимо выбрать соответствующие значения из раздела «Параметры прогнозирования». После отметки в соответствующем элементе управления становится доступным для выбора соответствующий диапазон параметра.

На рисунке 15 представлена форма для настройки параметров прогнозирования объема продаж с учетом сезонности.

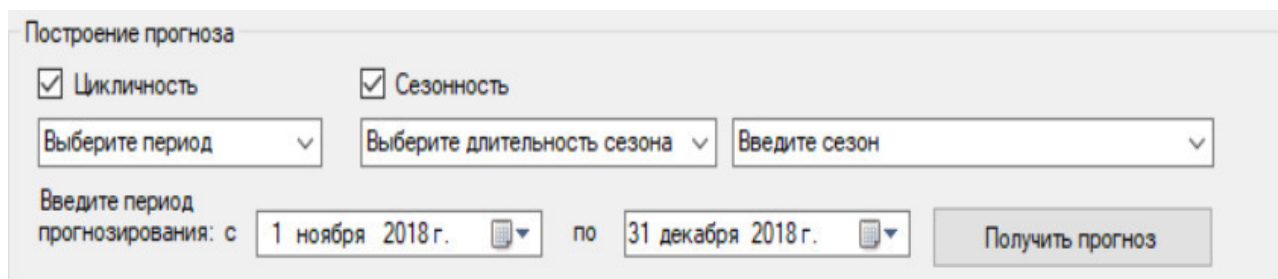


Рисунок 15 – Форма настроек параметров торговой стратегии

С помощью данной формы можно указать:

- цикличность, после чего станет доступным для выбора выпадающий список со значениями периодов циклов.

- сезонность, позволяющая системе более точно определить сезон продаж при помощи выбора типа периода (месяц, год, неделя), а также, если выбран месяц, выбора непосредственно месяцев сезонности.

При необходимости возможно изменить параметры, редактировать список инструментов в стратегии. Для изменения значений параметра необходимо изменить соответствующее значение в выпадающих списках и чек-листах.

Форма с измененными параметрами и раскрытым списком сезона изображена на рисунке 16.

Построение прогноза

Цикличность Сезонность

Год Месяц Введите сезон

Введите период прогнозирования: с 1 ноября 2018 г. по 31 декабря

Среднестатистический прогноз

Ноябрь: 33% - 1 924 667 руб., 66% - 6 000 000. руб
Декабрь: 33% - 1 924 667 руб., 66% - 6 000 000. руб

январь
февраль
март
апрель
май
июнь
июль
август
сентябрь
октябрь
ноябрь
декабрь

Рисунок 16 – Форма с открытым списком инструментов и измененными параметрами

На форме представлены все месяцы года с января по декабрь. Форма ввода позволяет выбрать несколько месяцев.

Форма с выпадающим списком значений цикличности изображена ниже на рисунке 17.

Рисунок 17 - Форма с выпадающим списком значений цикличности

Доступными значениями являются год, неделя и месяц.

Если все параметры указаны корректно или не указаны вовсе (в этом случае считается, что определение повторов отдано на усмотрение системы), выбран ряд для прогнозирования, то при нажатии кнопки «Получить прогноз» система начнет процесс поиска похожих подпоследовательностей в указанном временно ряде.

Форма с выпадающим списком значений сезонности изображена ниже на рисунке 18.

Рисунок 18 - Форма с выпадающим списком значений сезонности

После нажатия кнопки «Получить прогноз», в случае корректно добавленных параметров или не добавленных вовсе, успешной загрузки ряда данных, система начинает прогнозирование методом поиска похожей подпоследовательности временного ряда.

Ниже на рисунке 19 представлена форма с полученными результатами прогноза.

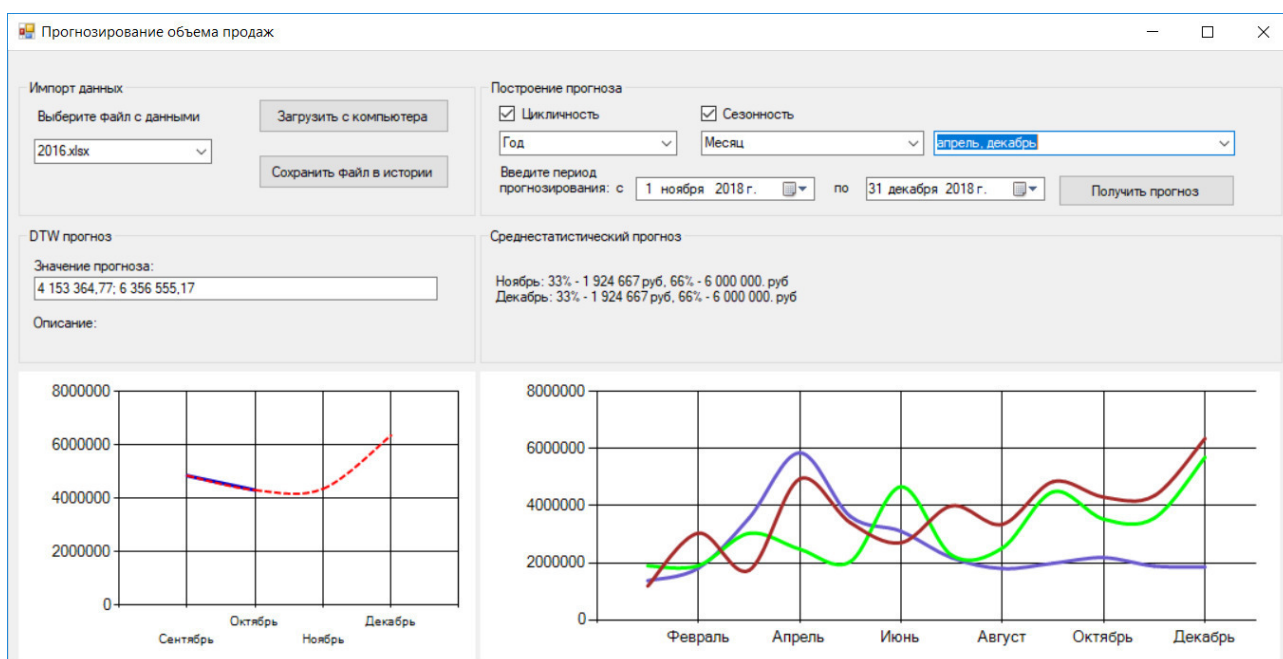


Рисунок 19 - Форма с полученными результатами прогноза

Результат прогноза, основанного на методе динамической трансформации временной шкалы, отображается в соответствующей секции. Данная секция представлена на рисунке 20.

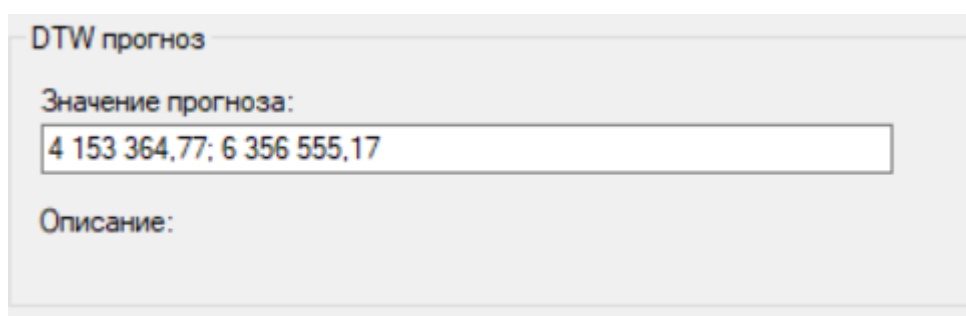


Рисунок 20 – Секция с результатами DTW прогноз

Власов Д.В., Божко В.П. в книге «Информационные технологии в экономике и управлении» указывают: «для настроек параметров оптимизации необходимо выбрать меню «Настройки» и «Настройки оптимизации». После

чего появится возможность выбора количества особей в популяции, вероятность применения оператора селекции, вероятность применения оператора мутации и вероятность применения оператора инверсии.

Секция содержит поле вывода, где отображаются прогнозные значения.

На рисунке 21 представлена секция «Среднестатистический прогноз».



Рисунок 21 – секция «Среднестатистический прогноз»

Эта секция отображает результаты прогноза, сделанного в том случае, если похожий отрезок адаптированным методом поиска подпоследовательности во временном ряде не был найден.

Примеры графика прогноза методом адаптированным методом поиска подпоследовательности во временном ряде при помощи алгоритма динамической трансформации временной шкалы можно видеть ниже на рисунке 22.

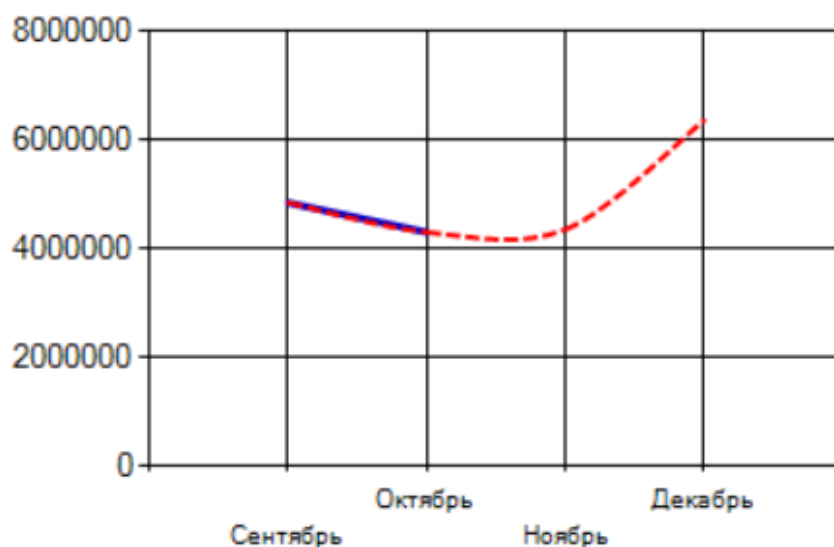


Рисунок 22 – График с результатами оптимизации

На данном графике отражен прогноз красной прерывистой линией, а синей отображен базовый отрезок, по которому ищутся похожие периоды в статистических данных о прогнозах продаж предприятия.

Ниже на рисунке 23 отображена общая статистика за прогнозируемые интервалы времени.

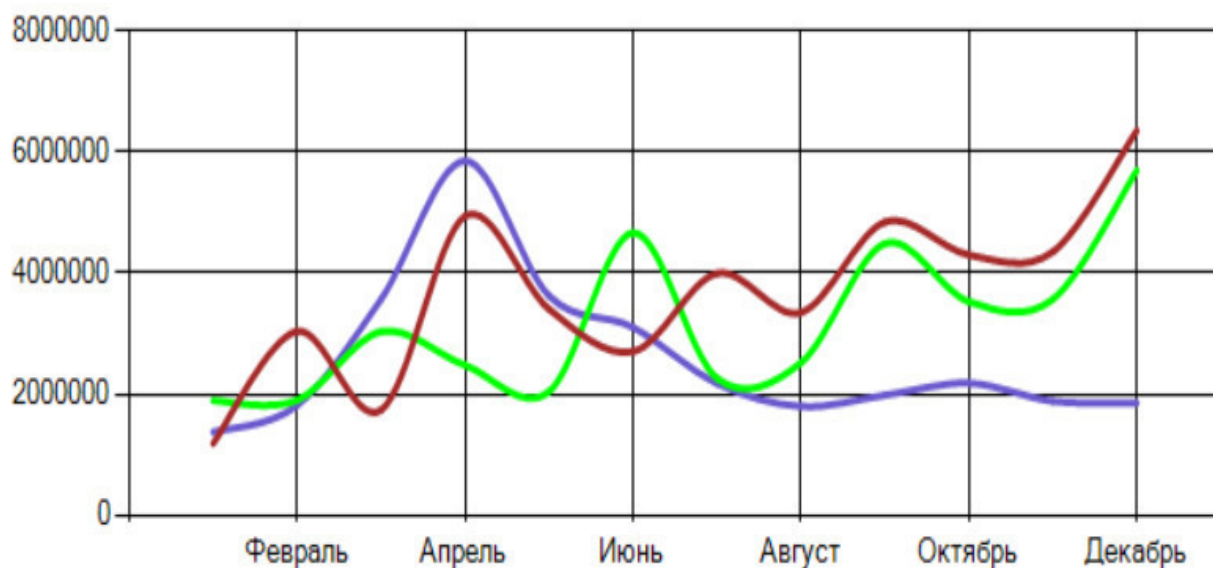


Рисунок 23 - Общая статистика за прогнозируемые интервалы времени

В данном случае представлены графики за 3 года: 2015, 2016 и 2017. На этой основе сделан прогноз на 2018 год, в частности на ноябрь и декабрь. По вертикальной оси отмечена сумма объема продаж в рублях, а по горизонтальной – месяцы, в которые производились продажи. Так как пользователь заранее отметил цикличность в год, то графики расположены таким образом – один над другим, каждое начало одного года соответствует началу другого и так далее.

На вкладке «Популяции», в таблице представлены результаты тестирования стратегий из указанной популяции. В столбце «Стратегия» указывается название стратегии, в столбце результат – мера приспособленности индивида и в столбце «Параметры» – полный список инструментов и параметров, которые используются в стратегии.

При нажатии кнопки «Открыть график» в верхнюю область добавляется график, визуализирующий ордера на исторических данных используемого торгового инструмента».

На рисунке 19 изображена информация о торговых стратегиях из первой популяции и представлен график сделок одной из стратегий.

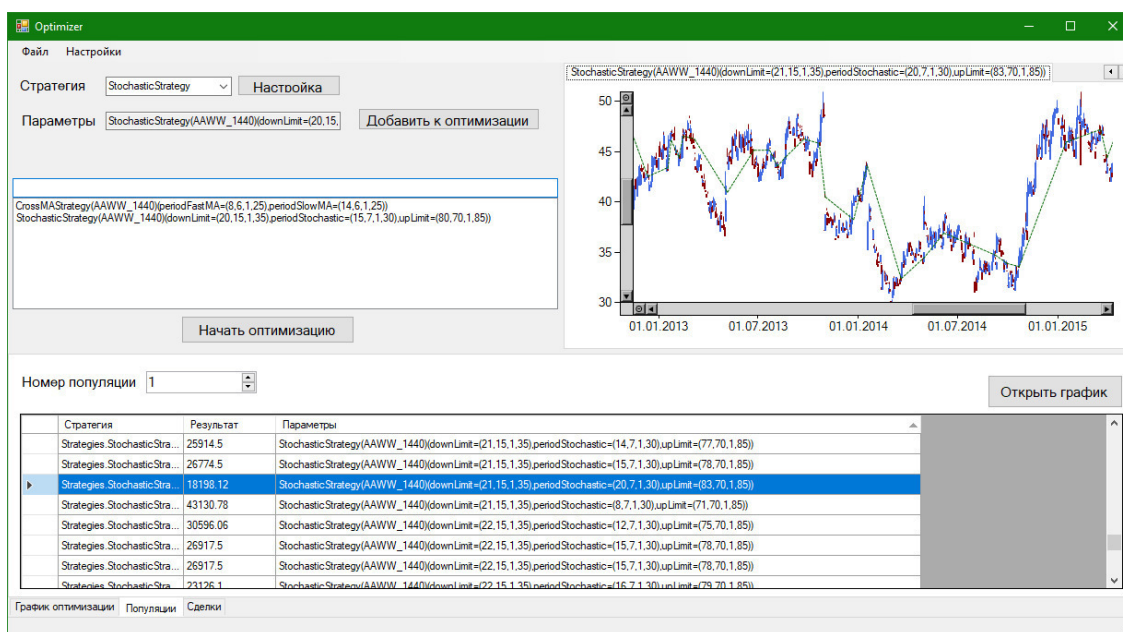


Рисунок 19 – Информация о стратегиях в первой популяции

В третьей вкладке под названием «Сделки» отображены все сделки после тестирования выбранной стратегии на вкладке «Популяции». В таблице отображены вся информация о каждой сделке, такие как: тикет (номер сделки), символ (название инструмента), тип (тип сделки, один из Buy, Sell, BuyLimit, SellLimit, BuyStop, SellStop), время открытия и закрытия, цена открытия и закрытия, уровень stoploss и TakeProfit, а также комментарий при открытии сделки и причина закрытия.

На рисунке 20 представлено общее окно программного кода, со сделками выбранной стратегии и их изображение на ценовом графике.

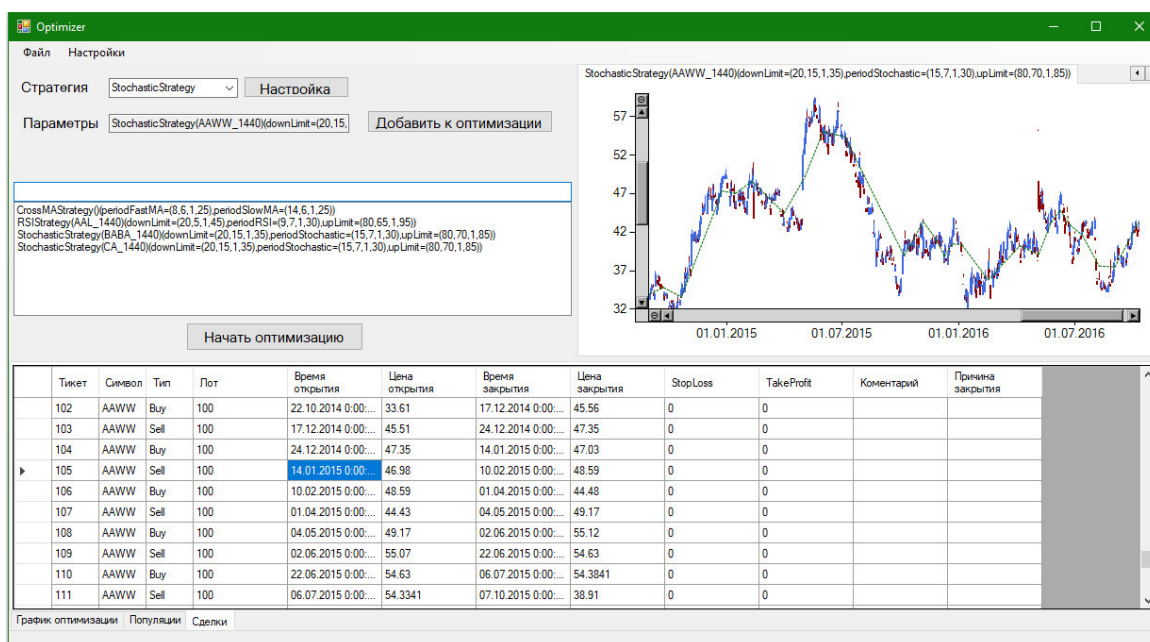


Рисунок 20 – Сделки стратегии и соответствующий график

Для отображения биржевых данных используется график с японскими свечами, «бычьи» свечи (цены закрытия в которых выше цен открытия) обозначены синим цветом, а «медвежьи» (цены открытия которых выше цен закрытия) обозначены красным цветом. Для графика цен имеется возможность изменять масштаб, изменять в окне отображаемый временной интервал.

Чтобы изменить масштаб необходимо выделить интересующую область графика, как представлено на рисунке 21.

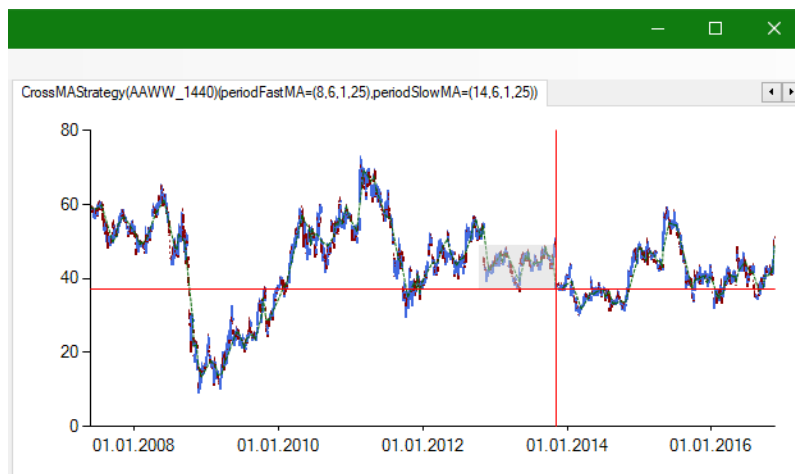


Рисунок 21 – Изменение масштаба

После этого масштаб будет изменен и появятся ленты прокручивания, для изменения отображаемого временного интервала.

На рисунке 22 изображен график с измененным масштабом.

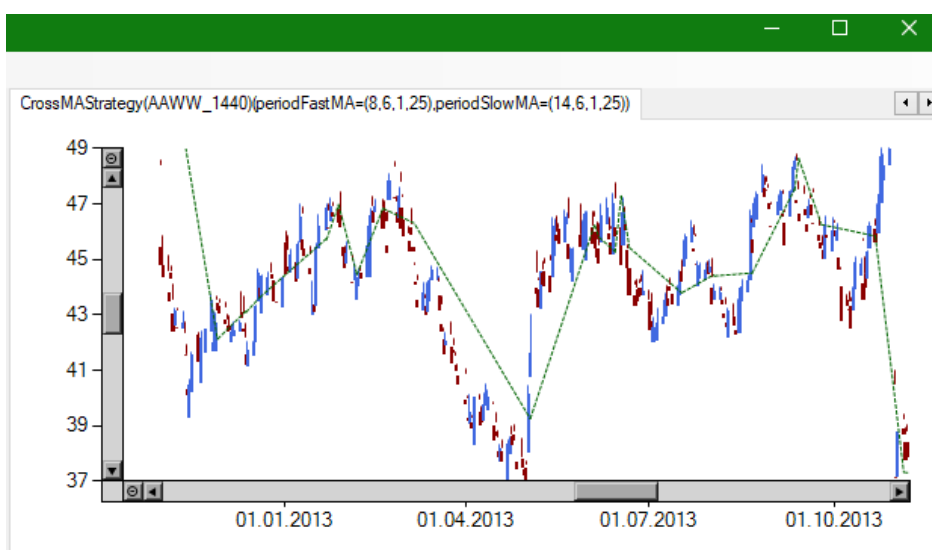


Рисунок 22 – График с измененным масштабом

Для возвращения графика в исходное состояние, необходимо нажать на кнопки рядом с полосами прокрутки, что приведет к изменению масштаба на исходное по соответствующим осям графика. На рисунке 23 стрелками показано расположение кнопок изменения масштаба.



Рисунок 23 – Расположение кнопок изменения масштаба на графике

Таким образом был реализован интерфейс программы, для ввода данных, изменения параметров стратегий и параметров оптимизации, код программы работы основных классов представлен в приложении.

Для выполнения тестирования контрольного примера использовались стратегии CrossMAStrategy и StochasticStrategy, в которых использовался инструмент AAWW (котировки акций компании «Atlas Air Worldwide Holdings») и CA (котировки акций компании «CA Technologies») сделки по которым выполняются в NASDAQ — американской бирже, которая специализируется на акциях высокотехнологичных компаний.

На рисунке 24 представлены параметры стратегий для оптимизации, а также настройки параметров генетического алгоритма.

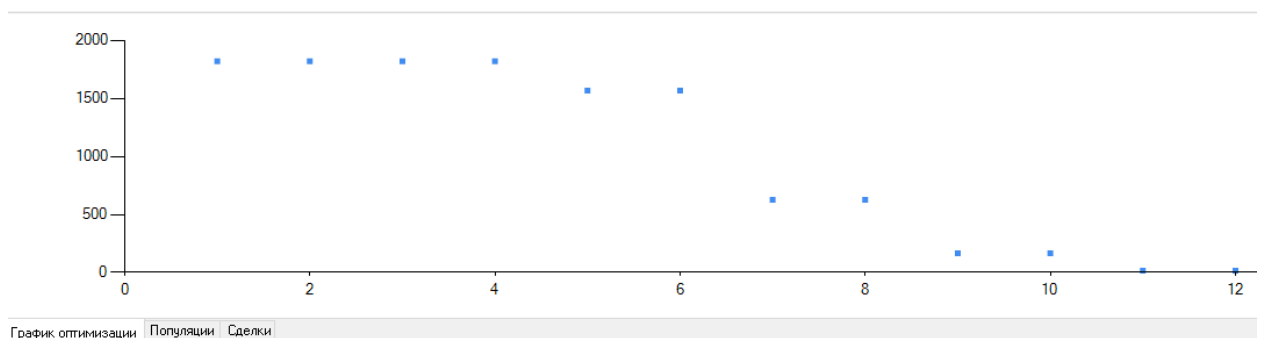


Рисунок 26 – График оптимизации

Из 400 стратегий в последней популяции лучшими характеристиками обладают следующие стратегии, параметры которых представлены на рисунке 27.

Стратегия	Результат	Параметры
Strategies.Stoch...	47029.00	StochasticStrategy(AAWW_1440)(downLimit=(32,15,1,35),periodStochastic=(11,7,1,30),upLimit=(74,70,1,85))
Strategies.Stoch...	43904.00	StochasticStrategy(AAWW_1440)(downLimit=(15,15,1,35),periodStochastic=(21,7,1,30),upLimit=(70,70,1,85))
Strategies.Stoch...	24338.00	StochasticStrategy(CA_1440)(downLimit=(15,15,1,35),periodStochastic=(7,7,1,30),upLimit=(70,70,1,85))
Strategies.Stoch...	23934.00	StochasticStrategy(CA_1440)(downLimit=(20,15,1,35),periodStochastic=(6,7,1,30),upLimit=(80,70,1,85))
Strategies.Stoch...	23908.00	StochasticStrategy(CA_1440)(downLimit=(16,15,1,35),periodStochastic=(7,7,1,30),upLimit=(71,70,1,85))
Strategies.Stoch...	23901.00	StochasticStrategy(CA_1440)(downLimit=(20,15,1,35),periodStochastic=(7,7,1,30),upLimit=(72,70,1,85))
Strategies.Stoch...	23858.00	StochasticStrategy(CA_1440)(downLimit=(20,15,1,35),periodStochastic=(8,7,1,30),upLimit=(74,70,1,85))
Strategies.Stoch...	23338.00	StochasticStrategy(CA_1440)(downLimit=(20,15,1,35),periodStochastic=(8,7,1,30),upLimit=(75,70,1,85))
Strategies.Stoch...	23338.00	StochasticStrategy(CA_1440)(downLimit=(20,15,1,35),periodStochastic=(8,7,1,30),upLimit=(80,70,1,85))
Strategies.Stoch...	23338.00	StochasticStrategy(CA_1440)(downLimit=(20,15,1,35),periodStochastic=(7,7,1,30),upLimit=(80,70,1,85))

Рисунок 27 – Лучшие стратегии последней популяции

Наибольшей эффективности по выбранному критерию получила стратегия StochasticStrategy с использованием символа AAWW и следующими значениями параметров downLimit=32, periodStochastic=11, upLimit=74.

Исходя из полученных на рисунке 13 результатов можно получить информацию о сделках лучшей стратегии. Для исследуемого варианта было получены данные о 188 сделках, часть из которых представлены ниже. На рисунке 28 представлена информация о первых десяти сделках лучшей стратегии.

Тикет	Символ	Тип	Лот	Время открытия	Цена открытия	Время закрытия	Цена закрытия
1	AAWW	Sell	100	16.05.2007 0:00:...	58.86	22.05.2007 0:00:...	58.76
2	AAWW	Buy	100	22.05.2007 0:00:...	58.76	25.05.2007 0:00:...	58.41
3	AAWW	Sell	100	25.05.2007 0:00:...	58.36	18.06.2007 0:00:...	57.92
4	AAWW	Buy	100	18.06.2007 0:00:...	57.92	24.07.2007 0:00:...	59.62
5	AAWW	Sell	100	24.07.2007 0:00:...	59.57	06.09.2007 0:00:...	50.63
6	AAWW	Buy	100	06.09.2007 0:00:...	50.63	01.10.2007 0:00:...	52.7976
7	AAWW	Sell	100	01.10.2007 0:00:...	52.7476	09.10.2007 0:00:...	53.82
8	AAWW	Buy	100	09.10.2007 0:00:...	53.82	07.11.2007 0:00:...	57.64
9	AAWW	Sell	100	07.11.2007 0:00:...	57.59	07.12.2007 0:00:...	53.1
10	AAWW	Buy	100	07.12.2007 0:00:...	53.1	17.12.2007 0:00:...	53.41
11	AAWW	Sell	100	17.12.2007 0:00:...	53.36	21.12.2007 0:00:...	53.3
12	AAWW	Buy	100	21.12.2007 0:00:...	53.3	03.01.2008 0:00:...	53.89

Рисунок 28 – Информация о первых десяти сделках

На рисунке 29 представлена информация о последних сделках лучшей стратегии.

Тикет	Символ	Тип	Лот	Время открытия	Цена открытия	Время закрытия	Цена закрытия
177	AAWW	Sell	100	16.06.2016 0:00:...	44.48	11.07.2016 0:00:...	40.37
178	AAWW	Buy	100	11.07.2016 0:00:...	40.37	26.07.2016 0:00:...	41.92
179	AAWW	Sell	100	26.07.2016 0:00:...	41.87	01.08.2016 0:00:...	42.33
180	AAWW	Buy	100	01.08.2016 0:00:...	42.33	03.08.2016 0:00:...	43.77
181	AAWW	Sell	100	03.08.2016 0:00:...	43.72	24.08.2016 0:00:...	35.02
182	AAWW	Buy	100	24.08.2016 0:00:...	35.02	15.09.2016 0:00:...	38.01
183	AAWW	Sell	100	15.09.2016 0:00:...	37.98	23.09.2016 0:00:...	39.86
184	AAWW	Buy	100	23.09.2016 0:00:...	39.86	13.10.2016 0:00:...	42.84
185	AAWW	Sell	100	13.10.2016 0:00:...	42.79	25.10.2016 0:00:...	42.6
186	AAWW	Buy	100	25.10.2016 0:00:...	42.6	03.11.2016 0:00:...	41.85
187	AAWW	Sell	100	03.11.2016 0:00:...	41.8	07.11.2016 0:00:...	42.65
188	AAWW	Buy	100	07.11.2016 0:00:...	42.65	15.11.2016 0:00:...	48.95

График оптимизации | Популяции | **Сделки**

Рисунок 29 – Последняя десятка стратегии

Чтобы посчитать число возможных вариантов наборов значений параметров при полном переборе, необходимо вычесть из максимального значение параметра минимальное и разделить на величину шага, после этого необходимо найти произведение полученных величин.

Выводы по третьему разделу: в данном разделе описан разработанный пользовательский интерфейс автоматизированной системы поиска оптимальной торговой стратегии для рынка ценных бумаг, а также описан контрольный пример реализации проекта.

Для ввода данных и отображений результатов было создано графическое приложение с помощью интерфейса программирования приложений Windows Form. Главное окно изображено на рисунке 30.

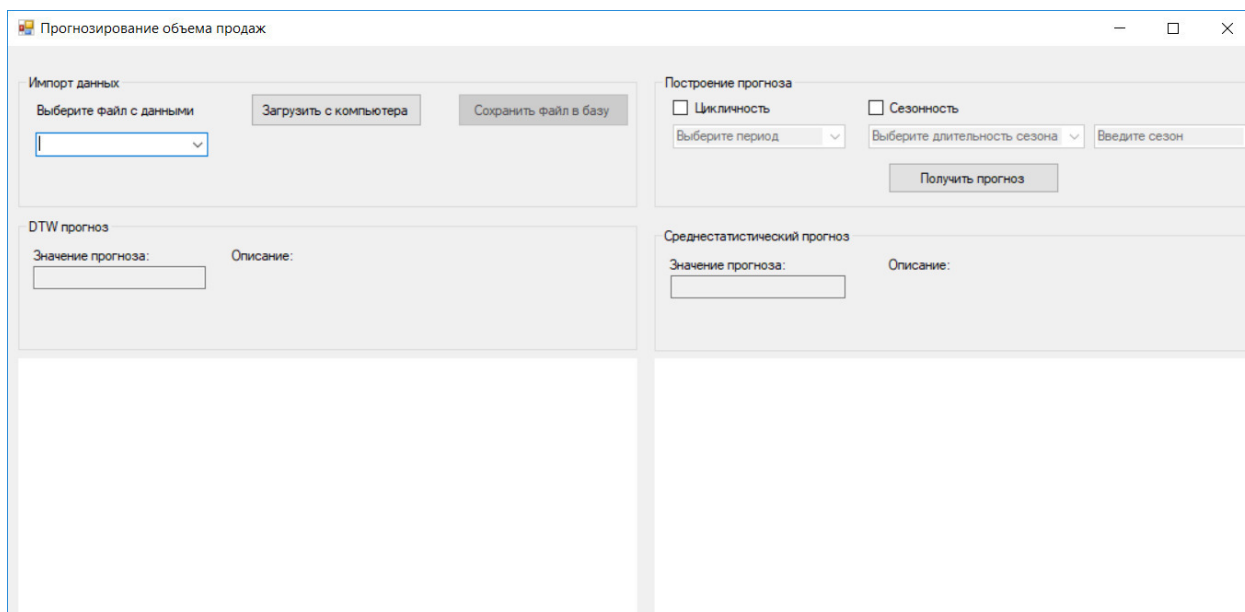


Рисунок 30 – Главное окно программы со стороны админа

Основное окно состоит из следующих частей: полосы меню, панель с настройками интерфейса, панель с графиком, на котором изображены сделки выбранной стратегии, панель с результатами оптимизации. Для удобства пользователя программой элементы находятся в контейнере SplitContainer, что позволяет изменять размер графиков и таблиц.

3.2 Оценка эффективности системы

Для оценки эффективности данного метода был проведен эксперимент, в рамках которого производилось сравнение метода последовательного перебора на основе динамической трансформации шкалы времени с методом экспоненциального сглаживания, методом построения линии тренда, методом авторегрессионного интегрированного скользящего среднего ARIMA.

Для эксперимента использовались данные, полученные на протяжении трех лет с 2015 по 2017 года. Прогноз рассматриваемых методов проводился в двух временных шкалах – прогноз на предстоящие 3 месяца и прогноз на предстоящие года.

Результаты эксперимента приведем в таблице 2.

Таблица 2 - Результаты эксперимента

Используемый метод	Критерий оценки	
	ME, %	MAPE, %
Адаптированный метод поиска похожих подпоследовательностей на основе динамической трансформации шкалы времени	7,79	3,52
Метод, используемый на предприятии	28,38	14,17
Построение полиномиальной линии тренда	13,31	3,99
Линейная регрессия с учетом сезонности	18,43	6,72

Для наглядности полученных результатов при проведении эксперимента построим график сравнения основных статистических методов, позволяющих спрогнозировать объем продаж с учетом не только внутренних факторов, но и внешних, количественных и даже качественных.

График сравнения основных статистических методов позволяющих спрогнозировать объем продаж с учетом не только внутренних факторов, но и внешних, количественных и даже качественных на три месяца вперед представлен на рисунке 33.

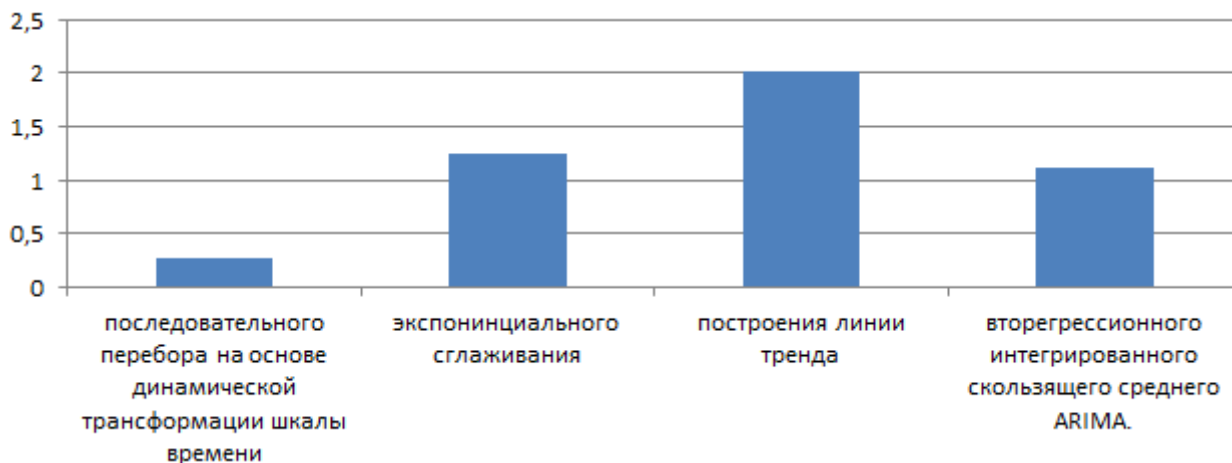


Рисунок 33 - График сравнения основных статистических методов позволяющих спрогнозировать объем продаж с учетом не только внутренних факторов за три года

График сравнения основных статистических методов позволяющих спрогнозировать объем продаж с учетом не только внутренних факторов, но и внешних, количественных и даже качественных на один год вперед представлен на рисунке 34.



Рисунок 34 - График сравнения основных статистических методов позволяющих спрогнозировать объем продаж с учетом не только внутренних факторов за год вперед

Анализируя полученные результаты, можно сделать вывод о том метод последовательного перебора на основе динамической трансформации шкалы времени дает более высокую точность прогноза по сравнению с методом экспоненциального сглаживания, методом построения линии тренда, методом авторегрессионного интегрированного скользящего среднего ARIMA, так как при прогнозе на разные промежутки времени имеет наименьшее среднее отклонение.

ЗАКЛЮЧЕНИЕ

В большинстве случаев профессиональные математические инструменты недоступны для компаний малого бизнеса и не соответствуют затратам, если прогнозирование необходимо для одного временного ряда. Разработанная информационная система и адаптированный метод прогнозирования с использованием алгоритма трансформации временной шкалы реализует простой функционал, необходимый для автоматизации деятельности менеджера по продажам при прогнозировании объема продаж предприятия с учетом сезонности.

В процессе выполнения магистерской диссертации в первом разделе было проведено общее исследование процесса прогнозирования объема продаж и описаны принципы оценки эффективности методов прогнозирования. Проведён анализ существующих аналогичных информационных систем и выявлены их достоинства и недостатки. Обоснована необходимость в разработке нового метода прогнозирования. Проведена общая характеристика организации решения задачи.

Во втором разделе описаны виды исследуемых методов прогнозирования. Были рассмотрены основные этапы и понятия классического DTW алгоритма. На основании проведенного анализа была разработан алгоритм прогнозирования объема продаж средствами генетического алгоритма для сокращения количества исследуемых вариантов. Для реализации разработанной модели было принято решение о необходимости разработки новой информационной системы с обоснованием требований, выбором архитектуры приложения и средства реализации. Выполнено проектирование автоматизированной системы с применением методологии IDEF, с разработкой контекстной диаграммы и выполнением декомпозиции.

В третьем разделе представлено описание графического интерфейса пользователя разработанной информационной системы с подробным

описание функционала используемых элементов. Программное обеспечение позволяет строить графики прогноза, отображать найденные похожие тенденции в прошлом.

Таким образом, все поставленные задачи выполнены. По результатам исследования и проектирования была реализована автоматизированная система прогнозирования объемов продаж с учетом сезонности. Для разработки системы использовалась современная среда разработки Microsoft Visual Studio 2017, что позволило быстро создать пользовательский интерфейс, вести удобную и качественную разработку, без применения дополнительных решений.

Использование разработанной системы позволит снизить временные затраты на прогнозирование, автоматизировать рутинные действия пользователя.

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. Бокс, Д., Дженкинс Г. Анализ временных рядов, прогноз и управление / Д.Бокс, Г.Дженкинс.– Москва: Мир, 2012. 406 с.
2. Бородачев,С. М. Прогнозирование и управление/ С.М. Бородачев. – Москва: Флинта, 2017. – 125 с.
3. Вдовин, В.М. Информационные технологии: Практикум / В.М. Вдовин, Л.Е Суркова - Москва: Дашков и К°, 2012. – 248 с.
4. Власов Д.В., Божко В.П. Информационные технологии / Д.В.Власов, В.П.Божко - Москва: Юна, 2013. - 167 с.
5. Воеводин В.В. Вычислительная математика и структура алгоритмов./ В.В. Воеводин. - Москва: Фрегат, 2010. – 157 с.
6. Гусятников, В.Н. Стандартизация и разработка программных систем: Учебное пособие: [Электронный ресурс]/ В.Н. Гусятников, А.И. Безруков. - Москва : Финансы и статистика : Инфра-М, 2014. – 288 с.
7. Душин, В.К. Теоритические основы информационных процессов и систем / В.К Душин - Москва: Дашков и К, 2010
8. Егошин А.В. Анализ и прогнозирование сложных стохастических сигналов на основе методов ведения границ реализаций динамических систем: Автореферат диссертации .канд. техн. наук. Санкт-Петербург, 2009. 19 с.
9. Иглин С.П. Математические расчеты на базе Matlab /С.П. Иглин.- СПб.: ВHV-Санкт- Петербург, 2005. - 640 с.
10. Коберн А. Быстрая разработка программного обеспечения / А. Коберн. – Москва: Лори, 2013 – 336с.
11. Линев А.В.: Технологии параллельного программирования для процессоров новых архитектур / А.В.Линев - Москва: Московский университет, 2010 -211с.
12. Мезенцев, К.Н. Автоматизированные информационные системы / К.Н. Мезенцев. – Москва: Академия, 2012. – 174 с.

13. Мерков А.Б. Распознавание образов: Введение в методы статистического обучения / А.Б. Меркова.- Москва:Едиториал УРСС, 2011. 254 с.
14. Муратов, Л.И. Нормативные системы в прогнозировании развития предпринимательского сектора экономики [электронный ресурс]. / Л.И. Муратова [и др.] // Управление экономическими системами. Москва: [б.и.], 2014. Режим доступа:
<http://uecs.mcnip.ru/modules.php?name=News&file=print&sid=145>.
15. Нейгел, К. С# 2012 и платформа NET для профессионалов / К. Нейгел, Б. Ивсен Б. и др. - Москва: Диалектика, 2012.-651с.
16. Тихонов Э.Е. Прогнозирование в условиях рынка. / Э.Е. Тихонов.- Невинномысск:Печать - 2006. - 221 с.
17. Троелсен, Э. Язык программирования С# 6.0 и платформа .NET 4.6 / Э. Троелсен. – Москва:Вильямс, 2016. – 1440 с.
18. Уоррен, Г. С. Алгоритмические трюки для программистов / Г. С. Уоррен. – Москва:Вильямс, 2014. – 512 с.
19. Фаулер, М. Архитектура корпоративных программных приложений / М. Фаулер, Д. Райс, М. Фоммел, Э. Хайет, Р. Ми, Р. Стаффорд – М.: Вильямс, 2007. – 544 с.
20. Чернецов, С.А. Методы прогнозирования оптимальных доз инсулина для больных сахарным диабетом I типа. Обзор [электронный ресурс] / С.А. Чернецов [и др.] // Москва: [б.и.], 2014. Режим доступа:
<http://technomag.edu.ru/doc/119663.html> (дата обращения 28.08.2011).
21. Арлазаров, В.Л. Теория и методы создания интеллектуальных компьютерных систем/ В.Л. Арлазаров, Ю.И. Журавлев. – Москва: Издание, 2011. – 189с.
22. Герчикова, И.Н. Процесс принятия и реализации управленческих решений/ И.Н. Герчикова. – Москва: Юрайт, 2013. – 130 с.

ПРИЛОЖЕНИЕ А
Исходный код

```
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Windows.Forms;

namespace WindowsFormsApp1
{
    public partial class Form1 : Form
    {
        public Form1()
        {
            InitializeComponent();
        }

        private void Form1_Load(object sender, EventArgs e)
        {
            periodCBox.Enabled = false;
            cycleCheckbox.Checked = false;
            seasonPeriodBox.Enabled = false;
            seasonCheckBox.Checked = false;
        }

        private void checkBox1_CheckedChanged(object sender, EventArgs e)
        {
            if (cycleCheckbox.Checked) periodCBox.Enabled = true;
            else periodCBox.Enabled = false;
        }

        private void seasonCheckBox_CheckedChanged(object sender, EventArgs e)
        {
            if (seasonCheckBox.Checked) seasonPeriodBox.Enabled = true;
            else seasonPeriodBox.Enabled = false;
        }
    }
}
```

```

}

private void button2_Click(object sender, EventArgs e)
{
    MessageBox.Show("Файл загружен успешно", "Оповещение");
    OpenFileDialog openFileDialog2 = new OpenFileDialog();
    openFileDialog2.Filter = "Files|*.cvs;*.xls;*.xlsx";
    openFileDialog2.Title = "Select a file";

    if (openFileDialog2.ShowDialog() ==
System.Windows.Forms.DialogResult.OK)
    {
        MessageBox.Show("Файл загружен успешно", "Оповещение");
        // button2.Text = openFileDialog1.FileName;
        // comboBox2.Items.Add(openFileDialog1.FileName);
        // comboBox2.SelectedValue = openFileDialog1.FileName;
    }
}

private void groupBox2_Enter(object sender, EventArgs e)
{

}

private void button1_Click(object sender, EventArgs e)
{
    OpenFileDialog openFileDialog2 = new OpenFileDialog();
    openFileDialog2.Filter = "Files|*.cvs;*.xls;*.xlsx";
    openFileDialog2.Title = "Select a file";

    if (openFileDialog2.ShowDialog() ==
System.Windows.Forms.DialogResult.OK)
    {
        MessageBox.Show("Файл загружен успешно", "Оповещение");
        // button2.Text = openFileDialog1.FileName;
        // comboBox2.Items.Add(openFileDialog1.FileName);
        // comboBox2.SelectedValue = openFileDialog1.FileName;
    }
}
}

```

```
}
```

```
using  
System;
```

```
using System.Linq;  
using System.Windows;  
using System.Windows.Controls;  
using OxyPlot;  
  
namespace NDtw.Visualization.Wpf  
{  
    public partial class DtwMatrixPlot : UserControl  
    {  
        public DtwMatrixPlot()  
        {  
            InitializeComponent();  
        }  
  
        public static DependencyProperty DtwProperty =  
            DependencyProperty.Register(  
                "Dtw",  
                typeof(IDtw),  
                typeof(DtwMatrixPlot),  
                new FrameworkPropertyMetadata(null, (d, e) =>  
                    ((DtwMatrixPlot)d).OnDataChanged()));  
  
        public IDtw Dtw  
        {  
            get { return (IDtw) GetValue(DtwProperty); }  
            set { SetValue(DtwProperty, value); }  
        }  
  
        public static DependencyProperty DrawDistanceProperty =  
            DependencyProperty.Register(  
                "DrawDistance",  
                typeof(bool),  
                typeof(DtwMatrixPlot),  
                new FrameworkPropertyMetadata(false, (d, e) =>  
                    ((DtwMatrixPlot)d).OnDataChanged()));  
  
        public bool DrawDistance
```

```

    {
        get { return (bool)GetValue(DrawDistanceProperty); }
        set { SetValue(DrawDistanceProperty, value); }
    }

    public static DependencyProperty DrawCostProperty =
        DependencyProperty.Register(
            "DrawCost",
            typeof(bool),
            typeof(DtwMatrixPlot),
            new FrameworkPropertyMetadata(false, (d, e) =>
                ((DtwMatrixPlot)d).OnDataChanged()));

    public bool DrawCost
    {
        get { return (bool)GetValue(DrawCostProperty); }
        set { SetValue(DrawCostProperty, value); }
    }

    public void OnDataChanged()
    {
        if(DrawCost && DrawDistance)
            throw new Exception("Only one of the values can be drawn
at once, 'cost' or 'distance'.");

        double[][] matrixValues = null;
        if(DrawCost)
            matrixValues = Dtw.GetCostMatrix();
        if (DrawDistance)
            matrixValues = Dtw.GetDistanceMatrix();

        var dtwPath = Dtw.GetPath();
        var xLength = Dtw.XLength;
        var yLength = Dtw.YLength;
        var cost = Dtw.GetCost();
        var costNormalized = Dtw.GetCost() / Math.Sqrt(xLength *
xLength + yLength * yLength);

        var plotModel = new PlotModel(String.Format("Dtw norm by
length: {0:0.00}, total: {1:0.00}", costNormalized, cost))
        {

```

```

        LegendTextColor = DrawCost || DrawDistance ?
OxyColors.White : OxyColors.Black,
    };

    if (matrixValues != null)
    {
        var maxMatrixValue = 0.0;
        for (int i = 0; i < xLength; i++)
            for (int j = 0; j < yLength; j++)
                maxMatrixValue = Math.Max(maxMatrixValue,
Double.IsPositiveInfinity(matrixValues[i][j]) ? 0 :
matrixValues[i][j]);

        for (int i = 0; i < xLength; i++)
            for (int j = 0; j < yLength; j++)
            {
                var value = matrixValues[i][j];
                var isValuePositiveInfinity =
Double.IsPositiveInfinity(value);

                var intensityBytes = isValuePositiveInfinity ? new
byte[] { 0, 0, 0 } : GetFauxColourRgbIntensity(value, 0,
maxMatrixValue);
                //var intensityByte = (byte)(255 - Math.Floor(255 *
intensity));
                plotModel.Annotations.Add(new PolygonAnnotation
                {
                    Points =
                        new[]
                        {
                            new DataPoint(i - 0.5, j - 0.5), new DataPoint(i
+ 0.5, j - 0.5),
                                new DataPoint(i + 0.5, j + 0.5), new
DataPoint(i - 0.5, j + 0.5),
                                    },
                    StrokeThickness = 0,
                    Selectable = false,
                    Layer = AnnotationLayer.BelowAxes,
                    Fill = OxyColor.FromArgb(255, intensityBytes[0],
intensityBytes[1], intensityBytes[2]),
                });
    }

```

```

    }

    for (int i = 0; i < 30; i++)
    {
        var intensityBytes = GetFauxColourRgbIntensity(i, 0,
29);

        plotModel.Annotations.Add(new RectangleAnnotation
        {
            MinimumX = -39,
            MaximumX = -25,
            MinimumY = -i - 6,
            MaximumY = -i - 5,
            Selectable = false,
            Fill = OxyColor.FromArgb(255, intensityBytes[0],
intensityBytes[1], intensityBytes[2])
        });
    }

    plotModel.Annotations.Add(new TextAnnotation
    {
        Position = new DataPoint(-24, -5),
        HorizontalAlignment = HorizontalTextAlign.Left,
        VerticalAlignment = VerticalTextAlign.Middle,
        StrokeThickness = 0,
        Text = "0"
    });

    plotModel.Annotations.Add(new TextAnnotation
    {
        Position = new DataPoint(-24, -34),
        HorizontalAlignment = HorizontalTextAlign.Left,
        VerticalAlignment = VerticalTextAlign.Middle,
        StrokeThickness = 0,
        Text = String.Format("{0:0.00}", maxMatrixValue),
    });
}

var matrixPathSeries = new LineSeries("Path")
{
    StrokeThickness = 1,

```



```

        Color = OxyColors.Red,
    };

    for (int i = 0; i < dtwPath.Length; i++)
        matrixPathSeries.Points.Add(new
        DataPoint(dtwPath[i].Item1, dtwPath[i].Item2));

    plotModel.Series.Add(matrixPathSeries);

    var seriesMatrixScale = (xLength + yLength) * 0.05;

    for (int variableIndex = 0; variableIndex <
    Dtw.SeriesVariables.Length; variableIndex++)
    {
        var variableA = Dtw.SeriesVariables[variableIndex];
        var variableASeries = variableA.OriginalXSeries;
        var variableB = Dtw.SeriesVariables[variableIndex];
        var variableBSeries = variableB.OriginalYSeries;

        var minSeriesA = variableASeries.Min();
        var maxSeriesA = variableASeries.Max();
        var normalizedSeriesA = variableASeries.Select(x => (x -
        minSeriesA) / (maxSeriesA - minSeriesA)).ToList();
        var matrixSeriesA = new
        LineSeries(variableA.VariableName);

        for (int i = 0; i < normalizedSeriesA.Count; i++)
            matrixSeriesA.Points.Add(new DataPoint(i, (-1 +
            normalizedSeriesA[i]) * seriesMatrixScale - 1 - seriesMatrixScale *
            (variableIndex + 1)));

        plotModel.Series.Add(matrixSeriesA);

        var minSeriesB = variableBSeries.Min();
        var maxSeriesB = variableBSeries.Max();
        var normalizedSeriesB = variableBSeries.Select(x => (x -
        minSeriesB) / (maxSeriesB - minSeriesB)).ToList();
        var matrixSeriesB = new
        LineSeries(variableB.VariableName);

        for (int i = 0; i < normalizedSeriesB.Count; i++)

```

```

        matrixSeriesB.Points.Add(new DataPoint( -
normalizedSeriesB[i] * seriesMatrixScale - 1 - seriesMatrixScale *
(variableIndex + 1), i));

        plotModel.Series.Add(matrixSeriesB);
    }

    plotModel.Axes.Add(new LinearAxis(AxisPosition.Bottom, "
Series A") { Maximum = Math.Max(xLength, yLength),
PositionAtZeroCrossing = true});
    plotModel.Axes.Add(new LinearAxis(AxisPosition.Left, "
Series B") { Maximum = Math.Max(xLength, yLength),
PositionAtZeroCrossing = true });

    MatrixPlot.Model = plotModel;
}

/// <summary>
/// Generate heatmap color
/// </summary>
/// <remarks>Thanks to Eddie Yee Tak Ma:
http://eddiema.ca/2011/01/21/c-sharp-heatmaps/ </remarks>
public static byte[] GetFauxColourRgbIntensity(double val,
double min, double max)
{
    byte r = 0;
    byte g = 0;
    byte b = 0;
    val = (val - min) / (max - min);
    if (val <= 0.2)
    {
        b = (byte)((val / 0.2) * 255);
    }
    else if (val > 0.2 && val <= 0.7)
    {
        b = (byte)((1.0 - ((val - 0.2) / 0.5)) * 255);
    }
    if (val >= 0.2 && val <= 0.6)
    {
        g = (byte)(((val - 0.2) / 0.4) * 255);
    }
}

```

```

else if (val > 0.6 && val <= 0.9)
{
    g = (byte)((1.0 - ((val - 0.6) / 0.3)) * 255);
}
if (val >= 0.5)
{
    r = (byte)(((val - 0.5) / 0.5) * 255);
}
return new byte[] { r, g, b };
}

/// <summary>
/// Generate heatmap color (grayscale)
/// </summary>
/// <remarks>Thanks to Eddie Yee Tak Ma:
http://eddiema.ca/2011/01/21/c-sharp-heatmaps/ </remarks>
public static byte[] GetGrayscaleRgbIntensity(double val,
double min, double max)
{
    byte y;
    val = (val - min) / (max - min);
    y = (byte)((1.0 - val) * 255);
    return new [] { y, y, y };
}

private const double AspectRatio = 1;
protected override void
OnRenderSizeChanged(SizeChangedInfo sizeInfo)
{
    //keep aspect ratio
    if (sizeInfo.NewSize.Width / sizeInfo.NewSize.Height >
AspectRatio)
    {
        MatrixPlot.Width = sizeInfo.NewSize.Height *
AspectRatio;
        MatrixPlot.Height = MatrixPlot.Width / AspectRatio;
    }
    else
    {
        MatrixPlot.Height = sizeInfo.NewSize.Width /
AspectRatio;

```

```

        MatrixPlot.Width = MatrixPlot.Height * AspectRatio;
    }
}
}
}

```

```

<?xml version="1.0" encoding="utf-8"?>
<Project ToolsVersion="4.0" DefaultTargets="Build"
xmlns="http://schemas.microsoft.com/developer/msbuild/2003">
  <PropertyGroup>
    <Configuration Condition=" '$(Configuration)' == "
">Debug</Configuration>
    <Platform Condition=" '$(Platform)' == " ">AnyCPU</Platform>
    <ProductVersion>8.0.30703</ProductVersion>
    <SchemaVersion>2.0</SchemaVersion>
    <ProjectGuid> {D8F86136-A2B6-48EB-A36A-
49B1265A07E2} </ProjectGuid>
    <OutputType>library</OutputType>
    <AppDesignerFolder>Properties</AppDesignerFolder>
    <RootNamespace>NDtw.Visualization.Wpf</RootNamespace>
    <AssemblyName>NDtw.Visualization.Wpf</AssemblyName>
    <TargetFrameworkVersion>v4.0</TargetFrameworkVersion>
    <TargetFrameworkProfile>Client</TargetFrameworkProfile>
    <FileAlignment>512</FileAlignment>
    <ProjectTypeGuids> {60dc8134-eba5-43b8-bcc9-
bb4bc16c2548}; {FAE04EC0-301F-11D3-BF4B-
00C04F79EFBC} </ProjectTypeGuids>
    <WarningLevel>4</WarningLevel>
  </PropertyGroup>
  <PropertyGroup Condition=" '$(Configuration)|$(Platform)' ==
'Debug|AnyCPU' ">
    <DebugSymbols>>true</DebugSymbols>
    <DebugType>full</DebugType>
    <Optimize>>false</Optimize>
    <OutputPath>bin\Debug\</OutputPath>
    <DefineConstants>DEBUG;TRACE</DefineConstants>
    <ErrorReport>prompt</ErrorReport>
    <WarningLevel>4</WarningLevel>
  </PropertyGroup>
  <PropertyGroup Condition=" '$(Configuration)|$(Platform)' ==
'Release|AnyCPU' ">

```

```

<DebugType>pdbonly</DebugType>
<Optimize>>true</Optimize>
<OutputPath>bin\Release\</OutputPath>
<DefineConstants>TRACE</DefineConstants>
<ErrorReport>prompt</ErrorReport>
<WarningLevel>4</WarningLevel>
</PropertyGroup>
<ItemGroup>
  <Reference Include="OxyPlot, Version=2013.1.2.1, Culture=neutral,
PublicKeyToken=638079a8f0bd61e9, processorArchitecture=MSIL">
    <SpecificVersion>False</SpecificVersion>

<HintPath>..\packages\OxyPlot.Wpf.2013.1.2.1\lib\NET40\OxyPlot.dll</
HintPath>
  </Reference>
  <Reference Include="OxyPlot.Wpf, Version=2013.1.2.1,
Culture=neutral, PublicKeyToken=75e952ba404cddb0,
processorArchitecture=MSIL">
    <SpecificVersion>False</SpecificVersion>

<HintPath>..\packages\OxyPlot.Wpf.2013.1.2.1\lib\NET40\OxyPlot.Wpf.
dll</HintPath>
  </Reference>
  <Reference Include="OxyPlot.Xps, Version=2013.1.2.1,
Culture=neutral, processorArchitecture=MSIL">
    <SpecificVersion>False</SpecificVersion>

<HintPath>..\packages\OxyPlot.Wpf.2013.1.2.1\lib\NET40\OxyPlot.Xps.
dll</HintPath>
  </Reference>
  <Reference Include="System" />
  <Reference Include="System.Data" />
  <Reference Include="System.Xml" />
  <Reference Include="Microsoft.CSharp" />
  <Reference Include="System.Core" />
  <Reference Include="System.Xml.Linq" />
  <Reference Include="System.Data.DataSetExtensions" />
  <Reference Include="System.Xaml">
    <RequiredTargetFramework>4.0</RequiredTargetFramework>
  </Reference>
  <Reference Include="WindowsBase" />
  <Reference Include="PresentationCore" />
  <Reference Include="PresentationFramework" />
</ItemGroup>

```

```

<ItemGroup>
  <Page Include="DtwByVariablePlot.xaml">
    <Generator>MSBuild:Compile</Generator>
    <SubType>Designer</SubType>
  </Page>
  <Page Include="DtwMatrixPlot.xaml">
    <Generator>MSBuild:Compile</Generator>
    <SubType>Designer</SubType>
  </Page>
  <Compile Include="DtwByVariablePlot.xaml.cs">
    <DependentUpon>DtwByVariablePlot.xaml</DependentUpon>
  </Compile>
  <Compile Include="DtwMatrixPlot.xaml.cs">
    <DependentUpon>DtwMatrixPlot.xaml</DependentUpon>
    <SubType>Code</SubType>
  </Compile>
</ItemGroup>
<ItemGroup>
  <Compile Include="Properties\AssemblyInfo.cs">
    <SubType>Code</SubType>
  </Compile>
  <Compile Include="Properties\Resources.Designer.cs">
    <AutoGen>True</AutoGen>
    <DesignTime>True</DesignTime>
    <DependentUpon>Resources.resx</DependentUpon>
  </Compile>
  <Compile Include="Properties\Settings.Designer.cs">
    <AutoGen>True</AutoGen>
    <DependentUpon>Settings.settings</DependentUpon>
    <DesignTimeSharedInput>True</DesignTimeSharedInput>
  </Compile>
  <EmbeddedResource Include="Properties\Resources.resx">
    <Generator>ResXFileCodeGenerator</Generator>
    <LastGenOutput>Resources.Designer.cs</LastGenOutput>
  </EmbeddedResource>
  <None Include="packages.config" />
  <None Include="Properties\Settings.settings">
    <Generator>SettingsSingleFileGenerator</Generator>
    <LastGenOutput>Settings.Designer.cs</LastGenOutput>
  </None>
  <AppDesigner Include="Properties\" />
</ItemGroup>
<ItemGroup>
  <ProjectReference Include="..\NDtw\NDtw.csproj">
    <Project>{572EB90F-7F5D-484D-A701-C7FF1556FE32}</Project>

```

```

    <Name>NDtw</Name>
    </ProjectReference>
</ItemGroup>
<Import Project="$(MSBuildToolsPath)\Microsoft.CSharp.targets" />
<!-- To modify your build process, add your task inside one of the targets
below and uncomment it.

```

Other similar extension points exist, see Microsoft.Common.targets.

```

<Target Name="BeforeBuild">
</Target>
<Target Name="AfterBuild">
</Target>
-->
</Project>

```

Класс, реализующий алгоритм DTW

```

using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Collections;

namespace DTW
{
    class SimpleDTW
    {
        double[] x;
        double[] y;
        double[,] distance;
        double[,] f;
        ArrayList pathX;
        ArrayList pathY;
        ArrayList distanceList;
        double sum;

        public SimpleDTW(double[] _x, double[] _y)
        {
            x = _x;
            y = _y;
            distance = new double[x.Length, y.Length];
            f = new double[x.Length+1, y.Length+1];

            for (int i = 0; i < x.Length; ++i){
                for (int j = 0; j < y.Length; ++j){

```

```

        distance[i, j] = Math.Abs(x[i] - y[j]);
    }
}

for (int i = 0; i <= x.Length; ++i)
{
    for (int j = 0; j <= y.Length; ++j)
    {
        f[i, j] = -1.0;
    }
}

for (int i = 1; i <= x.Length; ++i) {
    f[i,0] = double.PositiveInfinity;
}
for (int j = 1; j <= y.Length; ++j) {
    f[0, j] = double.PositiveInfinity;
}

f[0, 0] = 0.0;
sum = 0.0;

pathX = new ArrayList();
pathY = new ArrayList();
distanceList = new ArrayList();
}

public ArrayList getPathX(){
    return pathX;
}

public ArrayList getPathY() {
    return pathY;
}

public double getSum(){
    return sum;
}

public double[,] getFMatrix() {
    return f;
}

public ArrayList getDistanceList() {
    return distanceList;
}

```



```

}

public void computeDTW() {
    sum = computeFBackward(x.Length, y.Length);
    //sum = computeFForward();
}

public double computeFForward() {
    for (int i = 1; i <= x.Length; ++i) {
        for (int j = 1; j <= y.Length; ++j) {
            if (f[i - 1, j] <= f[i - 1, j - 1] && f[i - 1, j] <= f[i, j - 1]) {
                f[i, j] = distance[i - 1, j - 1] + f[i - 1, j];
            }
            else if (f[i, j - 1] <= f[i - 1, j - 1] && f[i, j - 1] <= f[i - 1, j]) {
                f[i, j] = distance[i - 1, j - 1] + f[i, j - 1];
            }
            else if (f[i - 1, j - 1] <= f[i, j - 1] && f[i - 1, j - 1] <= f[i - 1, j]) {
                f[i, j] = distance[i - 1, j - 1] + f[i - 1, j - 1];
            }
        }
    }
    return f[x.Length, y.Length];
}

public double computeFBackward(int i, int j)
{
    if (!(f[i, j] < 0.0)) {
        return f[i, j];
    }
    else {
        if (computeFBackward(i - 1, j) <= computeFBackward(i, j - 1) &&
computeFBackward(i - 1, j) <= computeFBackward(i - 1, j - 1)
&& computeFBackward(i - 1, j) < double.PositiveInfinity)
        {
            f[i, j] = distance[i - 1, j - 1] + computeFBackward(i - 1, j);
        }
        else if (computeFBackward(i, j - 1) <= computeFBackward(i - 1, j) &&
computeFBackward(i, j - 1) <= computeFBackward(i - 1, j - 1)
&& computeFBackward(i, j - 1) < double.PositiveInfinity)
        {
            f[i, j] = distance[i - 1, j - 1] + computeFBackward(i, j - 1);
        }
        else if (computeFBackward(i - 1, j - 1) <= computeFBackward(i - 1, j)
&& computeFBackward(i - 1, j - 1) <= computeFBackward(i, j - 1)
&& computeFBackward(i - 1, j - 1) < double.PositiveInfinity)

```

```
        {
            f[i, j] = distance[i - 1, j - 1] + computeFBackward(i - 1, j - 1);
        }
    }
    return f[i, j];
}

}

}
```