План введения:   
1.Описание ДМСД подхода и модели \\ стоит вынести в конец введения

2.Описание применения нейронной сети

Введение:

Математическое моделирование - удобный подход для описания комплексных экосистем. Описаный далее подход был разработан для исследования зоопланктона на определенном участке времени на озере Севан. Полное имя метода звучит следующим образом - дискретное моделирования динамического изменения системы. Моделирования с помощью данного метода позволяют определить влияние компонент системы друга на друга.

Данный метода описывает все возможные состояние межкомпонентного взаимодействия, а именно “плюс-плюс”, “минус - минус”, “плюс -минус”,  “плюс - ноль”, “минус - ноль”, “ноль - ноль”.

Основная суть метода состоит в том, что мы строим матрицу зависимости, которая отображает отношения каждой компоненты системы к каждой и на основе этой зависимости можем предсказывать следующее состояние системы, на основе наблюдения в текущий момент времени.

Для большего понимания рассмотрим пример исследования болезни у разных особей одного вида. Пускай имеются n особей, которые болеют одинаковой болезнью. Наблюдение за процессом болезни происходит через измерение нескольких характеристик организма особи, к примеру, пульс, температура, количество лейкоцитов в крови и т.п. - это и будет компонентами нашей системы. У разных особей болезнь может проходить по разному, в том плане, что один индивид прошел основную фазу за 2 дня и восстановительную фазу за 4, а другой за 3 и за 2 дня соответственно. При измерении показателей болезни мы знаем только “человеческое время”, но не биологическое, а для исследования необходимо много результатов, выстроенных в правильном порядке, а не просто много, не связанных друг с другом наблюдений. В данном случае метод помогает объединить  данные от многих особей в одну последовательность с корректной принадлежностью к временным промежуткам.

Идея заключается в выявлении матрицы зависимости компонент друг от друга .

В данной работе, для анализа данных и построение нужной нам матрицы было использовано программное воплощение математической модели -  
Искусственная нейронная сеть. Причины, которые побудили использовать соответствующую математическую модель описаны ниже.

В последние несколько лет мы наблюдаем взрыв интереса к нейронным сетям, которые успешно применяются в самых различных областях - бизнесе, медицине, технике, геологии , физике.

Нейронные сети возникли из исследований в области искусственного интеллекта, а именно, из попыток воспроизвести способность биологических нервных систем обучаться и исправлять ошибки, моделируя низкоуровневую структуру мозга (Patterson, 1996). Основной областью исследований по искусственному интеллекту в 60-е - 80-е годы были экспертные системы. Такие системы основывались на высокоуровневом моделировании процесса мышления (в частности, на представлении, что процесс нашего мышления построен на манипуляциях с символами). Скоро стало ясно, что подобные системы, хотя и могут принести пользу в некоторых областях, не ухватывают некоторые ключевые аспекты человеческого интеллекта. Согласно одной из точек зрения, причина этого состоит в том, что они не в состоянии воспроизвести структуру мозга. Чтобы создать искусственных интеллект, необходимо построить систему с похожей архитектурой.

Но, не смотря, на невозможность повторить полную структуру человеческого мозга, искусственные нейронные сети способны достичь замечательных результатов

Нейронные сети вошли в практику везде, где нужно решать задачи прогнозирования, классификации или управления. Такой впечатляющий успех определяется несколькими причинами:

Богатые возможности.  Нейронные сети - исключительно мощный метод моделирования, позволяющий воспроизводить чрезвычайно сложные зависимости. В частности, нейронные сети *нелинейны* по свой природе. На протяжение многих лет линейное моделирование было основным методом моделирования в большинстве областей, поскольку для него хорошо разработаны процедуры оптимизации. В задачах, где линейная аппроксимация неудовлетворительна (а таких достаточно много), линейные модели работают плохо. Кроме того, нейронные сети справляются с "*проклятием размерности*", которое не позволяет моделировать линейные зависимости в случае большого числа переменных

Простота в использовании.  Нейронные сети *учатся на примерах*. Пользователь нейронной сети подбирает представительные данные, а затем запускает *алгоритм обучения*, который автоматически воспринимает структуру данных. При этом от пользователя, конечно, требуется какой-то набор эвристических знаний о том, как следует отбирать и подготавливать данные, выбирать нужную архитектуру сети и интерпретировать результаты, однако уровень знаний, необходимый для успешного применения нейронных сетей, гораздо скромнее, чем, например, при использовании традиционных методов статистики.

Нейронные сети привлекательны с интуитивной точки зрения, ибо они основаны на примитивной биологической модели нервных систем. В будущем развитие таких нейро-биологических моделей может привести к созданию действительно мыслящих компьютеров.

Актуальность применения Искусственных нейронных сетей в контексте данной задачи обусловлена  тем, что решение задачи, базирующееся на применении генетического алгоритма (ссылка на работу Г.Н. Жолткевича), требует много времени и не может использовать потенциал много-поточных вычислений. А так же рассмотрения другого типа эмпирических алгоритмов, которые потенциально могут увеличить точность результатов. Соответственно применение нейронной сети призвано решить проблему невозможности расспаралеливания программной реализации и по возможности, исследуя различные архитектуры и алгоритмы обучения искусственной нейронной сети повысить точность результатов.

\\добавить описание, что такое цикл

\\НЕЙРОННЫЕ СЕТИ

<http://www.statsoft.ru/home/textbook/modules/stneunet.html>

https://ru.wikipedia.org/wiki/%D0%98%D1%81%D0%BA%D1%83%D1%81%D1%81%D1%82%D0%B2%D0%B5%D0%BD%D0%BD%D0%B0%D1%8F\_%D0%BD%D0%B5%D0%B9%D1%80%D0%BE%D0%BD%D0%BD%D0%B0%D1%8F\_%D1%81%D0%B5%D1%82%D1%8C

Основная часть

1. Описание модели
2. Теория нейронной сети
3. Моделирование идеальной системы
4. Описание применения нейронной сети
   1. Архитектура
   2. Подговка данных
   3. Реализация
5. Полученные результаты

**1.Описание модели**

Метод DMSD – описан в статье (ссылка на работу Г.Н. Жолткевича).  
У нас имеется некоторая биологическая система, которая описывается N компонентами … . Это может быть количество животных, в примере болезни – пульс, температура и т.д. Каждая компонента может принимать значение из конечного промежутка – 1,2...K. Система измеряется в определенные моменты времени, тоесть измеряется значение каждой компоненты. Поэтому мы можем определить матрицу наблюдения за нашей системой, как бесконечную вправо матрицу.

Данная матрица описывет все возможные состояние системы на на временном промежутке t = 0,1 …   
Следовательно мы можем определить всю систему на момент времени t,

Как - , которая может единолично определится из состояния системы на момент (*t-*1).

Поскольку кажданя компонента может принимать строго определннае количество состояний, следовательно существует некоторое – , называемоем периодом системы, такое, что ***A*() –** примет все возможные состояние системы.

Так же справедливо утверждение:

, где – некое целое число > 0.

Соответственно мы можем выделить минор матрицы :

Который полностью описывает поведение системы.

Теперь рассмотрим матрицу отношения компонент, на основе которой мы можем определять поведения системы, используя лиш одно состояние.  
Пусть имметеся множество – Ω = . Подробности взаимоотношений можно найти в статье (ссылка на статью Г.Н. Жолткевича).  
Соответственно мы можем определить матрицу взаимоотношений между компонентами –

D =

Где Ω и описывает влияние *i* компоненты на *j* –ю. Состояние компоненты в момент времени t+1 определяется из состояния компоненты t c учетом всех компонент, которые на нее влияют, используя матрицу D. Состояние компоненты может изменится только на целую величину из интервала - .

(t+1) = F((t+1))

В работе (ссылка на статью Г.Н. Жолткевича) рассматривается два подхода к функции F, в данной работе будет рассмотрен только один – «Weight Functions’ Approach» (стоит ли его описывать явно ?), соответственно моделирование будет использовать только данный подход. Моделирование идеальной системы(описано ниже), показало, что система может иметь полный цикл, в тех случаях, когда диагональные элементы равны нулю, что означает, что компонента не оказывает влияние на саму себя.

2.

**3. Моделирование идеальной системы**

Поскольку для работы нейронной сети необходимо данные, на которых сеть может обучатся – следовательно необходимо смоделировать идеальную систему, в которой мы можем гарантировать наличие всех состояний из полного цикла системы. Зная характеристики системы можно построить алгоритм по которому мы сможем построить данные о системе.

**Алгоритм моделирования идеальной системы:**

**Шаг 0:** Инициализировать матрицу D случайными значениями из множества – . Инициализировать количество итераций - IC , нулем.   
Перейти к шагу 2.

**Шаг 1:** Если IC < 10000, то перейти к шагу 2, иначе конец выполнения.

**Шаг 2:**Инциализивароть вектор – *A*(0) случайным образом. Присвоить t значение – нуль. Присвоить IC =IC+1.  
Перейти к шагу 3.

**Шаг 3:**Построить состояние (*t* +1).   
Перейти к шагу 4.

**Шаг 4:** Найти k , такое, что A(t) = A(k), где k = 1..t-1.  
Перейти к шагу 5.

**Шаг 5:** Если k то перейти к шагу 6,иначе перейти к шагу 3.

**Шаг 6:** Если k == t-1 , то перейти к шагу 1, иначе конец выполнения всего алгоритма.

Циклом системы будет все состояния системы от (A(k),A(k+1),…A(t-1))

**Доказательство конечности алгоритма:**

Алгоритм будет завершен за конечное количество шагов, поскольку, количество состояний системы конечно, значит мы на определенном t получим состояние, которое уже было.

**4.Применения ИНС**

В качестве входных сигналов мы используем матрицу корреляции Пирсона, для всех состояний системы из цикла.

R = ,

где -Коэффициент корреляции Пирсона, между *A(i), A(j)*.

Для каждого элемента матрицы отношений *D* строится отдельная нейронная сеть и в качестве входных параметров используется матрица *R*, в качестве выходных нейронов – слой состоящий из трёх нейронов , соответствующие множеству – Ω.

В качестве скрытого слоя используется слой из 200 нейронов.  
Число 200 определенно эмпирическим путем, ниже будет приведена сравнительная таблица точности результатов в зависимости от количества нейронов скрытого слоя.

В работе рассматривается три алгоритма обучения, а именно – BackProp, QuickProp, RPropMinus. Ниже будет превидена сравнительная характеристика алгоритмов, а так же характеристика применения каждого из алгоритмов. Все эксперементы проводились с использование языка программирования – Python, с применение библиотеки – PyBrain.

**4.1 Алгоритм BackProp**

Структура нейронной сети

Входной слой

**4.2 Алгоритм QuickProp**

**4.3 Алгоритм RPropMinus**

Выводы:

Что было сделано :  
1.Смоделирована идеальная система  
2.Определен подход, как испольсовать ИНС, так что бы была решена проблема расспаралеливания   
3.Протестированы различные алгоритмы обучения сети  
4.Протестированы различные архитектуры ИНС

Точноть больше 75 процентов

Решена проблема многопоточности