Российская Академия Наук

РОССИЙСКАЯ АКАДЕМИЯ НАУК

ИНСТИТУТ ПРОБЛЕМ БЕЗОПАСНОГО РАЗВИТИЯ АТОМНОЙ ЭНЕРГЕТИКИ

RUSSIAN ACADEMY OF SCIENCES

NUCLEAR SAFETY INSTITUTE

Препринт ИБРАЭ № IBRAE-1999-07

Preprint IBRAE-1999-07

Н.С. Грачев, В.В. Демьянов, М.Ф. Каневский

АНАЛИЗ ДАННЫХ ПО ОКРУЖАЮЩЕЙ СРЕДЕ ПРИ ПОМОЩИ НЕЙРОННЫХ СЕТЕЙ С ОБОБЩЕННОЙ РЕГРЕССИЕЙ И ГЕОСТАТИСТИКИ

Москва 1999

Moscow 1999

УДК 502.3

Грачев Н.С., Демьянов В.В., Каневский М.Ф., Савельева Е.А., Тимонин В.А., Чернов С.Ю. АНАЛИЗ ДАННЫХ ПО ОКРУЖАЮЩЕЙ СРЕДЕ ПРИ ПОМОЩИ НЕЙРОННЫХ СЕТЕЙ С ОБОБЩЕННОЙ РЕГРЕССИЕЙ И ГЕОСТАТИСТИКИ. Препринт № IBRAE-99-07. Москва: Институт проблем безопасного развития атомной энергетики РАН. Ноябрь 1999. 39 с. Библиогр.: 14 назв.

Аннотация

В работе исследуется методы картирования пространственных данных при помощи искусственных нейронных сетей и геостатистики. Изучается влияние пространственных свойств исходных данных и конфигурации сети на качество оценки. Модели применены к реальным данным по радиоактивному загрязнению Брянской области.

Grachev N.S., Demyanov V.V., Kanevski M.F., Savelieva E.A., Timonin V.A., Chernov S.Y. ANALYZING OF ENVIRONMENTAL DATA WITH GRNN AND GEOSTATISTICS. Preprint IBRAE-99-07. Moscow: Nuclear Safety Institute. November 1999. 39 p. — Refs.: 14 items.

Abstract

In this work methods of mapping spatial data by Neural Network and Geostatistic are analysed. Effects of structure in initial data and Network configuration on model quality is studied. All models are applied to real radioactive pollution data in Briansk region.

Анализ данных по окружающей среде при помощи нейронных сетей с обобщенной регрессией и геостатистики

Н.С. Грачев, В.В. Демьянов, М.Ф. Каневский

ИНСТИТУТ ПРОБЛЕМ БЕЗОПАСНОГО РАЗВИТИЯ АТОМНОЙ ЭНЕРГЕТИКИ
113191, Москва, ул. Б. Тульская, 52
тел.: (095) 955–22–31, факс: (095) 955–11–51, http: //www.ibrae.ac.ru/~mkanev, email: vasia@ibrae.ac.ru, jedi@ibrae.ac.ru

Содержание

1 Введение

В последние несколько лет активное развитие получили Искусственные Нейронные Сети (ИНС). Среди множества математических алгоритмов, реализациями которых являются ИНС, есть известный в статистике метод – Обобщенная Регрессия. ИНС, реализующая этот алгоритм, называется Нейронная Сеть с Обобщенной Регрессией (НСОР) или General Regression Neural Network (GRNN). В этой работе НСОР использовалась для пространственной интерполяции (картирования) зашумленных данных. В цели работы входило исследование зависимости качества оценки от пространственных свойств исходных данных. Основным критерием служила среднеквадратическая ошибка. В качестве дополнительных критериев рассматривались вариация и смещенность оценки, а также способность сети полностью воспроизвести пространственную корреляционную структуру исходных данных.

При помощи НСОР было проведено моделирование реальных данных по радиактивному загрязнению почвы (произошедшему после Чернобыльской аварии) цезием (^{137}Cs) в западной части Брянской области. Работы про пространственному картированию Чернобыльских выпадений проводились и ранее, но с использованием других моделей [6-9].

2 Обобщенная Регрессия

Регрессией зависимой переменной **Z** по независимой переменной **X** называется вычисление наиболее вероятного значения **Z** (в смысле минимизации среднеквадратической ошибки) для каждого значения **X**, базирующееся на ограниченном количестве возможно зашумленных измерений **X** и соответствующих значений **Z** [2]. Переменные **X** и **Z** обычно являются векторами. При описании системы зависимую переменную **Z** мы будем называть ее выходом, а независимую переменную **X** входом. Стандартная задача, которую можно решить при помощи Обобщенной Регрессии такова: ***в какой–то области пространства в определенных точках произведены замеры какой–то величины (например, радиактивного загрязнения см. рис. 1*), необходимо по этим данным предсказать значение этой величины во всей области.***

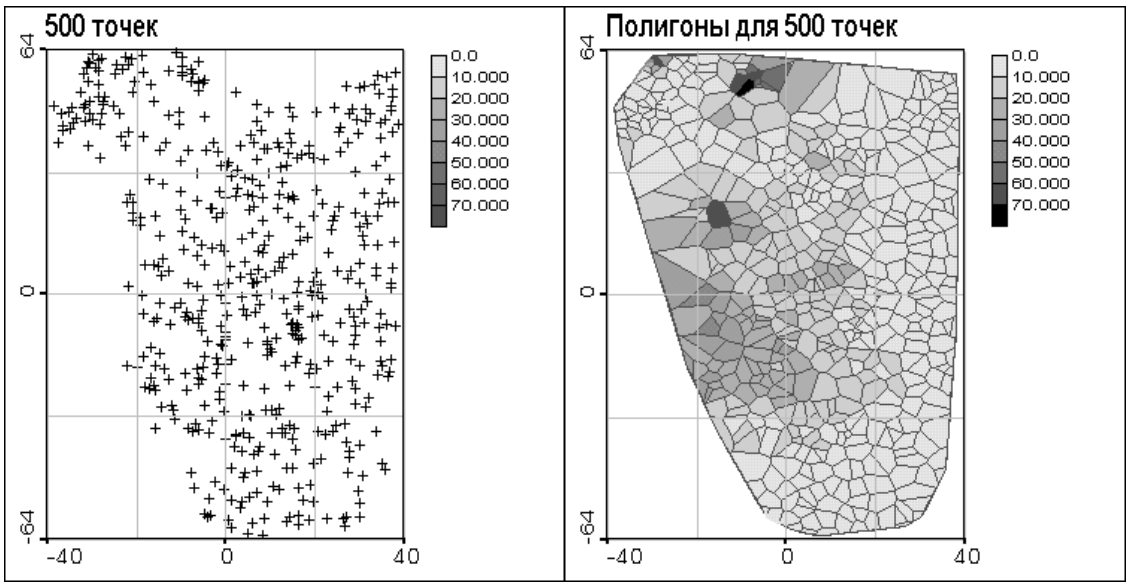


Рис. 1. Слева крестиками обозначены точки, в которых производились измерения, а справа полигоны Воронова. Каждой выделенной области соответствует одна точка измерения, цвет определяет примерное значение загрязнения в соответствии со шкалой

Решение будет наиболее простым, если предположить линейную зависимость между **Z** и **X**. Но если истинная зависимость квадратичная, то линейная оценка даст большую ошибку. Можно предполагать, что зависимость квадратичная, или имеет какой – либо иной вид, но, делая это заранее, мы сокращаем область применения нашего алгоритма, так как если истинная зависимость **Z** от **X** не будет угадана то ошибки "предсказаний" **Z** будут непростительно велики. Следовательно, этот подход предполагает, что у нас имеются некоторые специфические сведения о данных, которые мы хотим промоделировать. Только в этом случае сразу можно ограничиться определенным классом функций и искать решение задачи только в нем.

Обобщенная Регрессия является более гибким алгоритмом. **Z** и **X** предполагаются связанными посредством ***Функции Плотности Вероятности – ФПВ***. Никакой модельной зависимости не предполагается, так что этот алгоритм полностью оправдывает свое название. Если Функция Плотности Вероятности известна, то не сложно определить условное значение **Z** называемое также Регрессией **Z** по **X**.

* Здесь и в дальнейшем, на рисунках горизонтальная ось это направление на восток, а вертикальная на север. Размерность по осям - километры

В данной работе ФПВ будет найдена при помощи непараметрической оценки, впервые предложенной Парзеном. Результирующее уравнение Регрессии будет представлено параллельной нейронной сетью.

Дабы избежать путаницы и перейти к более привычным величинам, введем переобозначения:

X =(*x*,*y*) – пространственные координаты точки измерения

Z(X)=Z(*x*,*y*) – значение измеряемой величины

Теперь, также как и в стереометрии координаты “на плоскости” – “*x*” и “*y*” (входы сети) а “вертикальная” координата – Z (выход сети).

Допустим, что ФПВ уже известна. Тогда, условное значение Z(*x*,*y*) , называемое также регрессией Z по (*x*,*y*) найдется следующим образом [1]:

$$\mathrm{Z} = \frac{\int_{-\infty}^{+\infty} z\varphi(x, y, z)dz}{\int_{-\infty}^{+\infty} \varphi(x, y, z)dz} \quad , \tag{1}$$

Где $\varphi(x, y, z)$ есть функция условной вероятности. Для ее нахождения воспользуемся способом Парзена. Парзен предложил оценивать ФПВ следующим образом [4]:

$$\varphi(x, y, z) = \frac{1}{\left[(2\pi)^{3/2} h^3 n\right]} \sum_{i=1}^{n} \exp\left(-D_i^2 \Big/ 2h^2\right) \exp\left[-\left(\mathrm{Z} - \mathrm{Z}^i\right)^2 \Big/ 2h^2\right] \quad , \tag{2}$$

$$D_i^2 = \left(x - x^i\right)^2 + \left(y - y^i\right)^2 \quad . \tag{3}$$

Подставляя в (1) получим:

$$\mathrm{Z}_m = \frac{\sum_{i=1}^{n} \mathrm{Z}_i \exp\left(-D_i^2 \Big/ 2h^2\right)}{\sum_{i=1}^{n} \exp\left(-D_i^2 \Big/ 2h^2\right)} \quad , \tag{4}$$

$$\mathrm{Z}_m = \frac{\sum_{i=1}^{n} \mathrm{Z}_i w_i}{\sum_{i=1}^{n} w_i} \quad , \tag{5}$$

$$w_i = \exp\left(-D_i^2 \Big/ 2h^2\right) \quad . \tag{6}$$

Итак, по имеющемуся набору из n точек с координатами (x_i, y_i), в каждой из которых известно измеренное значение Z_i, мы строим ФПВ в которую входят как *i*–тые координаты точек измерения, так и переменные *x*,*y* определяющие точку Z_m. Формула (4) дает оценку Z_m. На качество этой оценки влияют как начальные данные, так и независимые параметры, входящие в уравнение.

По определению (1) это условное математическое ожидание Z при данных x и y. В числителе стоит произведение Z на совместную Функцию Плотности Вероятности. Знаменатель это просто нормировочный множитель на тот случай если интеграл по всей области от ФПВ не равен 1. При переходе к (4), интегрирование было заменено суммированием. Затем множитель, содержащий под экспонентой разность Z в точках измерения и Z в искомой точке, был вынесен из числителя и знаменателя. Таким образом, под экспонентой в формуле (4) остается функция, не зависящая от Z, что позволяет использовать ее для компьютерных расчетов. На оценку Z в произвольной точке влияют все начальные (измеренные) данные. Будем рассматривать только числитель (4), так как знаменатель это нормировочный множитель не зависящий от Z. Числитель (4) есть сумма произведений Z *i*–тых на числовой множитель (вес), который меняется от 0 до 1 и зависит от расстояния между *i*–той точкой

начальных данных и той точкой, где производится оценка. Зависимость экспоненциальная, следовательно, оценка для Z в произвольной точке, даваемая по формуле (4), будет обусловлена лишь ближайшими к ней точками начальных данных. Таким образом, Обобщенная Регрессия является ***локальным*** алгоритмом. При получении оценки в точке измерения (точке из набора начальных данных), наибольший вес, равный 1 имеет только эта точка. Если оценивать Z не в точке измерения, то весов равных 1 не будет. Ни один из весов не может ровняться 0, но может стать пренебрежимо малым, если расстояние между точкой измерения и точкой, где дается оценка достаточно велико.

Следует обратить внимание на важную особенность Обобщенной Регрессии. Несложно проверить по формуле (4), что в точке максимума, оценка Z будет строго меньше истинного значения, а в точке минимума – строго больше. Это явление называется ***сглаживанием*** и происходит из–за влияния соседних точек. Поскольку мы имеем дело с ***локальным*** алгоритмом, то оценка Z будет сглаженной и в *локальных* максимумах (минимумах), если они достаточно обособленны. Поэтому в задачах на оценку предельных значений, Обобщенную Регрессию нужно применять очень аккуратно [2,4].

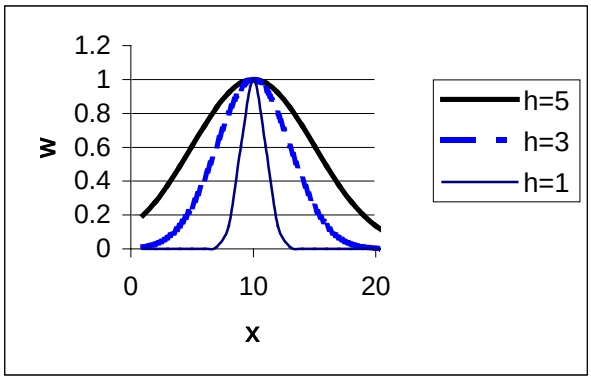


Рис. 2. На вертикальной оси отложен вес точки – w, который находится по формуле (6). Вес точки быстро убывает по мере удаления от нее. При этом h можно назвать характерным маштабом

Кратко обсудив влияние начальных данных, рассмотрим роль независимых параметров уравнения (4). Не теряя общности, ограничимся наиболее простым и наглядным случаем, когда Z=Z(x). Единственным параметром, входящим в уравнение Обобщенной Регрессии является *h*. От *h* зависит область влияния каждой точки. Как видно из рис. 2, *h* это характерный поперечный размер экспоненциальной “шапки” над точкой измерения. Под шапкой здесь понимается вид функции *w(x)* даваемый формулой (6). Оценочное значение для Z, даваемое формулой (4), есть просто сумма всех этих “шапок”, деленная на нормировочный множитель см. рис. 3. На рис. 3. Также приведена ненормированная оценка Z, получаемая по формуле (4) со знаменателем равным единице.

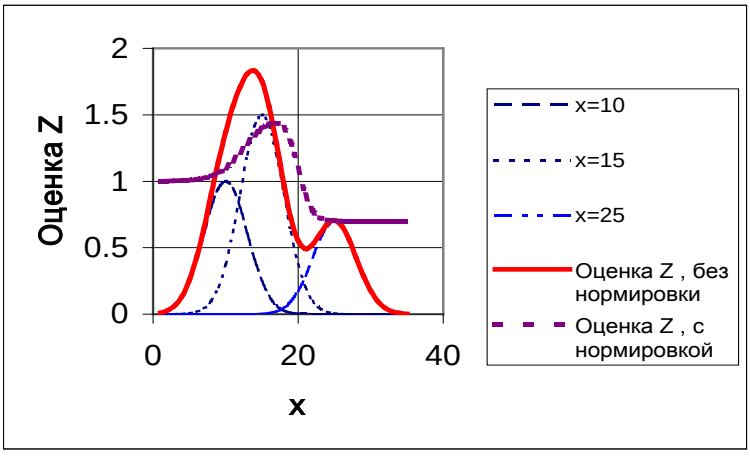


Рис. 3. Оценка Обобщенной Регрессией функции Z по трем точкам: x=10, x=15, x=25. Все вычисления были проведены при h = 3

Если в какой-нибудь области ненормированная оценка Z не совпадает с “шапкой” то это значит, что окончательная оценка Z (4) в каждой точке этой области будет определятся не одной, а несколькими точками значения Z в которых известны. Можно сказать, что оценка Z в точке *x* зависит от значений Z в тех точках, “шапки” или веса которых в точке *x* не равны нулю.

Очень важно, что правее и левее трех точек начальных данных, формула (4) дает константу не равную 0 (см. рис. 3). Это явление называется ***смещением***. Для того, чтобы понять как оно возникает, перепишем (4) для трех точек:

$$\mathrm{Z}_m = \frac{\mathrm{Z}_1 \exp\left(-D_1^2 \big/ 2h^2\right)}{\sum_{i=1}^{n} \exp\left(-D_i^2 \big/ 2h^2\right)} + \frac{\mathrm{Z}_2 \exp\left(-D_2^2 \big/ 2h^2\right)}{\sum_{i=1}^{n} \exp\left(-D_i^2 \big/ 2h^2\right)} + \frac{\mathrm{Z}_3 \exp\left(-D_3^2 \big/ 2h^2\right)}{\sum_{i=1}^{n} \exp\left(-D_i^2 \big/ 2h^2\right)} \quad . \tag{7}$$

Далее,

$$\mathrm{Z}_m = \frac{\mathrm{Z}_1}{1 + \exp\left(-D_2^2 + D_1^2 \big/ 2h^2\right) + \exp\left(-D_3^2 + D_1^2 \big/ 2h^2\right)} +$$

$$+ \frac{\mathrm{Z}_2}{1 + \exp\left(-D_1^2 + D_2^2 \big/ 2h^2\right) + \exp\left(-D_3^2 + D_2^2 \big/ 2h^2\right)} +$$

$$+ \frac{\mathrm{Z}_3}{1 + \exp\left(-D_1^2 + D_3^2 \big/ 2h^2\right) + \exp\left(-D_2^2 + D_3^2 \big/ 2h^2\right)} \quad . \tag{8}$$

Пусть $x_1 < x_2 < x_3$, рассмотрим произвольную точку, лежащую значительно левее x_1:

$$\Rightarrow D_1 > D_2 > D_3 \Rightarrow \exp\left(-D_1^2 + D_2^2 \big/ 2h^2\right) + \exp\left(-D_3^2 + D_2^2 \big/ 2h^2\right) \to \infty,$$

$$\exp\left(-D_1^2 + D_3^2 \big/ 2h^2\right) + \exp\left(-D_2^2 + D_3^2 \big/ 2h^2\right) \to \infty,$$

$$\exp\left(-D_2^2 + D_1^2 \big/ 2h^2\right) + \exp\left(-D_3^2 + D_1^2 \big/ 2h^2\right) \to 0.$$

$$\Rightarrow \mathrm{Z}_m \underset{x \to -\infty}{\to} \mathrm{Z}_1 \quad . \tag{9}$$

Из–за очень быстрого спадания экспоненты, на достаточно небольших расстояниях от “крайней” точки начальных данных см рис. 4, в вышеприведенных формулах можно заменить знак “$\to$”, знаком “=”. На рис. 4 хорошо видно, что оценка Z выходит на константу в том месте, где исчезает влияние всех точек множества начальных данных, кроме крайней. Если в наборе начальных данных только одна точка $\left(x_1, \mathrm{Z}_1\right)$, то формула (4) дает одно единственное значение для любого x , равное как раз Z_1:

$$\mathrm{Z}_m = \frac{\mathrm{Z}_1 \exp\left(-D_1^2 \big/ 2h^2\right)}{\exp\left(-D_1^2 \big/ 2h^2\right)} = \mathrm{Z}_1 \ .$$

Таким образом, результат (8) вполне закономерен.

При бесконечном увеличении *h*, оценка даваемая по формуле (4) стремится к математическому среднему по набору начальных данных и престает меняться (см. рис. 4).

$$\lim_{h\to\infty}(Z_m) = \lim_{h\to\infty} \frac{\sum_{i=1}^{n} Z_i \exp\left(-D_i^2 / 2h^2\right)}{\sum_{i=1}^{n} \exp\left(-D_i^2 / 2h^2\right)} = \frac{\sum_{i=1}^{n} Z_i}{n} = const \qquad (10)$$

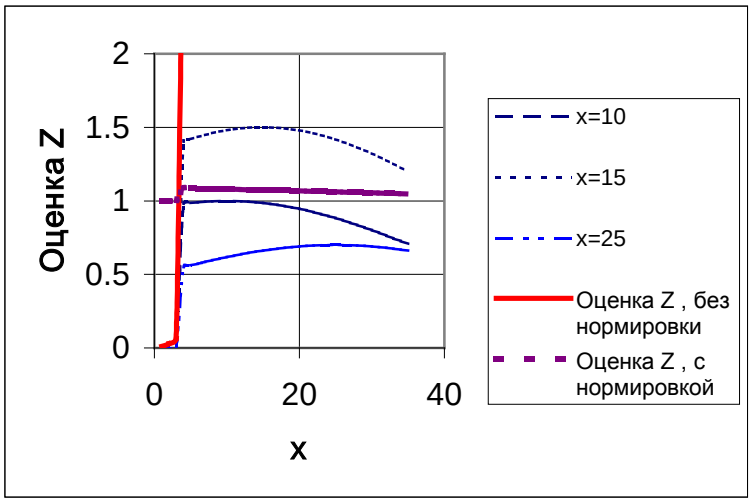


Рис. 4. Оценка Z для случая h=30

При стремлении *h* к 0, “шапки” над точками начальных данных становятся близкими к дельта–функциям. Оценка Z превращается в разрывную (в пределе) функцию, состоящую из отрезков, параллельных оси x и лежащих на высотах, равных значениям Z в точках начальных данных (см. рис. 5). На рисунке 4 не приведены функции *w(x)* для точек начальных данных по той причине, что ненормированная оценка Z с ними полностью совпадает.

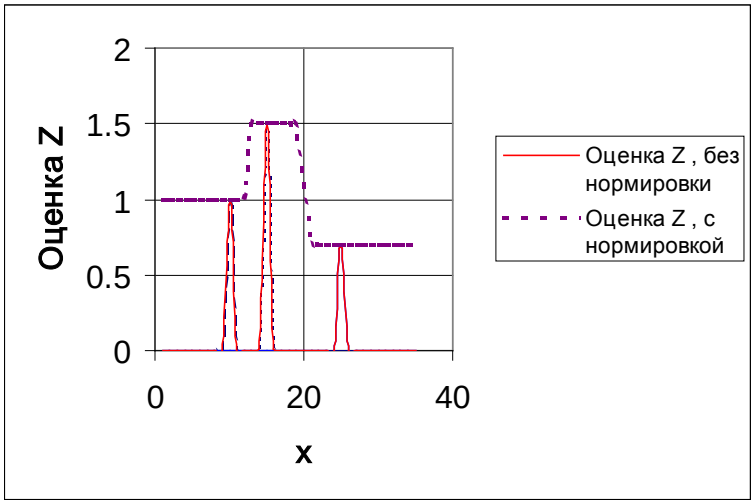


Рис. 5. На рисунке изображена оценка Z для случая h=0.3

В точках измерения получается точное значение Z. Оптимальное значение *h* лежит где–то между двумя вышеуказанными крайностями. При его нахождении, нужно помнить, что, уменьшая *h*, оценка становится менее гладкой, то есть повышается ее вариабильность. При увеличении *h*, оценка, стремясь к константе, становится все более и более гладкой, но уравнение (4) начинает все хуже и хуже отслеживать мелкомасштабные закономерности в наборе начальных данных. Все это приводит к росту ошибки (здесь и далее под ошибкой следует понимать средне квадратическую ошибку). Об алгоритмах нахождения

оптимального h мы поговорим в следующей главе, которая повествует о том, что такое Искусственные Нейронные Сети вообще и, в частности, что такое Искусственные Нейронные Сети с Обобщенной Регрессией.

3 Искусственные Нейронные Сети с Обобщенной Регрессией

3.1 Общая концепция Нейронных Сетей

Нейронные сети это параллельная распределяющая информацию структура, состоящая из элементарных элементов – нейронов (которые могут обладать локальной памятью и производить операции над локальной информацией), связанных посредством однонаправленных каналов называемых связями [1]. Связь просто умножает приходящий сигнал на определенный коэффициент и переправляет его к следующему нейрону. Нейрон представляет из себя ядро (выполняющее простую математическую операцию – действие активационной функции на сумму входов нейрона и выдачу результата на выходы) и собственно входы и выходы, количество которых может изменяться в зависимости от архитектуры сети. Важно, что входные сигналы нейрона могут быть различными, тогда как выход один и тот же для всех выходных связей. Нейроны обрабатывают любую самую произвольную информацию, единственным ограничением является ее полная локальность, то есть она должна зависеть только от значений тех величин, которые поступают на вход нейрона и хранятся в его локальной памяти. Работу каждого нейрона можно представить с помощью простой формулы:

$OUT = \sum_{k=1}^{l} f\left(w_k \bullet input_k\right)$ – выход нейрона

w_k – веса связей ,

$input_k$ – входы нейрона,

$f(\ )$ – активационная функция

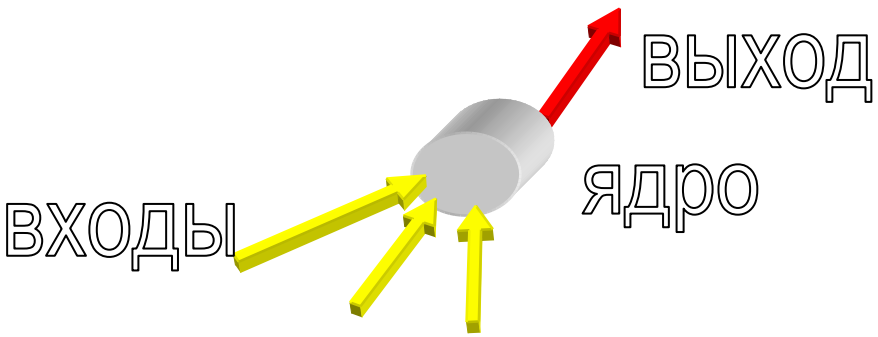


Рис. 6. Общий вид нейрона

При построении нейронных сетей нейроны обычно располагают слоями. Слой, на который подаются исходные данные, называется входным слоем. От этого слоя ведется нумерация. Номер входного слоя – 1. Самый последний слой называется выходным слоем сети. Слои между входным и выходным называются скрытыми. Связи и нейроны в каждом слое работают одновременно. Таким образом, работу Искусственной Нейронной Сети (ИНС) можно разбить на этапы – по числу слоев и рассматривать эти этапы по отдельности.

Выше, в самых общих чертах, была описана конструкция ИНС. Теперь несколько слов о том, каково их применение.

Основные области применения ИНС

- Распознавание образов. Распознавание и генерация речи.
- Временные ряды.
- Аппроксимация функций.
- Ассоциативная память.
- Оптимизация.
- Компьютерное зрение.
- Регрессия.

ИНС имеют преимущества перед прочими методами, если

- Данные чрезвычайно зашумлены или измерения содержат большие ошибки.
- Данные представляют сложные и непредсказуемые нелинейные зависимостями.
- Данные хаотичны в математическом смысле.
- ИНС способны грамотно анализировать, казалось бы, совершенно невероятные данные.

Работу с ИНС можно разбить на три этапа: ***обучение***, ***валидация***, ***обработка новых данных***. ИНС способна эффективно находить закономерности в обрабатываемой информации, для оптимизации этого процесса служат два первых этапа. Обычно, тех, кто использует ИНС, можно разделить на заказчиков и исполнителей. Заказчик объясняет исполнителю задачу, которую тот должен решить, и снабжает некоторым набором данных. Исполнитель должен предоставить заказчику готовый продукт (ИНС), с тем, чтобы заказчик мог его использовать без участия исполнителя. Следовательно, исполнитель полностью контролирует первые два этапа. После того как исполнитель находит работу ИНС удовлетворительной, все параметры определяющие сеть фиксируются и она, более не совершенствуясь, только обрабатывает информацию.

Рассмотрим подробнее этапы работы ИНС:

Обучение – на этом этапе сеть “знакомится” с данными определяющими задачу, для решения которой эта сеть создается и адаптируется для максимально корректной работы с ними. ИНС имеет ряд определяющих параметров (число нейронов, их расположение, активационные функции каждого нейрона, веса связей), которые варьируются в процессе обучения с целью уменьшить ошибку на выходе сети. Данные, которыми исполнитель располагает изначально, делятся на два набора: на ***тренировочный набор*** и на ***валидационный набор***. И тот и другой совершенно одинаковы по своей структуре, они оба состоят из некоторого количества пар (входы, выходы), одну такую пару мы будем называть ***примером***. В данной работе входов всего два, это две координаты точки измерения на плоскости (x,y) – независимые переменные. Выход только один – величина загрязнения ^{137}Cs Z – зависимая переменная. Во время обучения используется тренировочный набор. Таким образом, на входы сети подаются координаты (x,y), а выход сети – предсказанное значение загрязнения в этой точке, сравнивают с измеренным в этом месте значением, которое содержится в той же паре (входы, выходы) тренировочного набора. Если расхождение значительно, то параметры сети подвергаются изменению. То, какие это параметры, будет зависеть от алгоритма обучения.

Валидация – после того как ИНС была обучена, необходимо проверить, хорошо ли она может обобщать данные. Дело в том, что в процессе обучения сеть должна запомнить только закономерности, которые имелись в тренировочном наборе, но ИНС не должна его “выучивать”. Явление “выучивания” тренировочного набора или иначе говоря потеря сетью способности обобщать данные, называется ***перетренировка***. Это явление очень важно и мы остановимся на нем подробнее. В любые измерения обязательно входит ошибка: $Z = Z_{true} + \xi$, где Z – результат измерения, Z_{true} – истинное значение, а ξ – ошибка его измерения.

Хорошо, если ошибка измерения пренебрежимо мала. Но если это не так, то ни одно из измеренных значений не может быть признано точным. Поэтому, если выход сети точно совпадает со значением из тренировочного набора, значит, в нем присутствует ошибка равная ошибке измерения значения в этой точке. На рис. 7, исходная истинная зависимость Z(x) представлена прямой линией без маркеров, перетренированая сеть дает оценку проходящую через точки тренировочного набора (они помечены маркерами). Перетренированая ИНС почти в каждой точке дает неправильную оценку, так как разность между истинным и оцененным значением почти всюду сильно отлична от нуля. Величины этих

разностей в приведенном примере, возможно, будут несущественны. Но никакие реальные данные не застрахованы от аномалий. Картина резко меняется, если в какой–то точке появится “выброс” – рис. 8. Область вокруг точки выброса оценивается переученной ИНС с очень большой ошибкой, в то время как правильно обученная “хорошая” сеть дает лишь небольшое отклонение сравнимое в данном случае с ошибкой измерения. Следует обратить внимание на еще одну деталь. Наличие выброса сказалось только на близлежащей области, так как в ИНС, с помощью которой был получен рис. 8. использовался локальный математический алгоритм.

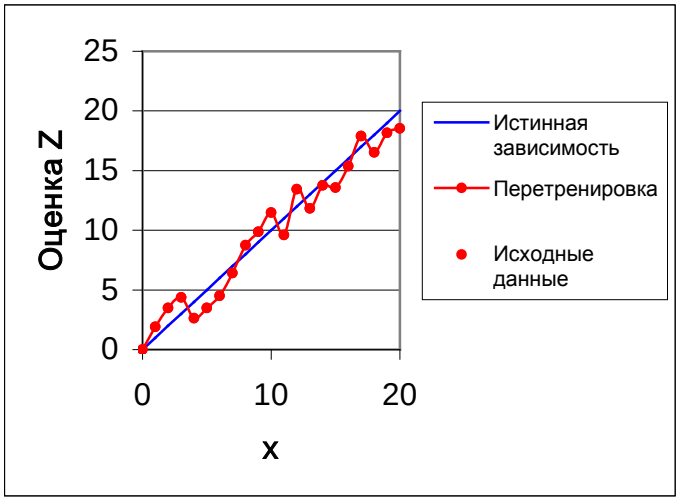


Рис. 7. Оценка перетренерованной сети проходит через точки начальных зашумленных данных. Оценка правильно обученной сети совпадает с истинной зависимостью Z(x)

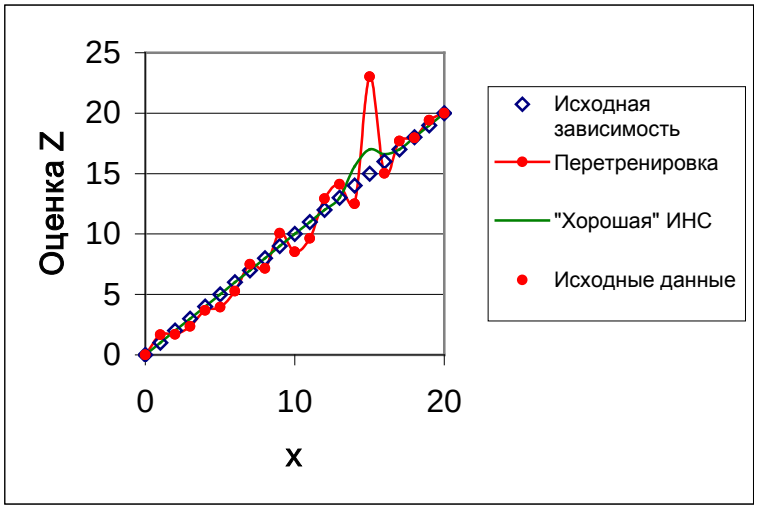


Рис. 8. Перетренированая сеть незначительно ошибается почти во всех точках. Но в точке выброса, ошибка уже сравнима с истинным значением Z(x). Правильно обученная “хорошая” сеть реагирует на выброс в гораздо меньшей степени

Итак, задача валидации – предотвратить перетренировку. Для этой цели используется валидационный набор данных, структура которого полностью совпадает со структурой тренировочного набора. Глубокое отличие между этими наборами в том, что в процессе обучения сеть не “знает” о существовании валидационного набора. В процессе обучения ИНС устанавливает для себя вид связи между входами и выходами, а валидация – первая проверка этой зависимости. Ошибка на валидационных данных, как правило, больше из-за того, что сеть, не смотря ни на что хоть немного, но выучивает начальные данные. Это, к сожалению неизбежно, так как других данных у сети нет, она приспосабливается к тренировочному набору. Но если эта ошибка приемлема, то параметры сети замораживаются и с ней начинают работать на новых данных. В противном случае нужно вернутся на этап обучения или еще дальше (поменять конфигурацию сети).

Можно контролировать способность сети обобщать данные и на этапе обучения (то есть до валидации). Для этого из тренировочного набора извлекается часть примеров. Образованный таким образом новый набор называется ***тестовым***, оставшиеся после извлечения данные образуют новый тренировочный набор. Обучение сети в этом случае можно разбить на два подэтапа. Вначале ИНС обучается только на примерах из нового тренировочного набора. После того как найдутся оптимальные параметры сети, к новому тренировочному набору добавляют один пример из тестового. Затем следует оптимизация параметров сети, после чего добавляется еще один пример из тестового набора и так далее, до тех пор, пока из тестового набора не будут использованы все данные. Отличие этого способа от валидации в том, что примеры тестового набора обрабатываются алгоритмом обучения ИНС, в то время как валидационный набор применяется для разовой проверки уже обученной сети.

Обработка новых данных – это последний этап, ради которого отдается столько сил во время обучения и валидации. После того, как исполнитель “максимально” используя данные, предоставленные заказчиком, сочтет работу сети приемлемой, он передает ее в руки заказчика и, следовательно, уже не может влиять на правильность ее работы. Заказчик же будет использовать полученную ИНС на любых данных касающихся вышеуказанной задачи. Значит, сети придется работать с данными, которые она “не видела” ни при обучении, ни при валидации. ИНС в руках заказчика пройдет как бы валидацию №2, но ее результаты уже не будут использованы для дальнейшего улучшения сети. Поэтому так важно избежать перетренировки.

Ко всему вышесказанному можно добавить, что сами по себе ИНС не являются каким–то физическим объектом. Они лишь помогают при визуализации того или иного алгоритма. С их помощью гораздо проще понять, как устроены некоторые, порой очень сложные, математические методы. ***Таким образом, своими прекрасными показателями работоспособности, ИНС обязаны, прежде всего, мощной математической базе.***

Теоретический обзор Обобщенной Регрессии уже был приведен выше, перейдем к Искусственным Нейронным Сетям с Обобщенной Регрессией.

3.2 Нейронные Сети с Обобщенной Регрессией

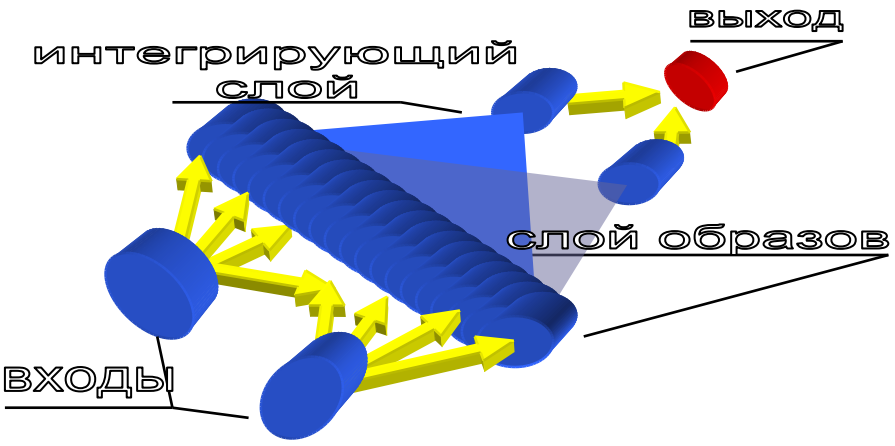


Рис. 9. Общий вид НСОР

Конфигурация ИНС зависит не только от выбора математического метода, но и от конкретной задачи. Метод уже выбран, а задача такова: используя определенное количество точек в Брянской области, в которых проведены замеры на предмет загрязнения ^{137}Cs, “научить” НСОР как можно точнее предсказывать величину загрязнения в любой другой точке принадлежащей Брянской области. Это означает, что, получив на вход координаты *x,y* НСОР должна по формуле (4) дать (используя оптимальное *h*) оценку загрязнения – Z в этой точке.

Архитектура НСОР продиктованная этой задачей очень проста (см. рис. 9). Входной слой содержит два нейрона, по одному на каждую из координат точки. Выход НСОР это всего лишь один нейрон, так как оценивается только одна переменная – Z. Между входом и выходом имеется два скрытых слоя. Первый из них называется слоем образов и содержит столько нейронов, сколько примеров в тренировочном наборе и называется – ***слой образов***. Во втором скрытом слое (он называется ***интегрирующим***) всего два нейрона. Действует НСОР так: сначала в слое образов запоминается тренировочный набор. Один нейрон слоя образов запоминает один пример из тренировочного набора

(только координаты x, y). При этом вес соответствующей связи, ведущей к одному из двух нейронов второго скрытого слоя, мы будем называть этот нейрон ***нумератором***, устанавливается равным значению Z в точке, координаты которой запомнил соответствующий нейрон слоя образов. Веса связей, ведущих к оставшемуся нейрону интегрирующего слоя (***денумератору***) и все оставшиеся веса, устанавливаются равными единице. Веса всех связей остаются постоянными. После "запоминания" все готово к обучению. Нейроны входного слоя доставляют каждому нейрону слоя образов координаты x и y. Каждый нейрон слоя образов находит расстояние от точки, поданной на вход НСОР до точки, координаты которой он запомнил:

$$D_i^2 = \left(x - x^i\right)^2 + \left(y - y^i\right)^2$$

Затем, действует на него экспоненциальной функцией:

$$\exp\left(-D_i^2 \Big/ 2h^2\right)$$

Переходим к интегрирующему слою. Выходы нумератора и денумератора равны суммам их входов. В результате выход нумератора будет равен:

$$\sum_{i=1}^{n} Z_i \exp\left(-D_i^2 \Big/ 2h^2\right) \quad .$$

Выход денумератора:

$$\sum_{i=1}^{n} \exp\left(-D_i^2 \Big/ 2h^2\right) \quad .$$

Нейрон выходного слоя делит первое из этих значений на второе, так что окончательный выход НСОР полностью совпадает с формулой (4).

Так как веса всех связей остаются постоянными, то единственным параметром НСОР является *h*. Следовательно, описанная выше модель математически полностью идентична методу Обобщенной Регрессии.

$$OUT_{HCOP} = \frac{\sum_{i=1}^{n} Z_i \exp\left(-D_i^2 \Big/ 2h^2\right)}{\sum_{i=1}^{n} \exp\left(-D_i^2 \Big/ 2h^2\right)}$$

3.3 Обучение НСОР

Обучение сети это нахождение оптимального значения *h*, используя исходные данные. Выбор алгоритма обучения зависит как от поставленной задачи, так и от типа ИНС. Для НСОР чаще всего используются ***генетические алгоритмы*** или ***кросс-валидация***. С генетическими алгоритмами можно подробно ознакомиться, прочитав [5]. Кросс-валидация прекрасно описана в [3].

4 Описание работы

4.1 Предмет исследования

После теоретического описания модели всегда полезно изучить практическую сторону. Далее, речь пойдет только о конкретном случае применения НСОР для решения конкретной задачи.

Исходными данными, о которых мы поговорим чуть позже, являются сведения о пробах почвы в 665– и различных точках области. Задача состояла в исследовании работы НСОР на этих (достаточно сложных) данных. В цели данной работы входило проведение пространственной интерполяции и анализ полученных результатов, в частности зависимость "качества" интерполяции от размера тренировочного набора, анализ ***невязок*** – разностей между измеренным и предсказанным значением в одной и той же точке, исследование того, как НСОР реагирует на "выбросы", влияние величины *h* и зависимость результата интерполяции от особенностей тренировочного и валидационного наборов.

4.2 Начальные данные

Из предоставленных 665–и точек, было собрано 6 пар наборов (тренировочный + валидационный). Все наборы кроме 200д – результаты случайных выборок. Набор 200д это выборка с декластеризацией. В дальнейшем мы будем указывать название набора, и называть ту его часть (валидационную или тренировочную) которую хотим рассмотреть. Ниже на рис. 10 изображена тренировочная часть набора 300 и, рядом, все 665 точек.

Таблица 1.
Разбиение исходных данных на тренировочный и валидационный наборы

№ набора	Условное обозначение набора	Тренировочный набор (кол–во точек)	Валидационный набор (кол–во точек)
1	100	100	565
2	200	200	465
3	200д	200	465
4	300	300	365
5	400	400	265
6	500	500	165

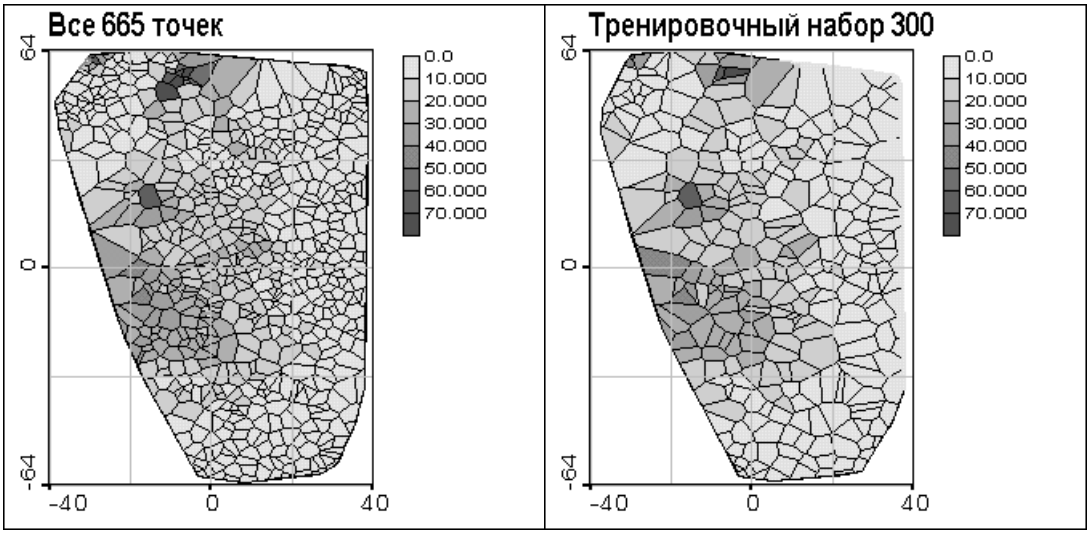


Рис. 10. Карты всех данных и тренировочного набора 300

4.3 Применение НСОР

Для каждого набора при помощи программы **Neuro Shell 2** была построена своя сеть. Обучение сетей проводилось методом, использующим генетические алгоритмы. Затем следовала проверка на валидационном наборе. То же было проделано и для 6-и наборов невязок. Результаты работы сетей исследовались при помощи программы **GeoStat Office** [10]. Были проведены вариография и статистический анализ.

5 Блок-схема работы

Для большей наглядности ход работы представлен на рисунке 11. Рассмотрим его последовательно:

1. Данные делятся на две группы: валидационные данные и данные для обучения.
2. Данные для обучения делятся на тренировочные и тестовые образуя, таким образом наборы 100, 200, 200д, 300, 400 и 500.
3. Для каждого набора данных стоится НСОР соответствующей конфигурации.
4. НСОР обучаются.
5. Обученные НСОР применяются к валидационным данным и данным для обучения.

6. Собирается статистика, по которой определяется качество работы НСОР.
7. Далее, следует очень важный этап – вариография невязок.
8. После оценки качества выдается окончательный прогноз.
9. Для большей наглядности результатов, все НСОР применялись к равномерной прямоугольной сетке.

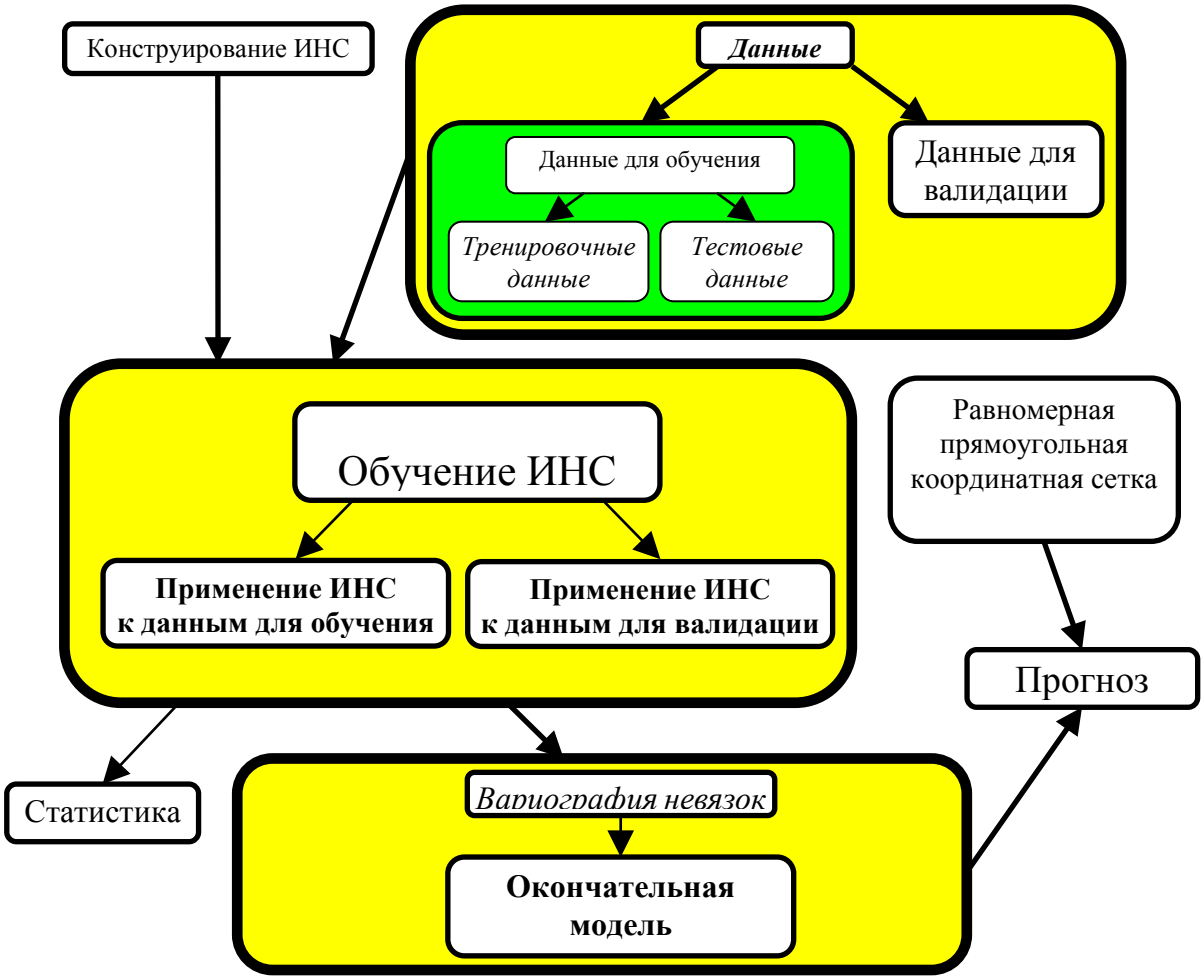


Рис. 11. Блок-схема работы

6 Анализ результатов

6.1 Пространственный структурный анализ (вариография)

Основным инструментом исследования в этой работе является вариография, перед обсуждением результатов, мы кратко напомним, что же это такое (приведенное ниже описание вариографии ни коем образом не претендует на полноту, более того описывается только та часть вариографии, которая имеет отношение к данной работе). Вариография используется для того, чтобы определить имеется или нет какая либо структурная зависимость в исследуемом наборе данных [3]. При помощи вариографии можно определить характерный размер областей, в которых указанная зависимость наблюдается. Предметом анализа вариографии является функция [3]:

$$\gamma(l) = \frac{1}{N(l)} \sum_{i=1}^{N(l)} \left(\mathrm{Z}(x_i) - \mathrm{Z}(l + x_i) \right)^2 \quad . \tag{11}$$

Эта функция называется вариограммой. По ее поведению можно судить о наличии или отсутствии корреляционной структуры в исследуемых данных. Если корреляционная структура имеет место, то по вариограмме можно судить о ее масштабе. Подробнее о вариографии можно узнать из [3].

6.2 Обработка исходных данных

Вначале поговорим о том, какие *h* * оказались оптимальными во время обучения и валидации и какие получались ошибки (под ошибкой здесь и далее, если не указано дополнительно, будет подразумеваться среднеквадратическая ошибка). Напомним, что обучались сети при помощи генетических алгоритмов [5]. Лучшее валидационное *h* находилось из условия минимизации среднеквадратической ошибки. В таблице 2 приведены оптимальные значения *h* для обучения и валидации:

$$RMSE = (1/N)\sqrt{\sum_i (Z\_net - Z)^2}$$ – среднеквадратическая ошибка.

Первый вывод, который можно сделать, достаточно очевиден, валидационная ошибка больше ошибки при обучении, как для каждой сети, так и в целом. Значит, как и отмечалось ранее, НСОР “немного выучивает” начальные данные, уменьшая при этом ошибку на валидационном наборе. Далее, следует обратить внимание на то, что для большинства наборов: 500, 400, 300 и 200 лучшее валидационное *h* больше оптимального *h* при обучении. Это тоже происходит из-за слабого выучивания тренировочных данных. Действительно, оптимальное значение *h* при обучении (с точки зрения минимизации среднеквадратической ошибки) равно нулю. В этом случае оценка НСОР, как было показано раньше, будет совпадать с соответствующими значениями начальных данных. При валидации график зависимости RMSE от *h* имеет один четкий минимум – рис. 12. Следовательно, оптимальное *h* для валидации определяется однозначно. В то время как в процессе обучения оптимальное *h* смещается в сторону меньших значений.

Таблица 2.
Ошибки обучения и валидации

Название набора данных	Обучение		Валидация	
	h	RMSE	*h*	RMSE
100	0.11	3.3800	0.07	8.6253
200	0.037176	3.7819	0.13	7.1677
200д	0.142	7.5696	0.021	5.9682
300	0.025529	3.5037	0.021	6.3911
400	0.037176	2.3030	0.057	6.7372
500	0.037176	4.2433	0.043	6.9773

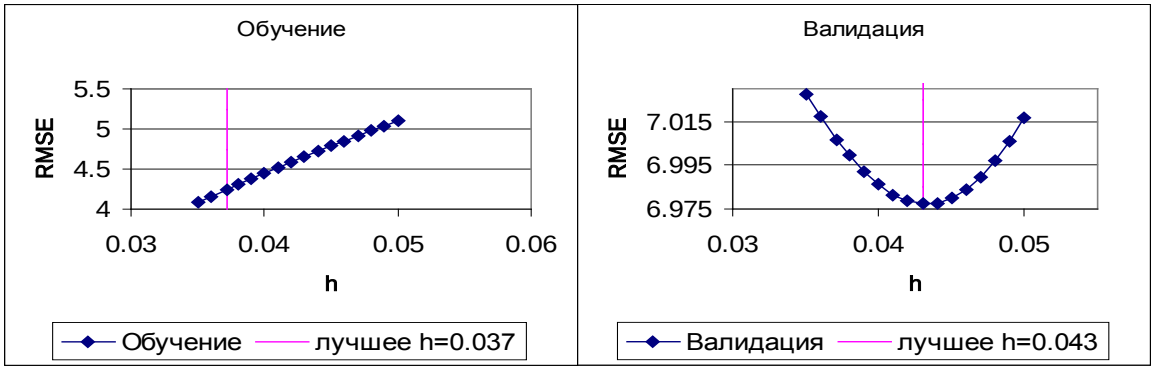


Рис. 12. Зависимость RMSE от параметра h для обучения и валидации

* О соответствии между единицами измерения *h* и километрами см. стр. 33.

Что касается двух наборов 100 и 200д, то они выпадают из общей тенденции. Это свидетельствует о недостаточной устойчивости алгоритма обучения сети. Результаты обучения гораздо менее важны, по сравнению с результатами валидации, поэтому далее в основном мы будем говорить именно о валидации.

До применения НСОР, можно было сделать простое и, казалось бы, правильное предположение: чем больше тренировочный набор, то есть чем больше у сети данных о поставленной задаче, тем лучше сеть с этой задачей справится. Но как видно из таблицы 2, это не совсем так. В данном случае все зависит в основном от внутренних особенностей как тренировочного так и валидационного наборов. Рассмотрим лучшую НСОР обучавшуюся на тренировочном наборе 200д и НСОР с тренировочным набором 500. Результатом декластеризации стало то, что в тренировочный набор 200д вошло большинство характерных точек (выбросов) из общего набора данных. Более того, в после декластеризации в тренировочный набор 200д вошли почти все точки определяющие разномасштабную пространственную структуру данных. Следовательно, в валидационный набор 200д вошли в основном точки с наиболее часто встречающимися значениями – точки лежащие в зонах гладкости Z(x,y). В случайно выбранный набор тренировочный набор 500, вошли далеко не все выбросы. Вклад вышеуказанных обстоятельств в большую разницу ошибок значительно усиливается тем, что размер валидационного набора 200д – 465 точек, тогда как в валидационном наборе 500 их всего 165 . То есть множество точек с маленькими ошибками в 200д “пересиливают” несколько сильных выбросов (см. рис. 13). В случае 500 из–за малого размера валидационного набора, вклад выбросов в ошибку гораздо существенней. Аналогичная тенденция прослеживается и для наборов 300 и 400. То есть ошибка на валидации растет с увеличением тренировочного набора. Это происходит из-за того, что НСОР получает избыток информации во время обучения, действительно, ведь корреляционная структура невязок пропадает уже на валидационном наборе 300. Выборки 400 и 500 случайны, поэтому не несут дополнительной информации а лишь сглаживают оценку.

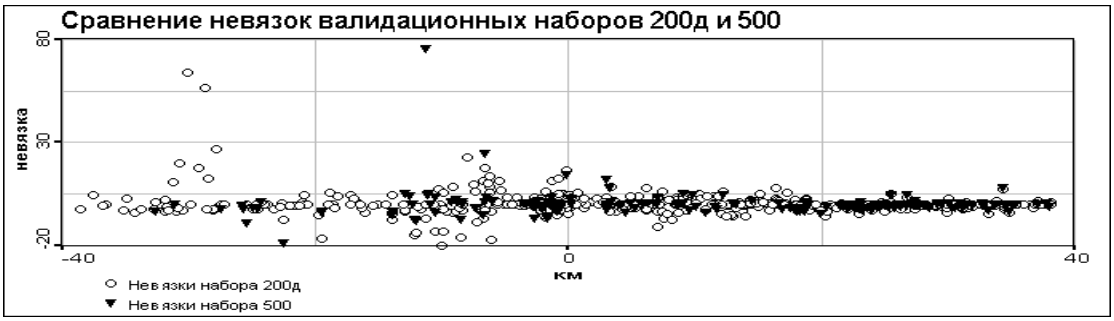


Рис. 13. Величины невязок для двух НСОР обучавшихся на наборах 200д и 500, спроектированные на x

Выше, о выбросах говорилось как об характерных точках, так как выброс всегда приводит к большой ошибке. Это уже было обосновано теоретически в главе посвященной Обобщенной Регрессии. Применение НСОР дало и практическое подтверждение данного обстоятельства. Достаточно взглянуть на картину невязок (см. рис. 14, 14а и 14б). Большие значения на левом рисунке, показывающем исходные данные, соответствуют большим невязкам на правом рисунке. В точках с небольшими значениями на невязки могут быть отрицательными. Особенно наглядны рисунки 14 и 14а. На них изображены невязки, полученные при оценке НСОР валидационного набора 300 и 100, относительно значения Z.

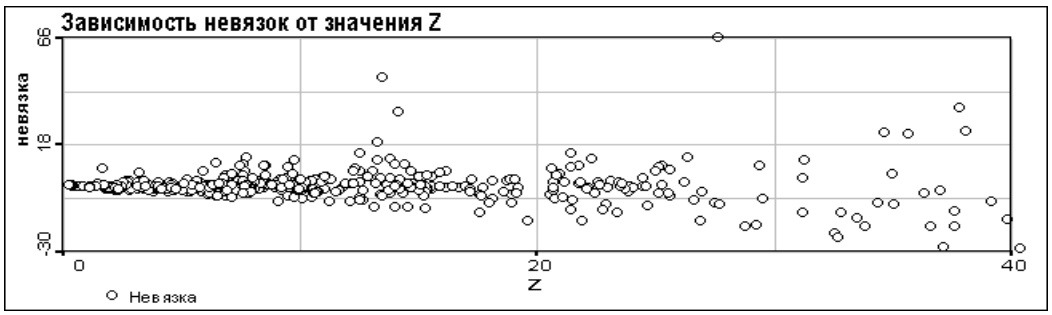


Рис. 14. Завимсимость невязок валидационного набора 300 от значения Z

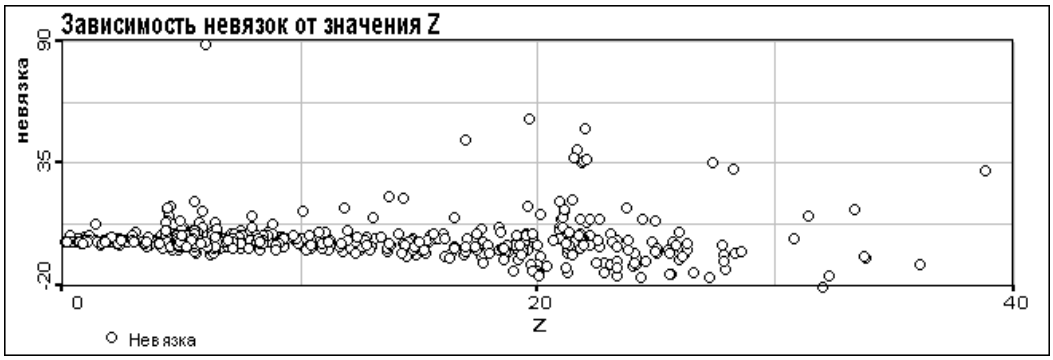


Рис. 14а. Завимсимость невязок валидационного набора 100 от значения Z

Рис. 14б. Полигоны Вороного невязок валидационного набора 300 (слева) и сам набор 300(справа)

Хотя увеличение тренировочного набора и не привело к улучшению результатов работы НСОР, с чисто визуальной точки зрения, в оценке даваемой НСОР появляются новые более мелкие детали. Особенно наглядно выглядит прогноз сети на регулярной сетке (см. рис. 15). В приложении содержатся карты невязок на валидации, карты невязок на тренировке и карты оценок для всех наборов.

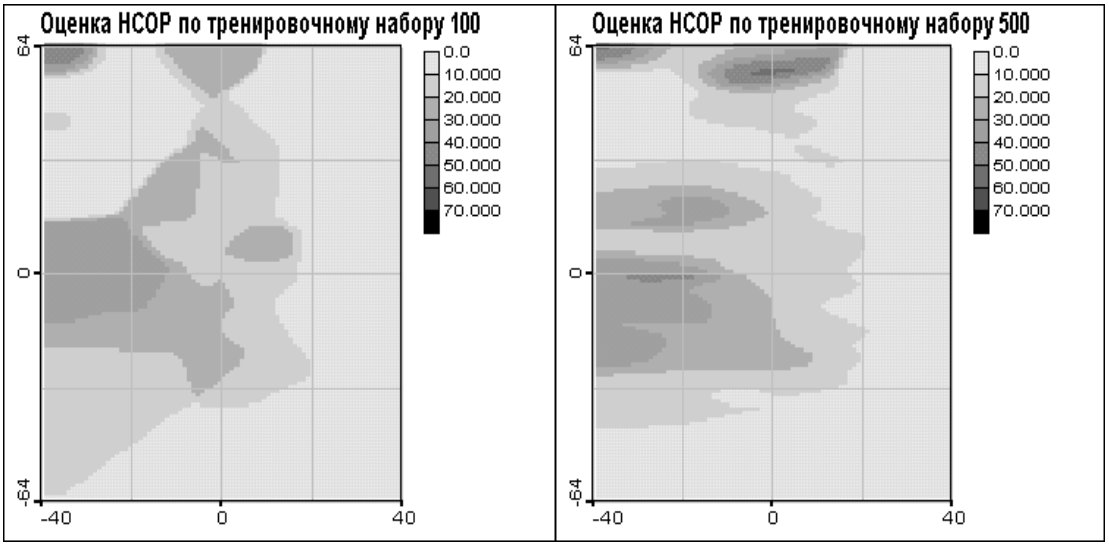


Рис. 15. Оценки НСОР обучавшихся на тренировочных наборах 100 (слева) и 500 (справа) на равномерной сетке

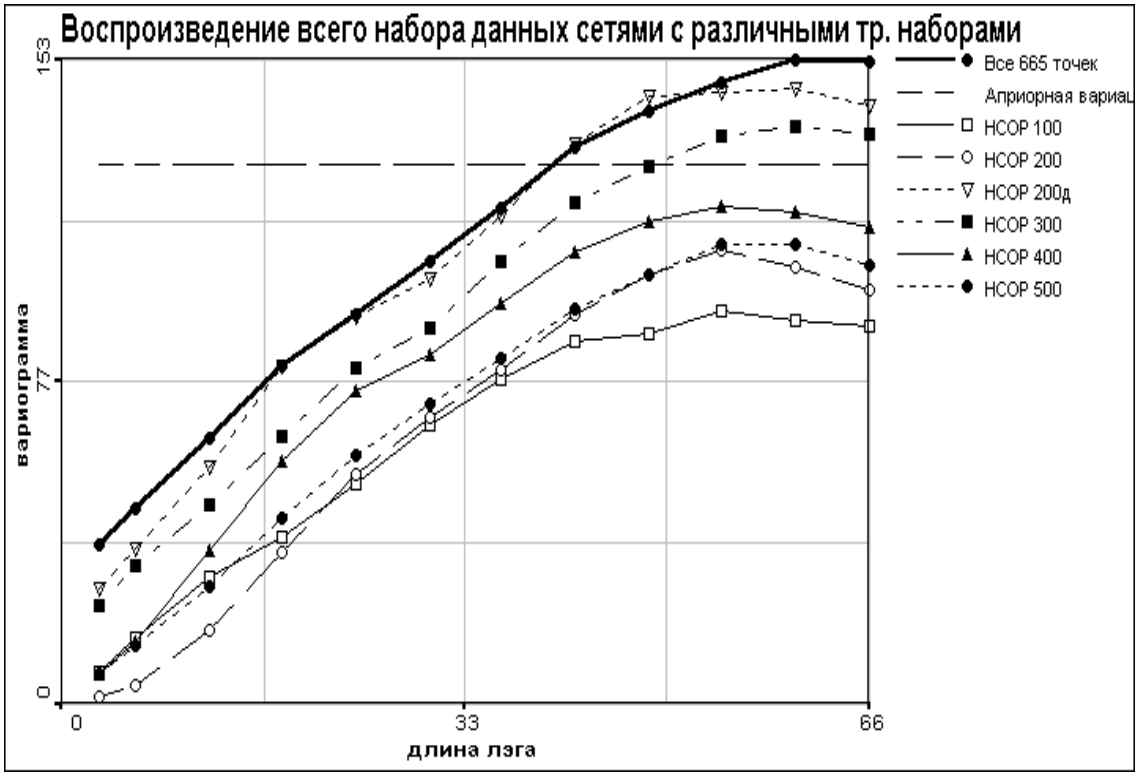


Рис. 16. Вариограммы оценок для всего набора данных различными НСОР

От визуализации результатов можно перейти к крупно и мелко масштабным зависимостям в данных то есть к вариографии. Вначале посмотрим, как воспроизводится вариограмма всех 665 точек – рис.16.

По ее виду можно предположить, что валидационная и тренировочная ошибки набора 500 будут больше соответствующих ошибок других наборов. Аналогичные вариограммы можно построить для данных обучения и валидации. Далее, мы в основном будем рассматривать только три набора 100, 200д и 400. НСОР, обучавшаяся на тренировочном наборе 100 дала самые плохие результаты на валидации. НСОР с набором 200д наоборот показала лучшие результаты. Оценки сети с набором 400 качественно ни чем не отличаются от результатов сетей использовавших наборы 300 и 500.

На рисунках 17, 18, и 19 построены вариограммы оценок различных НСОР в точках соответствующих тренировочных наборов. И для сравнения и контроля качества обучения там же приведены и вариограммы самих тренировочных наборов и их априорная вариация. Все вариограммы, в общем, похожи можно лишь сказать, что из-за большого количества точек, тренировочный набор 400 был оценен несколько хуже.

Рис. 17. Вариограмма оценок для тренировочного набора 100 соответствующей НСОР

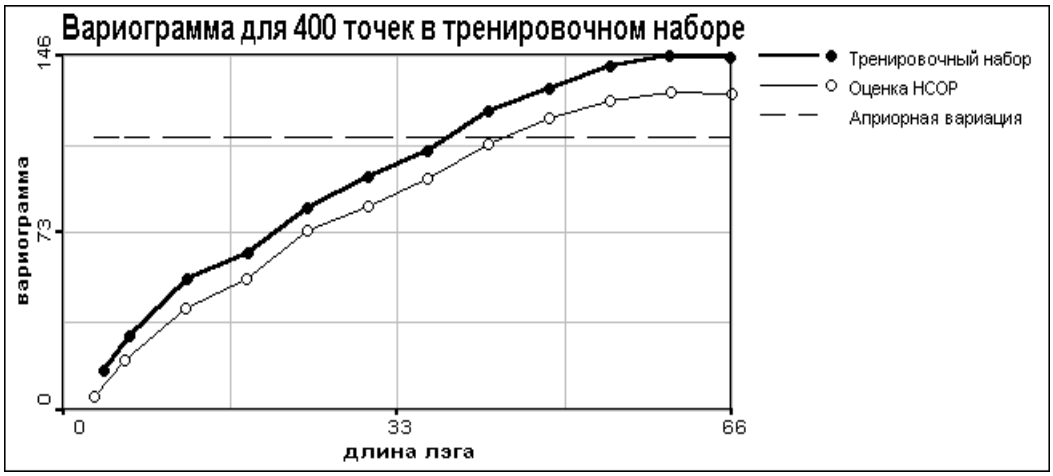


Рис. 18. Вариограмма оценок для тренировочного набора 400 соответствующей HCOP

Рис. 19. Вариограмма оценок для тренировочного набора 200д соответствующей HCOP

При рассмотрении аналогичных вариограмм для валидационных наборов (см. рис. 20, 21 и 22), сразу видно преимущество HCOP учившейся на 200д. Вариограммы оценки и истинных значений из валидационного набора почти всюду параллельны друг другу. Это говорит о том, что пространственная структура данных полностью воспроизводится, а расстояние между вариограммой оценки и вариограммой истинных значений определяется только шумом.

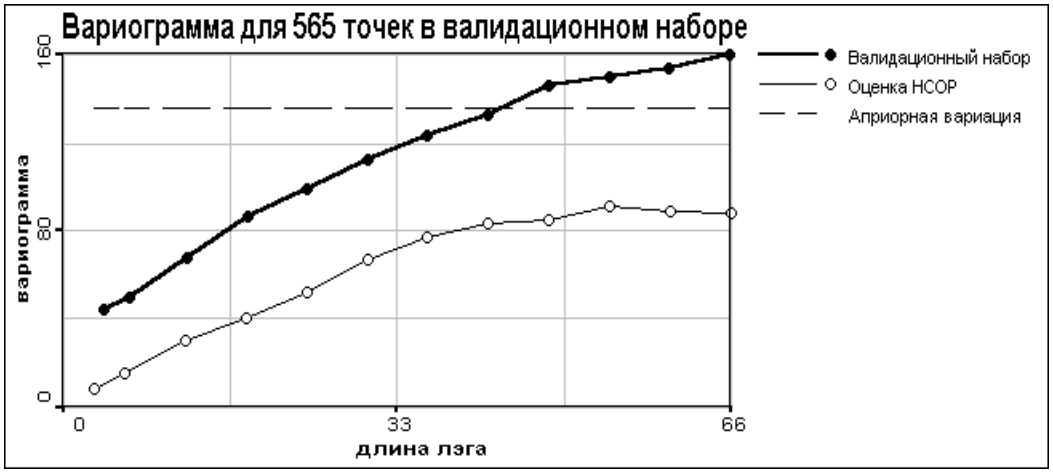


Рис. 20. Вариограмма оценок для валидационного набора 100 соответствующей HCOP

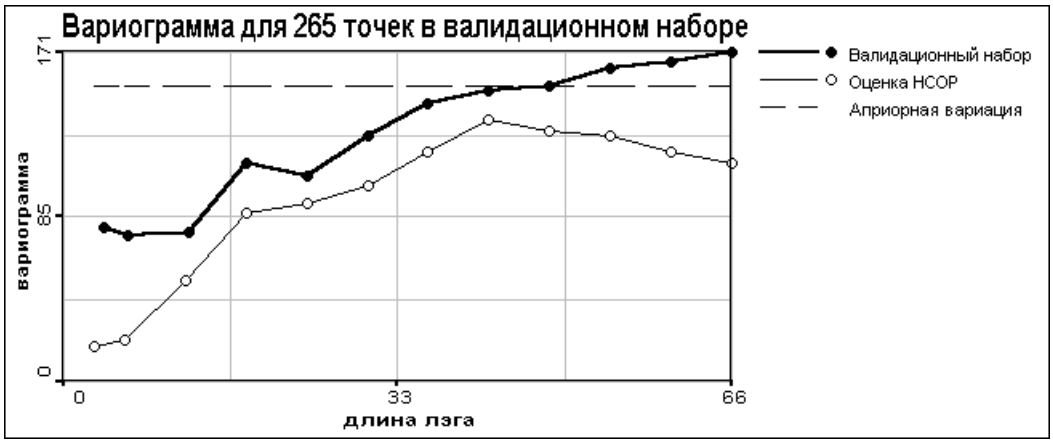


Рис. 21. Вариограмма оценок для валидационного набора 400 соответствующей HCOP.

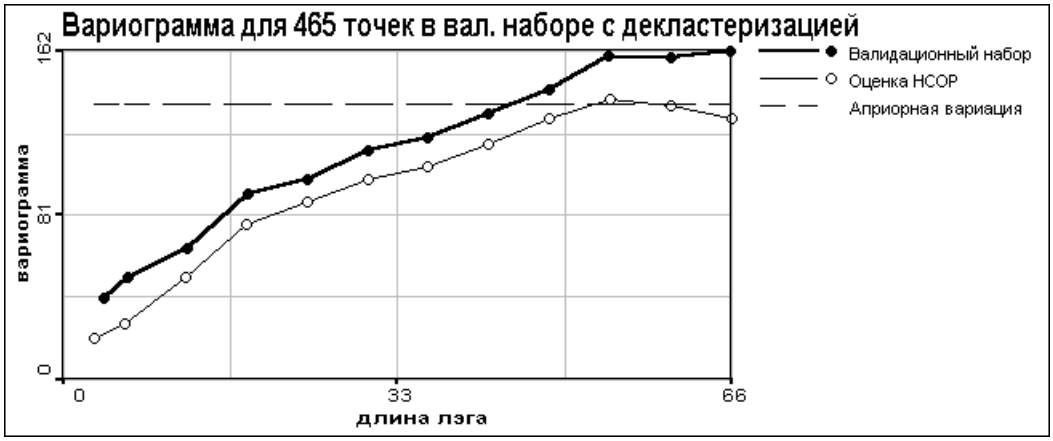


Рис. 22. Вариограмма оценок для валидационного набора 200д соответствующей HCOP

Осталось понять, смогла ли HCOP отследить все мелкомасштабные структуры в наборах 100, 200, 300, 400 и 500. Для этого необходимо провести вариографию невязок.

6.3 Анализ невязок

Если вариограмма колеблется около постоянного значения, то это означает, что данных не содержат ни каких зависимостей. HCOP оказалась очень хорошим интерполятором. Невязки на тренировочных наборах не содержали никакой структуры, вариограммы не имели значительных отличий, поэтому мы приведем здесь только вариограмму для набора 100 (см. рис. 23).

Рис. 23. Вариограмма невязок на тренировочном наборе 100

Иная картина получилась при рассмотрении вариограмм по оценке валидационных данных. Для двух сетей, обучавшихся на тренировочных наборах 100 и 200, вариография показала наличие структуры (см. рис 24 и рис. 25). В остальных случаях вариограмма невязок не давала структуры (см. рис. 26, 27, 28 и рис. 29).

Рис. 24. Вариограмма невязок на валидационном наборе 100

Рис. 25. Вариограмма невязок на валидационном наборе 200

Рис. 26. Вариограмма невязок на валидационном наборе 200д

Рассмотрим рис. 24. На расстояниях порядка 35 километров сохранилась структура. Раз есть структура, значит, ее можно попробовать промоделировать при помощи HCOP обученной на невязках набора 100. Результат применения HCOP с невязками набора 100 в качестве тренировочного набора изображен на рисунке 30. Рисунок 30 является прекрасной качественной иллюстрацией работы HCOP. Данные предоставленные сети имеют крупномасштабную структуру на расстояниях порядка 30 километров, Эта структура точно отслеживается обученной HCOP. Получившиеся невязки это только

шум, который при сложении с оценкой сети дает вариограмму исходных данных. Таким образом, HCOP проигнорировала шум, полностью распознав при этом структуру начальных данных.

Рис. 27. Вариограмма невязок на валидационном наборе 300

Рис. 28. Вариограмма невязок на валидационном наборе 400

Рис. 29. Вариограмма невязок на валидационном наборе 500

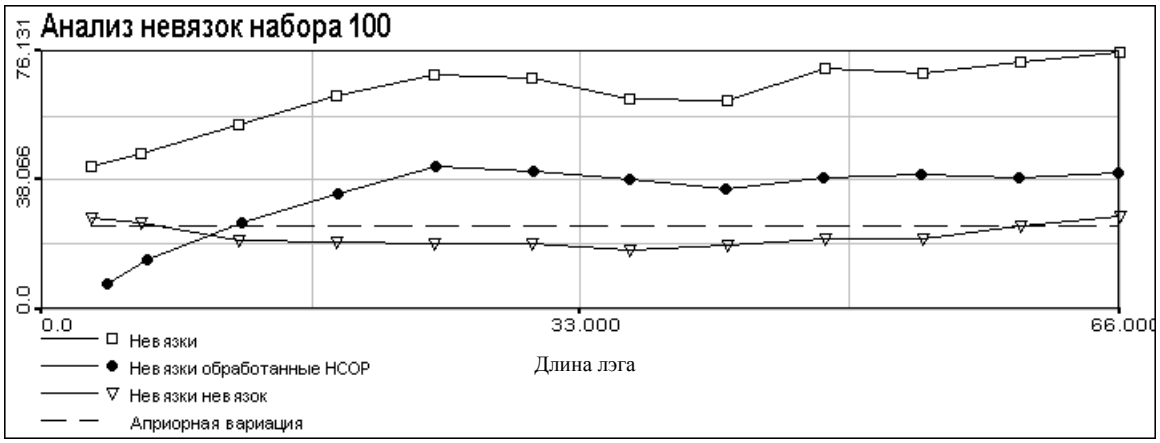


Рис. 30. Вариограмма невязок на валидационном наборе 100 и их повторная обработка HCOP

6.4 Резюме

В работе применен метод моделирования пространственно распределенных данных при помощи Нейронных Сетей с Обобщенной Регрессией. Анализировалась зависимость качества оценки даваемой Нейронной Сетью с Обобщенной Регрессией в зависимости от набора исходных данных. В качестве исходных были взяты данные по загрязнению брянской области.

Выводы:

1. Применение HCOP дает очень хорошие результаты.
2. Лучше всего для тренировки HCOP подходят данные, выбранные с учетом декластеризации.
3. При достаточно большом количестве данных, HCOP позволяет промоделировать полную пространственную структуру в этих данных как на крупном, так и на мелком масштабе. Невязки HCOP на тренировочных наборах не обладают корреляционной структурой. Невязки HCOP на валидационных наборах также не обладают корреляционной структурой, за исключением наборов 100 и 200, в которых было меньше всего данных для обучения.
4. Для каждого набора данных можно подобрать такое значение дисперсии функции совместной плотности вероятности h, что среднеквадратическая ошибка на валидационных данных имеет минимум.

7 Учет декластеризации в тренировочном наборе

В силу определенных пространственных свойств, выборки сделанные с учетом декластеризации наиболее хорошо подходят для обучения Нейронных Сетей с Обобщенной Регрессией. Нейронная сеть, обучавшаяся на наборе 200д, как и ожидалось, показала наилучшие результаты. Но делать выводы, основываясь лишь на одном примере несколько опрометчиво. В этой главе речь пойдет о более полном исследовании качества оценки сетей обучавшихся только на декластеризованных данных.

В задачу исследования входило изучение устойчивости оценки и ее качества в зависимости от выборки, критерием вновь служила среднеквадратическая ошибка. Дополнительными критериями служили смещенность и вариабильность оценки, а также способность сети полностью воспроизвести пространственную корреляционную структуру валидационного набора.

7.1 Декластеризация

Исходные 665 точек, из которых составлялись все наборы для тренировки и обучения содержались в строчках таблицы из трех колонок – x, y и z. Случайная выборка состояла из строчек со случайно выбранными номерами. Произвольность выбора срок в тренировочный набор позволяла надеяться, но вовсе не гарантировала, что в тренировочный набор войдут точки распределенные по всей области сети мониторинга, а не сосредоточенные в маленькой ограниченной области. На рисунке 31 наглядно показано, о чем идет речь.

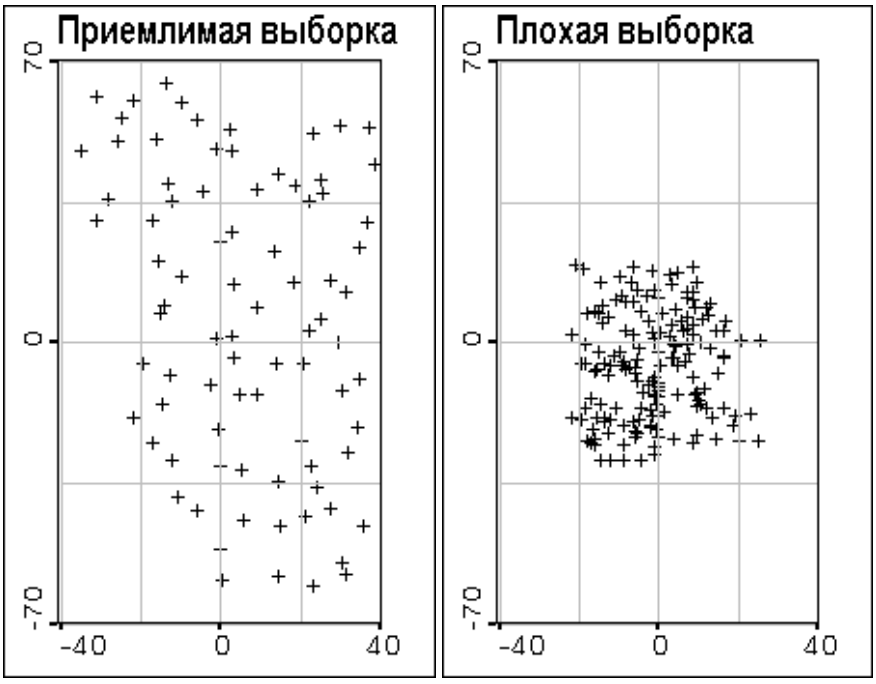


Рис. 31. Различные варианты выборок

Слева, точки почти равномерно покрывают область сети мониторинга, справа, хотя точек измерения и больше, но они сосредоточены в центре области, следовательно, из-за отсутствия измерений, в довольно большой зоне невязки могут быть очень высоки. Заметим, что речь пока идет только о местах, в которых производились измерения, о влиянии значений измеряемой величины мы поговорим позже.

Для того чтобы точки входящие в тренировочный набор пространственно отражали всю исследуемую область можно воспользоваться особым способом их выбора – декластеризацией. Вся область разбивается на одинаковые прямоугольники – ячейки размер которых может варьироваться. Затем, из каждого прямоугольника выбирается (каким-либо способом) определенное количество точек. Полученная выборка называется выборкой с учетом декластеризации. Как и сколько точек следует выбирать из каждой ячейки вопрос сложный и до конца не решенный. В данной работе из каждой ячейки выбиралась одна точка и выбор этот был чисто случайным, подробнее о преимуществах и недостатках такого подхода будет сказано в главе “ Результаты и обсуждение работы”. Размер ячейки выбирался пропорциональным размеру области (y/x = 3/2). Исходя из этого масштабного соотношения, уменьшеним или увеличением размера ячейки достигалось необходимое количество точек в выборке.

7.2 Ход работы

- Были сделаны 5 групп по 10 выборок содержащих 100, 200, 300, 400 и 500 точек с учетом декластеризации, в 1-й группе были только выборки по 100 точек, во второй по 200 и т.д.
- Для каждой группы выборок была найдена средняя RMSE на валидации
- Для дальнейшего, более подробного изучения были взяты 2-е группы выборок – по 200 и 400 точек
- Для каждой выборки была построена своя нейронная сеть
- Для каждой НСОР минимизировалась ошибка на валидации
- Строились вариограммы невязок на валидационном наборе
- Производился расчет среднего и вариации невязок, а также вариация оценок

7.3 Результаты и обсуждение работы

7.3.1 Представление результатов

Таблица 3.
Распределение средней RMSE

Кол-во точек в выборке	*h*	Средняя RMSE
500	0.02	5.7
400	0.02	5.7
300	0.02	5.8
200	0.03	5.9
100	0.04	6.3

Дальнейшее исследование проводилось только для групп с выборками по 200 и 400 точек. Сначала исследовались выборки по 400 точек. Полученные результаты сведены в таблицу 4.

Таблица 4.
Сводная таблица для выборок по 400 точек

Выборка	RMSE	*h*	Среднее невязки	Корр. оценки с данными для		Вариация вал. набора	Вариация оценки
				валидации	обучения		
1d	4.516	0.02	-0.31	0.91	0.96	128.8	123.9
2d	5.44	0.02	-0.49	0.89	0.94	127.54	122.11
3d	5.176	0.03	0.08	0.9	0.95	134.32	115.26
4d	5.403	0.02	-0.41	0.88	0.94	129.22	124.69
5d	5.208	0.02	-0.06	0.9	0.95	138.48	132.19
6d	5.205	0.02	-0.03	0.9	0.95	140.47	136.05
7d	4.833	0.02	0.17	0.9	0.95	144.9	103.52
8d	5.417	0.02	-0.42	0.87	0.94	126.38	128.45
9d	4.36	0.02	-0.03	0.91	0.95	133.98	112.81
10d	5.118	0.02	0.18	0.88	0.94	149.69	104.1

Аналогичная таблица для выборок по 200 точек представлена ниже.

Таблица 5.
Сводная таблица для выборок по 200 точек

Выборка	RMSE	*h*	Среднее невязки	Корр. оценки с данными для		Вариация вал. набора	Вариация оценки
				валидации	обучения		
*1d	6.781	0.03	0.32	0.88	0.96	140.85	75.76
2d	5.255	0.03	0.21	0.89	0.95	125.14	97.55
3d	5.44	0.03	0.29	0.88	0.95	123.54	109.31
4d	5.587	0.03	-0.31	0.89	0.94	118.13	106.77
*5d	6.482	0.02	0.87	0.87	0.94	135.95	77.32
*6d	6.504	0.03	0.94	0.88	0.97	134.68	79.81
7d	6.002	0.02	0.75	0.87	0.96	141.7	105.8
8d	5.67	0.02	0.34	0.88	0.94	137.59	102.22
9d	5.905	0.03	0.3	0.89	0.95	137.85	108.4
10d	5.652	0.03	0.41	0.88	0.95	142.76	109.27

* Звездочками в таблице 5 обозначены выборки в которые не вошла точка (-11,53,94). О влиянии этой точки речь пойдет в конце пункта 7.3.2.

Разброс результатов и в случае с обоими наборами выборок весьма невелик. Графические результаты еще более похожи, поэтому чтобы не обременять читателя, ниже приведены рисунки только для самых

характерных выборок по 400 точек. Аналогичные рисунки для выборок по 200 точек практически неотличимы от нижеприведенных, поэтому сводные таблицы гораздо информативней.

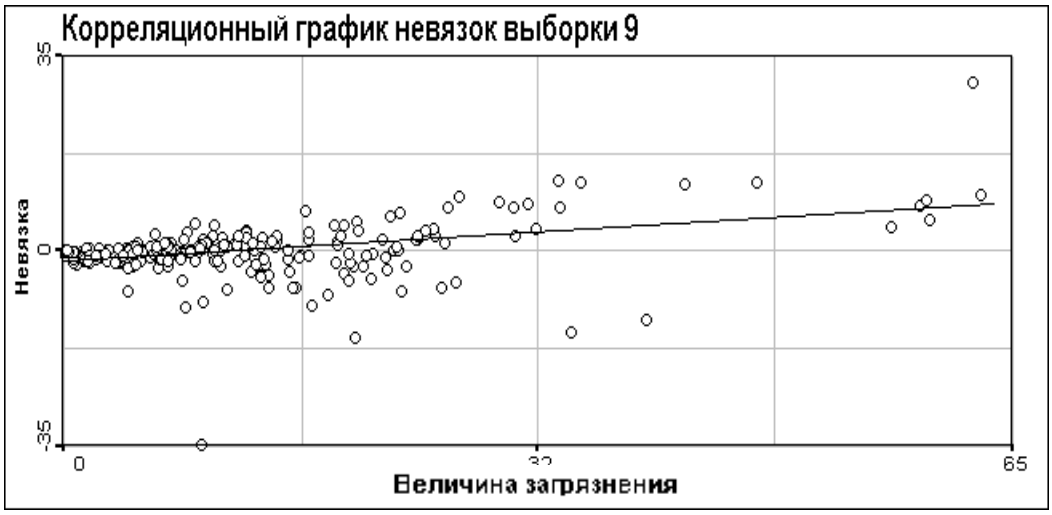


Рис. 32. Зависимость невязки от значения оцениваемой функции (для 400 точечной выборки)

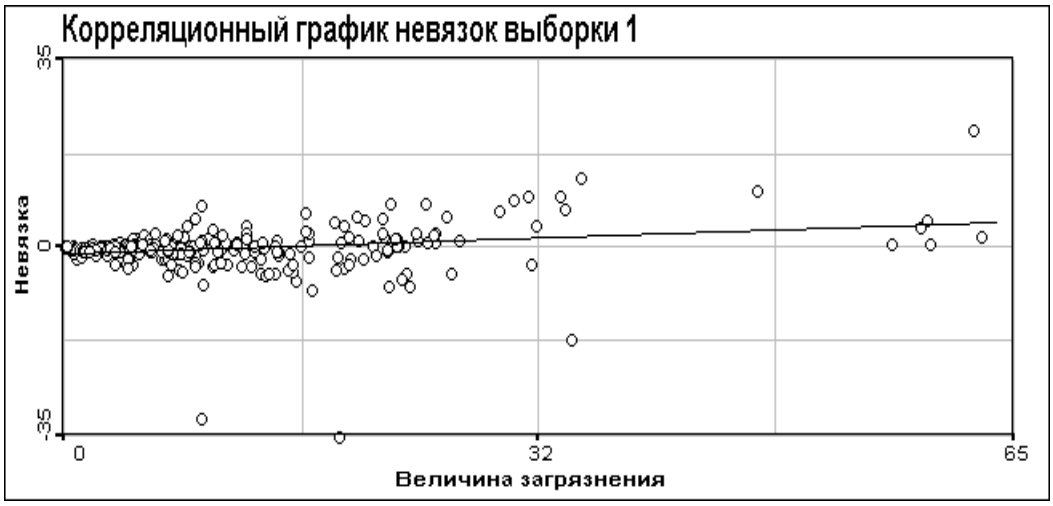


Рис. 33. Зависимость невязки от значения оцениваемой функции (для 400 точечной выборки)

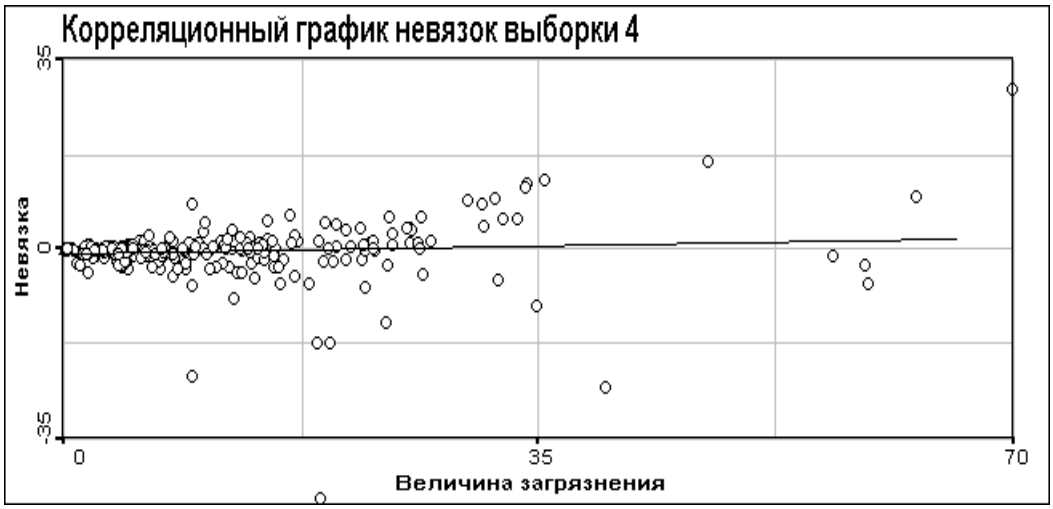


Рис. 34. Зависимость невязки от значения оцениваемой функции (для 400 точечной выборки)

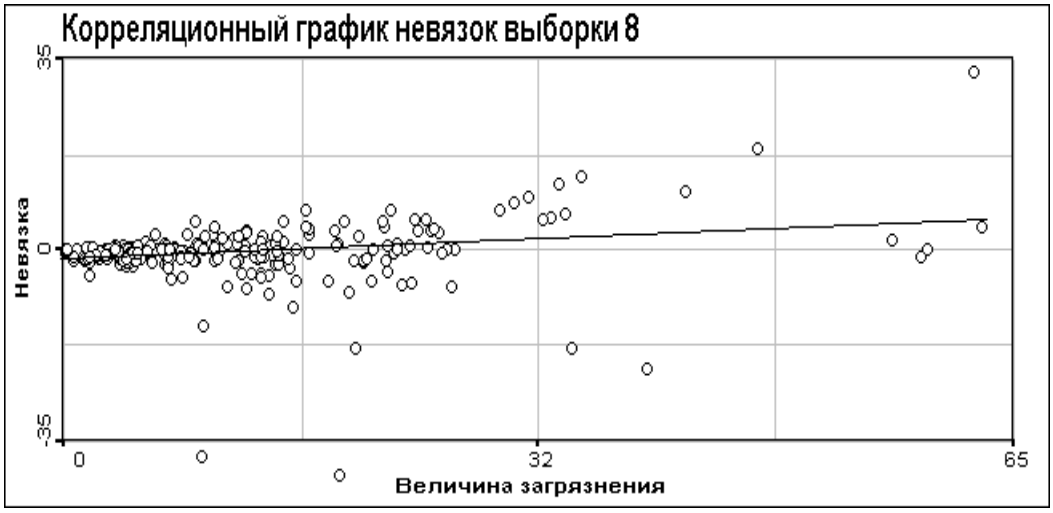


Рис. 35. Зависимость невязки от значения оцениваемой функции (для 400 точечной выборки)

Рис. 36. Корреляция между истинным значением и оценкой (для 400 точечной выборки)

Рис. 37. Корреляция между истинным значением и оценкой (для 400 точечной выборки)

Рис. 38. Корреляция между истинным значением и оценкой (для 400 точечной выборки)

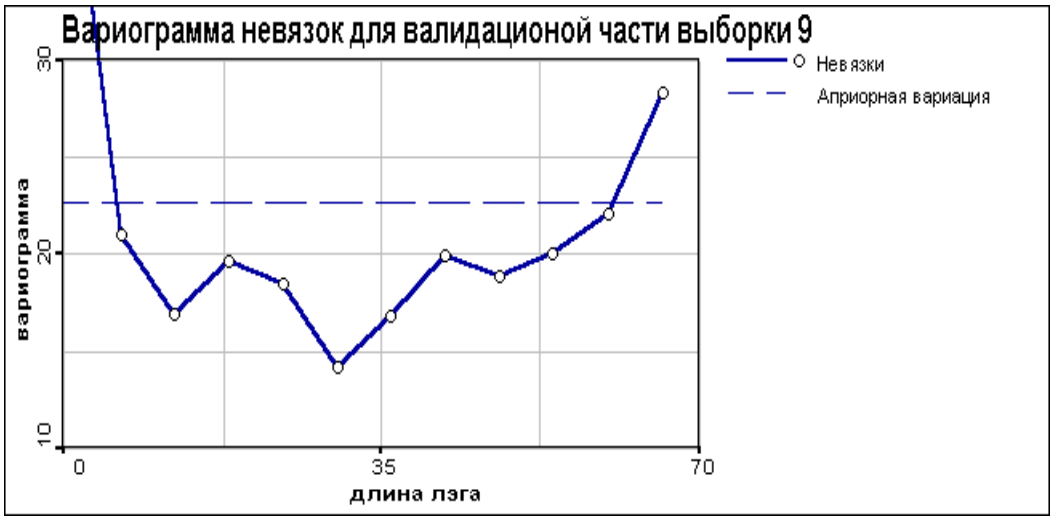


Рис. 39. Вариограмма невязок (для 400 точечной выборки)

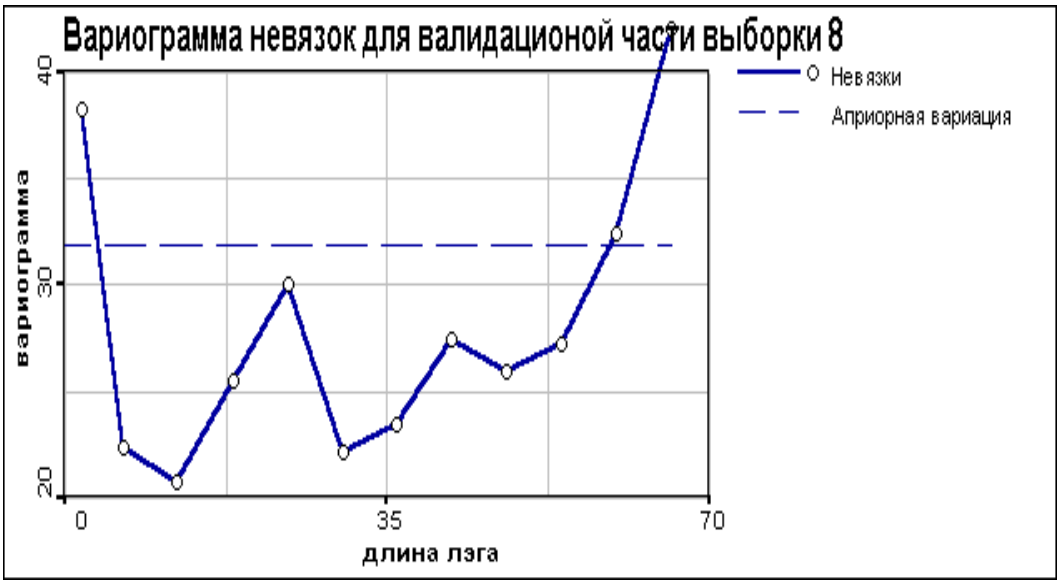


Рис. 40. Вариограмма невязок (для 400 точечной выборки)

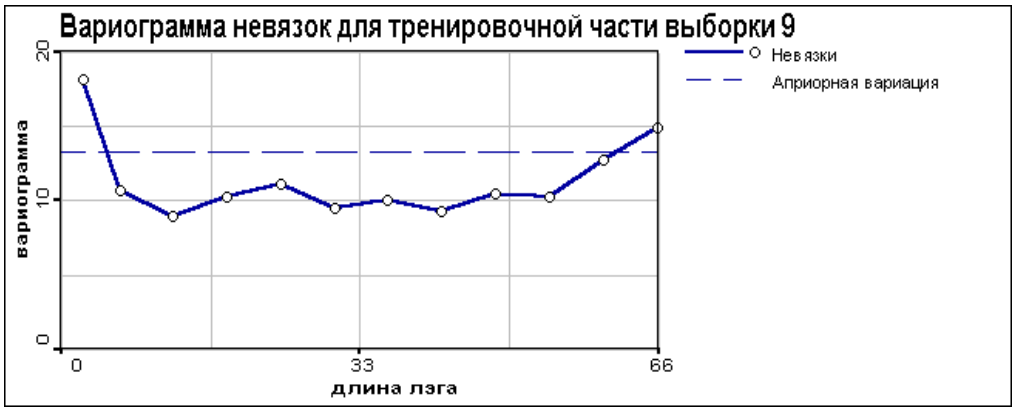


Рис. 41. Вариограмма невязок (для 400 точечной выборки)

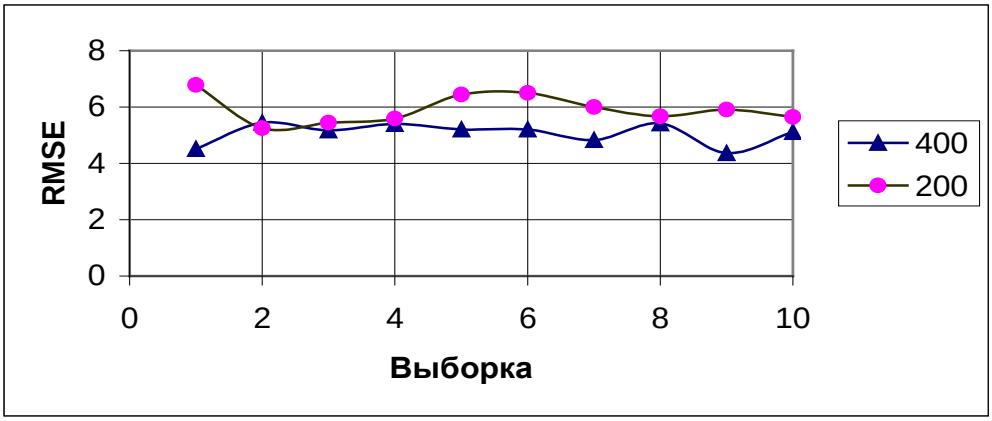


Рис. 42. Зависимость RMSE от номера выборки

7.3.2 Обсуждение

После представления результатов перейдем к их обсуждению. Сразу хочется отметить, что предположение об улучшении результатов в случае использования декластеризованных тренировочных данных блестяще подтвердилось. Максимальная RMSE по всем декластеризованным данным меньше минимальной RMSE на данных выбранных случайно (см. таблицы 3-5). Рассмотрим это явление подробнее. Преимущество декластеризованных данных не только в том, что они "покрывают" всю область. Рассмотрим такую ситуацию (см. рис. 43): имеется одна обособленная точка А и на некотором расстоянии от нее группа точек Б расположенных очень плотно.

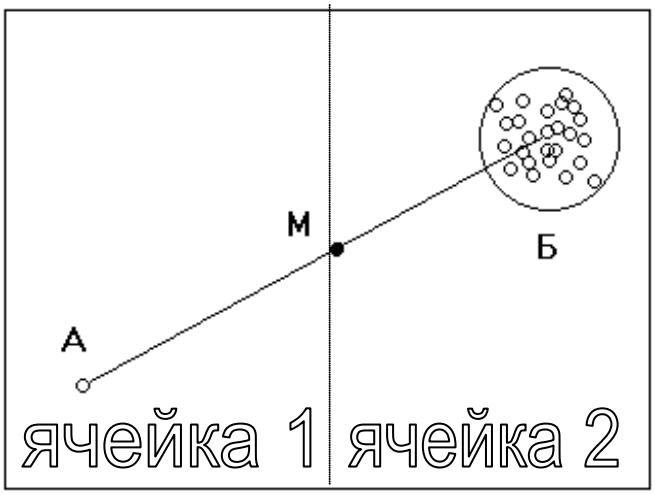


Рис. 43. Пример декластеризации данных имеющих области с высокой плотностью измерений

Предположим что значение функции в каждой точке группы Б примерно одно и то же, и существенно отличается от значение функции в точке А. Посмотрим какую оценку даст формула регрессии (4) в середине отрезка соединяющего точку А с группой точек Б.

$$Z_m = \frac{\sum_{i=1}^{n} Z_i \exp\left(-D_i^2 / 2h^2\right)}{\sum_{i=1}^{n} \exp\left(-D_i^2 / 2h^2\right)} \tag{4}$$

Выражение для оценки разобьется на два слагаемых, одно из которых будет вкладом точки А, другое, имеющее суммирование в числителе, вкладом группы Б.

$$Z_m = \frac{Z_A \exp\left(-D_A^2 / 2h^2\right)}{\sum_{i=1}^{n} \exp\left(-D_i^2 / 2h^2\right)} + \frac{\sum_{i=2}^{n} Z_Б \exp\left(-D_i^2 / 2h^2\right)}{\sum_{i=1}^{n} \exp\left(-D_i^2 / 2h^2\right)} \tag{4*}$$

Так как размер группы точек Б гораздо меньше расстояния длины отрезка АБ, то формулу можно подвергнуть дальнейшему упрощению.

$$Z_m = \frac{Z_A \exp\left(-D^2 / 2h^2\right)}{\sum_{i=1}^{n} \exp\left(-D_i^2 / 2h^2\right)} + \frac{(n-1) Z_Б \exp\left(-D^2 / 2h^2\right)}{\sum_{i=1}^{n} \exp\left(-D_i^2 / 2h^2\right)}, \tag{4**}$$

где D – половина отрезка АБ.

Таким образом вклад группы Б в значение оценки в точке М тем больше, чем больше в этой группе точек. Что же дает декластеризация? На рисунке 43 точка А и группа Б попали в разные ячейки, следовательно, из всего множества точек остаются только две. Вклад этих точек в оценку будет одинаков. Если выборку сделать случайно, то имеется большая вероятность, что точка А в нее вообще не попадет.

Вообще, при нерегулярной сети мониторинга, часто возникают подобные описанной выше ситуации. Область с повышенной плотностью измерений называется кластер. Кластеры с примерно одинаковыми значениями оцениваемой функции вносят серьезные изменения в результат работы НСОР. Важно понять негативно или положительно влияние этих изменений. Обозначим значения, которые может принимает оцениваемая функция в точке А квадратиком, в кластере Б кружком (см. рис. 44), причем пусть для определенности значение в точке А больше. Попытаемся ответить на вопрос: к какому значению (к кружку или квадратику) ближе значение функции в точке М? Будем обозначать оценку в т. М квадратиком, если она больше среднего арифметического между значениями в т. А и на кластере Б, если же оценка меньше – кружком.

К ответу на поставленный вопрос можно подойти с разных позиций. Если, например считать, что вероятность события кружок в какой либо точке тем выше чем большее количество раз оно выпадает в точках лежащих относительно недалеко, то в точке М нужно поставить кружок (методически аналогично действует и НСОР). Положительные стороны такого ответа налицо. Но в реальных условиях такой подход может не сработать. Допустим, необходимо оценить высоту местности. Можно провести множество замеров высоты на дне узкого отвесного ущелья (кластер Б) и всего одно на вершине (точка А). НСОР, оценивая карту высот по полученным тренировочным данным выдаст значение очень близкое к значению оцениваемой функции на кластере Б, то есть многометровые перепады высоты будут полностью потеряны.

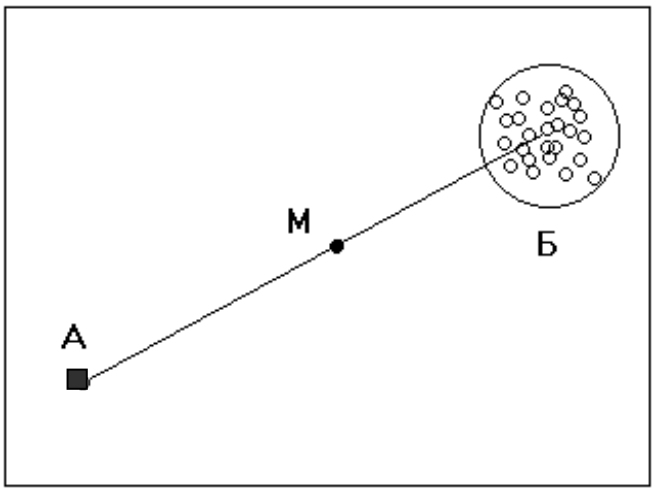


Рис. 44. Область содержащая обособленную точку А и кластер Б

Вообще, как видно из формулы 4**, при любом значении функции в точке А можно поместить в кластер Б столько точек и так подобрать *h*, что оценка НСОР в точке М будет не только принадлежать классу кружок, но будет неотличима от значения оцениваемой функции на кластере с любой заданной точностью. Таким образом, если говорить о работе нейронной сети вообще, а не о конкретном случае показанном на рисунке 44, то некоторые измерения для НСОР оказываются просто несущественными. Для того чтобы избежать указанного недоразумения и применяется декластеризация. Результат декластеризации в приведенном на рисунке случае - две точки (т. А и одна точка из кластера Б), каждая из которых отвечает за целую область. НСОР при таком тренировочном наборе выдаст в точке М среднее арифметическое между значением в точке А и значением в любой точке кластера Б (при условии равенства длины отрезка АМ расстоянию от точки М до точки из кластера Б), то есть уверенно поставить в точку М кружок или квадратик в этом случае нельзя. Область влияния (на результат оценки) каждой точки тем меньше, чем больше у этой точки близких соседей. В результате значимость в оценке обособленной точки становится равной значимости кластера. Таков второй подход к ответу на этот вопрос. О преимуществах этого варианта ответа уже практически все сказано, недостатки сводятся к преимуществам первого варианта плюс еще кое-что, о чем мы поговорим ниже.

Итак, для НСОР выборка с декластеризацией предпочтительней случайно выборки, так как область влияния каждой точки при этом изменяется в зависимости от количества соседних точек. Но параметры декластеризации (размер ячейки и кол-во выбираемых из ячейки точек) могут меняться в весьма широких пределах. Подбор этих параметров следует осуществлять исходя из специфики НСОР. Размер ячейки был выбран пропорциональным размеру области исходных данных для того чтобы учесть анизотропный характер области сети мониторинга. Перечислим причины, по которым из каждой ячейки выбиралась одна точка. Во-первых, для того чтобы получить равномерно пространственно распределенные точки для тренировочного набора. Во-вторых, так как выборки состояли из сравнимого с исходным количества точек (выборки по 400 точек и 200 точек, всего было 665 точек), то ячейки были довольно мелкими и количество точек в них было невелико, поэтому для того чтобы избавиться от кластеров, размеры ячеек уменьшались, но из каждой ячейки бралась только одна точка. В связи с приведенным выше витиеватым рассуждением, следует указать на еще один возможный недостаток декластеризации. Необходимо очень аккуратно следить за уменьшением размеров ячейки, может случиться так, что характерный размер кластеров будет сильно превосходить размер ячеек и тогда эффективность декластеризации заметно снизится.

Рассмотрим распределение средней RMSE по группам выборок (см. таблицу 3). Монотонное уменьшение RMSE с увеличением количеств точек в выборке – самое очевидное предположение, которое прекрасно подтвердилось на практике. Хотя указанного в предыдущем параграфе ухудшения оценки с уменьшением размера ячейки наблюдать не удалось, это легко объяснимо. Действительно, чем больше точек попадает в тренировочный набор, тем меньше их остается в валидационном, причем в валидационный набор после декластеризации попадают в основном точки гладкости оцениваемой функции. Поэтому валидационный набор для выборок по 500 точек не содержит ни одного выброса, а значит, невязка оценки будет невелика. Из сумм квадратов малых невязок получается и малое значение для RMSE. Это обстоятельство несколько портит чистоту эксперимента, поэтому для дальнейшего исследования была взята группа из выборок по 400 точек – достаточный в смысле своего объема

тренировочный набор, с адекватным валидационным набором. Выборки по 200 точек были взяты в качестве примера малых тренировочных наборов. 100 точек оказалось явно недостаточно для обучения сети, слишком много выбросов оставалось в валидационном наборе, отсюда и высокая RMSE. Вообще выяснилось, что критерий RMSE не совсем хорош, но об этом мы поговорим позже. Следует отметить что большинство (7 из 10) сетей обучавшихся на 100 точках, несмотря на высокие значения RMSE, сумели полностью воспроизвести пространственную корреляционную структуру валидационных данных, то есть вариограмма невязок оказалась чистым шумом.

При изучении зависимости RMSE от h выяснилось, что минимум для разных выборок с одинаковым количеством точек достигается практически при одном и том же значении h (см. таблицы 4 и 5). Этот факт, а также малый разброс RMSE в группах выборок по 200 и 400 точек (см. рис. 42) говорит о высокой устойчивости НСОР по отношению к выборке. Рост h с ростом количества точек в выборке, очевидно, связан с тем, что растет расстояние между точками тренировочного набора. Как показало исследование, оптимальное h всегда таково, что при оценке используются в среднем 5 близлежащих точек. Это нетрудно посчитать: при переходе к единицам измерения h оси координат сжимаются в отрезки [0,1]. Следовательно, если h равняется 0.02, то на оценку в определенной точке будут влиять лишь измерения попавшие в эллипс с радиусами $2h$ (сжатия по разным осям при переходе к отрезку [0,1] различны). При пересчете в километры радиусы эллипса равны трем и пяти.

Продолжим изучение таблиц 4 и 5. Следующие за оптимальным валидационным h два столбика содержат коэффициенты корреляции оцененных тренировочного и валидационного наборов с их измеренными значениями. Превосходные результаты по оценке тренировочных данных (при этом h бралось равным лучшему валидационному значению) связанны со спецификой работы НСОР (см рис 38). Так как после декластеризации точки тренировочного набора распределены по плоскости почти равномерно, то из-за того, что для всех точек тренировочных данных НСОР при расчете использует одно и тоже h, равномерно распределенные данные оказываются оптимальными для оценивания. Высокие коэффициенты корреляции при оценке валидационного набора показали, что НСОР с декластеризованным тренировочным набором прекрасно подходит для решения задач по оценке зашумленных данных на нерегулярной сети мониторинга (см. рис. 32-37). То, что аналогичные результаты для выборок по 200 точек немного хуже объясняется, прежде всего, уменьшением тренировочного набора и соответственно увеличением выбросов в валидационном наборе.

Несмотря на то, что при оценивании НСОР сглаживает исходные данные, следует обратить внимание на очень высокую вариабильность оценки валидационного набора – два последних столбика таблиц. Необходимо отметить, что это свойство происходит исключительно из-за декластеризованности тренировочного набора, так как в случае случайных выборок вариация была максимальной для НСОР обучавшейся на 500 точках и не превосходила 80. Как показывает сравнение таблиц для выборок по 400 и 200 точек, вариация более чувствительна к увеличению тренировочного набора, чем RMSE.

Вариограммы невязок при оценивании валидационного набора оказались чистым шумом для случая всех НСОР за исключением трех обучавшихся на 100 точках (см. рис. 39-41). Это был ожидаемые результат, учитывая то, что при случайно выбранных тренировочных наборах по 300 и более точек, невязки на валидации не содержали структуры.

В конце повествования, перед перечислением выводов несколько слов о двух очень интересных вопросах, которые не были исследованы в этой работе. Начнем с критерия качества работы сети. RMSE – среднеквадратическая ошибка представляется в виде:

$$RMSE = (1/N)\sqrt{\sum_i (Z\_net - Z)^2} \quad .$$

Эта нехитрая степенная зависимость от невязок очень чувствительна к выбросам. Обратите внимание, что в таблице выборки, в которые не попала точка с координатами (-11,53,94) помечены звездочкой. В выше указанной точке оцениваемая функция достигает своего, ярко выраженного, глобального максимума, эту точку можно назвать выбросом. В остальном, все выборки по 200 точек очень похожи. Но, максимальные RMSE точно соответствуют звездочкам. С другой стороны коэффициенты корреляции не выделяются на общем фоне. Вообще, если оценка данная одной моделью полностью совпадает с истинной зависимостью везде за исключением одной точки, в которой имеется сильное отклонение, а оценка данная другой моделью слабо похожа на истину, но не имеет значительных невязок, то RMSE для обоих случаев могут оказаться равными. Весь вопрос в том, что требуется от модели, конечно оценка должна прежде всего быть похожа на истину, но похожа в каком смысле? По-разному отвечая на этот вопрос можно получать разные математические критерии качества.

Перейдем ко второму вопросу. Итак, качество (в смысле RMSE) работы НСОР – локальной математической модели стало гораздо выше, когда данные для обучения наделили специальными, можно сказать локальными, пространственными свойствами. Вместо беспорядка и хаоса случайных выборок, НСОР получила тренировочный набор с четко прослеживаемой структурой. Вся исследуемая область разделилась на равные подобласти – ячейки, за каждую из которых отвечает (своим значением) одна единственная точка. Это соответствует тому, что мы одинаково доверяем каждому отдельно взятому измерению независимо от значений в соседних точках. Так как вклад в оценку НСОР каждой точки не зависит от количества соседей, то декластеризация избавляет от ошибок при плотно расположенных точках измерений. Осталось понять, как выбрать из ячейки одну единственную точку. Нужен какой-то критерий, который помог бы найти самую "характерную" для данной ячейки точку. Можно развить эту тему следующим образом. Выше по тексту часто применялось понятие выброс, хотя его точного определения не давалось. Интуитивно оно очевидно – точка со значением функции, которое сильно отличается от значений той же функции в соседних, близлежащих точках. Но для математической модели нужно математической, а не формальное описание. Выбросы являются наиболее трудными для оценивания точками, возможно, следует вообще исключить их из рассмотрения и тем самым резко увеличить качество работы НСОР. Таким образом, есть смысл говорить о некоторой гипотезе относительно приемлемости исходных данных. Используя эту гипотезу для предобработки исходных данных, мы убережем НСОР от больших невязок при оценке, что непременно отразится на качестве ее работы.

7.4 Резюме

1. При использовании декластеризованных данных RMSE на валидации заметно уменьшается по сравнению со случайными выборками.
2. RMSE уменьшатся с увеличением тренировочного набора.
3. Оценка НСОР устойчива по отношению к выборке, то есть для множества выборок одного размера оптимальное валидационное h имеет одно и то же значение при котором разброс RMSE незначителен по сравнению со среднем значением.
4. Коэффициент корреляции между тренировочным набором и его оценкой превышает 0.95 и практически не зависит от количества точек в тренировочном наборе.
5. Коэффициент корреляции между валидационным набором и его оценкой превышает 0.90 для 400 точечных тренировочных наборов.
6. Коэффициент корреляции между валидационным набором и его оценкой падает с уменьшением размера тренировочного набора.
7. Оценка НСОР обладает большой вариацией, практически равной вариации оцениваемых данных.
8. Вариация оценки падает с уменьшением размера тренировочного набора.
9. Начиная с 200 точек в тренировочном наборе, НСОР полностью воспроизводит пространственную корреляционную структуру данных.

8 Заключение

Работа состояла из двух больших частей – исследование качества работы НСОР обучавшейся на случайно выбранном тренировочном наборе и исследование качества работы НСОР обучавшейся на декластеризованных данных. Выводы по каждой из этих двух частей содержатся в пунктах 6.4 и 7.4. Сейчас мы постараемся свести их воедино. Итак выводы из всей работы в целом таковы:

1. НСОР прекрасно подходит для моделирования зашумленных данных нерегулярной сети мониторинга.
2. При достаточно большом тренировочном наборе НСОР позволяет полностью промоделировать как крупномасштабную так и мелкомасштабную пространственную корреляционную структуру.
3. Для каждого набора данных можно подобрать такое значение дисперсии функции совместной плотности вероятности – h, что среднеквадратическая ошибка на валидационных данных имеет минимум. Имеется тренировочный набор для которого среднеквадратическая ошибка на валидационных данных минимальная по отношению ко всем остальным наборам.

4. НСОР, тренировавшиеся на декластеризованных данных, показали гораздо лучшие результаты по сравнению с теми сетями, тренировочными наборами которых служили случайные выборки.
5. Для сетей, использовавших декластеризованный тренировочный набор, валидационная оценка очень точно воспроизводит исходные данные (совпадают вариограммы и очень близки значения вариации). Оценка валидационных данных имеет очень высокий коэффициент корреляции с валидационными данными.
6. Оценка НСОР тренировавшейся на декластеризованных данных устойчива по отношению к выборке, то есть для множества выборок равного размера оптимальное валидационное *h* имеет одно и то же значение и разброс RMSE в пределах множества подобных выборок при этом *h* крайне мал.

9 Благодарности

Работа выполнена при частичной поддержке грантов ИНТАС 96 – 1957, 96 – 31726 и гранта для молодых ученых РАН “Искусственные нейронные сети и генетические алгоритмы для анализа и моделирования пространственной информации по окружающей среде”.

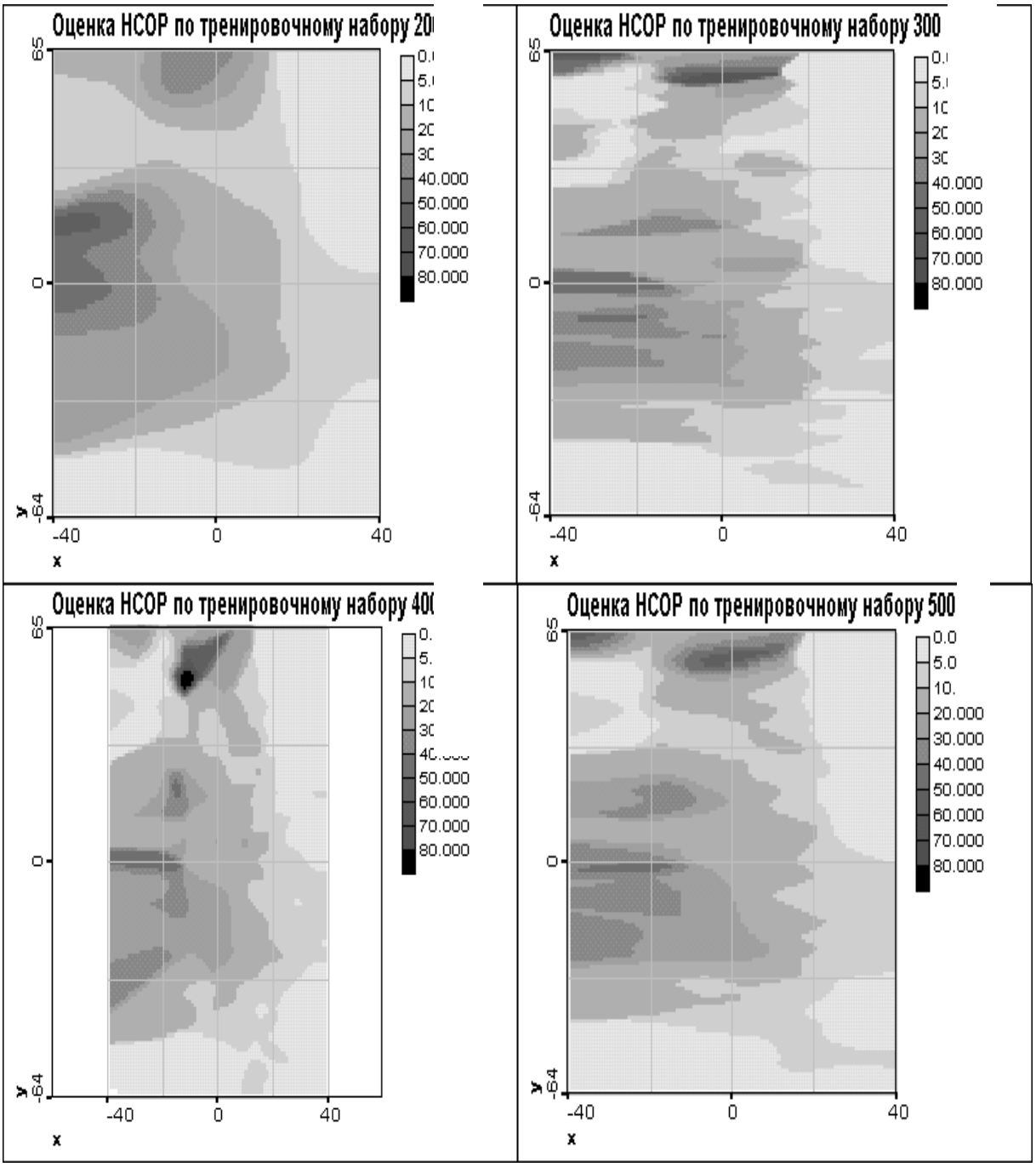
При работе использовалась программа GeoStat Office, разработанная высококвалифицированными программистами Черновым С.Ю., Савельевой Е.А., Тимониным В.А.

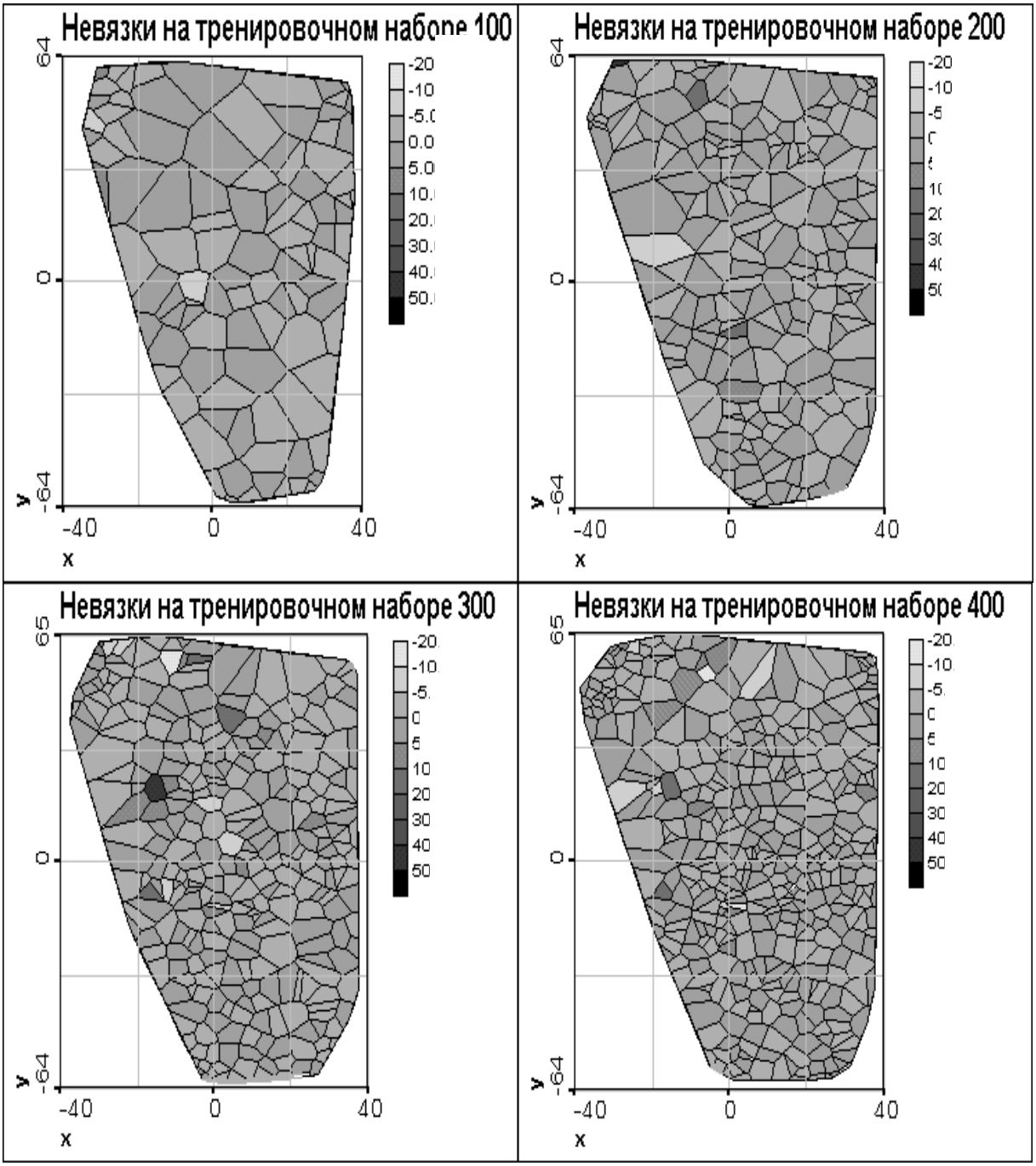
10 Литература

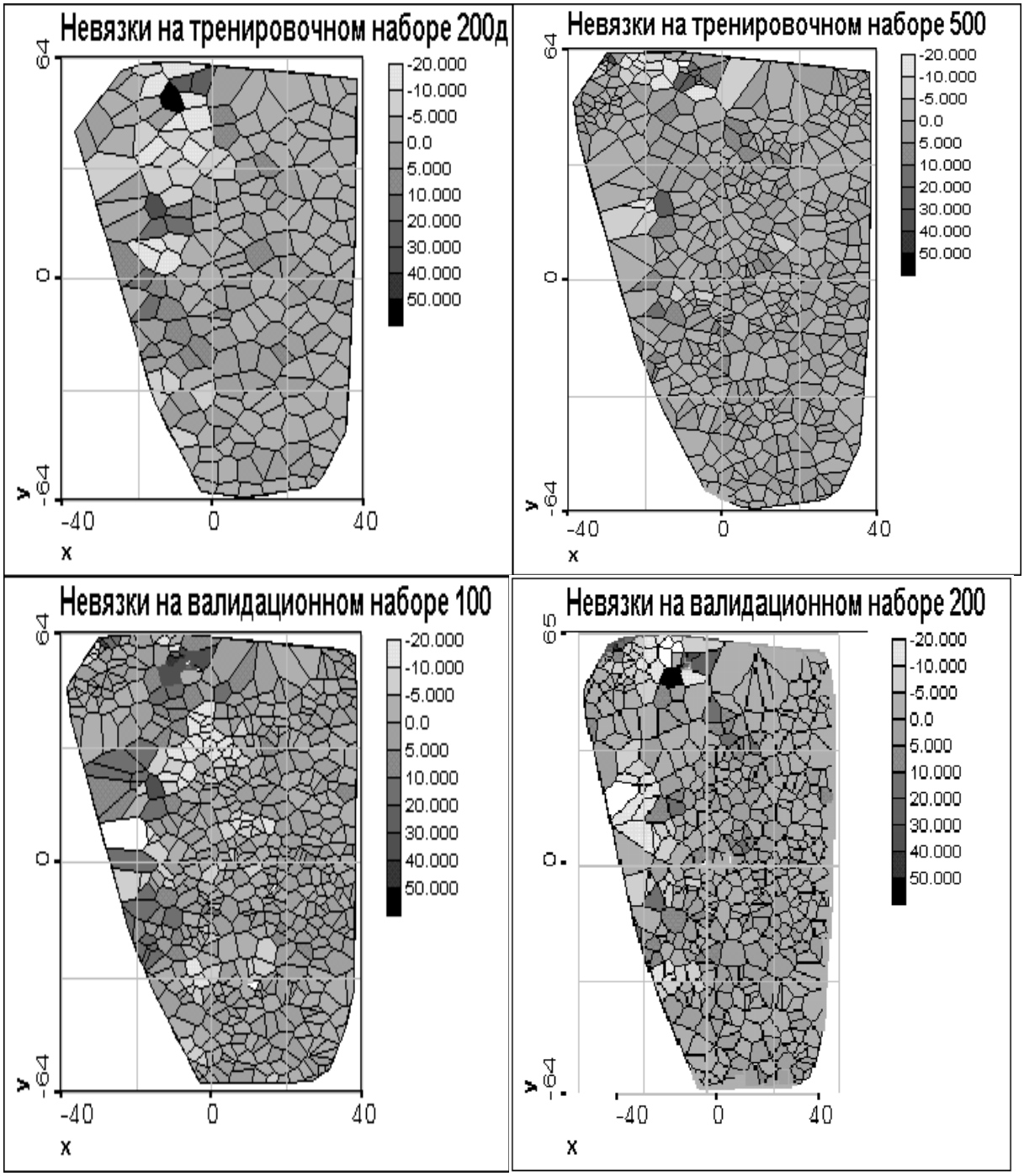
1. Masters T. Practical Neural Network Recipes in C+. Academic Press, 1993.
2. Specht, Donald A general regression neural networks 1991.
3. Isaaks E.H., Shrivastava R.M. An Introduction to Applied Geostatistics. Oxford University press, Oxford, 1989.
4. B. W. Silverman (1986) Density estimation for statistics and data analysis. School of Mathematics University of Bath UK.
5. M. Mitchell An introduction to genetic algorithms. The MIT Press Cambridge, Massachusetts, London, England 1996.
6. Kanevsky M., Arutyunyan R., Bolshov L., Demyanov V., Linge I., Savelieva E.,. Shershakov V., Haas T., Maignan M. Geostatistical Portrayal of the Chernobyl Fallout. Geostatistics Wollongong '96, ed. E.Y. Baafi, N.A. Schofield, Kluwer Academic Publishers, 1996, volume 2, pp.1043-1054.
7. Kanevsky M., Arutyunyan R., Bolshov L., Demyanov V., Maignan M. Artificial neural networks and spatial estimations of Chernobyl fallout. Annual Conference of International Association for Mathematical Geology. Osaka, Japan, 29 October - 2 November 1995. Abstracts for Technical Programs, p. 27-30.
8. Kanevsky M., Arutyunyan R., Bolshov L., Demyanov V., Maignan M. Artificial neural networks and spatial estimations of Chernobyl fallout. Geoinformatics. Vol.7, No.1-2, 1996, pp.5-11.
9. M. Kanevski, V. Demyanov, S. Chernov, E. Savelieva, A. Serov, V. Timonin. Geostat Office for Environmental and Pollution Spatial Data Analysis. Mathematische Geologie, band 3, April, 1999, pp. 73-83, CPress Publishing House, 1999.
10. M. Kanevski, R. Arutyunyan, L. Bolshov, V. Demyanov, S. Chernov, E.Savelieva, V. Timonin, M. Maignan, M.F. Mapping of Radioactively Contaminated Territories with Geostatistics and Artificial Neural Networks. In Contaminated Forests I. Linkov and W.R. Schell (eds.), pp. 249-256, Kluwer Academic Publishers, 1999, Printed in Netherlands.
11. M.F. Kanevski. Spatial Predictions of Soil Contamination Using General Regression Neural Networks. Int. Jour. Systems Research and Informational Science. 1998. In Print.
12. M. Kanevski, V. Demyanov, S. Chernov, E. Savelieva, V. Timonin. Neural Network Residual Kriging Application For Climatic Data. The Journal of Geographic Information and Decision Analysis (GIDA) ISSN 1480-8943.
13. Каневский М. Ф. Использование искусственных нейронных сетей для пространственных интерполяций радиоэкологических данных. Известия академии наук Энергетика. №3, стр. 26, Москва, 1995 г.
14. Горбань А.Н., Россиев Д.А. Нейронные сети на персональном компьютере. - Новосибирск: Наука. 1996.

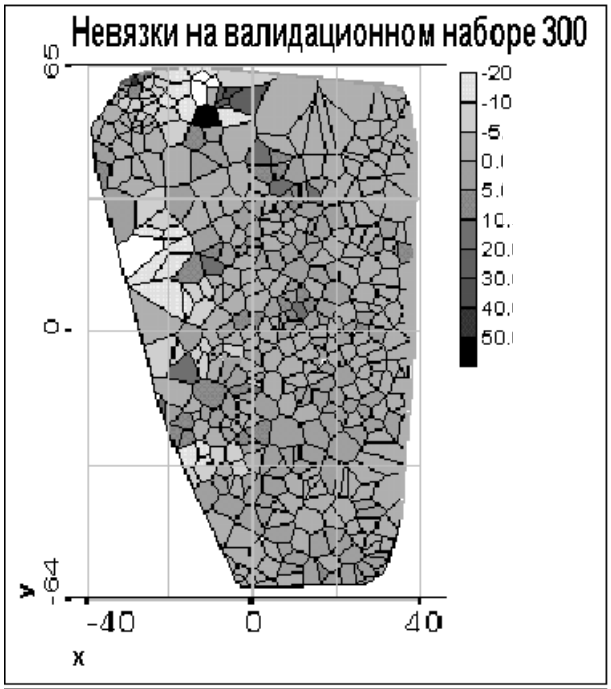
Приложение

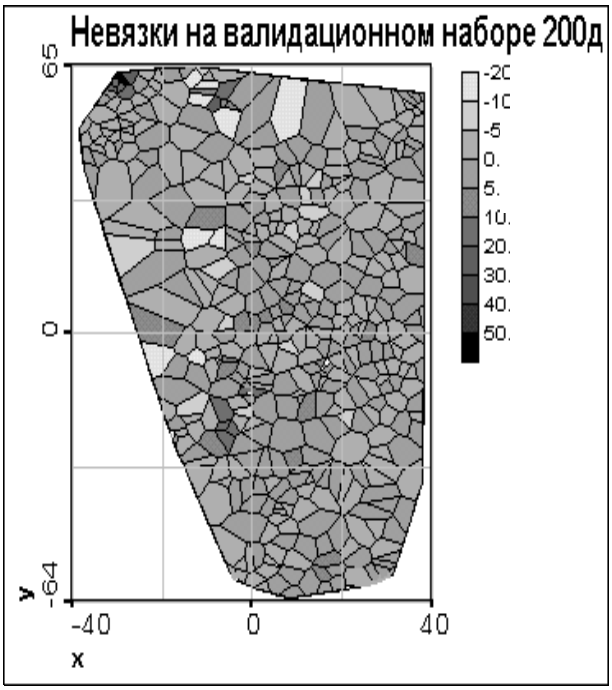
В данном приложении можно найти кар ценок и невязок НСОР для всех исследованнь первой части работы наборов данных.

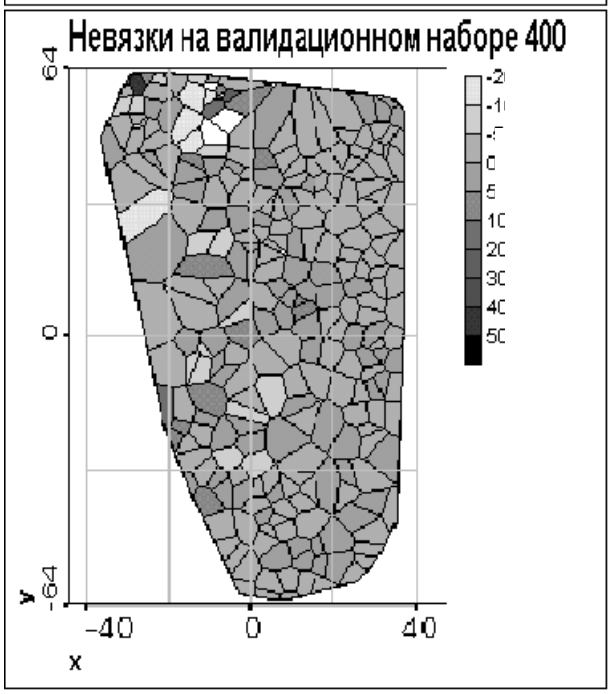


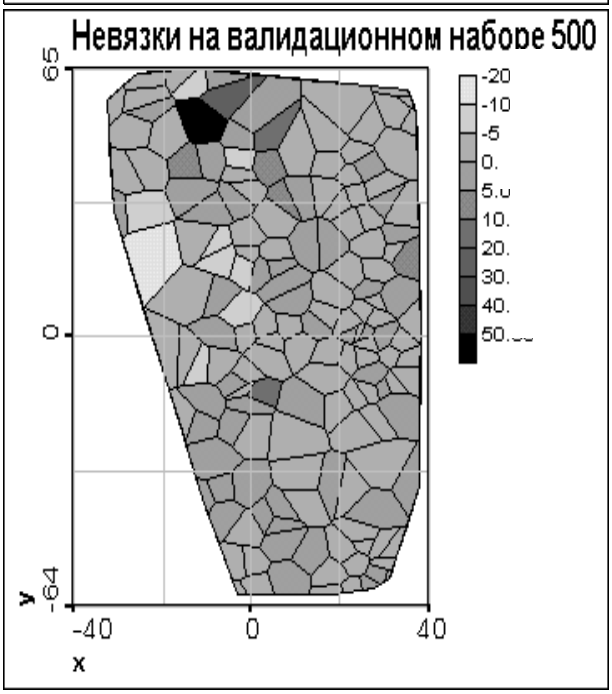

Невязки на тренировочном наборе 100
Невязки на тренировочном наборе 200
Невязки на тренировочном наборе 300
Невязки на тренировочном наборе 400
64
0
-64
y
-40
0
40
x
65
-20
-10
-5
0
5
10
20
30
40
50


Невязки на тренировочном наборе 200д
Невязки на тренировочном наборе 500
Невязки на валидационном наборе 100
Невязки на валидационном наборе 200
-20.000
-10.000
-5.000
0.0
5.000
10.000
20.000
30.000
40.000
50.000
64
0
-64
65
-40
0
40
x
y


Невязки на валидационном наборе 300
65
0
-64
y
-40
0
40
x
-20
-10
-5
0.
5.
10.
20.
30.
40.
50.


Невязки на валидационном наборе 200д
65
0
-64
y
-40
0
40
x
-20
-10
-5
0.
5.
10.
20.
30.
40.
50.


Невязки на валидационном наборе 400
64
0
-64
y
-40
0
40
x


Невязки на валидационном наборе 500
65
0
-64
y
-40
0
40
x
-20
-10
-5
0.
10.
20.
30.
40.