7 Нелинейные регрессионные модели

В главе 6 рассматривались регрессионные модели, являющиеся линейными по своей природе. Многие из них могут адаптироваться для нелинейных тенденций в данных посредством ручного добавления составляющих модели (например, квадратичных), что требует знаний о сути нелинейности данных.

Заметим, что существует множество нелинейных регрессионных моделей, применение которых, впрочем, возможно и без учета конкретной формы нелинейности. В этой главе рассматриваются некоторые из таких моделей: нейросети, многомерные адаптивные регрессионные сплайны (MARS), метод опорных векторов (SVM) и метод k ближайших соседей (KNN). Древовидные модели, рассматриваемые в главе 8, также нелинейны.

7.1. Нейросети

Нейросети, согласно заключениям Бишопа, Рипли и Титтерингтона (Bishop, 1995; Ripley, 1996; Titterington, 2010), представляют собой мощные методы нелинейной регрессии, в основу которых положены теории мозговой активности. Как и в случае с частными наименьшими квадратами, результат моделируется промежуточным набором ненаблюдаемых или *скрытых* переменных, представляющих собой линейные комбинации исходных предикторов, но, в отличие от моделей PLS, не оценивающихся в иерархическом контексте (рис. 7.1).

Как упоминалось, каждая скрытая переменная является линейной комбинацией, обычно преобразуемой нелинейной функцией `g(·)`, например логистической (сигмоидальной):

$$h_k(\mathrm{x}) = g(\beta_{0k} + \sum_{i=1}^{P} x_j \beta_{jk}), \quad \text{где}$$

$$g(u) = \frac{1}{1+e^{-u}}.$$

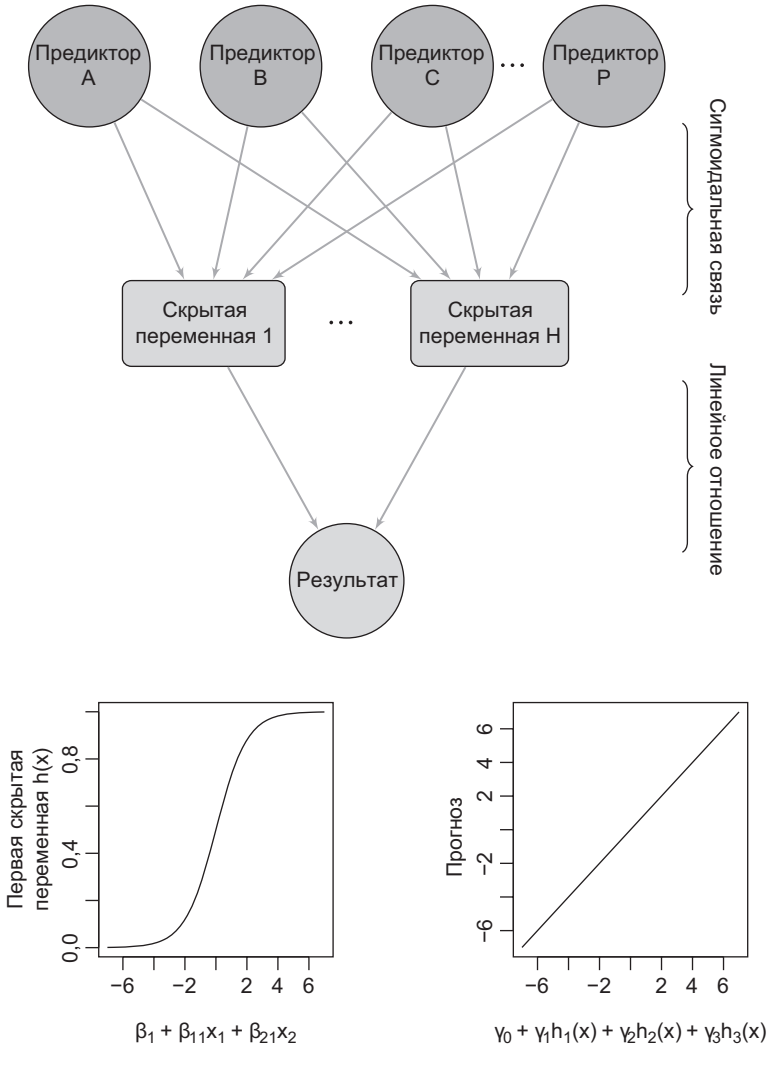

Рис. 7.1. Диаграмма нейросети с одним скрытым уровнем. Скрытые переменные являются линейными комбинациями предикторов, которые были преобразованы сигмоидальной функцией. Результат моделируется линейной комбинацией скрытых переменных

Коэффициенты β аналогичны коэффициентам регрессии; коэффициент β_{jk} описывает воздействие *j*-го предиктора на *k*-ю скрытую переменную. Модель нейросети обычно включает несколько скрытых переменных для моделирования результата. Следует учесть, что, в отличие от линейных комбинаций в PLS, не существует ограничений, которые бы помогали определять эти линейные комбинации. В этой

связи маловероятно, чтобы коэффициенты каждой переменной представляли некий осмысленный фрагмент информации.

После определения количества скрытых переменных каждая из них должна быть связана с результатом, например, следующим образом:

$$f(\mathrm{x}) = \gamma_0 + \sum_{k=1}^{H} \gamma_k h_k.$$

Для нейросетей такого типа и P предикторов оцениваются всего $H(P + 1) + H + 1$, причем эта величина быстро возрастает с ростом P. Вспомним в этой связи, что данные растворимости содержали 228 предикторов. Модель нейросети с тремя скрытыми переменными будет оценивать 691 параметр, тогда как у модели с пятью скрытыми переменными их число составит 1151. Если интерпретировать эту модель как нелинейную регрессионную, то при этом параметры обычно оптимизируются для минимизации суммы квадратов остатков. Обычно параметры инициализируются случайными значениями, после чего для решения уравнений используются специализированные алгоритмы. Представленный у Румелхарта (Rumelhart et al., 1986) алгоритм обратного распространения — высокоэффективная методология, работающая с производными для нахождения оптимальных параметров. Тем не менее решение уравнения не гарантирует, что полученный набор параметров будет однозначно лучше любого другого набора.

Кроме того, нейросети склонны к чрезмерному обучению связей между предикторами и реакцией из-за большого количества коэффициентов регрессии. Для решения этой проблемы существует несколько методов.

Во-первых, это описанный у Уанга и Венкатеша (Wang and Venkatesh, 1984) метод ранней остановки, основанный на заблаговременном прекращении работы итеративных алгоритмов по отысканию решения регрессионных уравнений. Процедура оптимизации прерывается, как только значение оценки параметров или частоты ошибок выходит за пределы допустимой стабильности. У этой процедуры есть очевидные минусы. Во-первых, как объективно оценить погрешность модели? Кажущаяся частота ошибок может оказаться завышенной (см. раздел 4.1). Определенные затруднения могут быть обусловлены и последующим разделением тренировочного набора. Кроме того, поскольку значениям измеренной частоты ошибок тоже присуща некоторая неопределенность, как убедиться в том, что она на самом деле возрастает?

Другой подход к преодолению чрезмерного обучения основан на *снижении весов* — штрафном методе регуляризации модели, сходном с гребневой регрессией (см. главу 6). Для больших коэффициентов регрессии добавляется штраф, с тем чтобы любое большое значение оказывало значительное влияние на погрешности модели. Формально это должно привести к минимизации альтернативной версии суммы квадратов погрешностей:

$$\sum_{i=1}^{n}(y_i - f_i(x))^2 + \lambda\sum_{k=1}^{H}\sum_{j=0}^{P}\beta_{jk}^2 + \lambda\sum_{k=0}^{H}\gamma_k^2$$

для заданного значения λ.

С ростом значения регуляризации аппроксимированная модель становится более плавной и с меньшей вероятностью вызовет чрезмерное обучение тренировочного набора. Этот параметр, наряду с количеством скрытых переменных, является параметром настройки модели. Допустимые значения λ лежат в диапазоне от 0 до 1. Поскольку коэффициенты регрессии суммируются, они должны измеряться по одной и той же шкале; следовательно, перед моделированием предикторы должны пройти центрирование и масштабирование.

Описываемая структура модели соответствует простейшей архитектуре нейросети: одноуровневой сети прямого распространения. Существует и много других разновидностей, включая модели с несколькими уровнями скрытых переменных (то есть уровень скрытых переменных, моделирующий другие скрытые переменные). В других вариантах архитектуры модели также присутствуют циклы, проходящие в обоих направлениях между уровнями. Для дальнейшей оптимизации моделей допускается удаление отдельных связей между объектами, в том числе посредством методов Байеса. Так, байесовская инфраструктура, описанная у Нила (Neal, 1996), автоматически интегрирует регуляризацию и выбор признаков. Этот метод считается высокоэффективным, хотя его реализация требует обстоятельного учета вычислительных аспектов оптимизируемой модели. Очень близка к нейросетям модель самоорганизующихся карт Кохонена (Kohonen, 1995), используемая и как неконтролируемый исследовательский метод, и как контролируемый метод для выполнения прогнозирования по Мельссену (Melssen et al., 2006).

С учетом проблем, связанных с оценкой большого количества параметров, аппроксимированная модель позволяет найти оценки параметров, являющиеся локально оптимальными. Заметим, что разные локально оптимальные решения могут приводить к формированию моделей, которые, сильно отличаясь друг от друга по формальным признакам, обладают при этом почти эквивалентной эффективностью. Подобная нестабильность модели иногда может затруднять ее применение. Альтернатива — создание нескольких моделей с разными начальными значениями и последующее усреднение результатов моделей для получения более стабильного прогноза, о чем упоминают, в частности, Перрон и Купер, Рипли, Тьюмер и Гош (Perrone and Cooper, 1993; Ripley, 1995; Tumer and Ghosh, 1996).

Усреднение моделей часто приводит к значительному положительному эффекту для нейросетей. Вместе с тем на такие модели не исключено отрицательное влияние высокой корреляции между предикторными переменными, использующими градиенты для оптимизации параметров модели.

Существуют два способа решения этой проблемы. Первый основан на предварительной фильтрации и удалении предикторов, связанных с высокими корреляциями, второй — на предшествующем моделированию выделении признаков для устранения корреляций (например, посредством анализа главных компонент). Оба способа требуют оптимизации меньшего количества составляющих модели, а это означает ускорение вычислений.

Для данных растворимости (см. ранее) использовались нейросети с усреднением моделей. Были вычислены три разных значения снижения весов (λ = 0,00, 0,01, 0,10) наряду с одним скрытым уровнем с размерами от 1 до 13 скрытых переменных. Итоговые прогнозы представляли результаты усреднения пяти разных нейросетей, созданных с разными исходными значениями параметров. Профили среднеквадратичной погрешности с перекрестной проверкой для этих моделей представлены на рис. 7.2.

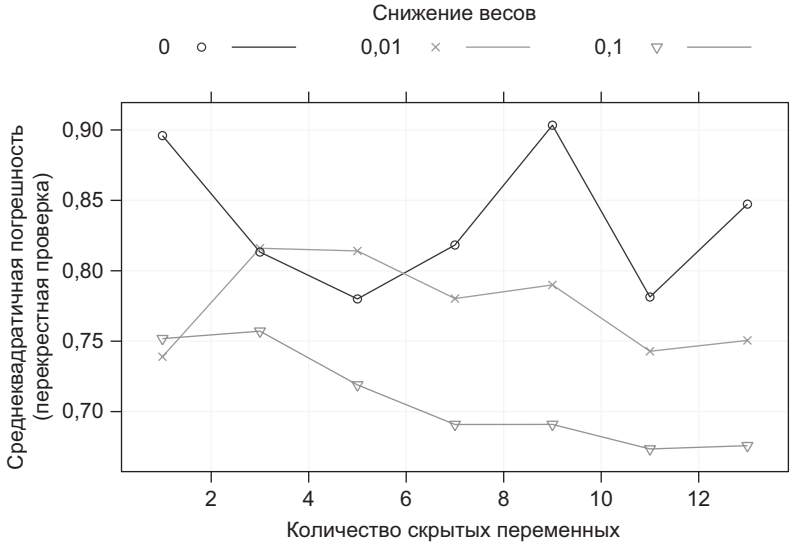

Рис. 7.2. Профили среднеквадратичной погрешности для модели нейросети. Оптимальная модель использует λ = 0,1 и 11 скрытых переменных

Нарастание снижения весов очевидным образом улучшает эффективность модели. Увеличение числа скрытых переменных сокращает ошибку модели. Оптимальная модель использовала 11 скрытых переменных с 2531 коэффициентом, и ее эффективность достаточно стабильна для высокой степени регуляризации (то есть λ = 0,1), что позволяет ожидать того же и от моделей с меньшим числом переменных и коэффициентов.

7.2. Многомерные адаптивные регрессионные сплайны

Как и нейросети и частные наименьшие квадраты, модель MARS, согласно Фридману (Friedman, 1991), использует суррогатные признаки вместо исходных предикторов. Но если PLS и нейросети базируются на линейных комбинациях предикторов, то MARS создает две контрастные версии предиктора для ввода в модель. Кроме того, суррогатные признаки MARS обычно являются функцией только одного или двух предикторов.

С учетом специфики MARS предикторы разделяются на две группы, в каждой из которых моделируются линейные связи между предиктором и результатом: в заданной точке разделения для предиктора двумя новыми признаками становятся «шарнирные» функции оригинала (рис. 7.3).

У одной функции нулевые значения следуют после точки разделения, у другой — до нее. Новые признаки добавляются в базовую модель линейной регрессии для оценки угла наклона и точки пересечения оси. Фактически эта схема создает *фрагментарную линейную модель*, в которой каждый новый признак моделирует изолированную часть исходных данных.

Для выбора точки разделения предварительно оценивается каждая точка данных этого предиктора. Для этого строится соответствующая модель линейной регрессии с потенциальными признаками и вычисляется соответствующая погрешность модели. После этого для модели используется комбинация предиктора/точки разделения, при которой достигается наименьшая погрешность. Способ преобразования предикторов допускает вычисление большого количества линейных регрессий. В некоторых реализациях MARS, включая приведенную здесь, кроме того, оценивается применимость простых линейных составляющих для каждого предиктора (то есть без шарнирной функции).

После создания исходной модели с первыми двумя признаками модель проводит другой исчерпывающий поиск для нахождения следующего набора признаков, которые для заданного исходного набора обеспечивают наилучшую аппроксимацию модели. Процесс продолжается до тех пор, пока не будет достигнута точка остановки (которая может задаваться пользователем).

При исходном поиске признаков в данных растворимости наименьшая частота ошибок достигалась с точкой разделения 5,9 для молекулярной массы. Полученные искусственные предикторы показаны на двух верхних областях рис. 7.3. У одного предиктора все значения, не достигающие точки разделения, обнуляются, а значения, превышающие точку разделения, остаются без изменений. Второй предиктор является зеркальным отражением первого. Вместо исходных данных два новых предиктора используются для прогнозирования результата в модели линейной регрессии.

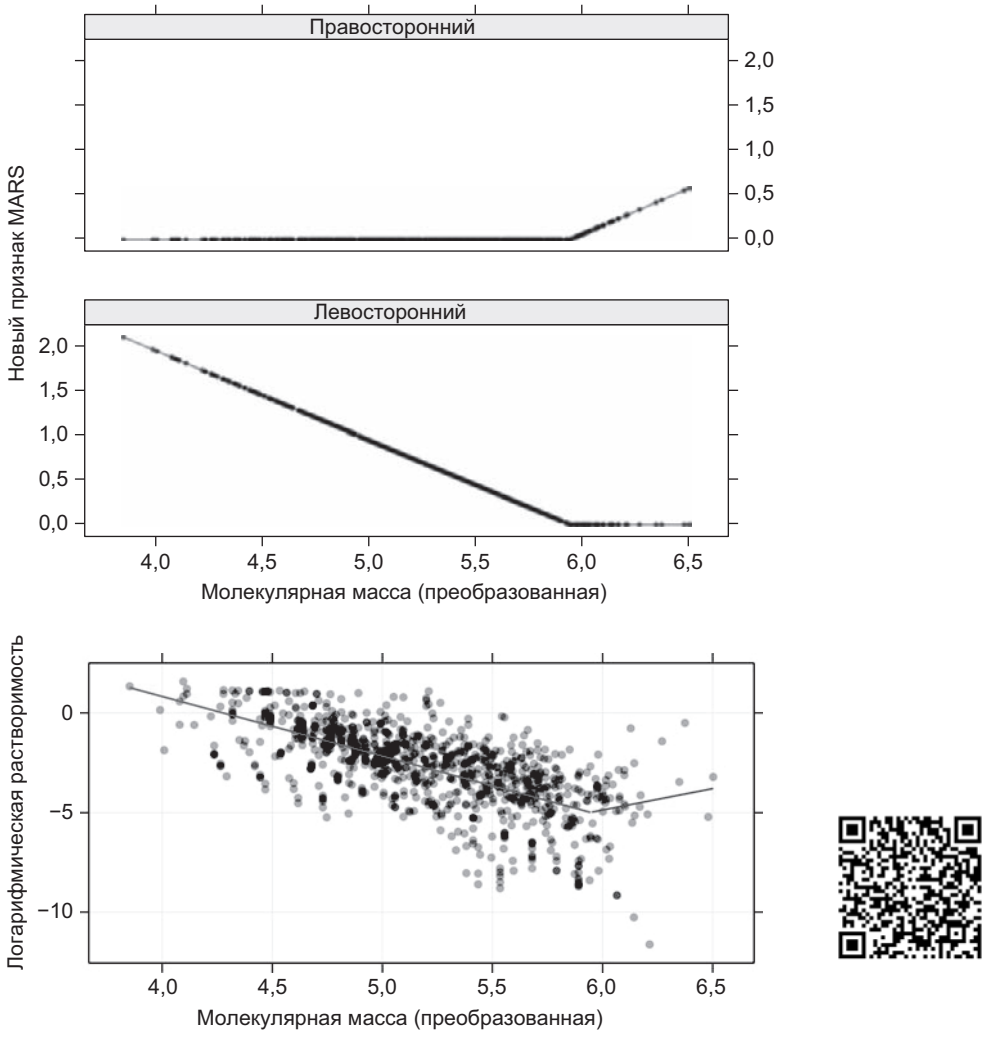

Рис. 7.3. Пример признаков, используемых MARS для данных растворимости. После нахождения точки разделения 5,9 для молекулярной массы были созданы два новых признака, которые использовались в модели линейной регрессии. Две верхние области показывают связь между исходным предиктором и двумя полученными признаками. На нижней области показано прогнозируемое соотношение при использовании двух признаков в модели линейной регрессии. Красная линия обозначает вклад «левосторонней» шарнирной функции, тогда как синяя линия связана с другим признаком

На нижней области рис. 7.3 показан результат линейной регрессии с двумя новыми признаками и фрагментарным характером связи. «Левосторонний» признак связывается с отрицательным наклоном, когда молекулярная масса меньше 5,9, тогда

как «правосторонний» признак оценивает положительный наклон для больших значений предиктора. Для новых признаков шарнирная функция может записываться в виде

$$h(x) = \begin{cases} x & x > 0 \\ 0 & x \le 0 \end{cases}. \tag{7.1}$$

Пара шарнирных функций обычно записывается в виде $h(x - a)$ и $h(a - x)$. Первая отлична от нуля при $x > a$, тогда как вторая отлична от нуля при $x < a$. Для модели MARS (см. рис. 7.3) фактическое уравнение модели имеет вид

$$-5+2{,}1 \times h(\text{МолМасса} - 5{,}94516) + 3 \times h(5{,}94516 - \text{МолМасса}).$$

Первое слагаемое в этой формуле (–5) определяет точку пересечения, второе слагаемое связано с правосторонним признаком на рис. 7.3, а третье — с левосторонним признаком. В табл. 7.1 представлены несколько первых этапов фазы генерирования признака (до усечения).

Таблица 7.1. Результаты нескольких итераций алгоритма MARS до усечения

Предиктор	Тип	Точка разделения	Среднеквадратичная погрешность	Коэффициент
Intercept			4,193	–9,33
MolWeight	Правосторонний	5,95	2,351	–3,23
MolWeight	Левосторонний	5,95	1,148	0,66
SurfaceArea1	Правосторонний	1,96	0,935	0,19
SurfaceArea1	Левосторонний	1,96	0,861	–0,66
NumNonHAtoms	Правосторонний	3,00	0,803	–7,51
NumNonHAtoms	Левосторонний	3,00	0,761	8,53
FP137	Линейный		0,727	1,24
NumOxygen	Правосторонний	1,39	0,701	2,22
NumOxygen	Левосторонний	1,39	0,683	–0,43
NumNonHBonds	Правосторонний	2,58	0,670	2,21
NumNonHBonds	Левосторонний	2,58	0,662	–3,29

Примечание. Для оценки среднеквадратичной погрешности использовалась статистика GCV.