# Линейная регрессия.

Брескану Никита.

## 1 Основные определения в регрессии

Задача регрессии заключается в приближении зависимости между данными функциональной зависимостью:

$$Y \approx f(X)$$

где $f(\cdot)$ — обучаемая функция. Чаще всего, она обучается на каком-то семействе функций, например, линейных. Обучение функции по сути является «подгонкой» под известные данные с целью дальнейшего предсказания на неизвестных.

В этой формуле $X$ может быть объектом любой природы, но чаще всего это вектор. Переменные этого вектора называются (независимыми) **признаками** (регрессорами; независимыми переменными).

Обычно $Y$ — это скаляр, который называется **таргетом** (зависимой переменной, целевой переменной).

Для обучения обычно создаётся выборка $\mathbb{D}$ — множество объектов, которые известны модели полностью. Будем называть это множество **обучающей выборкой**.

$$\mathbb{D} = \{(X_i, Y_i)\}_{i=1}^{L}$$

Здесь $\mathbb{D}$ — это обучающая выборка, пара $(X_i, Y_i)$ — это **объект** выборки, в котором $X_i$ — некоторое признаковое описание, а $Y_i$ — скалярный таргет.

Например, если дом является объектом, то его описание признаки могут быть числом комнат, площадью и т.д, а таргетом — цена.

## 2 Одномерный случай

Рассмотрим самый простой случай: необходимо исследовать зависимость одного признака от другого. На примере это датасет Old Faithful Geyser Data, исследующий зависимость между длительностью извержения гейзера и интервала между этими извержениями.

Обозначим гейзеры через пару: $(X_i, Y_i)$, где $X_i \in \mathbb{R}$ — длительность извержения, $Y_i \in \mathbb{R}$ — интервал, а также $L$ — число гейзеров.

В линейной регрессии мы предполагаем наличие зависимости, близкой к линейной. Судя по графику (1), такое предположение имеет смысл.

$$Y \approx w_0 + w_1 X$$

где параметры $w_0, w_1$ необходимо подобрать. Для удобства обозначим их за $w = (w_0, w_1)$. Традиционно $w_0$ называется **смещением** (bias, intercept). Знак приближённого равенства пока что не полностью описывает задачу, и поставлен для интуитивного понимания.

**Ошибкой** будем называть разность между линейным приближением таргета и истинным таргетом:

$$\epsilon = Y - (w_0 + w_1 X)$$

Таким образом, задача, ранее поставленная как «подгонка» параметров под данные, формализуется как минимизация ошибки $\epsilon$.

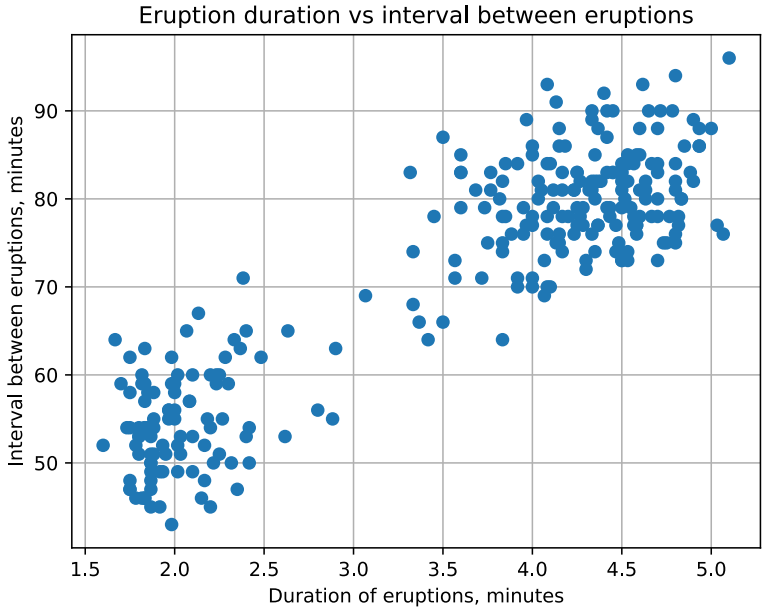


Рис. 1: Длительность извержения и интервал между извержениями гейзеров

Так как мы имеем дело с неидеальной линейной зависимостью, то просто приравнять $\epsilon$ к нулю нельзя. Обычно вместо этого задача сводится к минимизации какой-то функции от выборочных $\epsilon_i$. Будем называть такую функцию **эмпирическим риском** (лоссом, $Q$).

$$Q(\mathbb{D}, w) = \frac{1}{L} \sum_{i=1}^{L} \mathbb{L}(\epsilon_i)$$

где $\mathbb{L}(\cdot)$ — какая-нибудь (желательно положительно определённая) функция. Будем называть $\mathbb{L}$ **функцией потерь**. Также удобным свойством для оптимизации будет являться гладкость $\mathbb{L}$. Самая простая такая функция, это $\mathbb{L}(x) = x^2$ (квадратичная функция потерь), а эмпирический риск с этой функцией потерь будем называть **Mean Squared Error** (MSE).

$$\text{MSE}(\mathbb{D}, w) = \frac{1}{L} \sum_{i=1}^{L} (Y_i - (w_0 + w_1 X_i))^2$$

Тогда стоит следующая оптимизационная задача:

$$\text{MSE}(\mathbb{D}, w) \to \min_w$$

Такая постановка задачи является, вообще говоря, немного неполной и упрощённой. Полная постановка представлена в аппендиксе (A).

В одномерном случае эту задачу можно решить аналитически методами первого курса матанализа (B). Для датасета с гейзерами решение имеет вид:

$$\overline{Y} \approx 33.474 + 10.730 X$$

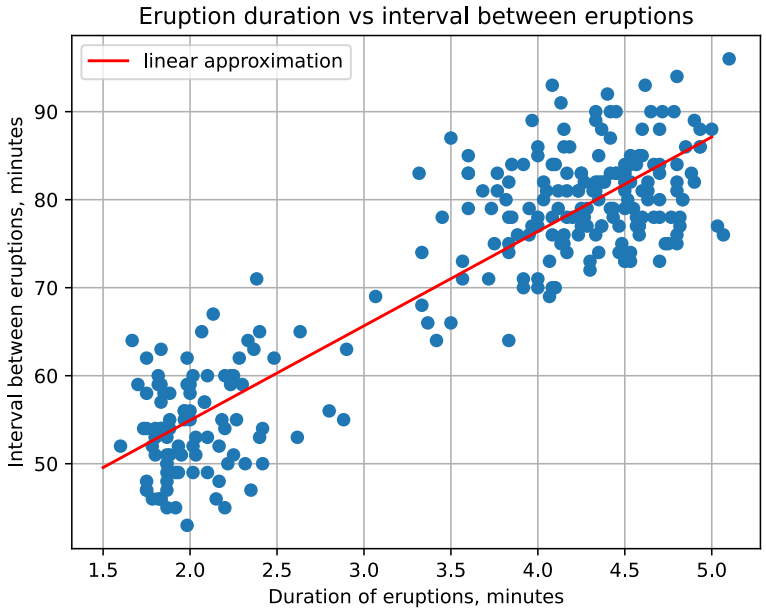


Рис. 2: Линейная регрессия для датасета с гейзерами

# 3 Многомерный случай

## 3.1 Постановка задачи

Перейдём к случаю, когда признаков много. Пусть $D$ - число признаков. Такая задача встречается гораздо чаще и является существенно более сложной. Поменяем обозначения: через $X_j$ будем обозначать j-ый признак, через $\mathbf{X}$ матрицу размера $L \times D$, где $X_{ij}$ - j-ый признак i-ого объекта, через $\mathbf{Y}$ - вектор размера $L$, состоящий из таргетов для каждого объекта. Теперь $w = (w_0, w_1, \ldots, w_D)$.

Мы хотим, чтобы выполнялось:

$$Y \approx w_0 + w_1 X_1 + \ldots + w_M X_D$$

Теперь вместо прямой, модель описывает проводит гиперплоскость в D-мерном пространстве.

Естественным образом обобщаются формулы для эмпирического риска и, в частности, MSE, при этом задача имеет такой же вид:

$$\text{MSE}(\mathbb{D}, w) \to \min_w$$

Стоит заметить, что если добавить искуственный признак $X_0 \equiv 1$, то тогда $w_0$ - коэффициент у этого признака, а смещение равно нулю.

Далее, не ограничивая общность, будем считать, что смещение равно нулю (через ввод искуственного признака), а $D$ - число признаков, учитывая искуственный.

## 3.2 Аналитическое решение

В таких обозначениях, задача принимает матричный вид:

$$\mathbf{Y} \approx \mathbf{X}w$$

То есть, это стандартная система линейных уравнений (СЛАУ).

При этом MSE будет выглядеть, как:

$$\text{MSE}(\mathbb{D}, w) = \frac{1}{L}\|\mathbf{Y} - \mathbf{X}w\|^2$$

где $\|\cdot\|$ - евклидова норма.

Невязкой уравнения будет ошибка $\epsilon$.

В линейной алгебре вы проходили, что если СЛАУ не имеет решения, а очень хочется чтобы оно было, то можно найти псевдорешение. При этом оно будет минимизировать невязку, что в нашем случае совпадает с минимизацией MSE.

Поэтому мы можем домножить обе части на $\mathbf{X}^T$:

$$\mathbf{X}^T\mathbf{X}w = \mathbf{X}^T\mathbf{Y}$$

Квадратная матрица размера $D \times D$ $\mathbf{X}^T\mathbf{X}$ обратима $\iff$ столбцы $\mathbf{X}$ линейно независимы.

Линейная независимость столбцов матрицы означает, если смотреть на каждый признак, как на столбец, то один признак не может выражаться через все остальные. Когда это свойство выполняется, или очень близко к выполнению, это называется проблемой мультиколлинеарности. О ней мы поговорим позже, затрагивая регуляризацию 5.

Отсутствие линейной зависимости — это разумное требование, т.к. при ней зависимые признаки можно выбросить.

Чаще всего, когда признаков больше объектов ($D > L$), это требование выполняется.

Если сделать предположение о линейной независимости признаков, то можно в явном виде найти псевдорешение:

$$w = (\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{Y}$$

# 4 Градиентный спуск

Мы разобрали аналитическое решение линейной регрессии. Теперь подумаем над алгоритмической сложностью. Так как $\mathbf{X}^T\mathbf{X}$ — матрица размера $D \times D$, то её обращение занимает $\text{O}(D^3)$ операций. Немного эффективнее вместо этого сразу решать СЛАУ $\mathbf{X}^T\mathbf{X}w = \mathbf{X}^T\mathbf{Y}$, но алгоритмическая сложность от этого не поменяется. Существуют также и итеративные методы решения СЛАУ, в которых на каждой итерации новый вектор $w^{(k)}$ уменьшает невязку и гарантированно сходится по норме к точному решению при $k \to \infty$.

Одним из таких методов является **градиентный спуск**. Он удобен, когда матрица чрезмерно большая.

Обозначим через $k$ номер итерации, а через $w^{(k)}$ — значения весов на k-ой итерации. $w^{(0)}$ - начальные веса. Рассмотрим самый простой пример градиентного спуска:

$$w^{(0)} = 0$$
$$w^{(k+1)} \leftarrow w^{(k)} - \gamma_k \nabla Q(w^{(k)})$$

где $\nabla f(\cdot)$ — градиент минимизируемой функции, $\gamma_k$ — **размер шага** (learning rate) на k-ой итерации.

Критерий остановки в простейшем случае выбирается среди следующих двух:

- $|Q(w^{(k+1)}) - Q(w^{(k)})| < T_f$
- $\|w^{(k+1)}) - w^{(k)}\| < T_w$
- $k = K_{\text{terminal}}$

где $T_f, T_w$ — какие-то зафиксированные пороги (tolerance).

Чтобы вывести формулу итерации градиентного спуска, нужно вначале посчитать градиент MSE (т.е. минимизируемой функции) по весам $w$. Это сделано в аппендиксе (C), а здесь сразу выпишем результат:

$$\nabla \text{MSE}(w) = \frac{2}{L}\mathbf{X}^T(\mathbf{X}w - \mathbf{Y})$$

Теперь, после подстановки градиента MSE формула итерации будет следующая:

$$w^{(k+1)} \leftarrow w^{(k)} - \frac{2\gamma_k}{L}\mathbf{X}^T(\mathbf{X}w^{(k)} - \mathbf{Y})$$

Подумаем над вычислительной сложностью итерации. Заранее можно посчитать матрицы $\mathbf{X}^T\mathbf{X}$ и $\mathbf{XY}$. Тогда самым сложным шагом будет умножение заранее посчитанной матрицы $\mathbf{X}^T\mathbf{X}$ на $w^{(k)}$. Так как размерность матрицы $D \times D$, а вектора — $D$, то сложность такого умножения будет $\text{O}(D^2)$. Если $K$ — общее число итераций, то вычислительная сложность градиентного спуска будет равна $\text{O}(KD^2)$.

Зачастую признаков много, поэтому $D > K$, и градиентный спуск оказывается эффективнее, чем решение СЛАУ. Присутсвует свобода выбора между точностью решения и временем оптимизации. К тому же, при большом количестве объектов можно использовать модификацию — стохастический градиентный спуск.

## 4.1 Гиперпараметры

В машинном обучении принято называть **параметрами** модели её обучаемые веса, а **гиперпараметрами** — необучаемые значения, влияющие на процесс обучения (можно сказать, конфигурацию обучения).

У классического градиентного спуска всего 2 гиперпараметра — это начальное приближение $w^{(0)}$ и размер шага $\gamma_k$.

Начальное приближение, мягко говоря, не так важно: оптимизационные свойства задачи таковы, что вы сойдётесь из любого вектора к решению. Единственная разница — если начали очень далеко от решения, то вам понадобится больше итераций. К счастью, в линейной регрессии, особенно после нормализации (о ней поговорим позже, ) веса находятся примерно около нуля. Какие-то признаки вносят положительный вклад (коэффициенты больше 0), какие-то — отрицательный (меньше 0). Поэтому начальное приближение $w^{(0)} = 0$ является простым и надёжным способом хорошо сойтись.

Помимо нулевых начальных весов, существуют чуть более экзотические способы инициализации, например:

$$w^{(0)} \sim \mathcal{N}(0, \sigma^2)$$
$$w^{(0)} \sim \mathcal{U}[-\epsilon, \epsilon]$$

где $\sigma$ и $\epsilon$ следует взять небольшими числами, где-то в диапазоне от 0.01 до 0.1. Обычно на практике хорошо подходит значение для $\epsilon$ или для $\sigma$, равное $\frac{1}{\sqrt{D}}$.

Самым важным гиперпараметром является размер шага $\gamma_k$: он непосредственно влияет на скорость сходимости. Вначале рассмотрим самый простой случай — $\gamma_k \equiv \gamma$, т.е. шаг постоянный. Если $\gamma$ слишком мал, то сходимость точная, но долгая; если слишком велик, то веса быстро приблизятся к окрестности решения, но могут так и не сойтись;

Эти наблюдения приводят к идее, что в начале стоит взять большой шаг, чтобы быстро сойтись к нужному месту, а уже позже — маленький, чтобы сойтись без перескакиваний. Эта интуиция реализовывается в **планировщике скорости обучения** (learning rate scheduler). Самый распространённый его вид — это следующий (inverse scale):

$$\gamma_k = \frac{\gamma_0}{(k+1)^\beta}$$

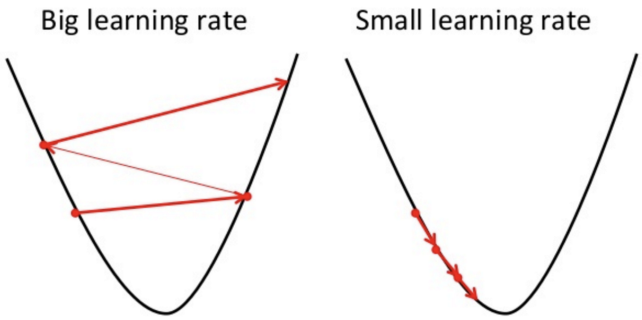


Рис. 3: Иллюстрация большого и малого постоянного шага

# 5 Регуляризация

Ранее мы получали аналитическую формулу и предполагали, что $\mathbf{X}^T\mathbf{X}$ невырождена. Однако даже если это так, матрица может оказаться плохо обусловленной (как бы, почти вырожденной). Давайте вспомним, почему это плохо:

- Плохая обобщающая способность. Плохо обусловленные СЛАУ имеют решения, сильно чувствительные к правой части (то есть, к таргетам). Из-за этого модель может слишком сильно подстроиться под обучающую выборку, и плохо показывать себя на других объектах. В машинном обучении такой эффект называют **переобучением**.
- Численная неустойчивость. В таких СЛАУ вычислительные погрешности могут быстро разрастаться, что приводит к плохому результату. Например, какие-то коэффициенты $w_i$ могут оказаться очень большими. А при малом изменении входных данных $w$ окажутся совершенно другими.

Можно сказать, матрица $\mathbf{X}^T\mathbf{X}$ плохо обусловлена $\iff$ столбцы матрицы «почти линейно зависимы». В машинном обучении это явление называется **мультиколлинеарностью**.

Как уже было сказано, мультиколлинеарность может приводить к большим коэффициентам. Поэтому, интуитивно можно понять, что штраф за большую норму коэффициентов — это хорошо. Поэтому добавим к эмпирическому риску слагаемое, отвечающее за этот штраф. Добавление такого штрафа называется **регуляризацией**.

$$Q_\alpha(\mathbb{D}, w) = Q(\mathbb{D}, w) + \frac{1}{2}\alpha\|w\|^2$$

где $\alpha$ — коэффициент регуляризации (обычно маленький, около 0.001), а множитель $1/2$ поставлен только ради удобства.

Если решить новую минимизационную задачу, то аналитический ответ будет таков:

$$w = (\mathbf{X}^T\mathbf{X} + \alpha I_D)^{-1}\mathbf{X}^T\mathbf{Y}$$

, где $I_D$ - единичная матрица размера $D \times D$. При достаточно большом $\alpha$ матрица $\mathbf{X}^T\mathbf{X} + \alpha I_D$ становится хорошо обусловленной, что делает её обращение гораздо более «приятным».

Регуляризация, которую мы рассмотрели, заключается в добавлении нормы L2 нормы (евклидовой). Поэтому её называют **L2 регуляризацией**, а линейную регрессию с L2 регуляризацией - **Ridge**. Название Ridge появилось из-за того, что в случае мультиколлинеарности ландшафт эмпирического риска имеет «хребет» (перевод слова Ridge), а добавление L2 штрафа его убирает и стабилизирует функцию.

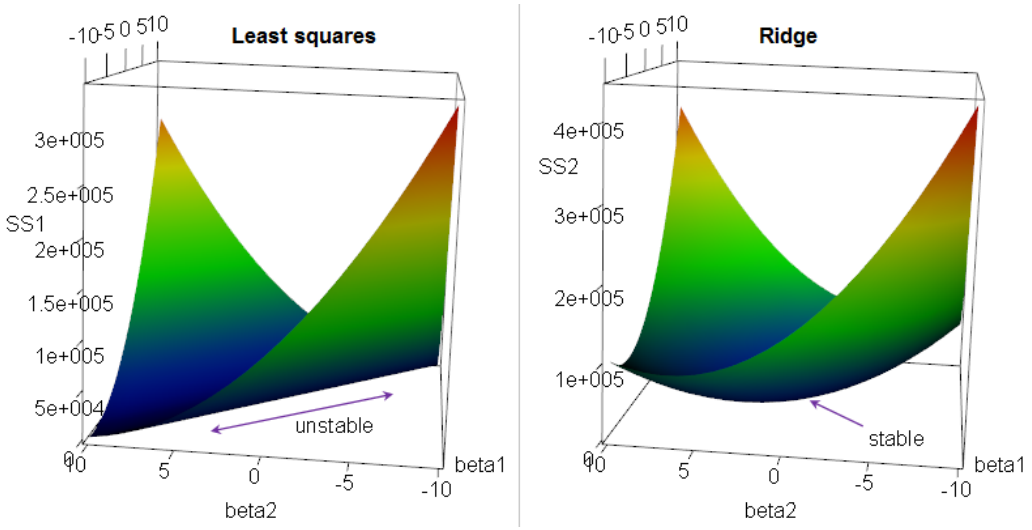


Рис. 4: Ландшафт эмпирического риска для стандартной линейной регрессии и для Ridge

## 5.1 Отбор признаков

Если в штрафе за большие параметры использовать L1 норму (сумму модулей), а не L2, то такой штраф называется **L1 регуляризацией (LASSO)**. LASSO — это абревиатура от Least Absolute Shrinkage and Selection Operator. Для понимания этого названия стоит объяснить все слова:

- Least - означает минимизацию (эмпирического риска)
- Absolute - использование L1 нормы (модулей)
- Shrinkage - штрафование больших весов, т.е. их «сокращение»
- Selection - отбор признаков (будет описан далее)
- Operator - процедура минимизации

$$Q_\gamma(\mathbb{D}, w) = Q(\mathbb{D}, w) + \gamma\|w\|_1$$

где $\|x\|_1 = \sum_{i=1}^{M} |x_i|$ - L1 норма.

Заметим, Lasso, в отличие от Ridge, не гладкая: $|\cdot|$ не дифференцируем в точке 0.

Самое простое, что можно сделать — доопределить производную нулём в точке 0, и применять обычный градиентный спуск. Тогда производную модуля можно записать через $\operatorname{sign} X$. Но у такого метода можно выделить недостаток: на каждой итерации от всех координат $w_i$ отнимается их знак $\operatorname{sign} w_i$ (домноженный на константу). Таким образом, легко перескочить через 0, и веса могут колебаться в окрестности нуля. Имеет смысл при малом значении весов сразу обнулять их. Эта идея воплощается в проксимальном градиентном спуске, о котором мы не будем говорить в этом семестре.

Само название LASSO (4 буква S: Selection) подчёркивает, что этот метод склонен к отбору признаков, т.е. к занулению коэффициентов у ненужных признаков. Если $w_i$ близко к 0, то $w_i^2 << |w_i|$, т.е. L1 штрафует гораздо больше, чем L2, поэтому в нём веса более склонны к занулению.

Использование LASSO — это один из простейших методов отбора признаков, причём число отобранных признаков можно контролировать с помощью коэффициента $\gamma$: чем больше $\gamma$, тем больше коэффициентов LASSO может занулить.

# 6 Вероятностная интерпретация

До сих пор мы не касались вероятностей, хотя конечно, использование термина выборка, наличие выборочных средних, дисперсий и ковариаций в формулах намекают на их присутсвие.

Пусть $Y$ — случайная величина, а $X$ — случайный вектор, состоящий из признаков объекта.

Заметим, что в теории одним и тем же признакам могут соответсвовать разные таргеты. Можно сказать, эти таргеты образуют условное распределение $Y|X$. Наша задача состоит в том, чтобы узнать хотя бы что-то об этом распределении. Самая простая характеристика распределения - это матожидание. Будем оценивать его с помощью линейной регрессии. А дисперсию посчитаем постоянной (см. далее).

$$\mathbb{E}(Y|X = x) = w_1 x_1 + \ldots + w_D x_D = \langle x, w \rangle$$

Обозначим модель через $f(X) = \langle X, w \rangle$. Сделаем предположение о нормальной распределённости таргетов:

$$Y|X \sim \mathcal{N}(f(X), \sigma^2) \iff \epsilon \sim \mathcal{N}(0, \sigma^2)$$

то есть, эквивалентно, все ошибки независимы и одинаково распределены с одним и тем же стандартным отклонением $\sigma$.

Наша задача — оценить параметры модели $w$. Для этого можно использовать метод максимума правдоподобия.

Пусть выбраны веса $\hat{w}$. Для таких весов мы можем получить выборку ошибок: $\{\hat{\epsilon_i}\}_{i=1}^{L}$. Запишем вероятность такой выборки:

$$\mathbb{P}(\epsilon_1 = \hat{\epsilon_1}, \ldots, \epsilon_L = \hat{\epsilon_L}) = \prod_{i=1}^{L} \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{\hat{\epsilon_i}^2}{2\sigma^2}} \to \max_w$$

Если прологарифмировать, и умножить на -1, это будет эквивалентно:

$$-\ln \mathbb{P}(\epsilon_1 = \hat{\epsilon_1}, \ldots, \epsilon_L = \hat{\epsilon_L}) = \sum_{i=1}^{L} (-\ln \frac{1}{\sqrt{2\pi}\sigma} + \frac{\hat{\epsilon_i}^2}{2\sigma^2}) \to \min_w$$

Убрав константы, не влияющие на минимизацию, получим:

$$\sum_{i=1}^{L} \hat{\epsilon_i}^2 \to \min_w \iff \sum_{i=1}^{L} (f(X_i) - Y_i)^2 \to \min_w$$

Таким образом, мы вывели MSE, причём не только для линейной регрессии, но и для любой другой модели, при условии, что выполнено предположение о нормальности ошибок.

Стоит заметить, что предположение о постоянной дисперсии условного распределения далеко не всегда может быть выполнено. Более сложные модели, такие как гаусовские процессы, способны оценивать и дисперсию тоже.

# 7 Преобразования признаков

Геометрически линейная регрессия проводит гиперплоскость в пространстве таргета и всех признаков. Но иногда гиперплоскости недостаточно, чтобы описать зависимость. В этом случае можно добавлять нелинейные преобразования признаков (линейные нельзя, т.к. возникнет мультиколлинеарность). Например, можно добавить квадрат признака, или другую степень.

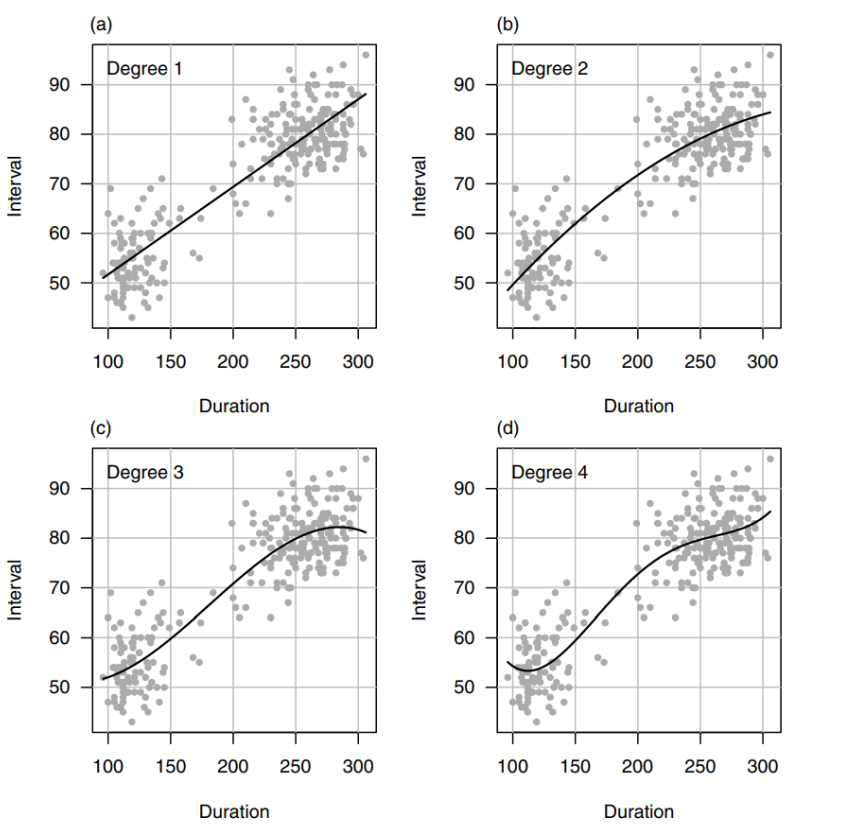


Рис. 5: Добавление степеней признака в датасете с гейзерами

Если добавить все перемножения пар признаков и все их квадраты, то такая линейная регрессия сможет проводить произвольные квадратичные поверхности. Если признак может принимать значения разного порядка, обычно вместо него берут его логарифм.

# 8 Нормализация

Разные признаки могут принимать значения разного масштаба. Например, цена дома в рублях исчисляется миллионами, а его площадь в квадратных метрах — до 3 порядка. Для использования линейной регрессии на практике хочется, чтобы признаки были такими, какие они давались изначально. Однако это может ухудшить оптимизацию и усложнить интерпретирацию решения:

- числа большого порядка негативно влияют на численную стабильность.
- физический смысл формулы может оказаться странным. например, если складываются метры и килограммы. Почему не сантиметры и граммы?

- сложнее интепретировать «важность» признаков. После нормализации все коэффициенты имеют один и тот же масштаб, поэтому признаки можно сравнивать по «важности», смотря на их коэффициенты: чем больше по модулю коэффициент, тем больше признак влияет на таргет (больше ковариация).

- самый главный минус: Ridge и LASSO являются **неинвариантными к масштабированию**, т.е. оптимизация даст другой результат, нежели масштабирование $\Longrightarrow$ оптимизация $\Longrightarrow$ масштабирование обратно. Причина этому — разная сила регуляризации в этих случаях: где-то больше, а где-то меньше.

Поэтому хорошим тоном является масштабирование признаков до одинакового диапазона, которое обычно называют **нормализацией**. Нормализация бывает разной. Можно привести все признаки к отрезку $[0, 1]$, но чаще всего используется другой способ:

$$\hat{X}_{ij} = \frac{X_{ij} - \overline{X}_j}{\sqrt{\mathrm{SD}_{X_j}}}$$

где $\hat{X}_{ij}$ — нормализованный признак. По сути, признак нормализуется до матожидания 0 и стандартного отклонения 1 по всем объектам.

# Аппендикс для ботанов

## A Формальная постановка задачи машинного обучения

Для формальной постановки задачи нужно будет пользоваться языком теории вероятностей.

Пусть $X$ — случайная величина, характеризующая признаковое описание объекта и принимающая значения из пространства $\mathbb{X}$; $Y$ — случайная величина, характеризующая таргет и принимающая значения из $\mathbb{Y}$ (в случае регрессии это $\mathbb{R}$). Введём функцию потерь $\mathbb{L} : \mathbb{Y} \times \mathbb{Y} \to \mathbb{R}$. Обозначим модель машинного обучения (алгоритм) через $a : \mathbb{X} \to \mathbb{Y}$.

Введём **риск** $R(a)$:

$$R(a) = \mathbb{E}_{X,Y}[\mathbb{L}(a(X), Y)]$$

Если функция потерь по смыслу означает «потерю» на одном объекте, то риск - это средняя «потеря» на всевозможных объектах.

Задача машинного обучения формулируется как минимизация риска:

$$R(a) \to \min_{a \in M}$$

где $M$ — множество перебираемых алгоритмов.

Естественно, в реальности мы не знаем все объекты, т.е. всё совместное распределение $X, Y$. Нам доступна только выборка из этого совместного распределения $\mathbb{D} = \{(X_i, Y_i)\}_{i=1}^{L}$, называемая **обучающей выборкой**.

Поэтому теперь мы можем работать только с **эмпирическим риском** $Q$ — статистической оценкой риска $R$ на выборке $\mathbb{D}$. Матожидание всегда оцениваниется средним арифметическим, поэтому эмпирический риск имеет вид:

$$Q(a) = \sum_{i=1}^{L} \mathbb{L}(a(X_i), Y_i)$$

Предполагается, что обучающая выборка хорошо описывает всё совместное распределение, поэтому минимизация риска заменяется на минимизацию эмпирического риска:

$$Q(a) \to \min_{a \in M}$$

# B Аналитическое решение одномерной линейной регрессии

Введём некоторые величины:

$$\overline{X} = \frac{1}{L}\sum_{i=1}^{L} X_i$$

$$\overline{Y} = \frac{1}{L}\sum_{i=1}^{L} Y_i$$

$$\mathrm{SD}_X = \frac{1}{L}\sum_{i=1}^{L}(X_i - \overline{X})^2 = \frac{1}{L}\sum_{i=1}^{L} X_i(X_i - \overline{X})$$

$$\mathrm{S}_{XY} = \frac{1}{L}\sum_{i=1}^{L}(X_i - \overline{X})(Y_i - \overline{Y}) = \frac{1}{L}\sum_{i=1}^{L} Y_i(X_i - \overline{X}) = \frac{1}{L}\sum_{i=1}^{L} X_i(Y_i - \overline{Y})$$

где $\mathrm{SD}_X$ — (смещенная) выборочная дисперсия $X$; $\mathrm{S}_{XY}$ — (смещённая) выборочная ковариация $X$ и $Y$.

Приступим к решению:

$$\frac{\partial Q}{\partial w_0} = \frac{2}{L}\sum_{i=1}^{L}(w_0 + w_1 X_i - Y_i) = 0 \implies w_0 = \overline{Y} - w_1\overline{X}$$

$$\frac{\partial Q}{\partial w_1} = \frac{2}{L}\sum_{i=1}^{L} X_i(w_0 + w_1 X_i - Y_i) = \frac{2}{L}\sum_{i=1}^{L} X_i(\overline{Y} - w_1\overline{X} + w_1 X_i - Y_i) = 0$$

$$2w_1\mathrm{SD}_X = 2\mathrm{S}_{XY} \implies w_1 = \frac{\mathrm{S}_{XY}}{\mathrm{SD}_X}$$

Мы решили оптимизационную задачу аналитически. Посмотрим на результаты:

Коэффициент $w_1$ пропрорционален выборочной ковариации, т.е. отражает, насколько поменяется $Y$ при изменении $X$.

А $w_0$ подобран так, чтобы при усреднении желаемая формула точно выполнялась: $\overline{Y} = w_0 + w_1\overline{X}$.

# C Вычисление градиента MSE по весам

Вначале посчитаем вспомогательные градиенты:

$$\frac{d}{dx_i}\langle x, a\rangle = a_i \implies \frac{d}{dx}\langle x, a\rangle = (\frac{d}{dx_1}\langle x, a\rangle, \ldots, \frac{d}{dx_n}\langle x, a\rangle) = (a_1, \ldots, a_n) = a$$

Пусть $A = A^T$ — симметричная матрица. Тогда примем за данное (без доказательства) формулу:

$$\frac{d}{dx}\langle Ax, x\rangle = (A + A^T)x = 2Ax$$

Возможным объяснением «на пальцах» этого равенства будет то, что мы как бы пользуемся формулой произведения, перекидываем $A$ на другую сторону скалярного произведения, она становится $A^T$, и пользуемся уже известным градиентом скалярного произведения.

Теперь применим всё вместе:

$$\frac{d}{dw}\|\mathbf{Y}-\mathbf{X}w\|^2 = \frac{d}{dw}\langle \mathbf{X}w-\mathbf{Y}, \mathbf{X}w-\mathbf{Y}\rangle = \frac{d}{dw}\langle \mathbf{X}^T\mathbf{X}w, w\rangle - 2\frac{d}{dw}\langle w, \mathbf{X}^T\mathbf{Y}\rangle = 2\mathbf{X}^T(\mathbf{X}w-\mathbf{Y})$$

$$Q(w) = \frac{1}{L}\|\mathbf{Y}-\mathbf{X}w\|^2 \implies \nabla Q(w) = \frac{2}{L}\mathbf{X}^T(\mathbf{X}w-\mathbf{Y})$$

Заметим, что если приравнять это к нулю, то получим тоже самое равенство 3.2, к которому мы пришли с помощью псевдорешений.