# Линейная регрессия: метод наименьших квадратов

Сергей Николенко


Центр Речевых Технологий, 2012

Центр Речевых Технологий

# Outline

Центр Речевых Технологий

# В предыдущей серии...

- Теорема Байеса:

$$p(\theta|D) = \frac{p(\theta)p(D|\theta)}{p(D)}.$$

- Две основные задачи байесовского вывода:
  1. найти апостериорное распределение на гипотезах/параметрах:

$$p(\theta \mid D) \propto p(D|\theta)p(\theta)$$

     (и/или найти гипотезу максимального правдоподобия $\arg\max_\theta p(\theta \mid D)$);
  2. найти апостериорное распределение исходов дальнейших экспериментов:

$$p(x \mid D) \propto \int_{\theta\in\Theta} p(x \mid \theta)p(D|\theta)p(\theta)\mathrm{d}\theta.$$

# Метод наименьших квадратов

- Линейная модель: рассмотрим линейную функцию

$$y(\mathbf{x}, \mathbf{w}) = w_0 + \sum_{j=1}^{p} x_j w_j = \mathbf{x}^\top \mathbf{w}, \quad \mathbf{x} = (1, x_1, \ldots, x_p).$$

- Таким образом, по вектору входов $\mathbf{x}^\top = (x_1, \ldots, x_p)$ мы будем предсказывать выход $y$ как

$$\hat{y}(\mathbf{x}) = \hat{w}_0 + \sum_{j=1}^{p} x_j \hat{w}_j = \mathbf{x}^\top \hat{\mathbf{w}}.$$

Центр Речевых Технологий

# Метод наименьших квадратов

- Как найти оптимальные параметры $\hat{\mathbf{w}}$ по тренировочным данным вида $(\mathbf{x}_i, y_i)_{i=1}^N$?
- Метод наименьших квадратов: будем минимизировать

$$\mathrm{RSS}(\mathbf{w}) = \sum_{i=1}^{N} (y_i - \mathbf{x}_i^\top \mathbf{w})^2.$$

- Как минимизировать?

# Метод наименьших квадратов

- Можно на самом деле решить задачу точно – записать как

$$\mathrm{RSS}(\mathbf{w}) = (\mathbf{y} - \mathbf{X}\mathbf{w})^\top (\mathbf{y} - \mathbf{X}\mathbf{w}),$$

где $\mathbf{X}$ – матрица $N \times p$, продифференцировать по $\mathbf{w}$, получится

$$\hat{\mathbf{w}} = (\mathbf{X}^\top \mathbf{X})^{-1} \mathbf{X}^\top \mathbf{y},$$

если матрица $\mathbf{X}^\top \mathbf{X}$ невырожденная.

- Замечание: $\left(\mathbf{X}^\top \mathbf{X}\right)^{-1} \mathbf{X}^\top$ называется *псевдообратной матрицей Мура–Пенроуза* (Moore–Penrose pseudo-inverse) матрицы $\mathbf{X}$; это обобщение понятия обратной матрицы на неквадратные матрицы.
- Много ли нужно точек, чтобы обучить такую модель?

# Метод наименьших квадратов

- Пример: задача классификации. Два класса; мы кодируем один ответ как $y = 1$, другой ответ как $y = 0$ и рисуем прямую $\mathbf{x}^\top \hat{\mathbf{w}} = \frac{1}{2}$.
- Мы видим, что данные разделяются не то чтобы совсем замечательно.
- Когда линейная модель работает хорошо, когда плохо?
- Предположим, что это была смесь нескольких нормальных распределений – что тогда?

# Метод ближайших соседей

- Линейная модель – очень сильные предположения, много точек не нужно.
- Совсем другой подход – давайте вообще никаких предположений не делать (это не совсем так, конечно :) ), а будем отталкиваться от данных.
- Давайте не будем строить вообще никакой модели, а будем классифицировать новые примеры как

$$\hat{y}(\mathbf{x}) = \frac{1}{k} \sum_{\mathbf{x}_i \in N_k(\mathbf{x})} y_i,$$

где $N_k(\mathbf{x})$ – множество $k$ ближайших соседей точки $\mathbf{x}$ среди имеющихся данных $(\mathbf{x}_i, y_i)_{i=1}^N$.

# Метод ближайших соседей

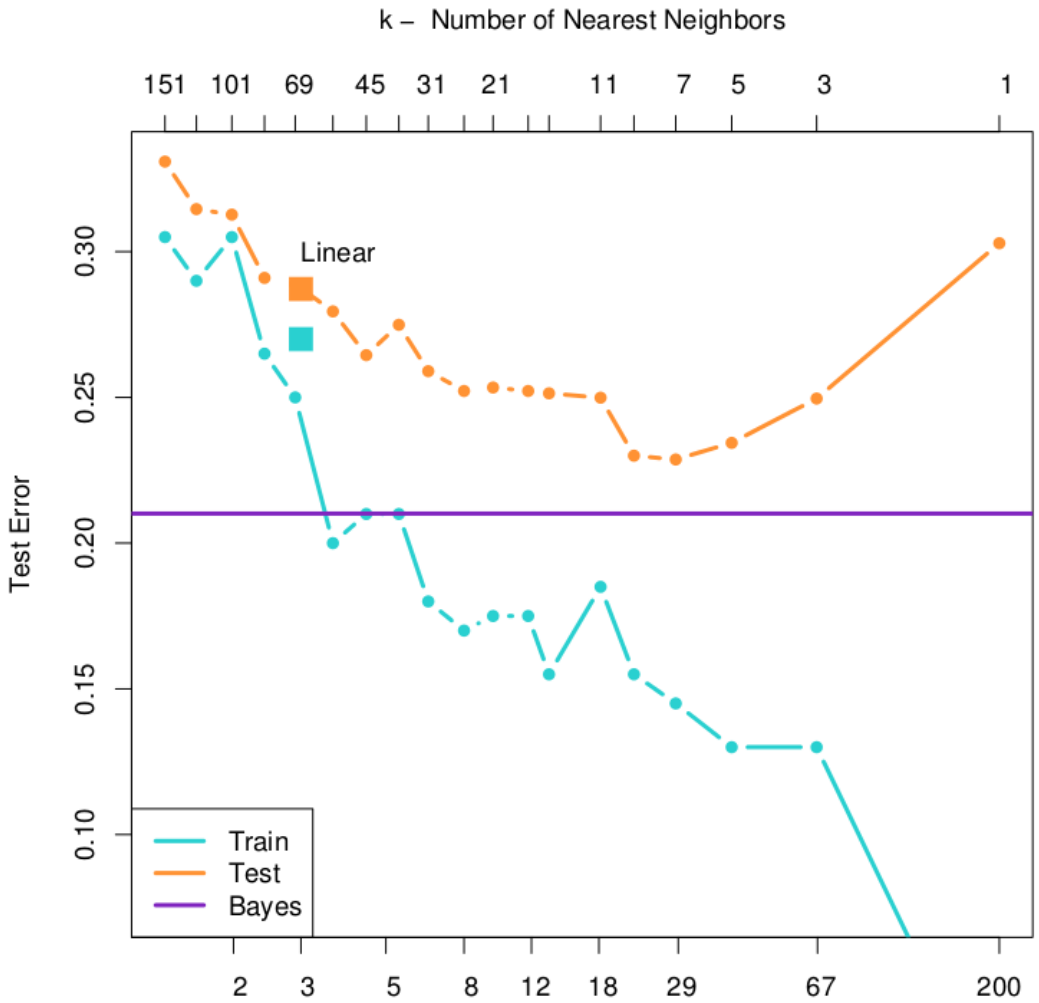
- Снова смотрим на примеры – теперь появился параметр $k$, от которого многое зависит.
- Для разумно большого $k$ у нас в нашем примере стало меньше ошибок.
- Но это не предел – для $k = 1$ на тестовых данных вообще никаких ошибок нету!
- Что это значит? В чём недостаток метода ближайших соседей при $k = 1$?
- Сколько параметров у метода $k$-NN?
- Как выбрать $k$? Можно ли просто подсчитать среднеквадратическую ошибку и минимизировать её?

# Метод ближайших соседей

- На самом деле данные были порождены так:
  - сначала по распределению $\mathcal{N}((1,0)^\top, \mathbf{I})$ породили 10 синих средних;
  - потом по распределению $\mathcal{N}((0,1)^\top, \mathbf{I})$ породили 10 красных средних;
  - потом для каждого из классов сгенерировали по 100 точек так: выбрать одно из 10 средних $m_k$ равномерно (с вероятностью $\frac{1}{10}$), потом породили точку $\mathcal{N}(m_k, \frac{1}{5}\mathbf{I})$.
- Получилось, что мы разделяем две смеси гауссианов.

Центр Речевых Технологий

# Качество метода $K$-NN



Центр Речевых Технологий

# Outline

# Функция потери

- Сейчас мы попытаемся понять, что же на самом деле происходит в этих методах.
- Начнём с настоящей регрессии – непрерывный вещественный вход $\mathbf{x} \in \boldsymbol{R}^p$, непрерывный вещественный выход $y \in \boldsymbol{R}$; у них есть некоторое совместное распределение $p(\mathbf{x}, y)$.
- Мы хотим найти функцию $f(\mathbf{x})$, которая лучше всего предсказывает $y$.

# Функция потери

- Введём *функцию потери* (loss function) $L(y, f(\mathbf{x}))$, которая наказывает за ошибки; естественно взять квадратичную функцию потери

$$L\left(y, f(\mathbf{x})\right) = \left(y - f(\mathbf{x})\right)^2.$$

- Тогда каждому $f$ можно сопоставить *ожидаемую ошибку предсказания* (expected prediction error):

$$\mathrm{EPE}(f) = \mathrm{E}(y - f(\mathbf{x}))^2 = \int\int (y - f(\mathbf{x}))^2 p(\mathbf{x}, y)\, d\mathbf{x} dy.$$

- И теперь самая хорошая функция предсказания $\hat{f}$ – это та, которая минимизирует $\mathrm{EPE}(f)$.

# Функция потери

- Это можно переписать как

$$\mathrm{EPE}(f) = \mathrm{E}_{\mathbf{x}}\mathrm{E}_{y|\mathbf{x}}\left[(y - f(\mathbf{x}))^2 \mid \mathbf{x}\right],$$

и, значит, можно теперь минимизировать $\mathrm{EPE}$ поточечно:

$$\hat{f}(\mathbf{x}) = \arg\min_c \mathrm{E}_{y|\mathbf{x}'}\left[(y - c)^2 \mid \mathbf{x}' = \mathbf{x}\right],$$

а это можно решить и получить

$$\hat{f}(\mathbf{x}) = \mathrm{E}_{y|\mathbf{x}'}(y \mid \mathbf{x}' = \mathbf{x}).$$

- Это решение называется *функцией регрессии* и является наилучшим предсказанием $y$ в любой точке $\mathbf{x}$.

# $k$-NN

- Теперь мы можем понять, что такое $k$-NN.
- Давайте оценим это ожидание:

$$f(\mathbf{x}) = \mathrm{E}_{y|\mathbf{x}'}(y \mid \mathbf{x}' = \mathbf{x}).$$

- Оценка ожидания – это среднее всех $y$ с данным $\mathbf{x}$. Конечно, у нас таких нету, поэтому мы приближаем это среднее как

$$\widehat{f}(\mathbf{x}) = \text{Average}\left[y_i \mid \mathbf{x}_i \in N_k(\mathbf{x})\right].$$

- Это сразу два приближения: ожидание через среднее и среднее в точке через среднее в ближних точках.
- Иначе говоря, $k$-NN предполагает, что в окрестности $\mathbf{x}$ функция $y(\mathbf{x})$ не сильно меняется, а лучше всего – она кусочно-постоянна.

# Линейная регрессия

- А линейная регрессия – это модельный подход, мы предполагаем, что функция регрессии линейна от своих аргументов:

$$f(\mathbf{x}) \approx \mathbf{x}^\top \mathbf{w}.$$

- Теперь мы не берём условие по $\mathbf{x}$, как в $k$-NN, а просто собираем много значений для разных $\mathbf{x}$ и обучаем модель.

# Классификация

- То же самое можно и с задачей классификации сделать. Пусть у нас переменная $g$ с $K$ возможными значениями $g_1, \ldots, g_k$ предсказывается.
- Введём функцию потери, равную 1 за каждый неверный ответ. Получим

$$\mathrm{EPE} = \mathrm{E}\left[L(g, \hat{g}(\mathbf{x}))\right].$$

- Перепишем как раньше:

$$\mathrm{EPE} = \mathrm{E}_{\mathbf{x}} \sum_{k=1}^{K} L(g_k, \hat{g}(\mathbf{x})) p(g_k \mid \mathbf{x}).$$

- Опять достаточно оптимизировать поточечно:

$$\hat{g}(\mathbf{x}) = \arg\min_g \sum_{k=1}^{K} L(g_k, \hat{g}(\mathbf{x})) p(g_k \mid \mathbf{x}).$$

# Классификация

- Опять достаточно оптимизировать поточечно:

$$\hat{g}(\mathbf{x}) = \arg\min_g \sum_{k=1}^{K} L(g_k, \hat{g}(\mathbf{x}))p(g_k \mid \mathbf{x}).$$

- Для 0-1 функции потери это упрощается до

$$\hat{g}(\mathbf{x}) = \arg\min_g \left[1 - p(g \mid \mathbf{x})\right], \text{ т.е.}$$

$$\hat{g}(\mathbf{x}) = g_k, \text{ если } p(g_k \mid \mathbf{x}) = \max_g p(g \mid \mathbf{x}).$$

- Это называется *оптимальным байесовским классификатором*; если модель известна, то его обычно можно построить.

# Классификация

- Байесовский классификатор: $\hat{g}(\mathbf{x}) = g_k$ для $p(g_k \mid \mathbf{x}) = \max_g p(g \mid \mathbf{x})$.
- Опять $k$-NN строит приближение к этой формуле – выбирает большинством голосов в окрестности точки.
- Что делает линейный классификатор, мы уже обсуждали – кодируем $g$ через 0-1 переменную $y$, приближаем $y$ линейной функцией, предсказываем.
- Правда, странновато получается – наше приближение может быть отрицательным или большим 1, например.

Центр Речевых Технологий

# Outline

Центр Речевых Технологий

# Поиск скрытых параметров

- Сначала – небольшое лирическое отступление о нормальном распределении. Кстати, почему все всё время предполагают нормальное распределение?
- Очень многие задачи машинного обучения можно представить как *поиск скрытых параметров*.
- Есть некоторое предположение о структуре задачи, т.е. о виде распределений, которыми набрасываются тестовые данные.
- Требуется найти наиболее правдоподобные неизвестные параметры этих распределений.

# Гауссиан

- Давайте решим эту задачу для нормального (гауссовского) распределения. У него два параметра:

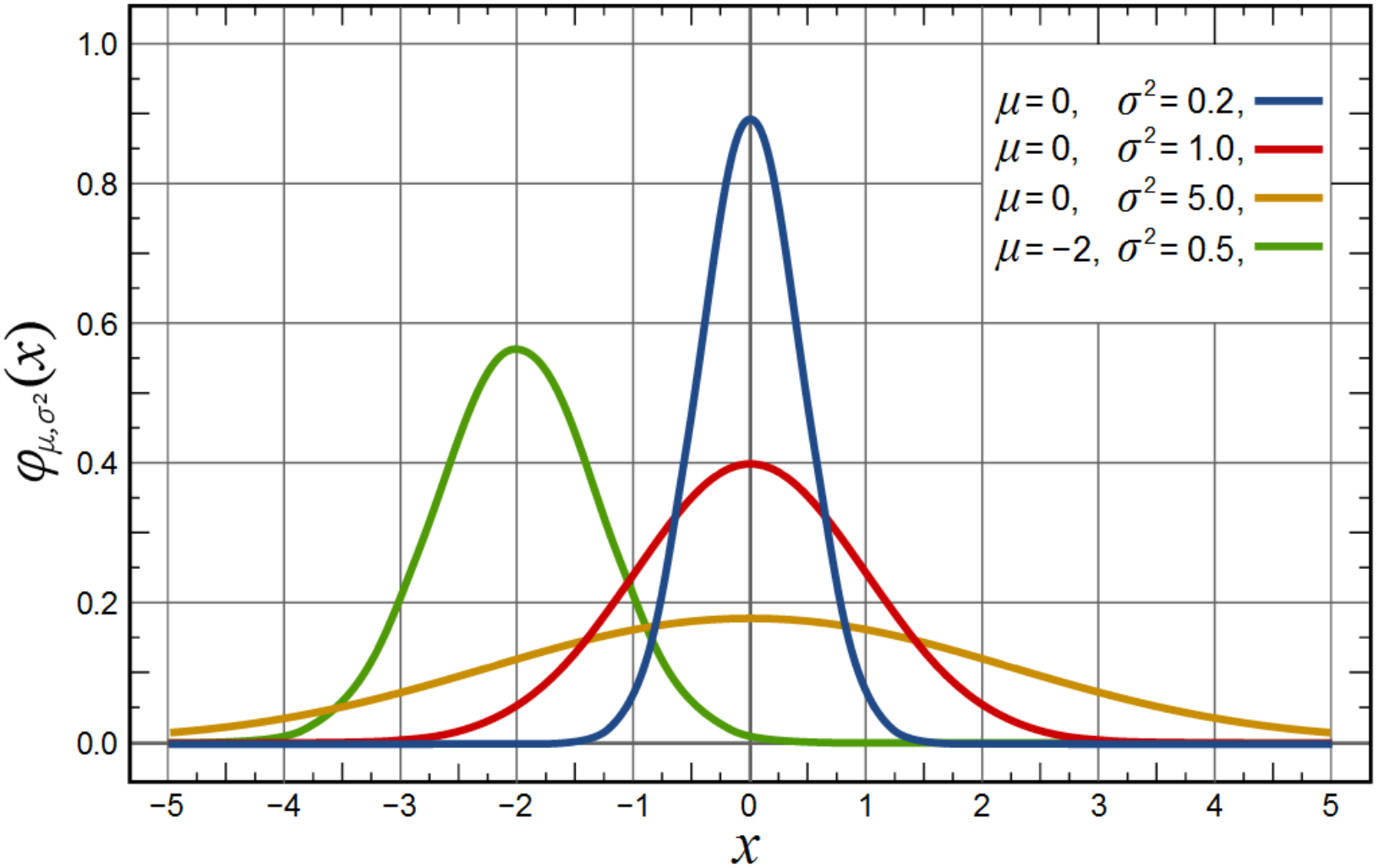
$$p(x) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{(x-\mu)^2}{2\sigma^2}}.$$

- Функция правдоподобия данных $x_1, \ldots, x_n$:

$$p(x_1, \ldots, x_n | \mu, \sigma) = \frac{1}{(\sigma\sqrt{2\pi})^n} e^{-\sum_{i=1}^{n} \frac{(x_i-\mu)^2}{2\sigma^2}}.$$

Центр Речевых Технологий

# Нормальное распределение



# Гауссиан: достаточные статистики

- Заметим, что функция эта зависит от двух параметров, а не от $n$:

$$p(x_1, \ldots, x_n | \mu, \sigma) = \frac{1}{(\sigma\sqrt{2\pi})^n} e^{-\frac{S + n(\bar{x} - \mu)^2}{2\sigma^2}},$$

где

$$\bar{x} = \frac{x_1 + \ldots + x_n}{n}, \quad S = \sum_{i=1}^{n} (\bar{x} - x_n)^2.$$

- Параметры $\bar{x}$ и $S$ называются *достаточными статистиками* (sufficient statistics).

Центр Речевых Технологий

# Гауссиан: ГМП

- Какие параметры лучше всего описывают данные?
- Перейдём, как водится, к логарифму:

$$\ln p(x_1, \ldots, x_n | \mu, \sigma) = -n \ln(\sigma\sqrt{2\pi}) - \frac{S + n(\bar{x} - \mu)^2}{2\sigma^2}.$$

- Как выяснить, при каких параметрах функция правдоподобия максимизируется?

Центр Речевых Технологий

# Гауссиан: ГМП

- Какие параметры лучше всего описывают данные?
- Перейдём, как водится, к логарифму:

$$\ln p(x_1, \ldots, x_n | \mu, \sigma) = -n \ln(\sigma\sqrt{2\pi}) - \frac{S + n(\bar{x} - \mu)^2}{2\sigma^2}.$$

- Как выяснить, при каких параметрах функция правдоподобия максимизируется?
- Взять частные производные и приравнять нулю.

Центр Речевых Технологий

# Гауссиан: ГМП

- По $\mu$:
$$\frac{\partial \ln p}{\partial \mu} = -\frac{n}{\sigma^2}(\mu - \bar{x}).$$
- То есть в гипотезе максимального правдоподобия $\mu_{ML} = \bar{x}$, независимо от $S$.
- Теперь нужно найти $\sigma$ из гипотезы максимального правдоподобия.
- Для этого мы продифференцируем по $\ln \sigma$ — полезный приём на будущее. Кстати, $\frac{\mathrm{d}x^n}{\mathrm{d}(\ln x)} = nx^n$.

# Гауссиан: ГМП

- $$\frac{\partial \ln p}{\partial \ln \sigma} = -n + \frac{\sum_{i=1}^{n} (x_i - \mu)^2}{\sigma^2}.$$
- Следовательно, в гипотезе максимального правдоподобия

$$\sigma^2_{ML} = \frac{1}{n} \sum_{i=1}^{n} (x_i - \mu_{ML})^2.$$

- Интересно, что это *смещённая* оценка, т.е. ожидание $\sigma^2_{ML}$ по распределению $\mathcal{N}(\mu, \sigma^2)$ не равно $\sigma$, а несмещённая оценка получается как

$$\mathrm{s}^2 = \frac{1}{n-1} \sum_{i=1}^{n} (x_i - \mu)^2.$$

# Несколько гауссианов

- Теперь то же самое для нескольких гауссианов сразу.
- Даны несколько точек $x_1, \ldots, x_n$, но они принадлежат смеси гауссианов с разными $\mu_k$ и $\sigma_k$.
- Обозначим коэффициенты смеси через $w_k$ (вероятность того, что точка порождена гауссианом со средним $\mu_k$).
- Тогда распределение будет

$$p(x|\mathbf{w}, \boldsymbol{\mu}, \boldsymbol{\sigma}^2) = \sum_k w_k \mathcal{N}(x \mid \mu_k, \sigma_k^2).$$

# Несколько гауссианов

- Мы будем ещё говорить о том, как находить параметры максимального правдоподобия в смесях распределений вроде этой;

$$p(x_1, \ldots, x_n \mid \mathbf{w}, \boldsymbol{\mu}, \boldsymbol{\sigma}^2) = \prod_{i=1}^{n} p(x_i|\mathbf{w}, \boldsymbol{\mu}, \boldsymbol{\sigma}^2) = \prod_{i=1}^{n} \sum_{k} w_k \mathcal{N}(x_i \mid \mu_k, \sigma_k^2).$$

- А пока просто – что мы сделаем, если найдём эти параметры? Как называется эта задача?

# Несколько гауссианов

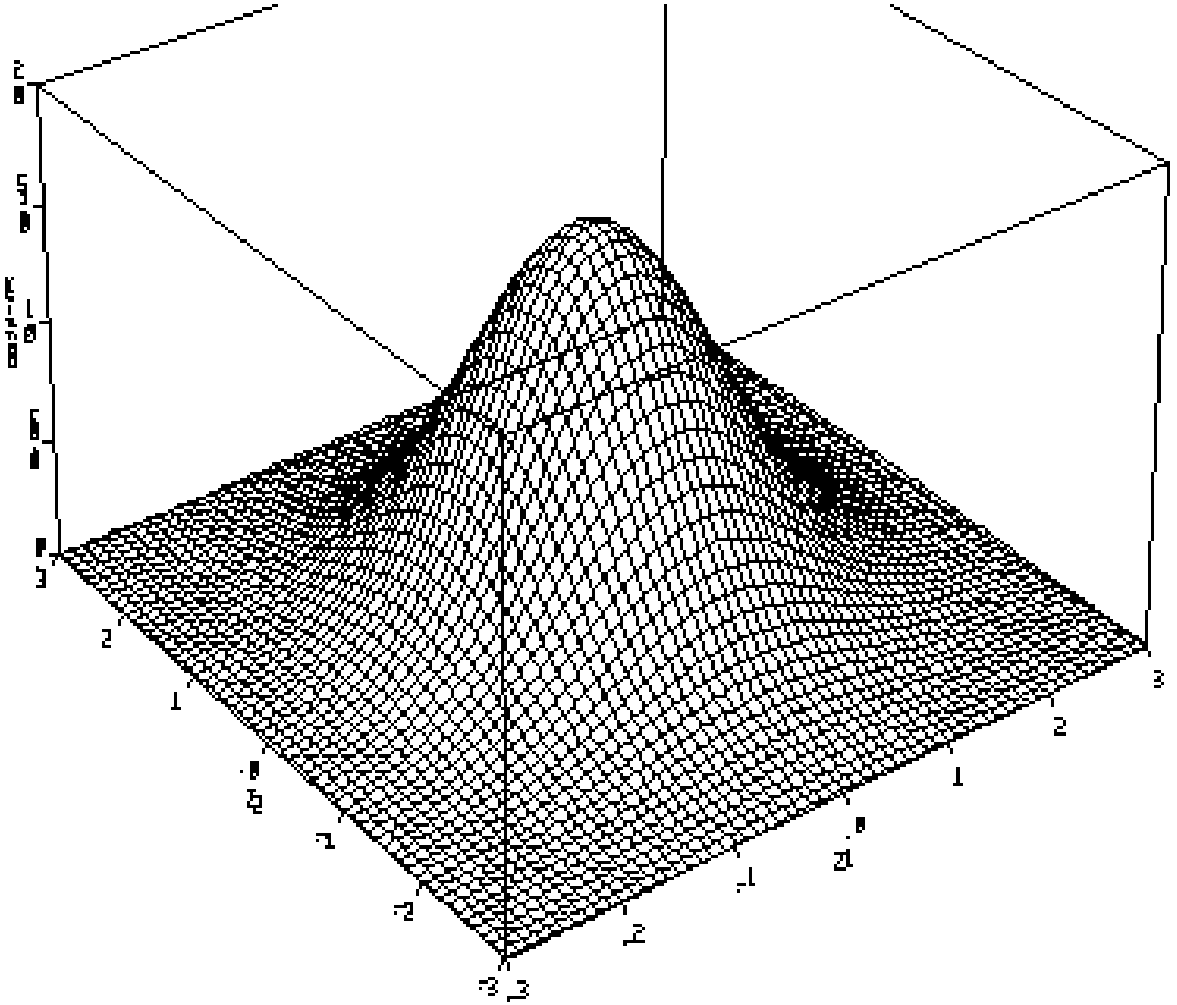
- Мы будем ещё говорить о том, как находить параметры максимального правдоподобия в смесях распределений вроде этой;

$$p(x_1, \ldots, x_n \mid \mathbf{w}, \boldsymbol{\mu}, \boldsymbol{\sigma}^2) = \prod_{i=1}^{n} p(x_i|\mathbf{w}, \boldsymbol{\mu}, \boldsymbol{\sigma}^2) = \prod_{i=1}^{n} \sum_{k} w_k \mathcal{N}(x_i \mid \mu_k, \sigma_k^2).$$

- А пока просто – что мы сделаем, если найдём эти параметры? Как называется эта задача?
- Мы решим задачу *кластеризации*: разделим данные (по наиболее правдоподобной компоненте смеси) на хорошо определённые (в данном случае гауссовские) кластеры.

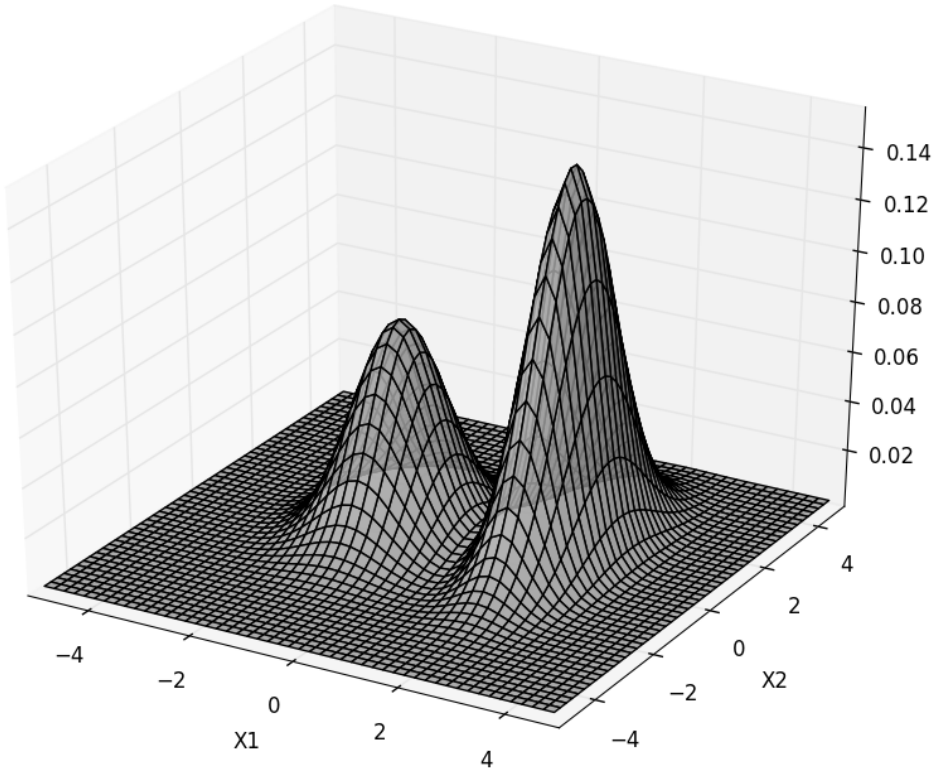
Центр Речевых Технологий

# Двумерный гауссиан



Центр Речевых Технологий

# Смесь двумерных гауссианов



# Байесовская регрессия

- Теперь давайте поговорим о линейной регрессии по-байесовски.
- Основное наше предположение – в том, что шум (ошибка в данных) распределён нормально, т.е. переменная $t$, которую мы наблюдаем, получается как

$$t = y(\mathbf{x}, \mathbf{w}) + \epsilon, \quad \epsilon \sim \mathcal{N}(0, \sigma^2).$$

Иными словами,

$$p(t \mid \mathbf{x}, \mathbf{w}, \sigma^2) = \mathcal{N}(t \mid y(\mathbf{x}, \mathbf{w}), \sigma^2).$$

- Здесь пока $y$ – любая функция.

# Байесовская регрессия

- Чтобы не повторять совсем уж то же самое, мы рассмотрим не в точности линейную регрессию, а её естественное обобщение – линейную модель с базисными функциями:

$$y(\mathbf{x}, \mathbf{w}) = w_0 + \sum_{j=1}^{M-1} w_j \phi_j(\mathbf{x}) = \mathbf{w}^\top \boldsymbol{\phi}(\mathbf{x})$$

($M$ параметров, $M-1$ базисная функция, $\phi_0(\mathbf{x}) = 1$).

Центр Речевых Технологий

# Байесовская регрессия

- Базисные функции $\phi_i$ – это, например:
  - результат feature extraction;
  - расширение линейной модели на нелинейные зависимости (например, $\phi_j(x) = x^j$);
  - локальные функции, которые существенно не равны нулю только в небольшой области (например, гауссовские базисные функции $\phi_j(\mathbf{x}) = e^{-\frac{(x-\mu_j)^2}{2s^2}}$);
  - ...

# Байесовская регрессия

- Рассмотрим набор данных $\mathbf{X} = \{\mathbf{x}_1, \ldots, \mathbf{x}_N\}$ со значениями $\mathbf{t} = \{t_1, \ldots, t_N\}$.
- Будем предполагать, что данные взяты независимо по одному и тому же распределению:

$$p(\mathbf{t} \mid \mathbf{X}, \mathbf{w}, \sigma^2) = \prod_{n=1}^{N} \mathcal{N}\left(t_n \mid \mathbf{w}^\top \boldsymbol{\phi}(\mathbf{x}_n), \sigma^2\right).$$

- Прологарифмируем (опустим $\mathbf{X}$, т.к. по нему всегда условная вероятность будет):

$$\ln p(\mathbf{t} \mid \mathbf{w}, \sigma^2) = -\frac{N}{2}\ln(2\pi\sigma^2) - \frac{1}{2\sigma^2}\sum_{n=1}^{N}\left(t_n - \mathbf{w}^\top \boldsymbol{\phi}(\mathbf{x}_n)\right)^2.$$

# Байесовская регрессия

- Прологарифмируем (опустим $\mathbf{X}$, т.к. по нему всегда условная вероятность будет):

$$\ln p(\mathbf{t} \mid \mathbf{w}, \sigma^2) = -\frac{N}{2}\ln(2\pi\sigma^2) - \frac{1}{2\sigma^2}\sum_{n=1}^{N}\left(t_n - \mathbf{w}^\top \boldsymbol{\phi}(\mathbf{x}_n)\right)^2.$$

- И вот мы получили, что для максимизации правдоподобия по $\mathbf{w}$ нам нужно как раз минимизировать среднеквадратичную ошибку!

$$\nabla_{\mathbf{w}} \ln p(\mathbf{t} \mid \mathbf{w}, \sigma^2) = \frac{1}{\sigma^2}\sum_{n=1}^{N}\left(t_n - \mathbf{w}^\top \boldsymbol{\phi}(\mathbf{x}_n)\right)\boldsymbol{\phi}(\mathbf{x}_n).$$

Центр Речевых Технологий

# Байесовская регрессия

- Решая систему уравнений $\nabla \ln p(\mathbf{t} \mid \mathbf{w}, \sigma^2) = 0$, получаем то же самое, что и раньше:

$$\mathbf{w}_{ML} = \left(\mathbf{\Phi}^\top \mathbf{\Phi}\right)^{-1} \mathbf{\Phi}^\top \mathbf{t}.$$

- Здесь $\mathbf{\Phi} = (\phi_j(\mathbf{x}_i))_{i,j}$.

Центр Речевых Технологий

# Байесовская регрессия

- Теперь можно и относительно $\sigma^2$ максимизировать правдоподобие; получим

$$\sigma^2_{ML} = \frac{1}{N}\sum_{n=1}^{N}\left(t_n - \mathbf{w}_{ML}^\top \boldsymbol{\phi}(\mathbf{x}_n)\right)^2,$$

т.е. как раз выборочная дисперсия имеющихся данных вокруг предсказанного значения.

Центр Речевых Технологий

# Проклятие размерности

- В прошлый раз $k$-NN давали гораздо более разумные результаты, чем линейная модель, особенно если хорошо выбрать $k$.
- Может быть, нам в этой жизни больше ничего и не нужно?
- Давайте посмотрим, как $k$-NN будет вести себя в более высокой размерности (что очень реалистично).

Центр Речевых Технологий

# Проклятие размерности

- Давайте поищем ближайших соседей у точки в единичном гиперкубе. Предположим, что наше исходное распределение равномерное.
- Чтобы покрыть долю $\alpha$ тестовых примеров, нужно (ожидаемо) покрыть долю $\alpha$ объёма, и ожидаемая длина ребра гиперкуба-окрестности в размерности $p$ будет $e_p(\alpha) = \alpha^{1/p}$.
- Например, в размерности 10 $e_{10}(0.1) = 0.8$, $e_{10}(0.01) = 0.63$, т.е. чтобы покрыть 1% объёма, нужно взять окрестность длиной больше половины носителя по каждой координате!
- Это скажется и на $k$-NN: трудно отвергнуть по малому числу координат, быстрые алгоритмы хуже работают.

# Проклятие размерности

- Второе проявление the curse of dimensionality: пусть $N$ точек равномерно распределены в единичном шаре размерности $p$. Тогда среднее расстояние от нуля до точки равно

$$d(p, N) = \left(1 - \frac{1}{2}^{1/N}\right)^{1/p},$$

  т.е., например, в размерности 10 для $N = 500$ $d \approx 0.52$, т.е. больше половины.
- Большинство точек в результаты ближе к границе носителя, чем к другим точкам, а это для ближайших соседей проблема – придётся не интерполировать внутри существующих точек, а экстраполировать наружу.

Центр Речевых Технологий

# Проклятие размерности

- Третье проявление: проблемы в оптимизации, которые и имел в виду Беллман.
- Если нужно примерно оптимизировать функцию от $d$ переменных, на решётке с шагом $\epsilon$ понадобится примерно $\left(\frac{1}{\epsilon}\right)^d$ вычислений функции.
- В численном интегрировании – чтобы интегрировать функцию с точностью $\epsilon$, нужно тоже примерно $\left(\frac{1}{\epsilon}\right)^d$ вычислений.

Центр Речевых Технологий

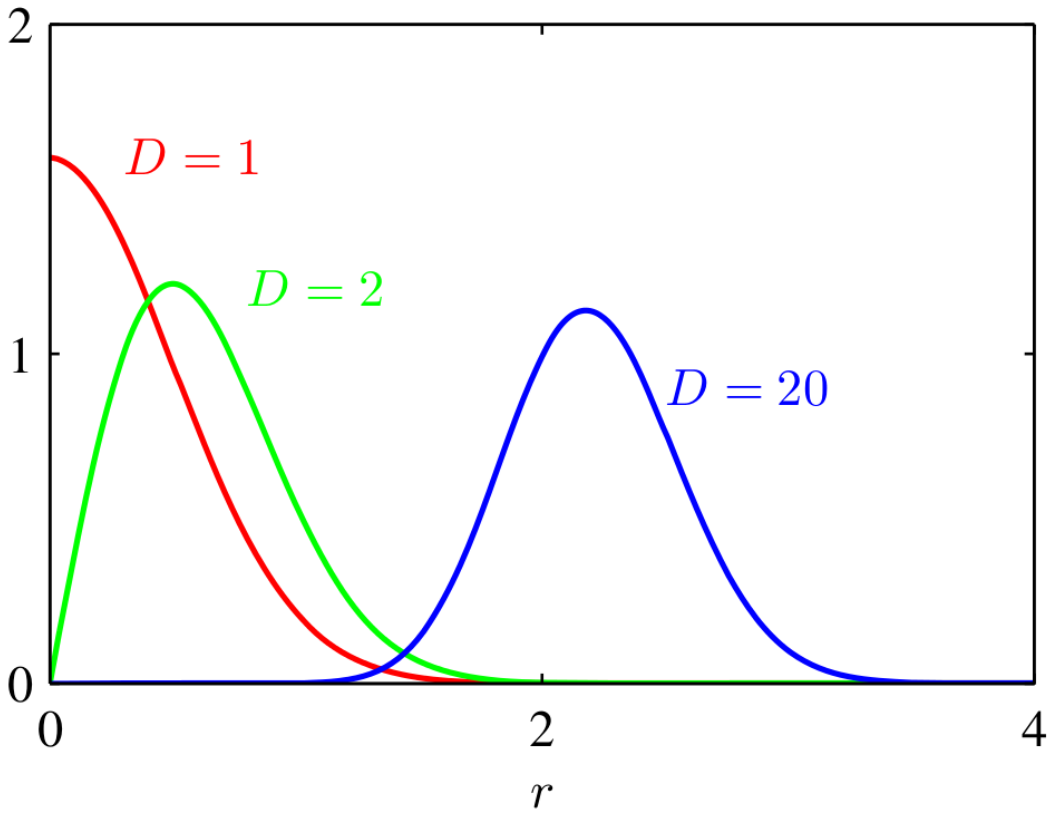
# Проклятие размерности

- Плотные множества становятся очень разреженными. Например, чтобы получить плотность, создаваемую в размерности 1 при помощи $N = 100$ точек, в размерности 10 нужно будет $100^{10}$ точек.
- Поведение функций тоже усложняется с ростом размерности – чтобы строить регрессии в высокой размерности с той же точностью, может потребоваться экспоненциально больше точек, чем в низкой размерности.
- А у линейной модели ничего такого не наблюдается, она не подвержена проклятию размерности.

Центр Речевых Технологий

# Проклятие размерности

- Ещё пример: нормально распределённая величина будет сосредоточена в тонкой оболочке.




**Упражнение.** Переведите плотность нормального распределения в полярные координаты и проверьте это утверждение.

Центр Речевых Технологий

# Thank you!

**Спасибо за внимание!**