Введение: AI и байесовский вывод

Сергей Николенко

Академический Университет, весна 2011

Outline

Первые мысли об искусственном интеллекте

- Гефест создавал себе роботов–андроидов, например, гигантского человекоподобного робота Талоса.
- Пигмалион оживлял Галатею.
- Иегова и Аллах — куски глины.
- Особо мудрые раввины могли создавать големов.
- Альберт Великий изготовил искусственную говорящую голову (чем очень расстроил Фому Аквинского).
- Начиная с доктора Франкенштейна, дальше AI в литературе появляется постоянно...

Тест Тьюринга

- AI как наука начался с *теста Тьюринга* (1950).
- Компьютер должен успешно выдать себя за человека в (письменном) диалоге между судьёй, человеком и компьютером.
- Правда, исходная формулировка была несколько тоньше и интереснее...

Тест Тьюринга

- Здесь уже очевидно, сколько всего надо, чтобы сделать AI:
  - обработка естественного языка;
  - представление знаний;
  - выводы из полученных знаний;
  - обучение на опыте (собственно machine learning).

Дартмутский семинар

- Термин AI и формулировки основных задач появились в 1956 на семинаре в Дартмуте.
- Его организовали Джон Маккарти (John McCarthy), Марвин Мински (Marvin Minsky), Клод Шеннон (Claude Shennon) и Натаниэль Рочестер (Nathaniel Rochester).
- Это была, наверное, самая амбициозная грантозаявка в истории информатики.

Дартмутский семинар

Мы предлагаем исследование искусственного интеллекта сроком в 2 месяца с участием 10 человек летом 1956 года в Дартмутском колледже, Гановер, Нью–Гемпшир. Исследование основано на предположении, что всякий аспект обучения или любое другое свойство интеллекта может в принципе быть столь точно описано, что машина сможет его симулировать. Мы попытаемся понять, как обучить машины использовать естественные языки, формировать абстракции и концепции, решать задачи, сейчас подвластные только людям, и улучшать самих себя. Мы считаем, что существенное продвижение в одной или более из этих проблем вполне возможно, если специально подобранная группа учёных будет работать над этим в течение лета.

1956-1960: большие надежды

- Оптимистическое время. Казалось, что ещё немного, ещё чуть-чуть...
- Allen Newell, Herbert Simon: *Logic Theorist*.
    - Программа для логического вывода.
    - Смогла передоказать большую часть *Principia Mathematica*, кое-где даже изящнее, чем сами Рассел с Уайтхедом.

1956-1960: большие надежды

- Оптимистическое время. Казалось, что ещё немного, ещё чуть-чуть...
- General Problem Solver – программа, которая пыталась думать как человек;
- Много программ, которые умели делать некоторые ограниченные вещи (microworlds):
  - Analogy (IQ-тесты на «выберите лишнее»);
  - Student (алгебраические словесные задачи);
  - Blocks World (переставляла 3D-блоки).

1970-е: knowledge-based systems

- Суть: накопить достаточно большой набор правил и знаний о предметной области, затем делать выводы.
- Первый успех: MYCIN – диагностика инфекций крови:
    - около 450 правил;
    - результаты как у опытного врача и существенно лучше, чем у начинающих врачей.

1980-е: коммерческие применения; индустрия AI

- Началось внедрение.
- Первый AI-отдел был в компании DEC (Digital Equipment Corporation);
- К 1986 году он экономил DEC $10 млн. в год;
- Бум закончился к концу 80-х, когда многие компании не смогли оправдать завышенных ожиданий.

1990-2010: data mining, machine learning

- В последние десятилетия основной акцент сместился на машинное обучение и поиск закономерностей в данных.
- Особенно — с развитием интернета.
- Сейчас про AI в смысле трёх законов робототехники уже не очень вспоминают.
- // Но роботика — процветает и пользуется machine learning на каждом шагу.

AI как междисциплинарный проект

- Искусственный интеллект использует огромное количество методов из самых разных наук.
  1. Математика.
  2. Биология.
  3. Экономика.
  4. Философия.
  5. Психология.
  6. Лингвистика.
  7. ...

Определение

- Что значит — обучающаяся машина? Как определить «обучаемость»?

Определение

- Что значит — обучающаяся машина? Как определить «обучаемость»?

Определение

Компьютерная программа обучается по мере накопления опыта относительно некоторого класса задач T и целевой функции P, если качество решения этих задач (относительно P) улучшается с получением нового опыта.

- Определение очень (слишком?) общее.
- Какие конкретные примеры можно привести?

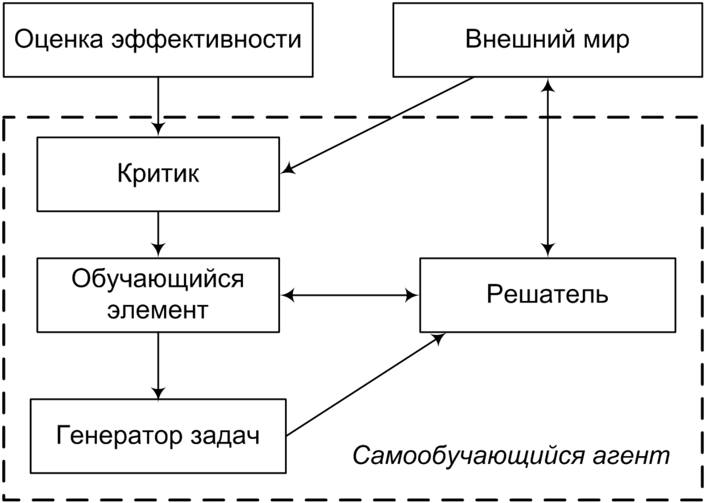
Самообучающиеся агенты

- В чём отличие самообучающегося от обычного интеллектуального агента?
- Обычный агент реагирует на информацию извне, а самообучающийся — не просто реагирует, а со временем реагирует всё лучше и лучше.

Самообучающиеся агенты

- Интеллектуальные агенты действуют в условиях неполной информации.
- Предсказать исход невозможно, но нужно научиться действовать оптимальным образом.
- Если бы можно было предсказать, обучение было бы не нужно, только вычислительные мощности.

Схема самообучающегося агента



Чем мы будем заниматься

- Мы, конечно, не будем разрабатывать готовых агентов.
- Мы будем рассматривать разные алгоритмы, которые решают ту ли иную задачу, причём решают тем лучше, чем больше начальных (тестовых) данных ему дадут.
- Сейчас мы перейдём к общей теории байесовского вывода, в которую обычно можно погрузить любой алгоритм машинного обучения.

Outline

Вероятностное пространство

Definition

Вероятностное пространство — это тройка $(\Omega, \mathcal{F}, p)$, где

- Ω — множество, элементы которого называются элементарными событиями, исходами или точками;
- $\mathcal{F}$ — сигма-алгебра подмножеств Ω, называемых (случайными) событиями;
- P — вероятностная мера или вероятность, т.е. такая σ–аддитивная конечная мера, что $p(\Omega) = 1$.

Definition

Случайная величина $X : \Omega \longrightarrow \mathbb{R}$ индуцирует *распределение случайной величины* X на борелевских $P^X(B) = P(X \in B)$.

Операции над вероятностями

- Совместная вероятность $p(x, y)$, $x \in \Omega_X$, $y \in \Omega_Y$ определяется на прямом произведении $\Omega_X \times \Omega_Y$.
- Маргинализация из совместной вероятности:

$$p(x = a) = \sum_{y \in \Omega_Y} p(x = a, y).$$

- Условная вероятность:

$$p(x = a | y = b) = \frac{p(x = a, y = b)}{p(y = b)}, \text{ если } p(y = b) \neq 0.$$

Об условных вероятностях

- В жизни на самом деле все вероятности условные.
- Всегда есть какие-то предположения, без которых, вообще говоря, и понятие «вероятность» будет не очень-то определено.
- Например, когда мы говорим «вероятность орла на честной монетке равна $\frac{1}{2}$», это вероятность в предположении, что монетка честная.
- В нашем курсе вероятности тоже будут обычно условными.

Общие правила

- Правило произведения:

$$p(x, y|H) = p(x|y, H)p(y|H) = p(y|x, H)p(x|H).$$

- Правило суммирования:

$$p(x|H) = \sum_y p(x, y|H) = \sum_y p(x|y, H)p(y|H).$$

- Теорема Байеса:

$$p(y|x, H) = \frac{p(x|y, H)p(y|H)}{p(x|H)} = \frac{p(x|y, H)p(y|H)}{\sum_{y'} p(x|y', H)p(y'|H)}.$$

- *Независимость* определим просто: две случайные переменные X и Y независимы, если

$$p(x, y) = p(x)p(y).$$

О болезнях и вероятностях

- Приведём классический пример из классической области применения статистики — медицины.
- Пусть некий тест на какую-нибудь болезнь имеет вероятность успеха 95% (т.е. 5% — вероятность как позитивной, так и негативной ошибки).
- Всего болезнь имеется у 1% респондентов (отложим на время то, что они разного возраста и профессий).
- Пусть некий человек получил позитивный результат теста (тест говорит, что он болен). С какой вероятностью он действительно болен?

О болезнях и вероятностях

- Приведём классический пример из классической области применения статистики — медицины.
- Пусть некий тест на какую-нибудь болезнь имеет вероятность успеха 95% (т.е. 5% — вероятность как позитивной, так и негативной ошибки).
- Всего болезнь имеется у 1% респондентов (отложим на время то, что они разного возраста и профессий).
- Пусть некий человек получил позитивный результат теста (тест говорит, что он болен). С какой вероятностью он действительно болен?
- Ответ: 16%.

Доказательство

- Обозначим через t результат теста, через d — наличие болезни.
- $p(d=1) = p(d=1|t=1)p(t=1) + p(d=1|t=0)p(t=0)$.
- Используем теорему Байеса:

$$p(d=1|t=1) = \\ = \frac{p(t=1|d=1)p(t=1)}{p(d=1|t=1)p(t=1) + p(d=1|t=0)p(t=0)} = \\ = \frac{0.95 \times 0.01}{0.95 \times 0.01 + 0.05 \times 0.99} = 0.16.$$

Вывод

- Вот такие задачи составляют суть вероятностного вывода (probabilistic inference).
- Поскольку они обычно основаны на теореме Байеса, вывод часто называют байесовским (Bayesian inference).
- Но не только поэтому.

Вероятность как частота

- Обычно в классической теории вероятностей, происходящей из физики, вероятность понимается как предел отношения количества определённого результата эксперимента к общему количеству экспериментов.
- Стандартный пример: бросание монетки.

Вероятность как степень доверия

- Мы можем рассуждать о том, «насколько вероятно» то, что
  - Дмитрий Медведев будет избран на второй срок;
  - «Одиссею» написала женщина;
  - Керенский бежал за границу в женском платье;
  - ...
- Но о «стремящемся к бесконечности количестве экспериментов» говорить бессмысленно — эксперимент здесь ровно один.

Вероятность как степень доверия

- Здесь вероятности уже выступают как *степени доверия* (degrees of belief). Это байесовский подход к вероятностям (Томас Байес так понимал).
- К счастью, и те, и другие вероятности подчиняются одним и тем же законам, а если не подчиняются — значит, не вероятности и были.

Прямые и обратные задачи

- Прямая задача: в урне лежат 10 шаров, из них 3 чёрных. Какова вероятность выбрать чёрный шар?
- Или: в урне лежат 10 шаров с номерами от 1 до 10. Какова вероятность того, что номера трёх последовательно выбранных шаров дадут в сумме 12?
- Обратная задача: перед нами две урны, в каждой по 10 шаров, но в одной 3 чёрных, а в другой — 6. Кто-то взял из какой-то урны шар, и он оказался чёрным. Насколько вероятно, что он брал шар из первой урны?
- Заметьте, что в обратной задаче вероятности сразу стали байесовскими (хоть здесь и можно переформулировать через частоты).

Прямые и обратные задачи

- Иначе говоря, прямые задачи теории вероятностей описывают некий вероятностный процесс или модель и просят подсчитать ту или иную вероятность (т.е. фактически по модели предсказать поведение).
- Обратные задачи содержат *скрытые переменные* (в примере — номер урны, из которой брали шар). Они часто просят по известному поведению построить вероятностную модель.
- Задачи машинного обучения обычно являются задачами второй категории.

Определения

- Запишем теорему Байеса:

$$p(x|y, H) = \frac{p(x)p(y|x, H)}{p(y|H)}.$$

- Здесь $p(x)$ — *априорная вероятность* (prior probability), $p(y|x, H)$ — *правдоподобие* (likelihood), $p(x|y)$ — *апостериорная вероятность* (aposterior probability), $p(y|H)$ — вероятность данных (evidence).
- Вообще, *функция правдоподобия* имеет вид

$$a \mapsto p(y|x = a)$$

для некоторой случайной величины y.

Постановка задачи

- Простая задача вывода: дана нечестная монетка, она подброшена N раз, имеется последовательность результатов падения монетки. Надо определить её «нечестность» и предсказать, чем она выпадет в следующий раз.

Первые замечания

- Если у нас есть вероятность p_h того, что монетка выпадет решкой (вероятность орла $p_t = 1 - p_h$), то вероятность того, что выпадет последовательность s, которая содержит n_h решек и n_t орлов, равна

$$p(s|p_h, H_1) = p_h^{n_h}(1 - p_h)^{n_t}.$$

- Сделаем предположение: будем считать, что монетка выпадает равномерно, т.е. у нас нет априорного знания p_h.
- Теперь нужно использовать теорему Байеса и вычислить скрытые параметры.

Применение теоремы Байеса

- $p(p_h|s, H_1) = \frac{p(s|p_h, H_1)p(p_h|H_1)}{p(s|H_1)}$.
- Здесь $p(p_h)$ следует понимать как непрерывную случайную величину, сосредоточенную на интервале $[0, 1]$, коей она и является. Наше предположение о равномерном распределении в данном случае значит, что априорная вероятность $p(p_h|H) = 1$, $p_h \in [0, 1]$ (т.е. априори мы не знаем, насколько нечестна монетка, и предполагаем это равновероятным). А $p(s|p_h, H_1)$ мы уже знаем.
- Итого получается:

$$p(p_h|s, H_1) = \frac{p_h^{n_h}(1 - p_h)^{n_t}}{p(s|H_1)}.$$

Применение теоремы Байеса

- Итого получается:

$$p(p_h|s,H_1)=\frac{p_h^{n_h}(1-p_h)^{n_t}}{p(s|H_1)}.$$

- Осталось подсчитать $p(s|H_1)$; её нужно маргинализовать из функции правдоподобия:

$$p(s|H_1)=\int_0^1 p_h^{n_h}(1-p_h)^{n_t}dp_a=$$

$$=\frac{\Gamma(n_h+1)\Gamma(n_t+1)}{\Gamma(n_h+n_t+2)}=\frac{n_h!n_t!}{(n_h+n_t+1)!}.$$

А теперь предскажем следующий исход

- Чтобы предсказать следующий исход, нужно вычислить $p(\mathrm{heads}|s)$:

$$p(\mathrm{heads}|s) = \int_0^1 p(\mathrm{heads}|p_h)p(p_h|s, H_1)dp_h =$$
$$= \int_0^1 \frac{p_h^{n_h+1}(1-p_h)^{n_t}}{p(s)}dp_h =$$
$$= \frac{(n_h+1)!n_t!}{(n_h+n_t+2)!} \cdot \frac{(n_h+n_t+1)!}{n_h!n_t!} = \frac{n_h+1}{n_h+n_t+2}.$$

- Это называется *правилом Лапласа*.

Постановка задачи

- Мы использовали предположение H_1 о равномерном априорном распределении p_h.
- Это было нашей моделью ситуации.
- Можно использовать другие модели; например, можно априори предположить, что монетка падает решкой с вероятностью $1/3$, т.е. $p_h = 1/3$. Обозначим через H_0
- Как сравнить, какая из моделей H_i лучше описывает данные?

Теорема Байеса

- Снова теорема Байеса:

$$p(H_1|s) = \frac{p(s|H_1)p(H_1)}{p(s)}, \quad p(H_0|s) = \frac{p(s|H_0)p(H_0)}{p(s)}.$$

- Нужно как-то оценить априорные вероятности $p(H_0)$ и $p(H_1)$; можно просто положить их равными $1/2$.
- Теперь можно составить отношение:

$$\frac{p(H_1|s)}{p(H_0|s)} = \frac{p(s|H_1)p(H_1)}{p(s|H_0)p(H_0)} = \frac{n_h!n_t!}{(n_h+n_t+1)!} \cdot \frac{1}{(1/3)^{n_h}(2/3)^{n_t}}.$$

- Это отношение можно использовать для сравнения моделей.

Outline

Суть

- Мы наконец-то подходим к тому, что мы будем делать всё время на протяжении нашего курса.
- Мы сейчас определим цель, которую пытается достичь любой алгоритм машинного обучения.
- Некоторые её достигают, некоторые только приближаются — но в любом случае это «золотой стандарт».

Определения

Definition

Гипотеза максимального правдоподобия (maximum likelihood hypothesis) — это набор параметров θ, который максимизирует функцию правдоподобия

$$p(D|\theta, H),$$

где D — имеющаяся информация (тестовые примеры).

- Гипотеза максимального правдоподобия — гипотеза, которая лучше всего описывает имеющиеся данные D в текущих предположениях H.

Зачем это нужно

- Гипотезы получают численные оценки, которые можно сравнивать; не просто «соответствует данным», а «лучше/хуже соответствует».
- Каждый наблюдаемый случай может увеличить или уменьшить вероятность гипотезы (а не просто отвергнуть или не отвергнуть её).
- Можно использовать гипотезы с вероятностными утверждениями.
- Новые случаи можно классифицировать, объединяя несколько гипотез.
- Рассуждения достаточно строгие, много можно доказать математически.

Поиск скрытых параметров

- 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

- Начнём с 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 $x_1, \ldots, x_n$:

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

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).

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?

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.

Гауссиан: ГМП

- По μ:
$$\frac{\partial \ln p}{\partial \mu} = -\frac{n}{\sigma^2}(\mu - \bar{x}).$$
- То есть в гипотезе максимального правдоподобия $\mu = \bar{x}$, независимо от S.
- Теперь нужно найти σ 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

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

$$\sigma = \sqrt{\frac{\sum_{i=1}^{n}(x_i - \mu)}{n}}.$$

Упражнение. Оценить ошибку в полученных нами значениях (найти доверительные интервалы).

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 μ_k (пусть σ будет одна и та же).
- Тогда распределение будет

$$p(x_1, \ldots, x_n | \{\mu_k\}, \sigma) = \prod_{i=1}^{n} p(x_i | \{\mu_k\}, \sigma).$$

Домашнее задание

Упражнение. Разработать алгоритм, который итеративно максимизировал бы функцию правдоподобия при данных тестовых примерах, т.е. находил бы μ_k. За основу можно взять метод итераций Ньютона, который для максимизации функции $f(x)$ итерирует её как

$$x \leftarrow x - \frac{f(x)}{\partial f / \partial x}.$$

- Что должно получиться?

Что должно получиться

- Что значит «найти скрытые параметры смеси гауссианов»?
- Это значит — найти центры, вокруг которых чаще всего будут появляться точки, а также дисперсии (читай — диаметр областей, в которых более-менее вероятно обнаружить точки вокруг этих центров).
- Это в точности задача *кластеризации*!
- Когда вы сделаете упражнение, у вас должен получиться фактически вариант алгоритма k–средних для сферических кластеров.
- Но о кластеризации мы потом ещё будем говорить гораздо подробнее.

Важные распределения

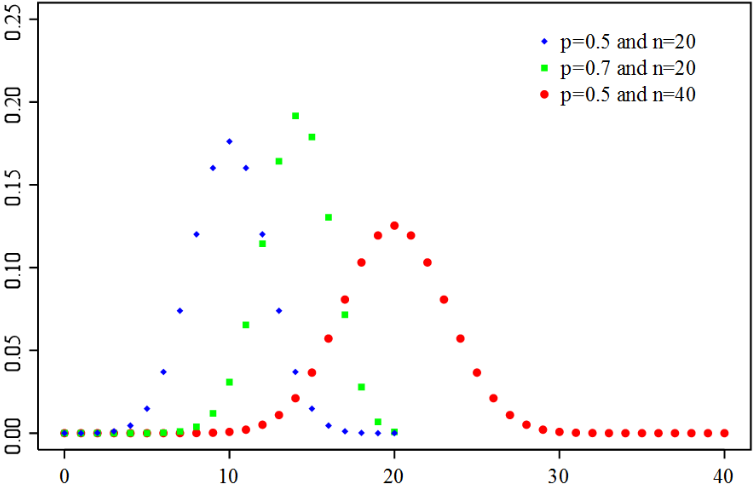
- Есть некоторое количество распределений, которые часто появляются в разных задачах и являются наиболее полезными на практике.
- Их полезно... ну, если не знать, то хотя бы быть знакомыми.
- Сейчас по ним и пробежимся.

Биномиальное распределение

- *Биномиальное распределение* возникает, когда мы подбрасываем нечестную монетку (вероятность решки p) n раз и хотим найти вероятность появления r решек.

$$p(r|p,n) = \binom{n}{r} p^r (1-p)^{n-r}.$$

Биномиальное распределение



Пуассоново распределение

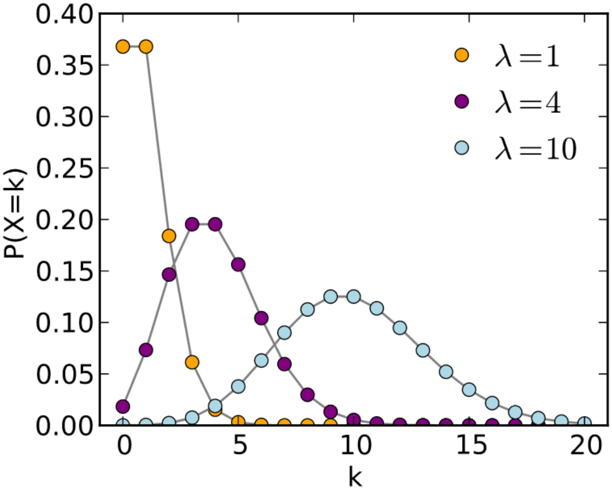
- *Распределение Пуассона* возникает, когда мы хотим подсчитать количество событий за фиксированный интервал, если нам дана средняя интенсивность этих событий.
- Если ожидается в среднем λ событий за этот интервал, то вероятность того, что произойдут ровно r событий, равна

$$p(r|\lambda) = e^{-\lambda}\frac{\lambda^r}{r!}.$$

- Распределение Пуассона – это предельный случай биномиального распределения. Если n очень велико, а p очень мало, $\mathrm{Binomial}(n, p)$ будет очень похоже на $\mathrm{Poisson}(np)$.

Упражнение. Проверьте это.

Пуассоново распределение



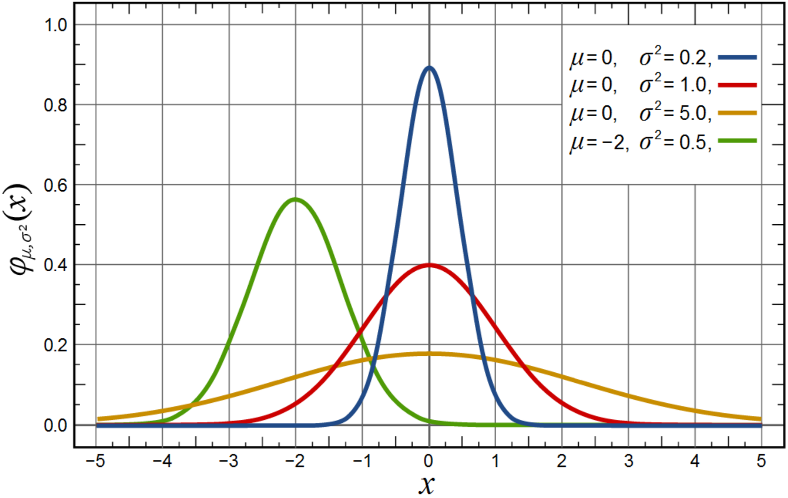
Нормальное распределение

- Мы уже давно знаем нормальное распределение:

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

- Очень многие процессы могут моделироваться нормальным (гауссовским) распределением; обычно возникает, когда есть некое среднее значение μ и шум вокруг него.

Нормальное распределение



Распределение Стьюдента

- Распределение Стьюдента применяется, когда нужно искать доверительные интервалы на параметры нормального распределения.
- Величина $T = \frac{\bar{x}_n - \mu}{S_n/\sqrt{n}}$ распределена по закону

$$f(t) = \frac{\Gamma(n/2)}{\sqrt{\pi(n-1)}\Gamma(\frac{n-1}{2})}\left(1 + \frac{t^2}{n-1}\right)^{-n/2}.$$

- Если мы выберем число A так, чтобы $p(-A < T < A) = 1 - \alpha$, то

$$\left[\bar{x}_n - A\frac{S_n}{\sqrt{n}}, \bar{x}_n + A\frac{S_n}{\sqrt{n}}\right]$$

будет доверительным интервалом для μ с вероятностью ошибки α.

Экспоненциальное распределение

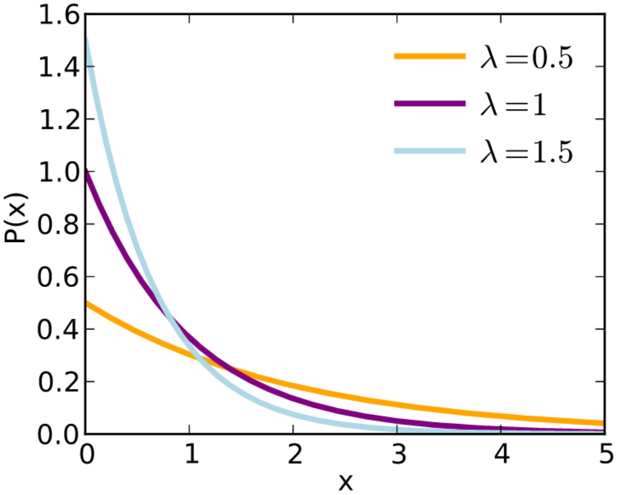
- Обычно возникает в ответах на вопрос «сколько надо ждать события».
- Дискретный вариант: вероятность того, что выпадения орла на нечестной монетке придётся ждать ровно r шагов, равна

$$p(r|p) = p^r(1-p) = (1-p)e^{-\lambda r}, \text{ где } \lambda = \ln\frac{1}{p}.$$

- Непрерывный вариант: если мы ждём события, которое происходит в среднем каждые $1/\lambda$ единиц времени (в пуассоновском процессе с интенсивностью λ), то распределение времени ожидания

$$p(x|s) = \lambda e^{-\lambda x}.$$

Экспоненциальное распределение

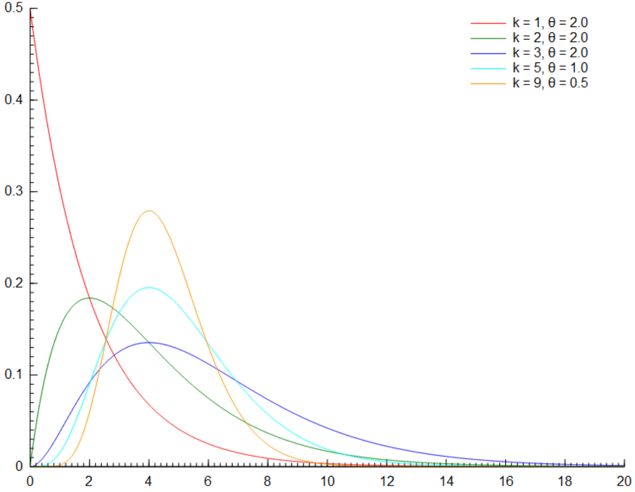


Гамма–распределение

- Может возникать как сумма нескольких экспоненциальных распределений.
- Плотность распределения

$$p(x|k,\theta) = x^{k-1}\frac{e^{-x/\theta}}{\theta^k\Gamma(k)}, \quad x > 0.$$

Гамма-распределение



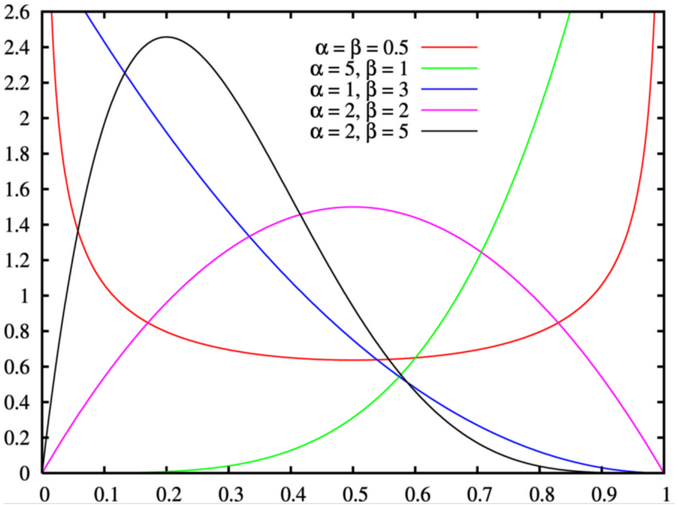
Бета–распределение

- Бета-распределение определяется над вероятностями, т.е. на интервале $[0, 1]$.
- Может служить априорным распределением для каких-либо вероятностей; является апостериорным распределением для испытаний Бернулли после

$$\mathrm{Beta}(x|\alpha,\beta) = \frac{1}{B(\alpha,\beta)} x^{\alpha-1}(1-x)^{\beta-1}, \quad \beta(\alpha,\beta) = \frac{\Gamma(\alpha)\Gamma(\beta)}{\Gamma(\alpha+\beta)}.$$

- $\mathrm{Beta}(i, j)$ — это распределение i-й по величине случайной величины из $i + j - 1$ случайных величин, распределённых равномерно на $[0, 1]$.

Бета–распределение



Порядковые статистики

- Бета–распределение — частный случай *порядковой статистики* (order statistics).
- Если n случайных величин $X_1, \ldots, X_n$ распределены одинаково с функцией распределения F, то k-я порядковая статистика $Y_k^{(n)}(y)$ — это функция распределения k-й сверху величины из этих n величин.
- Например, очевидно, что

$$Y_1^{(n)}(y) = F(y)^{n-1}.$$

Упражнение. Вывести формулу для второй порядковой статистики $Y_2^{(n)}$.

Распределение Дирихле

- Обобщение бета–распределения на многомерный случай.
- На k-мерном симплексе $\{x \mid \sum_{i=1}^{k} x_i = 1\}$ плотность распределения Дирихле с параметром $\alpha = (\alpha_1, \ldots, \alpha_k)$ равна

$$p(x|\alpha) = \frac{1}{B(\alpha)} \prod_{i=1}^{k} x_i^{\alpha_i - 1}, \quad B(\alpha) = \frac{\prod_{i=1}^{k} \Gamma(\alpha_i)}{\Gamma\left(\sum_{i=1}^{k} \alpha_i\right)}.$$

Thank you!

Спасибо за внимание!