Оптимизация и стохастика

gasnikov@yandex.ru

Задача 1 (SA vs SAA по Лану–Немировскому–Шапиро–Юдицкому)*. Дана задача выпуклой стохастической оптимизации: $f(x)=E_\xi\left[f(x,\xi)\right]\to\min\limits_{x\in Q}$, где $f(x,\xi)$ – выпуклая по $x\in\mathbb{R}^n$ ($n\gg 1$) функция. Будем считать, что п.н. $\left\|\nabla f(x,\xi)\right\|_2\le M$ ($\nabla=\nabla_x$ и E_ξ – перестановочны), а размер выпуклого замкнутого множества Q равен R (в действительности, достаточно считать, что R – расстояние от точки старта до решения, при этом множество Q может быть не ограничено). Предложите такой (SA) метод, на каждом шаге (итерации) которого считается проекция стохастического (суб-)градиента функции $f(x,\xi)$ (с независимой разыгранной с.в. ξ) по x на множество Q, что ($\sigma>0$ – малый доверительный уровень)[1]

$$P_{x_N}\left(E_\xi\left[f(x_N,\xi)\right]-\min_{x\in Q}E_\xi\left[f(x,\xi)\right]\ge CMR\sqrt{\frac{1+\ln\left(\sigma^{-1}\right)}{N}}\right)\le\sigma,$$

где C – константа (~ 10), а с.в. x_N – то, что выдает алгоритм после N итераций. Таким образом, для достижения точности по функции ε и доверительного уровня σ методу потребуется $\mathrm{O}\left(M^2R^2\ln\left(\sigma^{-1}\right)/\varepsilon^2\right)$ итераций (вычислений стохастического градиента и его проектирований). Покажите, что если использовать метод Монте-Карло (SAA), заключающийся в замене исходной задачи следующей задачей

$$\frac{1}{N}\sum_{k=1}^N f(x,\xi_k)\to\min_{x\in Q},$$

где с.в. ξ_k – i.i.d., и распределены также как и ξ, то для того, чтобы гарантировать, что абсолютно точное решение этой новой задачи является (ε,σ)-решением исходной задачи потребуется порядка $\mathrm{O}\left(M^2R^2\left(n\ln\left(MR/\varepsilon\right)+\ln\left(\sigma^{-1}\right)\right)/\varepsilon^2\right)$ итераций.

Указание. Воспользуйтесь неравенством больших уклонений для взвешенных сумм из задачи [ССЫЛКА] раздела [ССЫЛКА].

Замечание. Эта задача хорошо поясняет, что подход, связанный с усреднением случайности за счет самого метода более предпочтителен, чем замена задачи ее стохастической аппроксимацией (эта идея довольно стара; по-видимому, одним из первых кто количественно смог это прочувствовать был Б.Т. Поляк). Более предпочтителен не

[1] Эта оценка неулучшаема с точностью до мультипликативной константы C (даже в детерминированном случае $f(x,\xi)\equiv f(x)$), см. Немировский–Юдин.

только тем, что допускает адаптивность постановки и легко переносится на онлайн модификации исходной задачи, но, прежде всего, лучшей приспособленностью к большим размерностям. Подробнее о методах SA написано в статье *Juditsky A., Lan G., Nemirovski A., Shapiro A.* Stochastic approximation approach to stochastic programming // SIAM Journal on Optimization. 2009. V. 19. № 4. P. 1574–1609. О методах SAA написано в монографии *Shapiro A., Dentcheva D., Ruszczynski A.* Lecture on stochastic programming. Modeling and theory. MPS-SIAM series on Optimization, 2009. Здесь важно отметить, что в этой задаче "сидит" фундаментальная идея о том, что для получения (агрегирования) хороших оценок неизвестных параметров (особенно когда размерность пространства параметров велика) имеет смысл рассматривать задачу поиска оптимальных значений параметра, как задачу стохастической оптимизации и рассматривать выборку, как источник стохастических градиентов. Например, истинное значение неизвестного вектора параметров в предположении верности исходной параметрической гипотезы может быть записано как решение задачи стохастической оптимизации $\theta^* = \arg\max\limits_{\theta\in Q} E_\xi L(\theta,\xi)$ (метод наибольшего правдоподобия Фишера), где $L(\theta,\xi)$ – логарифм функции правдоподобия. Однако решать эту задачу обычными методами мы не можем, потому что математическое ожидание берется по с.в. ξ, распределение которой задается $L(\theta^*,\xi)$, а θ^* – не известно. Этот порочный круг распутывается, если мы будем решать задачу $E_\xi\left[-L(\theta,\xi)\right]\to\min\limits_{\theta\in Q}$ методами стохастической оптимизации, получая на каждом шаге новую реализацию (элемент выборки) ξ_k и рассчитывая значения градиента $\partial L(\theta,\xi_k)/\partial\theta$. То, что выдает алгоритм и будет (ε,σ)-оценкой вектора неизвестных параметров. Отметим, что, как правило, дополнительно известно, что $L(\theta,\xi)$ – гладкая и μ-сильно вогнутая (равномерно по ξ) функция по θ. Последнее обстоятельство позволяет получить лучшую скорость сходимости по функции $\mathrm{O}\left(M^2\ln\left(\ln(N)/\sigma\right)/(\mu N)\right)$, т.е. ($x=\theta$, $f=-L$)

$$P_{x_N}\left(E_\xi\left[f(x_N,\xi)\right]-\min_{x\in Q}E_\xi\left[f(x,\xi)\right]\geq\bar{C}M^2\frac{\ln\left(\ln(N)/\sigma\right)}{\mu N}\right)\leq\sigma.$$

Из неравенства Рао–Крамера будет следовать, что такая оценка не улучшаема (с точностью до фактора $\ln(\ln(N))$). Правда, тут возникают некоторые тонкости, когда мы говорим о неулучшаемости оценок с учетом вероятностей больших отклонений. Строго говоря, результаты типа Рао–Крамера, Ван-Трисса и т.п. (см., например, классическую монографию Ибрагимов–Хасьминский) позволяют лишь говорить о неулучшаемости в смысле сходимости полных математических ожиданий (без вероятностей больших отклонений), и именно в таком смысле можно получить неулучшаемую (с точностью до мультипликативной константы) оценку:

$$E\left[f(x_N,\xi)\right]-\min_{x\in Q}E_\xi\left[f(x,\xi)\right]\leq\frac{\breve{C}M^2}{\mu N}.$$

Можно обобщить рассмотренную задачу на случай, когда $\|\nabla f(x,\xi)\|_2$ имеет субгауссовский хвост. Тогда (в том числе в сильно выпуклом случае) вместо $\ln(\sigma^{-1})$ стоит писать $\ln^2(\sigma^{-1})$. Если же $\|\nabla f(x,\xi)\|_2^2$ имеет степенной хвост

$$P\left(\frac{\|\nabla f(x,\xi)\|_2^2}{M^2} \geq t\right) = \mathrm{O}\left(\frac{1}{t^\alpha}\right),$$

то ($\alpha > 2$)

$$P_{x_N}\left(E_\xi\left[f(x_N,\xi)\right] - \min_{x\in Q} E_\xi\left[f(x,\xi)\right] \geq C_\alpha MR\frac{\sqrt{N} + (N/\sigma)^{1/\alpha}}{N}\right) \leq \sigma.$$

Если дополнительно $f(x) = E_\xi\left[f(x,\xi)\right]$ – μ-сильно выпуклая функция, то ($\alpha > 1$)

$$P_{x_N}\left(E_\xi\left[f(x_N,\xi)\right] - \min_{x\in Q} E_\xi\left[f(x,\xi)\right] \geq \bar{C}_\alpha M^2\frac{\ln(\ln(N)) + \sigma^{-1/\alpha}}{\mu N}\right) \leq \sigma.$$

Если ничего не известно о $\|\nabla f(x,\xi)\|_2^2$, кроме $E\|\nabla f(x,\xi)\|_2^2 \leq M^2$, то по неравенству Маркова (второе неравенство подразумевает μ-сильную выпуклость $f(x)$)

$$P_{x_N}\left(E_\xi\left[f(x_N,\xi)\right] - \min_{x\in Q} E_\xi\left[f(x,\xi)\right] \geq \frac{\hat{C}MR}{\sigma\sqrt{N}}\right) \leq \sigma,$$

$$P_{x_N}\left(E_\xi\left[f(x_N,\xi)\right] - \min_{x\in Q} E_\xi\left[f(x,\xi)\right] \geq \frac{\breve{C}M^2}{\sigma\mu N}\right) \leq \sigma.$$

Все сказанное выше обобщается и на другие прокс-структуры (не обязательно евклидовы). Так в задачах 8, 9 рассматривается прокс-структура, порожденная "расстоянием" *KL* Кульбака–Лейблера (сильно выпуклом в 1-норме с константой 1 на единичном симплексе – неравенство Пинскера), по-видимому, "наилучшим" образом подходящая для симплекса (с некоторыми оговорками – см. замечание к задаче 10). Выгода от ее использования в том, что теперь $\|\nabla f(x,\xi)\|_\infty \leq M$, что в типичных ситуациях (см. задачу на теорему Б.С. Кашина о поперечниках) дает оценку константы M в $\sim\sqrt{n}$ раз лучше, а плата за это – увеличение оценки размера области в $\sim\ln n$.[2] Детали имеются

[2] Стоит обратить внимание, что если выбрана евклидова прокс-структура, то квадрат размера множества Q есть квадрат евклидова диаметра Q. При переходе к другой прокс-структуре в качестве квадрата размера Q фигурирует прокс-диаметр Q, поделенный на константу сильной выпуклости α прокс-функции, заданной на Q, относительно выбранной нормы в прямом пространстве. Скажем, в случае выбора KL-прокс структуры, 1-нормы в прямом пространстве и $Q = S_n(d)$, имеем

в упомянутой в начале замечания статье. Также все сказанное выше обобщается со схемы i.i.d. на случай, когда $\left\{\nabla f\left(x,\xi_k\right)-\nabla E_{\xi_k}\left[\,f\left(x,\xi_k\right)\right]\right\}$ – мартингал-разность (см. задачу 8). Это замечание существенно для перенесения (а это делается с сохранением всех оценок и даже констант в этих оценках) отмеченных результатов на задачи стохастической онланй оптимизации (сейчас это направления бурно развивается, см. обзоры S. Bubeck, E. Hazan, A. Rakhlin).

Подчеркнем, что все приведенные здесь оценки, вообще говоря (без дополнительных предположений), неулучшаемы (с точностью до мультипликативных констант). Один пример того, как это можно показать был рассмотрен выше (в общем случае следует смотреть монографию Немировского–Юдина). Сейчас же отметим, что если дополнительно известно, что $f\left(x,\xi\right)$ – гладкая по x функция, с константой Липшица градиента L и(или) сильно выпуклая с константой μ,[3] а вычисление и проектирование стохастического градиента на каждом шаге находится с неконтролируемой точностью δ, вообще говоря, не случайной природы (в этом месте есть неаккуратность, в действительности, определение δ-оракула, выдающего стохастический градиент, более тонкое[4]), то из недавних результатов Nesterov–Devolder–Glineur и Ghadimi–Lan можно получить (с некоторыми оговорками; кроме того, мы приводим немного огрубленный вариант для больше наглядности) такие оценки скорости сходимости (здесь стоит отметить большую работу, проделанную П.Е. Двуреченским, по получению нужного обобщения упомянутых выше результатов)

$$\min\left\{\mathrm{O}\left(\frac{LR^2}{N^p}+\sqrt{\frac{DR^2}{N}}+N^{p-1}\delta\right),\mathrm{O}\left(LR^2\exp\left(-NC\left(\frac{\mu}{L}\right)^{\frac{1}{p}}\right)+\frac{D}{\mu N}+\left(\frac{L}{\mu}\right)^{\frac{p-1}{p}}\delta\right)\right\},$$

$$R^2=KL\left(S_n(d)\right)/\alpha\left(S_n(d)\right)=d\cdot KL\left(S_n(1)\right)/\left(\alpha\left(S_n(1)\right)/d\right)=d^2\cdot KL\left(S_n(1)\right)/\alpha\left(S_n(1)\right)=d^2\cdot(\ln n)/1=d^2\ln n,$$

Для евклидовой прокс-структуры размер $Q=S_n(d)$ равнялся бы $2d^2$. Отсюда можно сделать вывод (верный и в общем случае), что выбор прокс-структуры имеет целью оптимально учесть структуру множества с точки зрения того как в итоговую оценку числа итераций будет входить размер пространства, в котором проходит оптимизация. При гомотетичном увеличении/уменьшении множества оценки числа итераций будут меняться одинаково, независимо от выбранной прокс-структуры.

[3] И гладкости и сильной выпуклости можно добиться искусственно: про гладкость мы скажем далее, а сильная выпуклость получается регуляризацией функционала в исходной задаче. Как правило, это не дает ничего нового с точки зрения выписанных оценок. Но в ряде специальных случаев, когда у задачи, есть дополнительная (например, седловая) структура, это может приносить определенные дивиденды. Пожалуй, наиболее ярким примером является метод двойственного сглаживания Ю.Е. Нестерова (регуляризации двойственной задачи с целью улучшения гладких свойств прямой).

[4] (δ,L)-оракул выдает такие $\left(F(x,\xi),G(x,\xi)\right)$, что $D_\xi\left[G(x,\xi)\right]\le D$, и для любых $x,y\in Q$

$$0\le E_\xi\left[\,f(y,\xi)\right]-E_\xi\left[F(x,\xi)\right]-\left\langle E_\xi\left[G(x,\xi)\right],y-x\right\rangle\le\frac{L}{2}\|y-x\|^2+\delta.$$

где $C \geq 1$ – некоторая константа, D – дисперсия $\nabla f(x,\xi)$, а параметр $p \in [1,2]$ подбирается оптимально исходя из масштаба шума δ. Дисперсию можно уменьшать, запрашивая на одном шаге реализацию стохастического градиента не один раз, а m раз, и заменяя стохастический градиент средним арифметическим, мы уменьшаем дисперсию в m раз. Это имеет смысл делать, если слагаемое, отвечающее стохастичности, доминирует. Важно, что мы при этом не увеличиваем число итераций, и слагаемое $N^{p-1}\delta$ остается прежним. Выписанные оценки характеризуют скорость сходимости в среднем. Они с одной стороны не улучшаемы[5] (причем это остается верно при $\delta = 0$ и(или) $D = 0$; при $D = 0$ мы считаем $\varepsilon \gg n^{-2}$, в противном случае оценки улучшаемы – методы эллипсоидов и внутренней точки, с оценками числа итераций типа $\sim n^{\alpha} \ln(R/\varepsilon)$, $\alpha \geq 1$) с точностью до мультипликативной константы, с другой стороны достигаются. В терминах больших отклонений возникает аналогичные оговорки тем, которые мы выше делали с одним исключением – в сильно выпуклом случае появляется дополнительная зависимость от N в множители, содержащем σ (см. выше). Эти результаты переносятся и на прокс-структуры отличные от евклидовой. При этом рассмотрение какой-либо отличной от евклидовой структуры в сильно выпуклом случае (когда минимум достигается на втором выражении), как правило, не имеет смысла, поскольку квадрат евклидовой асферичности p-нормы, возникающий в оценках числа обусловленности прокс-функции в p-норме (это число, в свою очередь, оценивает увеличение числа итераций метода при переходе от евклидовой норме к p-норме), больше либо равен 1. Равенство достигается на евклидовой норме. Скажем (см. задачу 15), для 1-нормы эта асферичность оценивается снизу размерностью пространства.

Множество Q должно быть достаточно простой структуры, чтобы на него можно было эффективно проектироваться. Однако в приложениях часто возникают задачи условной минимизации, в которых есть ограничения вида $g(x) \leq 0$, где $g(x)$ – выпуклые функции. “Зашивать” эти ограничения в Q, как правило, не представляется возможным в виду вышесказанного требования. Тем не менее, на основе описанного выше можно строить (за дополнительную логарифмическую плату) двухуровневые методы условной оптимизации подобно тем методам, которые описаны в конце главы 2 и 3 монографии *Нестеров Ю.Е.* Методы выпуклой оптимизации. М.: МЦНМО, 2010. При этом на каждом шаге такого метода потребуется проектироваться на пересечение множества Q с некоторым полиэдром, вообще говоря, зависящим от номера шага.

Все сказанное выше переносится в полной мере на задачи композитной оптимизации (Ю.Е. Нестеров) и частично на монотонные вариационные неравенства (Ю.Е. Нестеров).

Отметим также, что параметры R и μ могут быть не известны априорно или процедуры их оценивания приводят к слишком соответственно завышенным и заниженным результатам. Это может быть проблемой, поскольку знание этих и других

[5] В нижнюю оценку во втором выражении под знаком минимума при экспоненте вместо LR^2 входит μR^2, а константа $C = 1$.

параметров требуется методу для расчета величин шагов. Из этой ситуации можно выйти за логарифмическое (по этим параметрам) число рестартов метода. Стартуя, скажем, с $R=1$ и делая, предписанное этим R число шагов, мы проверяем выполняется ли условие ε-близости. Если нет, то полагаем $R:=2R$ и т.д. Это константное время и не отразится на общих трудозатратах. Аналогичное можно сказать про L и D. Однако если убрать стохастичность, тогда L можно не только эффективнее адаптивно подбирать по ходу самих итераций (увеличив в среднем число итераций не более чем в 4 раза), но и в некотором смысле оптимально самонастраиваться на гладкость функционала на текущем участке пребывания метода (речь идет о недавно предложенном универсальном методе Ю.Е. Нестерова). Для оценки D в ряде случаев бывает эффективнее воспользоваться какой-нибудь статистической процедурой.

В число итераций описанных методов не входит размерность пространства n. Это наталкивает на мысль о возможности использовать эти методы, например, в гильбертовых пространствах. Оказывается, это, действительно, можно делать. В частности, концепция неточного оракула позволяет привнести сюда элемент новизны, существенно мотивированной практическими нуждами (много материала о градиентных методах решения задач оптимизации в гильбертовых пространствах собрано во втором томе учебника Ф.П. Васильева по методам оптимизации).

Наконец, полезно иметь в виду, что за счет допускаемой неточности оракула, выдающего (стохастический) (суб-)градиент, можно "погрузить" задачу с гельдеровым градиентом $\left\|\nabla f(x,\xi)-\nabla f(y,\xi)\right\|_* \le L_\nu \left\|x-y\right\|^\nu$ (в том числе и негладкую задачу с ограниченной нормой разности субградиентов $\nu=0$) в класс гладких задач с оракулом, характеризующимся точностью δ и $L=L_\nu\left[\frac{L_\nu(1-\nu)}{2\delta(1+\nu)}\right]^{\frac{1-\nu}{1+\nu}}$.

В стохастической онлайн ситуации оценки будут такими:

$$\min\left\{\mathrm{O}\left(\sqrt{\frac{M^2R^2}{N}}+\delta\right),\mathrm{O}\left(\frac{M^2\ln N}{\mu N}+\delta\right)\right\}.$$

Эти оценки достигаются и неулучшаемы. Как видно из этих оценок наличие гладкости ничего не дает. Все что ранее говорилось про прокс-структуру и большие отклонения полностью и практически без изменений (в μ-сильно выпуклом случае в оценках вероятностей больших уклонений $\ln(\ln(N))\Rightarrow\ln N$) переносится и на онлайн случай. Отметим также, что онлайн методы, как правило, допускают прямо-двойственную модификацию (с теми же оценками скорости сходимости) при применении к задачам условной оптимизации.

В заключение отметим, что следует различать задачи стохастической оптимизации и задачи, в которые мы сами искусственно привносим случайность (рандомизацию) с целью более эффективного решения задачи. К этой ситуации, скажем, можно отнести случай, когда (негладкий) выпуклый функционал в задаче детерминированный, но

представляет собой трудно вычислимый интеграл от параметров, который компактно представим в виде математического ожидания по некоторой не сложной вероятностной мере. Тогда выгоднее считать стохастический градиент. Существенно экономя на вычислениях на каждом шаге и лишь не много теряя на логарифмическом увеличении числа шагов. Такой пример будет рассмотрен в следующей задаче. Однако если мы можем вычислять значение функции в детерминированной задаче, в которую мы решали рандомизированным методом, то ни о каких тяжелых хвостах можно не заботиться. Поскольку, выбрав число шагов так, чтобы метод находил ε-решение с вероятностью $\geq 1/2$ и запустив $\log_2\left(\sigma^{-1}\right)$ реализаций такого метода мы за дополнительную $\log_2\left(\sigma^{-1}\right)$ плату (мультипликативную) получим с вероятностью $1-\sigma$ среди выданных ответов хотя бы одно ε-решение. Однако предположение о возможности вычислять значения функции в ряде задач "натыкается" на существенные вычислительные сложности (значительно большие, чем при расчете стохастического градиента). Таким примером является задача поиска вектора PageRank $P^T p = p$ (P – стохастическая матрица), сводящаяся к негладкой задаче выпуклой оптимизации (матричной игре) $\max\limits_{u\in S_n(1)} \left\langle u, P^T p - p \right\rangle \to \min\limits_{p\in S_n(1)}$ (см. задачи 3, 14). Кроме того, если рандомизация осуществляется каким-то специальным образом, например, таким, что

$$E\left[\left\|\nabla f\left(x,\xi\right)\right\|_*^2\right] \leq C_n \left\|\nabla f\left(x\right)\right\|_*^2 + \textit{малый добавок}\left(\varepsilon\right)$$

и в точке минимума $\nabla f\left(x\right)=0$, то приведенные выше оценки можно существенно улучшить (Б.Т. Поляк). Такая рандомизация возникает, например, в связи с изучением безградиентных методов и методов спуска по случайному направлению для гладких функционалов (см. задачу 6) или специальных функционалов вида суммы гладких сильно выпуклых функций (Konecny–Richtarik, Shalev-Shwartz–Zhang, LeRoux–Schmidt–Bach).

Задача 2 (рандомизация суммы)*. Пусть необходимо решить задачу выпуклой оптимизации $f\left(x\right)=\dfrac{1}{m}\sum\limits_{k=1}^{m} f_k\left(x\right) \to \min\limits_{x\in Q}$, где $f_k\left(x\right)$ – негладкие выпуклые функции с ограниченной нормой субградиента M, Q – выпуклое замкнутое множество простой структуры (можем эффективно на него проектироваться) размера R. Введем новую функцию

$$f\left(x,\xi\right)=\begin{cases} f_1\left(x\right), \text{с вероятностью } 1/m \\ \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \\ f_m\left(x\right), \text{с вероятностью } 1/m \end{cases}.$$

Ее градиент легко считается. Разыгрывается за $\mathrm{O}\left(\ln m\right)$ с.в. ξ, принимающая значения $1,\ldots,m$ с равными вероятностями (см. задачу [ССЫЛКА] раздела [ССЫЛКА]). Затем считается субградиент $f_\xi\left(x\right)$ (и проектируется на Q). Покажите, что метод из

предыдущей задачи находит (ε,σ)-решение за $\mathrm{O}\left(M^2R^2\ln\left(\sigma^{-1}\right)/\varepsilon^2\right)$, со стоимостью одной итерации равной $\mathrm{O}(\ln m)$ + затраты на вычисления градиента $f_\xi(x)$ + затраты на проектирование. Покажите, что если решать задачу без рандомизации, то число итераций будет $\mathrm{O}\left(M^2R^2/\varepsilon^2\right)$, строго говоря, здесь M должно быть не много меньше за счет того, что

$$\max_{x\in Q}\left\|\frac{1}{m}\sum_{k=1}^{m}\nabla f_k(x)\right\|_* \le \max_{\substack{k=1,\dots,m\\ x\in Q}}\left\|\nabla f_k(x)\right\|_*,$$

но мы считаем, что обе части неравенства одного порядка. Зато шаг итерации будет теперь почти в m раз дороже. И если $m\gg 1$ это может оказаться существенным.

Замечание. Приведенную в задаче постановку можно распространить на случай, когда взвешивание функций не равномерное (тогда первое разыгрывание/приготовление займет $\mathrm{O}(m)$, а все последующие $\mathrm{O}(\ln m)$) и $f_k(x) := E_{\xi_k}\left[f_k(x,\xi_k)\right]$ с равномерно ограниченными (по k, x и ξ) нормами субградиентов. При этом все приведенные оценки сохранятся. Причем требование равномерной ограниченности норм субградиентов можно существенно ослабить за небольшую плату (см. замечание к задаче 1). Отметим также, что если на задачу 1, полученную по методу Монте-Карло (SAA) смотреть в контексте рандомизации, предложенной в настоящей задаче, то "все встанет на свои места" в смысле одинаковости двух подходов SA и SAA с рандомизацией (с точностью до логарифмического фактора).

Описанная в этой задаче рандомизация, по-видимому, была одной из первых, которые предлагались в стохастической оптимизации (Ю.М. Ермольев). Однако и сейчас в связи с приложениями к анализу данных эта рандомизация активно используется (см., например, цикл работ P. Richtarik с соавторам). Так, например, если дополнительно известно, что все функции гладкие с константой липшица L, то в недавней работе Konecny–Richtarik был предложен специальный рандомизированный метод, в котором число вычислений градиентов ($R=1$)

$$\mathrm{O}\left(\left(m+L/\varepsilon\right)\left(\ln\left(\varepsilon^{-1}\right)+\ln\left(\sigma^{-1}\right)\right)\right),$$

а если дополнительно имеется еще и μ-сильная выпуклость, то (считаем $m\gg\sqrt{L/\mu}$)

$$\mathrm{O}\left(\left(m+L/\mu\right)\left(\ln\left(\varepsilon^{-1}\right)+\ln\left(\sigma^{-1}\right)\right)\right).$$

Результат с … L/ε получается из … L/μ при регуляризации $\mu R^2\sim\varepsilon$, т.е. $\mu\sim\varepsilon$.

Отметим также, что сначала получается результат о сходимости средних

$$E\left(f(x_N)-\min_{x\in Q}f(x)\right)\le\varepsilon,$$

где

$$N(\varepsilon) = \mathrm{O}\Big(\big(m + L/\mu\big)\ln\big(\varepsilon^{-1}\big)\Big),$$

Потом из неравенства Маркова получают оценку больших уклонений

$$P\Big(f\big(x_{N(\varepsilon)}\big) - \min_{x\in Q} f(x) \geq \sigma\Big) \leq \varepsilon/\sigma,$$

которую переписывают в виде

$$P\Big(f\big(x_{N(\varepsilon\sigma)}\big) - \min_{x\in Q} f(x) \geq \sigma\Big) \leq \varepsilon,$$

где

$$N(\varepsilon\sigma) = \mathrm{O}\Big(\big(m + L/\mu\big)\Big(\ln\big(\varepsilon^{-1}\big) + \ln\big(\sigma^{-1}\big)\Big)\Big).$$

Мы привели здесь это наблюдение, потому что оно оказывается полезным и во многих других контекстах. Так при рандомизации, возникающей в покомпонентных спусках и безградиентных методах в гладком сильно выпуклом случае, также возникает геометрическая скорость сходимости (Ю.Е. Нестеров). Описанная выше конструкция используется для оценки больших уклонений там. Причем за счет регуляризации функционала все это переносится просто на гладкий случай.

Задача 3 (матричная рандомизация Лана–Немировского–Шапиро–Юдицкого)*. Дана матрица A большого размера $m\times n$ (вообще говоря, не разреженная) с элементами ограниченными по модулю M, соответствующая (антагонистической) матричной игре (см. задачу 14). Покажите, что

$$Ax = EA^{\langle i[x]\rangle}, \text{ где } S_n(1) = \left\{x \in \mathbb{R}^n_+ : \sum_{i=1}^n x_i = 1\right\},$$

$A^{\langle i\rangle}$ – i-й столбец матрицы A, а с.в. $i[x]$ имеет мультиномиальное распределение с вектором параметров x. Важным следствием является тот факт, что левая часть равенства вычисляется за $m\times n$ операций, а правая часть всего лишь за $\mathrm{O}(n)$. Используя это наблюдения (и аналогичное для умножения матрицы A на вектор слева), предложите такой способ поиска равновесия Нэша в этой игре, который давал бы после

$$\mathrm{O}\Big(M^2(n+m)\Big(\ln(n+m) + \ln\big(\sigma^{-1}\big)\Big)\Big/\varepsilon^2\Big)$$

элементарных арифметических операций такие $x \in S_n(1)$ и $y \in S_m(1)$, что

$$\max_{\tilde{y}\in S_m(1)} \tilde{y}^T A x - \min_{\tilde{x}\in S_n(1)} y^T A \tilde{x} \leq \varepsilon$$

с вероятностью $\geq 1-\sigma$.

Замечание. При не высоких требованиях к точности и огромных размерах матрицы может получиться, что не потребуется считывать все элементы матрицы A. В отличие от рандомизированных процедур, любые детерминированные требуют просмотра, как минимум половины элементов матрицы A. По-видимому, первым эффективным (с такими же оценками скорости сходимости и с возможностью практически полной параллелизации шага итерации) рандомизированным методом для матричных игр был метод Григориадиса–Хачияна, см. *Хачиян Л.Г.* Избранные труды / сост. С. П. Тарасов. М.: МЦНМО, 2009. С. 38–48.[6] Этот метод, по-сути, описан в задаче 14. Важным в этих подходах является то, что мы живем на множестве $S_n(1)$. Для более сложных множеств эффективность рандомизации резко падает. Тем не менее, в определенных ситуациях рандомизация по-прежнему остается полезной. А с учетом того, что большой класс задач оптимизации, приходящих из high-dimensional statistics (например, возникающих в связи compressed sensing и т.п.) может быть представлен в виде седловых задач, то возникает целая индустрия рандомизированых методов решения седловых задач, активно развиваемых А.С. Немировским с коллегами.

Задача 4 (FGM – быстрый градиентный метод; Ю.Е. Нестеров).** Дана гладкая (с константой Липшица градиента L) μ-сильно выпуклая функция $f(x)$. Требуется найти ее минимум. Метод[7] (с $x_0 = y_0$)

$$x_{k+1} = y_k - \frac{1}{L}\nabla f(y_k),$$

[6] В этой книге также написано о том, что такое битовая сложность применительно к решению задач оптимизации. Выпускник ФУПМ МФТИ лауреат премии Фалкерсона Л.Г. Хачиян (1952–2005) в конце 70-х годов показал, что в битовой сложности задача линейного программирования полиномиальна по размеру входа. Этот результат получил мировую известность. Стоит отметить, что встречающаяся у нас в замечании к задаче 1 концепция неточного оракула (если ее интерпретировать как неточность, возникающую при округлении из-за конечности длины мантиссы) не позволяет напрямую работать в концепции битовой сложности. Тем не менее, это связанные вещи.

[7] Полезно сначала посмотреть на этот метод, когда $x \in \mathbb{R}^1$, учитывая, что переход (шаг обычного градиентного метода, сходящегося как $O(LR^2/k)$) $x_{k+1} = y_k - \nabla f(y_k)/L$ имеет следующий геометрический смысл. Через точку $(y_k, f(y_k))$ мы проводим параболу, касающуюся в этой точке графика функции $f(y)$, и имеющую кривизну L. Тогда (по условию) график функции $f(y)$ везде кроме точки $(y_k, f(y_k))$ лежит ниже это параболы. В частности, в точке минимума этой параболы x_{k+1}. Эту точку и предлагается брать в качестве следующего приближения к минимуму в обычном градиентном спуске. Но, очевидно, что мы “не дошли” до минимума функции $f(x)$. Однако более правильную картинку того, как работает этот метод можно получить, если рассмотреть $x \in \mathbb{R}^2$ и $f(x) = 1/2(\mu x_1^2 + L x_2^2)$ с $\mu/L \ll 1$, и стартовать, скажем, с точки (x_1^0, x_2^0), с $0 < x_2^0 \ll x_1^0$. По-сути, речь идет об овражном методе (см., например, 1-й параграф главы 5 учебника Васильев Ф.П. Методы оптимизации. Т. 1. М.: МЦНМО, 2011).

$$y_{k+1} = x_{k+1} + \frac{\sqrt{L}-\sqrt{\mu}}{\sqrt{L}+\sqrt{\mu}}\left(x_{k+1} - x_k\right)$$

является "оптимальным"[8] в том смысле, что после k итераций (R – расстояние от точки старта до решения)

$$f\left(x_k\right) - \min_x f\left(x\right) = \mathrm{O}\left(LR^2 \exp\left(-k\sqrt{\frac{\mu}{L}}\right)\right),$$

что с точностью до замены LR^2 на μR^2 перед экспонентной, соответствует нижней оценки для этого класса функций (Немировский–Юдин). Обобщите оценку скорости сходимости этого метода на случай задачи стохастической оптимизации

$$f\left(x\right) \coloneqq E_\xi\left[f\left(x,\xi\right)\right],\ \nabla f\left(x\right) \to \nabla_x f\left(x,\xi\right),$$

где

$$E_\xi\left[\nabla_x f\left(x,\xi\right)\right] = \nabla_x E_\xi\left[f\left(x,\xi\right)\right] \text{ и } D_\xi\left[\nabla_x f\left(x,\xi\right)\right] \le D.$$

Указание. Для получения неравенств больших уклонений полезно ознакомиться с задачей 1, а также с работой *Rakhlin A., Shamir O., Sridharan K.* Making gradient descent optimal for strongly convex stochastic optimization // e-print, 2012.

Замечание. Прежде чем читать далее рекомендуется посмотреть замечание к задаче 1. FGM имеет один недостаток – чувствительность к неточности оракула δ, выдающего градиент. А именно

$$\min\left\{\mathrm{O}\left(\frac{LR^2}{N^2} + \sqrt{\frac{DR^2}{N}} + N\delta\right), \mathrm{O}\left(LR^2\exp\left(-N\sqrt{\frac{\mu}{L}}\right) + \frac{D}{\mu N} + \sqrt{\frac{L}{\mu}}\delta\right)\right\}$$

В то время как обычный градиентный метод (с усреднением, см., например, PhD Thesis O. Devolder'a) давал бы

$$\min\left\{\mathrm{O}\left(\frac{LR^2}{N} + \sqrt{\frac{DR^2}{N}} + \delta\right), \mathrm{O}\left(LR^2\exp\left(-N\frac{\mu}{L}\right) + \frac{D}{\mu N} + \delta\right)\right\}.$$

[8] В таком виде записанный FGM не допускает предельного перехода по $\mu \to 0+$ с сохранением свойства оптимальности. Однако это проблема довольно легко решается путем представления FGM в немного более общем виде (не столь наглядном, поэтому мы и использовали в условии задачи упрощенный вариант). Это приводит к оценке $\mathrm{O}\left(\sqrt{LR^2/k}\right)$. Об этом написано в книге Ю.Е. Нестерова, 2010 и в обзоре *Bubeck S.* Theory of convex optimization for Machine Learning // e-print, 2014. Этот обзор можно также рекомендовать как наиболее современное и аккуратное изложение численных методов выпуклой оптимизации в пространствах больших и сверх больших размеров (huge-scale optimization).

Что лучше зависит от δ. В замечании к задаче 1 описан результат работы некоторый "выпуклой комбинации" этих методов. Нам представляется важным возникающее в этой связи понятие алгебры над алгоритмами. Мы используем здесь этот термин в немного другом (более частном) контексте, чем, скажем, в школе Ю.И. Журавлева.

Задача 5 (покомпонентный спуск; Ю.Е. Нестеров)*. Дана гладкая выпуклая функция $f(x)$, $x \in \mathbb{R}^n$. Требуется найти ее минимум x^*, который (для простоты) предполагается единственным. Считаются известными такие константы $\{L_i\}_{i=1}^n$, что

$$\left| \frac{\partial f(x + he_i)}{\partial x_i} - \frac{\partial f(x)}{\partial x_i} \right| \le L_i |h_i|, \; e_i = (\underbrace{0,...,0,1}_{i},0,...,0)^T.$$

На основе этих констант определим с.в.

$$\xi: \; P(\xi = i) = L_i \left(\sum_{i=1}^n L_i \right)^{-1}.$$

Введем взвешенную евклидову норму

$$\|x\|_\alpha = \left[\sum_{i=1}^n L_i^\alpha x_i^2 \right]^{1/2}, \; \alpha \in [0,1].$$

Положим

$$R_\beta(x^0) = \max_{x \in \mathbb{R}^n} \left\{ \|x - x^*\|_\beta : \; f(x) \le f(x^0) \right\}.$$

Покомпонентный градиентный спуск определяется следующим образом ($(k+1)$-я итерация):

1. Независимо разыгрываем с.в. ξ;
2. $x^{k+1} = x^k - \frac{1}{L_i} \frac{\partial f(x^k)}{\partial x_\xi^k} e_\xi$.

Покажите, что

$$E\left[f(x^N) \right] - \min_{x \in \mathbb{R}^n} f(x) \le \frac{2}{N} \left[\sum_{i=1}^n L_i^\alpha \right] R_{1-\alpha}^2(x^0).$$

Пусть дополнительно известно, что $f(x)$ – сильно выпуклая функция относительно нормы $\|\cdot\|_{1-\alpha}$, с параметром сильной выпуклости $\sigma_{1-\alpha} > 0$. Покажите, что

$$E\left[f\left(x^N\right)\right]-\min_{x\in\mathbb{R}^n} f(x)\le\left(1-\frac{\sigma_{1-\alpha}}{S_\alpha}\right)^N\left(f\left(x^0\right)-\min_{x\in\mathbb{R}^n} f(x)\right).$$

Получите оценки вероятностей больших отклонений.

Указание. Для получения оценки вероятности больших отклонений можно воспользоваться замечанием к задаче 2.

Замечание. Начальное приготовление памяти под генерирование (на каждой итерации) с.в. ξ стоит $\mathrm{O}(n)$ (делается один раз). При этом генерирование с.в. ξ на каждой итерации стоит $\mathrm{O}(\ln n)$, см. задачу [ССЫЛКА] раздела [ССЫЛКА]. Из задачи 4 можно заключить, что обычный градиентный спуск – не самый оптимальный метод. Можно ожидать этого и от обычного покомпонентного спуска. И, действительно, используя FGM (см. задачу 4) мы можем улучшить оценки стандартным образом. Детали имеются в статье *Nesterov Yu.* Efficiency of coordinate descent methods on huge-scale optimization problems // CORE Discussion paper. 2010/2.

Задача 6 (безградиентные методы).** Рассматривается задача стохастической выпуклой оптимизации $E_\xi\left[f(x,\xi)\right]\to\min_{x\in Q}$, где множество Q простой структуры (на него не сложно проектироваться) (прокс-)размера R, $\left\|\nabla_x f(x,\xi)\right\|_*\le M$ (рассмотрите другие, "вероятностные", варианты этого ограничения, см. замечание к задаче 1), ∇_x и E – перестановочны. Однако оракул не может выдавать стохастический (суб-)градиент функции. На каждой итерации мы можем запрашивать оракула только значения реализации функции $f(x,\xi)$ в нескольких точках (принципиальная разница есть только между одной или двумя точками). При этом оракул выдает нам значения реализации функции не точно, а с шумом $\left|\delta(x,\xi)\right|\le\delta=\mathrm{O}(\varepsilon)$. По этим значениям мы можем оценить стохастический градиент:

$$g_{\mu,\delta}(x,s,\xi)=\frac{n}{\mu}\left(f(x+\mu s,\xi)+\delta(x+\mu s,\xi)-\left(f(x,\xi)+\delta(x,\xi)\right)\right)s,$$

где s – случайный вектор (независимый от ξ), равномерно распределенный на единичной сфере в норме $\|\cdot\|$ (предложите способы генерирования за $\mathrm{O}(n)$ с.в. с равномерным распределением на единичной сфере в 1-норме и 2-норме), $\mu\sim\varepsilon/M$. Покажите, что[9]

$$E_{s,\xi,\delta}\left[g_{\mu,\delta}(x,s,\xi)\right]=\nabla f_\mu(x)+\nabla_x E_{\tilde{s},\delta}\left[\delta\left(x+\mu\tilde{s}\right)\right],$$

[9] Взятие математического ожидания по δ подчеркивает, что $\delta(x,\xi)$ может быть случайной величиной не только потому, что может зависеть от ξ, но и потому, что может иметь собственную случайность.

где $\tilde{s}$ – случайный вектор, равномерно распределенный на единичном шаре в норме $\|\cdot\|$, а $f_\mu(x) = E_{\tilde{s},\xi}\left[f(x+\mu\tilde{s},\xi)\right]$ – сглаженная[10] версия функции $f(x) = E_\xi\left[f(x,\xi)\right]$. Причем,

$$0 \le f_\mu(x) - f(x) \le M\mu.$$

Покажите также, что

$$\left\|g_{\mu,\delta}(x,s,\xi)\right\|_* \le n\left(M + \frac{2\delta}{\mu}\right).$$

Считая, что[11]

$$\left\|\nabla_x E_{\tilde{s},\xi,\delta}\left[\delta(x+\mu\tilde{s},\xi)\right]\right\|_* = \mathrm{O}(\varepsilon/R),$$

предложите метод (применимый (с такой же оценкой) и к задачам стохастической онлайн оптимизации), который для достижения точности ε в среднем требовал бы $\mathrm{O}\left(M^2R^2n^2/\varepsilon^2\right)$ итераций, на каждой из которых мы считаем $g_{\mu,\delta}$. Предложите метод, который бы в случае γ-сильно выпуклой функции $f(x)$ (прокс-структура евклидова), требовал бы в среднем $\mathrm{O}\left(M^2n^2/(\gamma\varepsilon)\right)$ итераций. Предложите модификацию этого метода, применимого к задачам стохастической онлайн оптимизации, для которого в среднем $\varepsilon = \mathrm{O}\left(M^2n^2\ln N/(\gamma N)\right)$.[12] Оцените вероятности больших отклонений предложенных методов.

Указание. Можно воспользоваться методом зеркального спуска (описанным ранее для Q – симплекса и прокс-структуры, порожденной энтропией и $\|\cdot\|_1$) применительно к задаче $E_{\tilde{s},\xi}\left[f(x+\mu\tilde{s},\xi)\right] \to \min\limits_{x\in Q}$, учитывая приведенные в условии задачи оценки. Для сильно выпуклого случая ситуация сложнее. Не в онлайн ситуации можно воспользоваться, например, методом из работы Rakhlin–Shamir–Sridharan. В онлайн ситуации (A. Rakhlin, E. Hazan, S. Bubeck) предлагается брать (также как и в случае, когда доступен стохастический градиент) обычный градиентный спуск с усреднением и

---

[10] Все свойства функции $f(x)$ при переходе к $f_\mu(x)$ могут только улучшиться. В частности, $f_\mu(x)$ также выпуклая функция (можно перенести и на сильную выпуклость с не меньшей константой), с константой липшица и константой липшица градиента не большей чем у $f(x)$.

[11] Если это условие не выполняется, то все что написано далее (в том числе в замечании) останется верным, правда, при более ограничительных условиях на допустимый уровень неточности. Так, если не налагать это ограничение, то в условии задачи потребуется считать $\delta = \mathrm{O}\left(\varepsilon^2/R\right)$ или $\delta = \mathrm{O}\left(\varepsilon^{3/2}/\left(R\sqrt{L}\right)\right)$ – в случае, если $f(x)$ имеет L-липшицев градиент (см. замечание). Выписанное условие выполняется (с точностью до R), если $E_{\xi,\delta}\left[\delta(x,\xi)\right]/\delta$ – $\mathrm{O}(1)$-липшицева функция (по x). Еще один случай будет рассмотрен ниже.

[12] По-видимому, все приведенные оценки оптимальны (и при $\delta = 0$).

выбором шага $h_k = (\gamma k)^{-1}$. Некоторые сложности в сильно выпуклом случае появляются с получением оценок вероятностей больших отклонений. Тем не менее, можно сказать, что ничего принципиально нового по сравнению со сказанным в замечании к задаче 1 здесь не возникает.

Замечание. Если на каждой итерации разрешается запрашивать значение функции только в одной точке (такие постановки возникают, например, в онланй оптимизации[13])

$$g_{\mu,\delta}(x,s,\xi) = \frac{n}{\mu}\big(f(x+\mu s,\xi) + \delta(x+\mu s,\xi)\big)s,$$

то оценка $\mathrm{O}\left(M^2R^2n^2/\varepsilon^2\right)$ ухудшается[14] $\mathrm{O}\left(M^2R^2n^2/\varepsilon^4\right)$.

Если дополнительно известно, что $f(x)$ – гладкая и $\|\nabla f(x) - \nabla f(y)\|_* \le L\|x-y\|$, то

$$\left|f_\mu(x) - f(x)\right| \le \frac{L\mu^2}{2},$$

как следствие, можно ослабить требование к неточности: допускать неточность оракула масштаба[15] $\delta \sim M\mu \sim M\sqrt{\varepsilon/L}$. Но при сделанных дополнительных условиях гладкости за счет ужесточения требований к масштабу допускаемой неточности можно улучшить скорость сходимости: фактор n^2 во всех выписанных оценках перейдет в n. Фактически это означает, что мы можем выбрать настолько маленькое μ (насколько маленьким мы можем его выбрать определяется δ), что конечная разность "превращается" (с нужной точностью) в производную по направлению.[16] Для объяснения отмеченного перехода полезно заметить, что (ниже мы поясним, откуда следует эта оценка)

---

[13] См. обзор *Bubeck S., Cesa-Bianchi N.* Regret analysis of stochastic and nonstochastic multi-armed bandit problems // Foundation and Trends in Machine Learning. 2012. V. 5. no. 1. P. 1–122.

[14] Поскольку в худшем случае можно лишь гарантировать, что $E_{s,\xi}\|g_{\mu,\delta}(x,s,\xi)\|_*^2 \le C(n/\mu)^2$.

[15] Здесь мы дополнительно считаем, что $\left\|\nabla_x E_{\tilde{s},\xi,\delta}\left[\delta(x+\mu\tilde{s},\xi)\right]\right\|_* = 0$. В частности, это условие выполняется, если неточность $\delta(x,\xi)$ имеет независимое от x распределение. Предположим теперь, что оракул может считать абсолютно точно значение липшицевой функции, но вынужден нам выдавать лишь конечное (предписанное) число первых бит. Таким образом, в последнем полученном бите есть некоторая неточность (причем мы не знаем по какому правилу оракул формирует этот последний выдаваемый значащий бит). Однако мы всегда можем прибавить (по mod 1) к этому биту случайно приготовленный (независимый) бит. В результате, не ограничивая общности, можно считать, что оракул последний бит выбирает просто случайно в независимости от отброшенного остатка.

[16] Если расшифровать то как входит размерность пространства в оценки скорости сходимости (число итераций (точность)) метода покомпонентного спуска (см. задачу 5), то там, в типичной ситуации, также возникает дополнительный (к оценкам полных градиентных методов) фактор n, причем в независимости от нюансов (есть ли сильная выпуклость, используется ли FGM или обычный метод).

$$E_{s,\xi}\left\|g_{\mu,\delta}(x,s,\xi)\right\|_*^2 \le 4nM^2 + L\mu^2 n^2 + \frac{8\delta^2 n^2}{\mu^2}.$$

Далее рассматривается не стохастический и не онлайн вариант постановки с $Q=\mathbb{R}^n$, R – расстояние от точки старта до решения. В этом варианте, выписанная оценка может быть уточнена

$$E_s\left\|g_{\mu,\delta}(x,s)\right\|_*^2 \le 4n\left\|\nabla f(x)\right\|_*^2 + L\mu^2 n^2 + \frac{8\delta^2 n^2}{\mu^2}.$$

Последняя оценка следует из явления концентрации равномерной меры на сфере с выделенными полюсами вокруг экватора (см. задачу [ССЫЛКА] раздела [ССЫЛКА]):

$$E_s\left[\left\langle \nabla f(x), s\right\rangle^2\right] = \frac{1}{n}\left\|\nabla f(x)\right\|_*^2.$$

Уточненная оценка позволяет (если неточность δ должным образом мала; речь идет об оценках типа: $\delta \sim \varepsilon^\beta/n$, $\beta \ge 1$) получить оценку $\mathrm{O}\left(n\sqrt{LR^2/\varepsilon}\right)$ и в γ-сильно выпуклом случае $\mathrm{O}\left(n\sqrt{L/\gamma}\ln\left(LR^2/\varepsilon\right)\right)$. Оценки вероятностей больших уклонений можно получить исходя из замечания к задаче 2. Детали имеются в работе *Nesterov Yu.* Random gradient-free minimization of convex function // CORE Discussion paper. 2011/1. Насколько нам известно, все выписанные оценки не улучшаемые, даже если предполагать, что нет неточности, и можно считать производные по направлению вместо конечных разностей.

Отметим в заключении, что при заданном уровне шума (неточность $\delta(x,\xi)$ имеет независимое от x распределение) мы всегда можем достичь сколь угодно высокой точности решения (по функции), см. также задачу 16.

Задача 7 (PageRank от Яндекс). а) (Е.Ю. Клочков)* Имеется ориентированный web-граф, соответствующий сети Интернет с $n \approx 10^9$ вершинами. Случайно (независимо и равновероятно) выбираются $N = \left(4 + 6\ln\left(\sigma^{-1}\right)\right)\varepsilon^{-2}$ вершин, из которых независимо начинают параллельно блуждать человечки. Блуждают человечки согласно правилам:

- за такт времени каждый человечек с вероятностью $\alpha \approx 0.15$ телепортируется в одну из вершин графа согласно заданному распределению вероятностей $\chi(\omega)$,
- с вероятностью $1-\alpha$ перейдет в соседнюю вершину, разыгранную согласно распределению вероятностей, из соответствующей строки заданной стохастической матрицы $P(\varphi)$.

Покажите, что найдется такое $C>0$, что после $T=C\ln(n/\varepsilon)\alpha^{-1}$ тактов (шагов) вектор $n(T)$, характеризующий сколько в какой вершине человечков, будет давать PageRank π в следующем смысле:

$$P\left(\left\|\frac{n(T)}{N}-\pi\right\|_2\le\varepsilon\right)\ge 1-\sigma,$$

где $\pi=\pi(\omega,\varphi)$ – единственное (в классе распределений вероятностей) решение уравнения

$$\alpha\chi(\omega)+(1-\alpha)P^T(\varphi)\pi=\pi. \qquad \text{(PR)}$$

Оцените общее количество затраченных арифметических операций, требующихся для нахождения PageRank с "точностью" (ε,σ) в указанном выше смысле. Как изменится ответ, если известно, что в каждой строке матрицы $P(\varphi)$ все отличные от нуля элементы одинаковы? Сравните описанный в этом пункте метод (Markov chain Monte Carlo) с обычным методом простой итерации (Power method).

б) (безградиентные методы и huge-scale optimization)** В п. а) мы определили зависимость $\pi(\omega,\varphi)$. Будем считать, что $\omega,\varphi\in S_m(1)$, где $m\approx 10^3$. Эти вектора характеризуют "состояние мира", которое мы хотим восстановить:

$$\left\|\pi(\omega,\varphi)-\pi_{\text{expert}}\right\|_2^2\to\min_{\omega,\varphi\in S_m(1)},$$

где π_{expert} – известный вектор, отражающий ранжирование web-страниц экспертами. Считая, что зависимости $\chi(\omega)$ и $P(\varphi)$ таковы, что эта задача выпуклая, предложите эффективный численный способ ее решения.

Указание. Воспользуйтесь безградиентным методом (см. задача 6), в котором оракул, выдающий значение функции использует MCMC или Power method для решения вспомогательной задачи. Необходимо определить каким методом и насколько точно оракулу требуется решать вспомогательную задачу на каждом шаге в зависимости от того какой вариант безградиентного метода выбран. Перебрав различные варианты (их не большое число) предложите "оптимальное сочетание". Причем важно, что (кроме первой итерации) нам надо пересчитывать PageRank, а не считать его каждый раз заново. Другими словами, если мы уже посчитали $\pi(\omega,\varphi)$, а хотим посчитать $\pi(\omega+\Delta\omega,\varphi+\Delta\varphi)$, то вектор $\pi(\omega,\varphi)$ можно использовать в качестве начального приближения (точки старта для метода, численно решающего задачу поиска $\pi(\omega+\Delta\omega,\varphi+\Delta\varphi)$).

Задача 8 (стохастическая онлайн оптимизация и метод зеркального спуска / двойственных усреднений; А.С. Немировский и др., Ю.Е. Нестеров).** Рассмотрим задачу стохастической онлайн оптимизации[17]

$$\frac{1}{N}\sum_{k=1}^{N} E_{\xi^k}\left[f_k\left(x;\xi^k\right)\right] \to \min_{x\in S_n(1)},\ S_n(1)=\left\{x\geq 0:\ \sum_{i=1}^{n} x_i = 1\right\}, \qquad (*)$$

при следующих **условиях**:

1. $E_{\xi^k}\left[f_k\left(x;\xi^k\right)\right]$ – выпуклые функции (по x), для этого достаточно выпуклости по x функций $f_k\left(x;\xi^k\right)$;

2. Существует такой вектор $\nabla_x f_k\left(x;\xi^k\right)$, который для компактности будем называть субградиентом, хотя последнее верно не всегда (см. замечание к задаче 10), что

$$E_{\xi^k}\left(\nabla_x f_k\left(x;\xi^k\right) - \nabla_x E_{\xi^k}\left[f_k\left(x;\xi^k\right)\right] \middle| \Xi^{k-1}\right) \equiv 0,$$

где Ξ^{k-1} – σ-алгебра, порожденная случайными величинами $\xi^1,...,\xi^{k-1}$ (при использовании этого условия иногда нужно предполагать $x\in\Xi^{k-1}$, очевидно, что это предположение никак не меняет справедливость этого условия). Далее везде будем использовать обозначения обычного градиента для векторов, которые мы назвали здесь субградиентами. В частности, если мы имеем дело с обычным субградиентом, то запись $\nabla_x f_k\left(x;\xi^k\right)$ в вычислительном контексте (например, в итерационной процедуре МЗС, описанной ниже) означает какой-то его элемент (не важно какой именно), а если в контексте проверки условий (например, в условии 3 ниже), то $\nabla_x f_k\left(x;\xi^k\right)$ пробегает все элементы субградиента (говорят также, субдифференциала);

3. $\left\|\nabla_x f_k\left(x;\xi^k\right)\right\|_\infty \leq M$ – (равномерно, с вероятностью 1) ограниченный субградиент. Для справедливости части утверждений достаточно требовать одно из следующих (более слабых) условий:

$$\text{а) } E_{\xi^k}\left[\left\|\nabla_x f_k\left(x;\xi^k\right)\right\|_\infty^2\right] \leq M^2;\quad \text{б) } E_{\xi^k}\left[\exp\left(\frac{\left\|\nabla_x f_k\left(x;\xi^k\right)\right\|_\infty^2}{M^2}\right)\middle|\Xi^{k-1}\right] \leq \exp(1).$$

---

[17] Запись $E_{\xi^k}\left[f_k\left(x;\xi^k\right)\right]$ означает, что математическое ожидание берется по ξ^k, то есть x и f_k понимаются в такой записи не случайными. Отметим, что ξ^k может зависеть от ξ^1, …, ξ^{k-1}, а распределение ξ^k может зависеть от x (многорукие бандиты). Слово “онлайн” будет пояснено не много позднее. Отметим также, что если дополнительно что-то известно про онлайн постановку, то оценки могут быть улучшены. Например, оценки улучшаются, если дополнительно предполагать, что все функции $f_k(x)=E_{\xi^k}\left[f_k\left(x;\xi^k\right)\right]$ достигают минимума на $S_n(1)$ в одной и той же точке.

Онлайновость постановки задачи допускает, что на каждом шаге k функция f_k может подбираться из рассматриваемого класса функций враждебно по отношению к используемому нами методу генерации последовательности $\left\{x^k\right\}$. В частности, f_k может зависеть от $\left\{x^1,\xi^1;...;x^{k-1},\xi^{k-1};x^k\right\}$, если выбор x^k осуществляется исходя только из информации $\left\{x^1,\xi^1;...;x^{k-1},\xi^{k-1}\right\}$, т.е. без дополнительной рандомизации. Ситуация с дополнительной рандомизацией при выборе x^k рассматривается в следующем пункте.

Для решения задачи (*) воспользуемся адаптивным методом зеркального спуска (точнее двойственных усреднений) в форме. Положим $x_i^1 = 1/n$, $i = 1,...,n$. Пусть $t = 1,..., N-1$.

Алгоритм МЗС1-адаптивный / Метод двойственных усреднений

$$x_i^{t+1} = \frac{\exp\left(-\frac{1}{\beta_{t+1}}\sum_{k=1}^{t}\frac{\partial f_k\left(x^k;\xi^k\right)}{\partial x_i}\right)}{\sum_{l=1}^{n}\exp\left(-\frac{1}{\beta_{t+1}}\sum_{k=1}^{t}\frac{\partial f_k\left(x^k;\xi^k\right)}{\partial x_l}\right)}, \quad i=1,...,n, \quad \beta_t = \frac{M\sqrt{t}}{\sqrt{\ln n}}.$$

Не сложно показать, что этот метод представим также в виде:

$$x^{t+1} = \arg\min_{x\in S_n(1)}\left\{\sum_{k=1}^{t}\left\{f_k\left(x^k\right)+\left\langle\nabla f_k\left(x^k\right),x-x^k\right\rangle\right\}+\beta_{t+1}V\left(x\right)\right\}$$

или

$$\begin{cases} y^k = y^{k-1} - \gamma_k \nabla_x f_k\left(x^k;\xi^k\right) \\ x^{k+1} = \nabla W_{\beta_{k+1}}\left(y^k\right) \end{cases}, \; y^0 = 0, \; \gamma_k \equiv 1, \; \beta_k = \frac{M}{\sqrt{\ln n}}\sqrt{k}, \; k = 1,...,N,$$

где

$$W_\beta\left(y\right) = \sup_{x\in S_n(1)}\left\{\left\langle y,x\right\rangle - \beta V\left(x\right)\right\} = \beta\ln\left(\frac{1}{n}\sum_{i=1}^{n}\exp\left(y_i/\beta\right)\right),$$

$$V\left(x\right) = \ln n + \sum_{i=1}^{n} x_i \ln x_i \, .$$

Покажите справедливость следующих результатов.

Пусть справедливы условия 1, 2, 3.а, тогда

$$\frac{1}{N}\sum_{k=1}^{N}E\left[\, f_k\left(x^k;\xi^k\right)\right] - \min_{x\in S_n(1)}\frac{1}{N}\sum_{k=1}^{N}E_{\xi^k}\left[\, f_k\left(x;\xi^k\right)\right] \le 2M\sqrt{\frac{\ln n}{N}} \, .$$

Если $f_k \equiv f$, а $\left\{\xi^k\right\}$ – независимы и одинаково распределены, как ξ, то

$$E\left[f\left(\frac{1}{N}\sum_{k=1}^{N} x^k;\xi\right)\right] - \min_{x\in S_n(1)} E_\xi\left[f\left(x;\xi\right)\right] \le 2M\sqrt{\frac{\ln n}{N}}\,.$$

Пусть справедливы условия 1, 2, 3, тогда[18] при $\Omega \ge 0$ (выражение в левой части неравенства под вероятностью называют псевдо регретом в онлайн оптимизации)

$$P_{x^1,\ldots,x^N}\left\{\frac{1}{N}\sum_{k=1}^{N} E_{\xi^k}\left[f_k\left(x^k;\xi^k\right)\right] - \min_{x\in S_n(1)} \frac{1}{N}\sum_{k=1}^{N} E_{\xi^k}\left[f_k\left(x;\xi^k\right)\right] \ge \frac{2M}{\sqrt{N}}\left(\sqrt{\ln n}+\sqrt{8\Omega}\right)\right\} \le \exp\left(-\Omega\right).$$

Если $f_k \equiv f$, а $\left\{\xi^k\right\}$ – независимы и одинаково распределены, как ξ, то

$$P_{x^1,\ldots,x^N}\left\{E_\xi\left[f\left(\frac{1}{N}\sum_{k=1}^{N} x^k;\xi\right)\right] - \min_{x\in S_n(1)} E_\xi\left[f\left(x;\xi\right)\right] \ge \frac{2M}{\sqrt{N}}\left(\sqrt{\ln n}+\sqrt{8\Omega}\right)\right\} \le \exp\left(-\Omega\right).$$

Как изменятся последние два неравенства, если вместо условия 3 предполагать более слабое условие 3.б?

Указание. См. статьи *Juditsky A., Lan G., Nemirovski A., Shapiro A.* Stochastic approximation approach to stochastic programming // SIAM Journal on Optimization. 2009. V. 19. № 4. P. 1574–1609, *Nesterov Y.* Primal-dual subgradient methods for convex problems // Math. Program. Ser. B. 2009. V. 120(1). P. 261–283; а также обзоры A. Rakhlin, E. Hazan, S. Bubeck. Возможность распространения обычных методов оптимизации на онлайн постановку связана с их прямо-двойственной структурой. Если метод состоит в подборе последовательности $\left\{x^k\right\}$ так, чтобы зазор (двойственности)[19]

$$\max_{x\in Q}\left\{\sum_{k=1}^{N}\left\langle \nabla_x f\left(x^k;\xi^k\right), x^k - x\right\rangle\right\}$$

---

[18] Запись

$$\text{“ } P_{x^1,\ldots,x^N}\left\{\frac{1}{N}\sum_{k=1}^{N} E_{\xi^k}\left[f_k\left(x^k;\xi^k\right)\right] - \ldots \text{ ”}$$

означает, что под вероятностью мы считаем математическое ожидание по ξ^k, которое, вообще говоря, зависит и от $\xi^1,\ldots,\xi^{k-1}$ (мы не предполагаем независимости $\left\{\xi^k\right\}$), как бы “замораживая” (считая не случайными) x^k, то есть забывая про то, что x^k тоже зависит от $\xi^1,\ldots,\xi^{k-1}$. А вероятность берется как раз по $\left\{x^k\right\}$, с учетом того, что такая зависимость есть (см. определение алгоритма МЗС1).

[19] Получивший свое название, по-видимому, потому, что характеризует (с точностью до фактора N^{-1}) разницу между прямой и двойственной задачей, на решении, полученном после N итераций. При этом предполагается, что двойственная задача может быть построена, исходя из свойств множества Q. Соответствующий зазор двойственности оценивает левую часть последнего неравенства в условии задачи 3, см. также задачу 14.

был мал, то для онлайн постановки все сохраняется с той лишь разницей, что зазор будет иметь вид

$$\max_{x\in Q}\left\{\sum_{k=1}^{N}\left\langle \nabla_x f_k\left(x^k;\xi^k\right), x^k - x\right\rangle\right\}.$$

Далее из оценок зазора и следующего простого наблюдения

$$\sum_{k=1}^{N}\left\{E_{\xi^k}\left[f_k\left(x^k;\xi^k\right)\right]-E_{\xi^k}\left[f_k\left(x;\xi^k\right)\right]\right\}\le\sum_{k=1}^{N}\left\langle \nabla_x E_{\xi^k}\left[f_k\left(x^k;\xi^k\right)\right], x^k - x\right\rangle$$

выводятся требуемые неравенства.

Замечание. Труднее обстоит дело, если мы хотим оценить не псевдо регрет, а регрет (см. задачу 10)

$$\frac{1}{N}\sum_{k=1}^{N} f_k\left(x^k;\xi^k\right)-\min_{x\in S_n(1)}\frac{1}{N}\sum_{k=1}^{N}E_{\xi^k}\left[f_k\left(x;\xi^k\right)\right]$$

или

$$\frac{1}{N}\sum_{k=1}^{N} f_k\left(x^k;\xi^k\right)-\min_{x\in S_n(1)}\frac{1}{N}\sum_{k=1}^{N} f_k\left(x;\xi^k\right).$$

Тем не менее, при дополнительных оговорках и такие выражения можно вероятностно оценивать, см., например, *Lan G., Nemirovski A., Shapiro A.* Validation analysis of mirror descent stochastic approximation method // Mathematical Programming. 2012. V. 134. № 2. P. 425–458 не в онлайн случае.

Задача 9 (зеркальный спуск с рандомизацией при проектировании; Григориадис–Хачиян). Снова рассмотрим постановку задачи стохастической онлайн оптимизации (*) из задачи 8. Но на этот раз будем допускать, что метод генерирования последовательности $\left\{x^k\right\}$ может допускать (внешнюю, дополнительную) рандомизацию. Также как и раньше онлайновость постановки задачи допускает, что на каждом шаге k функция f_k может подбираться из рассматриваемого класса функций враждебно по отношению к используемому нами методу генерации последовательности $\left\{x^k\right\}$. В частности, f_k и ξ^k могут зависеть от $\left\{x^1,\xi^1;...;x^{k-1},\xi^{k-1}\right\}$, и даже от распределения вероятностей p^k (многорукие бандиты), согласно которому осуществляется выбор x^k. Чтобы можно было работать с таким классом задач, нам придется наложить дополнительное **условие**:

4. На каждом шаге генерирование случайной величины x^k согласно распределению вероятностей p^k осуществляется независимо ни от чего.

Положим $p_i^1 = x_i^1 = 1/n$, $i = 1,...,n$. Пусть $t = 1,...,N-1$.

Алгоритм МЗС2-адаптивный / Метод двойственных усреднений

Согласно распределению вероятностей

$$p_i^{t+1}=\frac{\exp\left(-\frac{1}{\beta_{t+1}}\sum_{k=1}^{t}\frac{\partial f_k\left(x^k;\xi^k\right)}{\partial x_i}\right)}{\sum_{l=1}^{n}\exp\left(-\frac{1}{\beta_{t+1}}\sum_{k=1}^{t}\frac{\partial f_k\left(x^k;\xi^k\right)}{\partial x_l}\right)},\quad i=1,\dots,n,\ \beta_t=\frac{M\sqrt{t}}{\sqrt{\ln n}},$$

получаем случайную величину $i(t+1)$, $x_{i(t+1)}^{t+1}=1,\ x_j^{t+1}=0,\ j\neq i(t+1)$.

Алгоритм МЗС2-неадаптивный (заранее известно N)

Согласно распределению вероятностей

$$p_i^{t+1}=\frac{\exp\left(-\frac{1}{\beta_{t+1}}\sum_{k=1}^{t}\gamma_k\frac{\partial f_k\left(x^k;\xi^k\right)}{\partial x_i}\right)}{\sum_{l=1}^{n}\exp\left(-\frac{1}{\beta_{t+1}}\sum_{k=1}^{t}\gamma_k\frac{\partial f_k\left(x^k;\xi^k\right)}{\partial x_l}\right)},\quad i=1,\dots,n,$$

$$\gamma_k\equiv M^{-1}\sqrt{2\ln n/N},\ \beta_t\equiv 1,$$

получаем случайную величину $i(t+1)$, $x_{i(t+1)}^{t+1}=1,\ x_j^{t+1}=0,\ j\neq i(t+1)$.

К сожалению, не делая относительно функций $f_k\left(x;\xi^k\right)$ дополнительно никаких предположений, не удается доказать для МЗС2 аналог утверждения задачи 8. Чтобы можно было сформулировать такой аналог, мы вынуждены будем предполагать, что $f_k\left(x;\xi^k\right)$ – линейные функции по x (можно обобщить и на сублинейные). С одной стороны это существенно сужает класс задач, к которым применим МЗС2. С другой стороны, как будет продемонстрировано в последующих задачах, даже такой узкий класс функций за счет “онлайновости” позволяет применять МЗС2 к довольно широкому кругу задач. Для того чтобы лучше чувствовалась преемственность методов и доказательств их сходимости, далее мы по-прежнему будем использовать общие обозначения $f_k\left(x;\xi^k\right)$, не подчеркивая в формулах линейность. Итак, подобно задачи 8, докажите следующие утверждения.

Пусть справедливы условия 1, 2, 3.а, 4 и $f_k\left(x;\xi^k\right)$ – линейные функции по x, тогда

$$\frac{1}{N}\sum_{k=1}^{N}E\left[f_k\left(x^k;\xi^k\right)\right]-\min_{x\in S_n(1)}\frac{1}{N}\sum_{k=1}^{N}E\left[f_k\left(x;\xi^k\right)\right]\le 2M\sqrt{\frac{\ln n}{N}}.$$

Для неадаптивного метода “2”-у перед M можно занести под знак корня.

Кроме того, если справедливы условия 1, 2, 3, 4, то при $\Omega \geq 0$

$$P_{x^1,\ldots,x^N}\left\{\frac{1}{N}\sum_{k=1}^{N}E_{\xi^k}\left[f_k\left(x^k;\xi^k\right)\right]-\min_{x\in S_n(1)}\frac{1}{N}\sum_{k=1}^{N}E_{\xi^k}\left[f_k\left(x;\xi^k\right)\right]\geq\frac{2M}{\sqrt{N}}\left(\sqrt{\ln n}+\sqrt{18\Omega}\right)\right\}\leq\exp\left(-\Omega\right).$$

Если $f_k\left(x;\xi^k\right)\equiv f_k\left(x\right)$, то это неравенство можно уточнить

$$\frac{2M}{\sqrt{N}}\left(\sqrt{\ln n}+\sqrt{18\Omega}\right)\to\frac{2M}{\sqrt{N}}\left(\sqrt{\ln n}+\sqrt{2\Omega}\right),$$

при этом же условии для неадаптивного метода можно еще больше уточнить

$$\frac{2M}{\sqrt{N}}\left(\sqrt{\ln n}+\sqrt{2\Omega}\right)\to\frac{\sqrt{2}M}{\sqrt{N}}\left(\sqrt{\ln n}+2\sqrt{\Omega}\right).$$

Как изменится неравенство для вероятностей больших уклонений, если вместо условия 3 предполагать более слабое условие 3.б?

Замечание (мотивация). Приведем мотивацию описанного метода (ограничимся детерминированным случаем, $\gamma_k\equiv 1$). Аппроксимируя

$$\min_{x\in S_n(1)}\frac{1}{N}\sum_{k=1}^{t}f_k\left(x\right)\approx\min_{x\in S_n(1)}\frac{1}{N}\sum_{k=1}^{t}\left\{f_k\left(x^k\right)+\left\langle\nabla f_k\left(x^k\right),x-x^k\right\rangle\right\},$$

получим

$$P\left(x_j^{t+1}=1;\,x_i^{t+1}=0,\,i\neq j\right)\overset{def}{=}P_{\varsigma}\left(j=\arg\max_{i=1,\ldots,n}\left\{\sum_{k=1}^{t}\left[-\nabla f_k\left(x^k\right)\right]_i+\varsigma_{t,i}\right\}\right),$$

где $\varsigma_{t,i}$ – независимые одинаково распределенные случайные величины по закону Гумбеля с параметром β_{t+1}, характеризующим среднеквадратичное отклонение $\varsigma_{t,i}$:[20]

$$P\left(\varsigma_{t,i}<\tau\right)=\exp\left\{-e^{-\tau/\beta_{t+1}}\right\}.$$

[20] Поскольку случайные величины $\varsigma_{t,i}$ получаются в результате суммирования t слагаемых (невязок в аппроксимации выпуклой функции линейными минорантами), то можно ожидать, что среднеквадратичное отклонение $\varsigma_{t,i}$ имеет порядок $\sqrt{t}$. Условие одинаковой распределенности $\varsigma_{t,i}$, которое не понятно за счет чего может иметь место, в действительности, не очень здесь и нужно. Достаточно, чтобы $E\left[\varsigma_{t,i}-\varsigma_{t,j}\right]=o\left(\sqrt{t}\right)$ и $\mathrm{Var}\left[\varsigma_{t,i}\right]\sim\sqrt{t}$. Более того, если, с некоторой натяжкой, наряду с независимостью $\left\{\varsigma_{t,i}\right\}_{i=1}^{n}$ также считать, что при формировании $\varsigma_{t,i}$ суммируются независимые слагаемые, то можно ожидать, что $\varsigma_{t,i}$ нормальные случайные величины (центральная предельная теорема). Следовательно, в контексте последующих операций при $n\gg 1$, $\varsigma_{t,i}$ могут быть заменены на случайные величины, распределенные по закону Гумбеля с соответствующими среднеквадратичными отклонениями.

Тогда (см. задачу [ССЫЛКА] раздела [ССЫЛКА])

$$E_{\varsigma}\left[x^{t+1}\right]=\nabla W_{\beta_{t+1}}\left(-\sum_{k=1}^{t}\nabla f_k\left(x^k\right)\right).$$

Распределение Гумбеля возникает именно в таком контексте совсем не случайно, и связано это с тем, что оно max-устойчиво (см. задачу [ССЫЛКА] раздела [ССЫЛКА]): пусть $\varsigma_1,...,\varsigma_n$ – независимые одинаково распределенные случайные величины, и существую такие константы $\alpha,\beta>0$, что

$$\lim_{y\to\infty}e^{y/\beta}\left[1-P\left(\varsigma_k<y\right)\right]=\alpha,$$

тогда (для $\alpha=0$ при незначительных оговорках также будет сходимость к распределению Гумбеля, но формула будет немного другой)

$$\max\left\{\varsigma_1,...,\varsigma_n\right\}-\beta\ln\left(\alpha n\right)\xrightarrow[n\to\infty]{d}\xi\text{, где }P\left(\xi<\tau\right)=\exp\left\{-e^{-\tau/\beta}\right\}.$$

Другими словами, при некоторых довольно общих предположениях (типа Крамера) $\varsigma_{t,i}$ могут быть распределенными произвольно, тем не менее, при $n\gg1$ с хорошей точностью можно считать, что мы в итоге имеем дело с соответствующим распределением Гумбеля. Более того, если задаться вопросом: а какое распределение “наиболее подходит” для $\varsigma_{t,i}$, чтобы в случае “враждебной Природы” (то есть в минимаксном смысле) иметь наилучшие оценки, то ответом будет: симметричное показательное распределение (Лапласа), которое ведет себя в интересном для анализа диапазоне подобно распределению Гумбеля, но в отличие от распределения Лапласа, в случае Гумбеля мы явно можем посчитать интересующие нас вероятности. Как правило, такого рода задачи явно не решаются, и распределение Гумбеля является приятным исключением, для которого есть явные формулы. Приведенная выше мотивация имеет одно интересное приложение в содержательной интерпретации равновесного распределения транспортных потоков (см. задачу [ССЫЛКА] раздела [ССЫЛКА]).

Замечание (рандомизация). Идея рандомизации (искусственного введения случайности), положенная в основу описанных алгоритмов, чрезвычайно продуктивна: против нас играет, возможно, враждебная “Природа”, которая, зная историю игры, и наши текущие намерения старается нам “предложить вариант похуже”. С этим можно “бороться” за счет случайного независимого осуществления своих намерений на каждом шаге, с реализацией неизвестной “Природе”. За счет этой случайности мы переходим от анализа по худшему случаю (роль которого в онлайн оптимизации играет враждебная

“Природа”) к анализу “в среднем”. Такая рандомизация дает возможность получать оценки, которые в детерминированном случае получить невозможно. Причем, если в онлайн постановке такая рандомизация прописывается в “правилах игры”, то применительно к задачам обычной оптимизации все это возникает совершенно естественным образом, как желание с большой вероятностью обезопасить себя от “самых худших случаев”[21] детерминированной версии метода. Описанный МЗС2 естественно также понимать как покомпонентный метод субградиентного спуска со случайным выбором компоненты. Как будет отмечено в задаче 14, такой метод не только оптимален с точки зрения числа итераций, но и в некотором смысле с точки зрения затрат на выполнение одной итерации. Отметим в связи с этим замечанием курс лекций *Rakhlin A., Sridharan K.* Statistical Learning Theory and Sequential Prediction. STAT928, 2014 и диссертацию *Sridharan K.* Learning From An Optimization Viewpoint, 2014.

Задача 10 (взвешенные многорукие бандиты)*. Имеется n различных ручек. Игра повторяется $N \gg 1$ раз (это число может быть заранее неизвестно). С каждой ручкой i на шаге k связаны случайные потери $r_i^k \in [0, M]$, зависящие от номера шага, номера ручки и от того, какой стратегии мы придерживались до шага k. На каждом шаге k мы должны выбрать вектор $x^k \in S_n(1)$. Наши потери на шаге k будут $\langle x^k, r^k \rangle$. Все, чем мы располагаем на шаге k при выборе x^k, это набор

$$\left(\left(x^1, \langle x^1, r^1 \rangle\right); \ldots; \left(x^{k-1}, \langle x^{k-1}, r^{k-1} \rangle\right)\right).$$

Целью является таким образом организовать процедуру выбора x^k, чтобы ожидаемые суммарные потери (псевдо регрет) были бы минимальны. Предложите эффективный способ выбора $x^k \in S_n(1)$ и оцените (в смысле вероятностей больших уклонений) получающийся при таком способе псевдо регрет.

Указание. Воспользуйтесь задачами 6, 8.

Замечание (классические многорукие бандиты). Приведем для сравнения классическую постановку задачи о многоруких бандитах, в которой накладываются еще более жесткие ограничения на доступную информацию и на выбор стратегии. Имеется n различных ручек. Игра повторяется $N \gg 1$ раз (это число может быть заранее неизвестно). На каждом шаге k мы должны выбрать ручку $i(k)$, которую “дергаем”. Дергание ручки приносит нам некоторые, вообще говоря, случайные потери $r_{i(k)}^k$ (считаем, для определенности, что всегда $r_{i(k)}^k \in [0, M]$), зависящие от номера шага, номера ручки и от того, какой стратегии мы придерживались до шага k включительно. Наша стратегия на

[21] Речь идет о, так называемых, массовых задачах, то есть, исследуя тот или иной метод, мы точно не знаем какой конкретно объект поступит на вход, поэтому, чтобы гарантировано что-то иметь, мы исходим из худшего (наименее благоприятного для данного метода) случая входных данных.

шаге k описывается вектором распределения вероятностей $x^k \in S_n(1)$, согласно которому мы независимо ни от чего выбираем ручку, которую будем дергать. Все, чем мы располагаем на шаге k, это вектором

$$\left(\left(x^1, i(1), r^1_{i(1)}\right);...;\left(x^{k-1}, i(k-1), r^{k-1}_{i(k-1)}\right)\right).$$

Мы считаем, что потери на k-м шаге r^k зависят от x^k, но не результата разыгрывания, от $\left(x^1,...,x^{k-1}\right)$ и результатов разыгрываний, а также от $\left(r^1,...,r^{k-1}\right)$. Целью является таким образом организовать процедуру дергания ручек, чтобы ожидаемые суммарные потери были бы минимальны. Введем функцию (r^k и результат разыгрывания, согласно распределению вероятностей, заданному вектором x, – независимы; обе эти "случайности" мы обозначаем ξ^k)

$$f_k\left(x;\xi^k\right) = r_i^k \text{ с вероятностью } x_i,\ i = 1,...,n,$$

и её обобщенный (в смысле удовлетворения условию 2 задачи 8) стохастический градиент

$$\nabla_x f_k\left(x;\xi^k\right) = (\underbrace{0,...,r_i^k}_{i} / x_i,...,0)^T \text{ с вероятностью } x_i,\ i = 1,...,n.$$

Тогда выполнены условия 1 и 2 (задачи 8). Однако имеется проблема: константа M в условии 3 (задачи 8) получается слишком большой (например, в 3.а $M := M \sup\limits_{x \in S_n(1)} \sqrt{\sum\limits_{i=1}^n x_i^{-1}} = \infty$), то есть утверждение задачи 8 ничего дать не может в том виде, в котором она была приведена. Возникает желание "что-то подкрутить", чтобы можно было им воспользоваться. Это можно сделать. В обзоре *Bubeck S., Cesa-Bianchi N.* Regret analysis of stochastic and nonstochastic multi-armed bandit problems // Foundation and Trends in Machine Learning. 2012. V. 5. no. 1. P. 1–122 описана общая схема, обобщающая изложенную нами схему МЗС1, позволяющая получить следующие оценки псевдо регрета[22]

$$\mathrm{O}\left(M\sqrt{\frac{n\ln n}{N}}\right) \text{ – в среднем; } \mathrm{O}\left(M\sqrt{\frac{n\ln(n/\sigma)}{N}}\right) \text{ – с вероятностью } \geq 1-\sigma.$$

[22] Стоит обратить внимание, что если использовать более специальную прокс-структуру (Audibert–Bubeck), то для псевдо регрета можно получить оценки без логарифмического фактора $\ln n$ под корнем, что уже соответствует нижним оценкам. В частности, это обстоятельство означает, что KL прокс структура для симплекса не всегда оптимальна (но близка к таковой). Тем не менее, в последующих нескольких задачах мы убедимся, что для ряда других постановок, оценки, полученные с помощью KL прокс структуры, – оптимальны. Кроме того, есть еще плата за избавление от фактора $\ln n$ под корнем – удорожание процедуры вычисления проекции на симплекс в смысле этой прокс-структуры. Другими словами, это ускорение оправдано только для онлайн постановок, в которых, как правило, стремятся минимизировать (псевдо) регрет, не сильно учитывая общую вычислительную трудоемкость.

В основе подхода лежит идея использования в доказательстве специальной локальной нормы (зависящей от x^k и свойств функции W_{β_k}) при оценке субградиента

$$\left\|\nabla_x f_k\left(x^k;\xi^k\right)\right\|_{loc}^2 \le 2Mx_j^k\left(1-x_j^k\right)\left(r_j^k/x_j^k\right)^2.$$

Труднее обстоит дело с оценкой регрета (в среднем и вероятностей больших уклонений). Пример из шестой лекции *Mansour Y.* Algorithmic Game Theory and Machine Learning, 2011 http://www.tau.ac.il/~mansour/advanced-agt+ml/ показывает ($n=2$), что МЗС1 может давать регрет $\sim cN^{-1/4}$, что значительно хуже оценки псевдо регрета $\sim cN^{-1/2}$. По сути, речь идет о том, что написано в замечании к задаче 8. Здесь уже требуется некая игра "bias–variance trade off": отказаться от несмещенности оценки градиента для уменьшения дисперсии этой оценки. Этот популярный трюк в математической статистике и машинном обучении позволяет с некоторыми оговорками распространить приведенные выше оценки псевдо регрета $O\left(M\sqrt{n\ln n/N}\right)$ и на случай оценок регрета. Кое-что на эту тему применительно к многоруким бандитам можно найти в упомянутом обзоре Bubeck–Cesa-Bianchi.

Интересно заметить, что на результаты, описанные выше, можно было бы посмотреть и с помощью МЗС2 из задачи 9. Для этого нужно взять

$$f_k\left(x;\xi^k\right)=\left\langle r^k,x\right\rangle$$

и её обобщенный (в смысле удовлетворения условию 2 из задачи 8) стохастический градиент

$$\nabla_x f_k\left(x;\xi^k\right)=(\underbrace{0,...,r_i^k}_{i}/\,p_i,...,0)^T \text{ с вероятностью } p_i,\ i=1,...,n,$$

где $x=(\underbrace{0,...,1}_{i},...,0)^T$ с вероятностью p_i, $i=1,...,n$.

Задача 11 (взвешивание экспертных решений, линейные потери). Рассмотрим задачу взвешивание экспертных решений. Имеется n различных Экспертов. Каждый Эксперт играет на рынке. Игра повторяется $N\gg 1$ раз (это число может быть заранее неизвестно). Пусть l_i^k – проигрыш Эксперта i на шаге k ($\left|l_i^k\right|\le M$). На каждом шаге k мы распределяем один доллар между Экспертами, согласно вектору $x^k\in S_n(1)$. Потери, которые мы при этом несем, рассчитываются по потерям экспертов $\left\langle l^k,x^k\right\rangle$. Целью является таким образом организовать процедуру распределения доллара на каждом шаге, чтобы наши суммарные потери были бы минимальны. Допускается, что потери экспертов l^k могут зависеть еще и от текущего хода x^k. Проверьте, что к данной постановке применимо утверждение из задачи 8 в детерминированном варианте с функциями

$$f_k\left(x;\xi^k\right)\equiv f_k\left(x\right)=\left\langle l^k,x\right\rangle.$$

Замечание. Отметим, что получаемая при этом оценка регрета

$$\mathrm{O}\left(M\sqrt{\frac{\ln n}{N}}\right)$$

– оптимальна для данного класса задач. Подробнее об этом можно прочитать в монографии *Lugosi G., Cesa-Bianchi N*. Prediction, learning and games. New York: Cambridge University Press, 2006. Эту же монографию можно рекомендовать и к следующим двум задачам.

Задача 12 (взвешивание экспертных решений, выпуклые потери). В условиях предыдущей задачи предположим, что на k-м шаге i-й эксперт использует стратегию $\zeta_i^k \in \Delta$ (множество Δ – выпуклое), дающую потери $\lambda\left(\omega^k, \zeta_i^k\right)$, где ω^k – "ход", возможно, враждебной "Природы", знающей, в том числе, и нашу текущую стратегию. Функция $\lambda(\cdot)$ – выпуклая по второму аргументу и $\left|\lambda(\cdot)\right| \le M$. На каждом шаге мы должны выбирать свою стратегию

$$x \overset{def}{=} \sum_{i=1}^{n} x_i \cdot \zeta_i^k \in \Delta,$$

дающую потери $\lambda\left(\omega^k, x\right)$ так, чтобы наши суммарные потери были минимальны. Покажите, что для данной постановки также применимо утверждение из задачи 8 в детерминированном варианте с

$$f_k\left(x;\xi^k\right) \equiv f_k\left(x\right) = \sum_{i=1}^{n} x_i \lambda\left(\omega^k, \zeta_i^k\right) \ge \lambda\left(\omega^k, x\right).$$

Указание. Полезно заметить, что функция $\lambda\left(\omega^k, \zeta\right)$ – выпуклая по ζ для любого ω^k, поэтому

$$\sum_{k=1}^{N} \lambda\left(\omega^k, x^k\right) - \min_{i=1,\ldots,n} \sum_{k=1}^{N} \lambda\left(\omega^k, \zeta_i^k\right) \le \sum_{k=1}^{N} f_k\left(x^k\right) - \min_{x \in S_n(1)} \sum_{k=1}^{N} f_k\left(x\right).$$

Замечание. Отметим, что получаемая при этом оценка регрета

$$\mathrm{O}\left(M\sqrt{\frac{\ln n}{N}}\right)$$

также оптимальна для данного класса задач.

Полезно, на наш взгляд, будет здесь привести другой способ (более типичный для данного класса приложений) получения аналогичного результата, не связанный на прямую с МЗС схемой вывода, но фактически, приводящий к точно такому же алгоритму.

Этот способ также весьма популярен в статистической теории обучения (см. G. Lugosi и др.).[23]

Введем обозначение $L_i^N = \sum_{k=1}^{N} \lambda\left(\omega^k, \zeta_i^k\right)$, $\tilde{L}^N = \sum_{k=1}^{N} \lambda\left(\omega^k, x^k\right)$, по определению считаем $L_i^0 \equiv 0$. Рассмотрим

$$W_\beta\left(\left\{-L_i^N\right\}_{i=1}^n\right) = \beta \ln\left(\frac{1}{n}\sum_{i=1}^{n} \exp\left(-L_i^N/\beta\right)\right) \geq -\min_{i=1,\ldots,n} L_i^N - \beta \ln n .$$

С другой стороны, вводя дискретную случайную величину (с.в.) z^k, имеющую (независящее ни от чего) распределение x^k (рассчитанное также как и раньше исходя из МЗС1, примененного к набору функций $\left\{f_k(x)\right\}_{k=1}^N$, определенных выше), можно заметить, что

$$W_\beta\left(\left\{-L_i^N\right\}_{i=1}^n\right) = \sum_{k=1}^{N}\left(W_\beta\left(\left\{-L_i^k\right\}_{i=1}^n\right) - W_\beta\left(\left\{-L_i^{k-1}\right\}_{i=1}^n\right)\right) = \beta\sum_{k=1}^{N} \ln\left(E_z\left(e^{-\lambda\left(\omega^k, z^k\right)/\beta}\right)\right).$$

Используя далее неравенство Хеффдинга (для с.в. $X \in [-M, M]$), см. задачу [ССЫЛКА] раздела [ССЫЛКА]

$$\ln\left(E_X\left(e^{sX}\right)\right) \leq sE_X(X) + s^2\frac{M^2}{2},$$

Получим

$$W_\beta\left(\left\{-L_i^N\right\}_{i=1}^n\right) \leq -\tilde{L}^N + (2\beta)^{-1} M^2 N .$$

Таким образом,

$$\tilde{L}^N \leq \min_{i=1,\ldots,n} L_i^N + \beta \ln n + (2\beta)^{-1} M^2 N .$$

Минимизация правой части по $\beta > 0$ приводит нас к уже известному ответу. Аналогичные, но чуть более тонкие рассуждения, позволяют избавиться от зависимости β от N, то есть сделать алгоритм адаптивным.

Задача 13 (взвешивание экспертных решений, невыпуклые потери). Предположим, что в условиях предыдущей задачи мы не можем гарантировать

[23] Максимум из независимых случайных величин, который сложно исследовать, заменяется (с хорошей точностью, контролируемой малостью параметра β) логарифмом от суммы экспонент от этих независимых случайных величин. А сумму независимых случайных величин (их экспонент) исследовать уже на много проще. В оптимизации эту процедуру называют сглаживанием (Ю.Е. Нестеров).

выпуклость $\lambda(\cdot)$ – по второму аргументу. Тогда мы выбираем стратегию – распределение вероятностей на множестве стратегий Экспертов, и разыгрываем случайную величину согласно этому распределению вероятностей. Другими словами мы просто пользуемся МЗС2 с $f_k(x;\xi^k) \equiv f_k(x) = \sum_{i=1}^{n} x_i \lambda(\omega^k, \zeta_i^k)$, применимость которого обосновывается утверждением из задачи 9. Получите оценки регрета

$$\mathrm{O}\left(M\sqrt{\frac{\ln n}{N}}\right) \text{ – в среднем; } \mathrm{O}\left(M\sqrt{\frac{\ln(n/\sigma)}{N}}\right) \text{ – с вероятностью } \geq 1-\sigma,$$

Замечание. Отметим, что эти оценки оптимальны для данного класса задач. Ключевая разница в задачах 10 и 13, "сто́ящая" $\sim\sqrt{n}$ в оценке M для многоруких бандитов (задача 10), заключается в том, что в многоруких бандитах мы имеем только свою историю дергания ручек (нам не известно, какие бы потери нам принесли другие ручки, кабы мы их выбрали), а в постановке взвешивания экспертных решений это все известно, и называется потерями экспертов.

Как будет видно из следующего примера описанный только что подход вполне успешно работает (дает не улучшаемые результаты) и в случае выпуклой по второму аргументу функции $\lambda(\cdot)$.

Задача 14 (антагонистические матричные игры; Григориадис–Хачиян). Пусть есть два игрока А и Б. Задана матрица игры $A = \|a_{ij}\|$, где $|a_{ij}| \leq M$, a_{ij} – выигрыш игрока А (проигрыш игрока Б) в случае когда игрок А выбрал стратегию i, а игрок Б стратегию j. Отождествим себя с игроком Б. И предположим, что игра повторяется $N \gg 1$ раз (это число может быть заранее неизвестно). Мы находимся в условиях предыдущей задачи с $\lambda(\omega^k, \zeta_j^k) = \sum_{i=1}^{n} \omega_i^k a_{ij}$, то есть

$$f_k(x) = \langle \omega^k, Ax \rangle, \; x \in S_n(1),$$

где ω^k – вектор[24] со всеми компонентами равными 0, кроме одной компоненты, соответствующей ходу А на шаге k, равной 1. Хотя функция $f_k(x)$ определена на единичном симплексе, по "правилам игры" вектор x^k имеет ровно одну единичную компоненту, соответствующую ходу Б на шаге k, остальные компоненты равны нулю. Обозначим цену игры

[24] Вообще говоря, зависящий от всей истории игры до текущего момента включительно, в частности, как-то зависящий и от текущей стратегии (не хода) игрока Б, заданной распределением вероятностей (результат текущего разыгрывания (ход Б) игроку А не известен).

$$C = \max_{\omega\in S_n(1)} \min_{x\in S_n(1)} \langle \omega, Ax\rangle = \min_{x\in S_n(1)} \max_{\omega\in S_n(1)} \langle \omega, Ax\rangle .$$ (теорема фон Неймана о минимаксе[25])

Пару векторов (ω, x), доставляющих решение этой минимаксной задачи, назовем равновесием Нэша. По определению (это неравенство восходит к Ханнану)

$$\min_{x\in S_n(1)} \frac{1}{N}\sum_{k=1}^{N} f_k(x) \le C .$$

Покажите, что если мы (игрок Б) будем придерживаться рандомизированной стратегии МЗС2, то из утверждения задачи 9 следует: с вероятностью $\ge 1-\sigma$ (в случае когда N заранее известно оценку можно уточнить)

$$\frac{1}{N}\sum_{k=1}^{N} f_k(x^k) - \min_{x\in S_n(1)} \frac{1}{N}\sum_{k=1}^{N} f_k(x) \le \frac{2M}{\sqrt{N}}\left(\sqrt{\ln n} + \sqrt{2\ln\left(\sigma^{-1}\right)}\right),$$

т.е. с вероятностью $\ge 1-\sigma$ наши потери ограничены

$$\frac{1}{N}\sum_{k=1}^{N} f_k(x^k) \le C + \frac{2M}{\sqrt{N}}\left(\sqrt{\ln n} + \sqrt{2\ln\left(\sigma^{-1}\right)}\right).$$

Самый плохой для нас случай (с точки зрения такой оценки) – это когда игрок А тоже "знает" утверждение задачи 9, и действует согласно МЗС2 (точнее версии МЗС2 для максимизации вогнутых функций на симплексе).

Покажите, что если и А и Б будут придерживаться МЗС2, то они сойдутся к равновесию Нэша, причем чрезвычайно быстро:

$$\frac{8M\left(\ln n + 2\ln\left(\sigma^{-1}\right)\right)}{\varepsilon^2} \text{ – итераций;}$$

$$\mathrm{O}\left(n + M\,\frac{s\ln n\left(\ln n + \ln\left(\sigma^{-1}\right)\right)}{\varepsilon^2}\right) \text{ – общее число арифметических операций,}$$

где $s \le n$ – среднее число элементов в строках и столбцах матрицы A.

Замечание. Отсюда видно, что если ε (зазор двойственности) – не очень малое, то может случиться, что общее число арифметических операций будет много меньше числа элементов матрицы A отличных от нуля, в то время как любой детерминированный способ поиска равновесия Нэша потребовал бы прочтения как минимум половины элементов матрицы A. Есть основания полагать, что описанный здесь метод оптимален не только с точки зрения числа итераций, но и с точки зрения "стоимости" шага. Особенно

[25] Отметим, что с помощью онлайн оптимизации и экспоненциального взвешивания можно похожим образом проинтерпретировать и вариант теоремы о минимаксе для векторнозначной функции выигрыша – теорему Блэкуэлла о достижимости (см., например, лекции Rakhlin–Sridharan, В.В. Вьюгина или монографию Lugosi–Cesa-Bianchi), которая используется, например, при построении калибруемых предсказаний.

ярко это проявляется, когда $s \ll n$. Нам кажется, что описанная в этом примере методология может оказаться полезной в huge-scale optimization (не только для рассмотренной здесь задачи). В частности, при изучении того, как оптимальным образом можно учитывать разреженность задачи и как оптимально организовывать покомпонентный спуск. Отметим в связи с вышесказанным другой пример задачи выпуклой оптимизации, когда удается получить ответ (с требуемой точностью), с большим запасом не просматривая весь объем имеющихся данных:[26] метод наименьших квадратов с разреженной структурой (см. S. Bubeck, 2014). Нам представляется, что именно эта ветвь более общего и бурно развивающегося в последнее время направления huge-scale оптимизации является сейчас одной из наиболее интересных, как с практической, так и с теоретической точки зрения, и основные результаты здесь еще впереди. Уточним, что речь идет о задачах, приходящих из: машинного обучения (в частности, распознавания изображений), моделирования различных сетей огромных размеров типа сети Интернет или транспортных сетей, биоинформатики, численных методов (проектирование конструкций методом конечных элементов) и ряда других приложений. Все эти задачи отличают колоссальные размеры. Скажем, для задачи ранжирования web-страниц необходимо решать вспомогательную оптимизационную задачу в пространстве, размерность которого больше миллиарда (см., например, Назин–Поляк). Но помимо размеров их отличает некоторые релаксированные требования к решению. Например, нет никакой необходимости ранжировать абсолютно все web-страницы по заданному запросу и делать это очень точно. Достаточно, чтобы качественно выдавалась только первые сто наиболее значимых (больших) компонент ранжирующего вектора, причем важны не столько сами значения этих компонент, сколько их порядок. Также эти задачи довольно специальные, то есть использовать концепцию черного ящика для оценки числа требуемых итераций, как правило, не представляется возможным. Более того, оценки имеются, в основном, только на число шагов (итераций). Поскольку общее время работы алгоритма определяется произведением числа итераций на стоимость одной итерации, то возникает игра между числом итераций и стоимостью итерации. Описанный в этой задаче метод как раз играет в эту игру. А именно, он увеличивает за счет рандомизации число итераций в $\ln\left(\sigma^{-1}\right)$ раз, при этом итерация становится в $\sim n/\ln n$ раз дешевле. Наконец, важной составляющей многих задач является разреженная структура. Тут имеется два варианта: разреженная структура решения и данных. В первом случае (частично уже затронутым выше на примере ранжирования web-страниц) часто речь идет о подмене исходной задачи, вычислительно более привлекательной задачей, по решению которой можно получить приближенное решение исходной. Пожалуй, наиболее ярким примером здесь является сжатие измерений (см. задачу [ССЫЛКА] раздела [ССЫЛКА]). В случае разреженных данных представляется перспективным использование специальных покомпонентных спусков и исследование вычислительных особенностей пересчета различных классов функций многих переменных в случае изменения лишь не большого числа их аргументов (см., например, *Nesterov Y.E.* Subgradient methods for huge-scale optimization problems // Math. Program., Ser. A. 2014). В заключение отметим, что важными составляющими анализа эффективности методов (в виду специфики описываемых задач) является исследование возможности распараллеливания (в рассматриваемом нами задаче это возможно сделать) и вероятностный анализ в среднем (или для почти всех входов). В отличие от Computer Science, в численных методах выпуклой оптимизации такой анализ можно встретить не часто (все же кое-что есть, например, вероятностный анализ симплекс-метода *Spielman D.A., Teng S.-H.* Smoothed analysis of algorithms: Why the simplex algorithm usually takes polynomial time // Journal of the ACM. 2004. V. 51 P. 385–463). Хотя уже сейчас (в связи с бурным развитием идей

[26] Общая проблема здесь: как понять, что просматривать, а что нет.

концентрации меры) становится все более и более ясно, что в пространствах огромных размеров такой анализ необходим, и может многое дать *Tao T.* Структура и случайность. М.: МЦНМО, 2013.

Задача 15 (Нестеров–Юдицкий). Покажите, что число обусловленности (отношение константы Липшица градиента L_1 к константе сильной выпуклости μ_1) квадратичных функций $f(x)=\frac{1}{2}x^TAx-b^Tx$, $x\in\mathbb{R}^n$, посчитанное в 1-норме не может быть меньше n.

Указание. Пусть $\xi=(\xi_1,....,\xi_n)$, где ξ_k – i.i.d., $P(\xi_k=1/n)=P(\xi_k=-1/n)=1/2$. Тогда, учитывая, что $\|\xi\|_1\equiv 1$,

$$\mu_1(f)\le E\left(\xi^TA\xi\right)=\frac{1}{n^2}\operatorname{tr}(A)\le\frac{1}{n}\max_{i,j=1,..,n}|A_{ij}|=\frac{1}{n}\|A\|_{1,\infty}=\frac{1}{n}L_1(f).$$

Задача 16 (задача об обнаружении сигнала на фоне не случайных помех; Фишер–Граничин–Поляк). Имеется известный (наблюдаемый) центрированный случайный процесс φ_n (с конечными моментами до четвертого порядка включительно). Мы знаем, что реализация $\varphi_1\neq 0$. Имеется неизвестное значение некоторого параметра $\theta\in[0,1]$. Мы наблюдаем значения случайного процесса $y_n=\varphi_n\theta+\varepsilon_n$, где $|\varepsilon_n|\le C$, но "природа" ε_n нам неизвестна. Известно лишь, что $\{\varepsilon_n\}$ и $\{\varphi_n\}$ – независимы. Покажите, что оценка

$$\hat{\theta}_n=\frac{\sum_{k=1}^{n}\varphi_k y_k}{\sum_{k=1}^{n}\varphi_k^2}$$

состоятельна п.н., т.е. $\hat{\theta}_n\xrightarrow[n\to\infty]{п.н.}\theta$.

Указание. Приведите $y_n=\varphi_n\theta+\varepsilon_n$, $n=1,2,....$ к виду

$$\frac{1}{n}\sum_{k=1}^{n}\varphi_k y_k=\frac{1}{n}\sum_{k=1}^{n}\varphi_k^2\theta+\frac{1}{n}\sum_{k=1}^{n}\varphi_k\varepsilon_k,\ n=1,2,.....$$

Эта задача является пожалуй одной из самых простых по теме оценивания неизвестных параметров при почти произвольных помехах. Идея (восходящая к Р. Фишеру) введения дополнительной рандомизации, которая бы устраняла за счет усреднения независимые от этой рандомизации помехи не случайной природы, весьма плодотворна во многих областях, например, в безградиентной оптимизации (см. задачу 6). Подробнее об этом

написано в монографии *Граничин О.Н., Поялк Б.Т.* Рандомизированные алгоритмы оценивания и оптимизации при почти произвольных помехах. М.: Наука, 2003.

Задача 17 (distributed optimization in networked systems; A. Nedic)*. Имеется сильно связанная вычислительная сеть (ориентированный граф, из любой вершины которого существует путь в любую другую вершину). В узлах сети (вершинах графа) "находятся" гладкие сильно выпуклые функции $f_k(x)$. Система живет в дискретном времени. Каждый узел может вычислять градиент (и его проекцию на выпуклое компактное множество Q) только своей функции, и посылать информацию в соседние узлы по сети. Задачей всех узлов является так организовать процесс своих вычислений и обмена информацией, чтобы система сошлась к консенсусу x^*, который определяется как единственное решение задачи $\sum_{k=1}^{m} f_k(x) \to \min_{x \in Q}$. Предложите способ решения этой задачи.

Указание. Пусть ориентированному графу соответствует дважды стохастическая матрица A ($A_{ij} > 0$ тогда и только тогда, когда из узла i идет ребро в узел j). В каждом узле i на шаге k хранятся два вектора $x_i(k)$ и $w_i(k)$, которые обновляются по правилам

$$w_i(k+1) = \sum_j A_{ij} x_j(k), \text{ (consensus-like step)}$$

$$x_i(k+1) = \Pi_Q\left[w_i(k+1) - h_k \nabla f_i\left(w_i(k+1)\right)\right], \text{ (local gradient-base step)}$$

где Π_Q – проектирование на множество Q. Покажите, что если $\sum_{k=1}^{\infty} h_k = \infty$, $\sum_{k=1}^{\infty} h_k^2 < \infty$, то какая бы не была точка старта (из Q), $\lim_{k\to\infty} x_i(k) = x^*$ – для всех узлов i.

Замечание. Сравнивая приведенное выше указание и замечание к задаче 1, естественно, возникает желание в таких же категориях (с точными оценками скорости сходимости, игрой на гладкости и сильной выпуклости, прокс-структуре задачи, задаваемой свойствами множества Q, учетом стохастических постановок, в том числе, рандомизации и т.п.) попытаться классифицировать всевозможные варианты и в данной задаче. К сожалению, насколько нам известно, это пока еще не сделано (причем с хорошим "запасом"). И сложность здесь в том, что в теоретических оценках скорости сходимости (а именно от них стараются "плясать") распределенных алгоритмов оптимизации требуется учитывать также время достижения консенсуса, что является в некотором смысле двойственной задачей к задаче оценки mixing time (характерного времени выхода на стационарное распределение) соответствующей марковской цепи. И хотя на сегодняшний день разобрано много случаев, когда такие оценки можно получать (см., например, *Levin D.A., Peres Y., Wilmer E.L.* Markov chain and mixing times. AMS, 2009. http://pages.uoregon.edu/dlevin/MARKOV/markovmixing.pdf, а также задачу 19), все-таки каких-то простых (компактных) и универсальных способов получения точных оценок нет. Тем не менее, если мы быстро приходим к консенсусу (хорошее перемешивание, быстрый

обмен информацией), то скорость сходимости в основном определяется уже чисто оптимизационной составляющей, и тогда уже замечание к задаче 1 может “заиграть”. Имеются существенно более точные результаты, чем приведенные в этой задаче (см., например, работы A. Nedic, P. Richtarik), причем в последнее время все чаще рассматриваются и более общие схемы, в которых вычислительная сеть и матрица A меняются со временем.

Задача 18 (глобальная оптимизация и монотонный симметричный марковский поиск; Некруткин–Тихомиров).** Рассматривается задача глобальной оптимизации $f(x) \to \min\limits_{x \in \mathbb{R}^n}$. Считаем, что глобальный минимум достигается в единственной точке x^* (причем для любых $\varepsilon > 0$ выполняется условие[27] $\inf\left\{ f(x) \colon x \in B_\varepsilon^c(x_*) \right\} > f(x_*)$), $f(x)$ – непрерывная функция, дважды гладка в точке x^*, причем матрица Гессе G функции $f(x)$ в этой точке положительно определена. Опишем алгоритм (с точностью до выбора функции плотности распределения $g(r)$, $r \in [0, \infty)$).

1. **(начальный шаг)** Выбираем точку старта $x_0 = x$;
2. **(шаг k)** Независимо генерируем с.в. ξ_k из центрально симметричного распределения с заданной плотностью $g(r)$. Если
   - $f(x_k + \xi_k) \le f(x_k)$, то $x_{k+1} = x_k + \xi_k$,
   - иначе $x_{k+1} = x_k$;
3. Если $k < N$, то полагаем $k := k+1$ go to 2; иначе останавливаем алгоритм, выдаем $x_k = x_N$.

Введем обозначения

$$M_r = \inf\left\{ x \in B_r(x_*) \colon \ f(x) < f(y) \text{ для всех } y \in B_r^c(x_*) \right\},$$

$$\tau_\varepsilon = \min\left\{ n \in \mathbb{N} \colon \ x_n \in M_\varepsilon \right\}, \ \delta(x) = \inf\left\{ r \ge 0 \colon \ x \in M_r \right\}, \ \Gamma = \prod_{i=1}^{n} \left(\frac{\lambda_i}{\lambda_{\min}} \right)^{1/2},$$

где λ_i – собственные числа G. Покажите, что, как бы мы не выбирали функцию плотности $g(r)$, всегда при $\varepsilon < \rho(x, x^*)$ имеет место следующая оценка снизу:

$$E\left[\tau_\varepsilon \middle| x_0 = x \right] \ge \ln\left(\rho(x, x^*) / \varepsilon \right) + 2.$$

[27] $B_\varepsilon^c(x_*)$ – дополнение шара $B_\varepsilon(x_*)$ радиуса ε с центром в точке x_*.

Покажите, что метод с плотностью (есть много других вариантов)

$$g(r)=\nu(r)r^{-d},\ \nu(r)=\frac{c}{\left(e+n\left|\ln r\right|\right)\ln^2\left(e+n\left|\ln r\right|\right)},$$

где c – находится из условия нормировки, дает оценку ($\varepsilon<\delta(x)/2$)

$$E\left[\tau_\varepsilon\middle|x_0=x\right]\le b^n\Gamma\ln^2(\varepsilon)\ln^2\left(\ln(\varepsilon)\right)\left|\ln\left(\delta(x)\right)\right|, \qquad (*)$$

где $b\in(2,3)$ (для простоты восприятия мы привели здесь огрубленный вариант).

Замечание. Если отказаться от гладкости и(или) положительной определенности матрицы G, то вместо Γ, которое в типичных ситуациях растет с размерностью пространства экспоненциально быстро, в (*) можно использовать $F_{\varepsilon,x}^{-1}$, где

$$F_{\varepsilon,x}=\inf_{r:\,\varepsilon\le r<\delta(x)}\left\{\mathrm{vol}(M_r)\Big/\mathrm{vol}\left(B_r\left(x^*\right)\right)\right\}.$$

Отметим, что все приведенные результаты сохраняются с небольшими поправками и для оценок вероятностей больших уклонений, т.е. для

$$n(x,\varepsilon,\gamma)=\min\left\{n:\ P\left(x_n\in M_\varepsilon\middle|x_0=x\right)\ge\gamma\right\}=\min\left\{n:\ P\left(\tau_\varepsilon\le n\middle|x_0=x\right)\ge\gamma\right\}.$$

Детали имеются в работах А.С. Тихомирова, опубликованных за последние 20 лет в ЖВМ и МФ.

Изложенные в этой задаче результаты могут вызвать на первых порах удивление. И, действительно, как такое возможно, чтобы в задачах глобальной оптимизации зависимость числа итераций от точности была логарифмическая, в то время как известны нижние оценки, в которых эта точность входит в степени размерности пространства (в случае равномерной гладкости высокого порядка, степень можно понижать) в знаменателе, см., например, *Zhigljavsky A., Zilinskas A.* Stochastic global optimization. Springer Optimization and Its Applications, 2008? Тут стоит отметить, что, во-первых, нижние оценки получаются для детерминированных методов, ну и самое главное, что “проклятие размерности” здесь также никуда не делось. Даже при самом благоприятном раскладе, в оценку (*) входит фактор 2^n, экспоненциально растущий с ростом размерности пространства. В отличие от глобальной оптимизации, в выпуклой оптимизации такие проблемы можно решать, что было продемонстрировано в задачах, приведенных выше.

Задача 19 (Markov Chain Monte Carlo Revolution и состоятельность оценок максимального правдоподобия; P. Diaconis)*. В руки опытных криптографов попалось закодированное письмо (10 000 символов). Чтобы это письмо прочитать нужно его декодировать. Для этого берется стохастическая матрица переходных вероятностей

$P = \left\| p_{ij} \right\|$ (линейный размер которой определяется числом возможных символов (букв, знаков препинания и т.п.) в языке на котором до шифрования было написано письмо – этот язык известен и далее будет называться базовым), в которой p_{ij} – отвечает за вероятность появления символа с номером j сразу после символа под номером i. Такая матрица может быть идентифицирована с помощью статистического анализ какого-нибудь большого текста, скажем, “Войны и мира” Л.Н. Толстого.

Пускай способ (де)шифрования (подстановочный шифр) определяется некоторой, неизвестной, дешифрующей функцией $\bar{f}$ – преобразование (перестановка) множества кодовых букв во множество символов базового языка.

В качестве, “начального приближения” выбирается какая-то функция f, например, полученная исходя из легко осуществимого частотного анализа. Далее рассчитывается вероятность выпадения полученного закодированного текста $\vec{x}$, сгенерированного при заданной функции f (функция правдоподобия):

$$L(\vec{x}; f) = \prod_k p_{f(x_k), f(x_{k+1})} . \qquad (*)$$

Случайно выбираются два аргумента у функции f и значения функции при этих аргументах меняются местами. Если в результате получилась такая f^*, что $L(\vec{x}; f^*) \geq L(\vec{x}; f)$, то $f := f^*$, иначе независимо бросается монетка с вероятностью выпадения орла $p = L(\vec{x}; f^*) / L(\vec{x}; f)$, и если выпадает орёл, то $f := f^*$, иначе $f := f$. Далее процедура повторяется (в качестве f выбирается функция, полученная на предыдущем шаге).

Объясните, почему предложенный алгоритм после некоторого числа итераций с большой вероятностью и с хорошей точностью восстанавливает дешифрующую функцию $\bar{f}$? Почему сходимость оказывается такой быстрой (0.01 сек. на современном PC)?

Указание. Описанный в задаче пример взят из обзора *Diaconis P.* The Markov chain Monte Carlo revolution // Bulletin (New Series) of the AMS. 2009. V. 49. № 2. P. 179–205. Детали того, что будет написано далее можно почерпнуть из работ *Jerrum M., Sinclair A.* The Markov chain Monte Carlo method: an approach to approximate counting and integration // Approximation Algorithms for NP-hard Problems / D.S. Hochbaum ed. Boston: PWS Publishing, 1996. P. 482–520; *Levin D.A., Peres Y., Wilmer E.L.* Markov chain and mixing times. AMS, 2009; *Joulin A., Ollivier Y.* Curvature, concentration and error estimates for Markov chain Monte Carlo // Ann. Prob. 2010. V. 38. № 6. P. 2418–2442; *Paulin D.* Concentration inequalities for Markov chains by Marton couplings // e-print, arXiv:1212.2015v2, 2013.

Для того чтобы построить однородный дискретный марковский процесс с конечным числом состояний, имеющий наперед заданную инвариантную (стационарную) меру π, переходные вероятности ищутся в следующем виде: $p_{ij} = p_{ij}^0 b_{ij}$, $i \neq j$; $p_{ii} = 1 - \sum_{j:\ j \neq i} p_{ij}$, где p_{ij}^0 – некоторая "затравочная" матрица, которую будем далее предполагать симметричной. Легко проверить, что матрица p_{ij} имеет инвариантную (стационарную) меру π, если при $p_{ij}^0 > 0$

$$\frac{b_{ij}}{b_{ji}} = \frac{\pi_j p_{ji}^0}{\pi_i p_{ij}^0} = \frac{\pi_j}{\pi_i}.$$

Чтобы найти b_{ij}, достаточно найти функцию $F : \mathbb{R}_+ \to [0,1]$ такую, что

$$\frac{F(z)}{F(1/z)} = z \text{ и положить } b_{ij} = F\left(\frac{\pi_j p_{ji}^0}{\pi_i p_{ij}^0}\right) = F\left(\frac{\pi_j}{\pi_i}\right).$$

Пожалуй, самый известный пример (именно он и использовался в задаче) такой функции $\tilde{F}(z) = \min\{z,1\}$ – алгоритм (Хастингса–)Метрополиса. Заметим, что для любой такой функции $F(z)$ имеем $F(z) \leq \tilde{F}(z)$. Другой пример дает функция $F(z) = z/(1+z)$. Заметим также, что p_{ij}^0 обычно выбирается равным $p_{ij}^0 = 1/M_i$, где M_i число "соседних" состояний у i, или

$$p_{ij}^0 = 1/(2M),\ i \neq j;\ p_{ii}^0 = 1/2,\ i \neq j.$$

При больших значениях времени t, согласно эргодической теореме, имеем, что распределение вероятностей близко к стационарному π. Действительно, при описанных выше условиях имеет место условие детального баланса (марковские цепи, для которых это условие выполняется, иногда называют обратимыми):

$$\pi_i p_{ij} = \pi_j p_{ji}, i, j = 1,...,n,$$

из которого сразу следует инвариантность меры π, т.е.

$$\sum_i \pi_i p_{ij} = \pi_j \sum_i p_{ji} = \pi_j,\ j = 1,...,n.$$

Основное применение замеченного факта состоит в наблюдении, что время выхода марковского процесса на стационарную меру (mixing time) во многих случаях оказывается

удивительно малым.[28] При том, что выполнение одного шага по времени случайного блуждания по графу, отвечающему рассматриваемой марковской цепи, как следует из алгоритма Кнута–Яо, также может быть быстро сделано. Таким образом, довольно часто можно получать эффективный способ генерирования распределения дискретной случайной величины с распределением вероятностей π за время полиномиальное от логарифма числа компонент вектора π.

Для лучшего понимания происходящего в условиях задачи, отметим, что одним из самых универсальных способов получения асимптотически наилучших оценок неизвестных параметров по выборке является метод наибольшего правдоподобия (Ибрагимов–Хасьминский, В.Г. Спокойный). Напомним вкратце в чем он заключается. Пусть имеется выборка из распределения, зависящего от неизвестного параметра – в нашем случае выборкой $\vec{x}$ из 10 000 элементов будет письмо, а неизвестным "параметром" будет функция f. Далее считается вероятность (или плотность вероятности в случае непрерывных распределений) $L(\vec{x}; f)$ того что выпадет данный $\vec{x}$ при условии, что значение параметра f. Если посмотреть на распределение $L(\vec{x}; f)$, как на распределение в пространстве параметров ($\vec{x}$ – зафиксирован), то при большом объеме выборки (размерности $\vec{x}$) при естественных условиях это распределение концентрируется в малой окрестности наиболее вероятного значения

$$f(\vec{x}) = \arg\max_f L(\vec{x}; f),$$

которое "асимптотически" совпадает с неизвестным истинным значением $\overline{f}$.

Замечание. Для оценки mixing time нужно оценить спектральную щель стохастической матрицы переходных вероятностей, задающей исследуемую марковскую динамику, то есть нужно оценить расстояние от максимального собственного значения этой матрицы равного единицы (теорема Фробениуса–Перрона) до следующего по величине модуля. Именно это число определяет основание геометрической прогрессии, мажорирующей исследуемую последовательность норм разностей расстояний (по вариации) между распределением в данный момент времени и стационарным (финальным) распределением. Для оценки спектральной щели разработано довольно много методов, из которых мы упомянем лишь некоторые: неравенство Пуанкаре (канонический путь), изопериметрическое неравенство Чигера (проводимость), с помощью техники каплинга (получаются простые, но, как правило, довольно грубые оценки), с помощью каплинга Мертона, с помощью дискретной кривизны Риччи и теорем о концентрации меры (Мильмана–Громова). Приведем некоторые примеры применения МСМС: Тасование n карт, разбиением приблизительно на две равные кучи и

[28] Более того, задача поиска такого симметричного случайного блуждания на графе (с равномерной инвариантной мерой в виду симметричности) заданной структуры, которое имеет "наименьшее" mixing time (другими словами, наибольшую спектральную щель), сводится к задаче полуопределенного программирования, которая, как известно, полиномиально (от числа вершин этого графа) разрешима.

перемешиванием этих куч (mixing time $\sim \log_2 n$);[29] Hit and Run (mixing time $\sim n^3$); Модель Изинга – n спинов на отрезке, стационарное распределение = распределение Гиббса, Глауберова динамика (mixing time $\sim n^{2\log_2 e/T}$, $0 < T \ll 1$); Проблема поиска кратчайших гамильтоновых путей; Имитация отжига (см. задачу 20) для решения задач комбинаторной оптимизации, MCMC для решения задач перечислительной комбинаторики. Но, пожалуй, самым известным примером (Dyer–Frieze–Kannan) является полиномиальный вероятностный алгоритм (работающий быстрее известных "экспоненциальных" детерминированных) приближенного поиска центра тяжести выпуклого множества и вычисления его объема. Одна из работ в этом направлении была удостоена премии Фалкерсона – аналога Нобелевской премии в области Computer Science. Близкие идеи используются и при применении экспандеров в Computer Science (см. раздел [ССЫЛКА]). В частности, в 2010 году премия Неванлинны была вручена Д. Спилману, в частности, за сублинейное (по числу элементов матрицы, отличных от нуля) решение системы линейных уравнений с помощью экспандеров.

Полезно сравнить эту задачу с задачей 7 и с задачей 19 раздела [ССЫЛКА].

Задача 20 (глобальная оптимизация и simulated annealing). Пожалуй, самым популярным сейчас методом глобальной оптимизации (правда, с очень плохими на данный момент теоретическими оценками скорости сходимости) является simulated annealing (имитация затвердевания или отжига), представляющий собой дискретное приближение решения стохастического дифференциального уравнения[30]

$$dx_t = -\nabla f(x_t)\,dt + \sqrt{2T(t)}\,dw_t,$$

где w_t – винеровский процесс. Покажите, что при неограничительных условиях и $T(t) \equiv T$ траектория x_t имеет при $t \to \infty$ стационарное распределение с плотностью Гиббса

$$\frac{\exp\left(-f(x)/T\right)}{\int \exp\left(-f(z)/T\right) dz},$$

[29] Здесь контраст проявляется, пожалуй, наиболее ярко. Скажем для колоды из 52 карт пространство состояний марковской цепи будет иметь мощность 52! (если сложить времена жизней в наносекундах каждого человека, когда либо жившего на Земле, то это число на много порядков меньше 52!). В то время как такое тасование: взять сверху колоды карту, и случайно поместить ее во внутрь колоды, отвечающее определенному случайному блужданию, с очень хорошей точностью выйдет на равномерную меру, отвечающую перемешанной колоде, через каких-то 200–300 шагов. Если брать тасование разбиением на кучки, то и того меньше – за 8–10 шагов.

[30] Детально изученного в статье *German S., Hwang C.P.* Diffusions for global optimization // SIAM J. Control and Optimization. 1986. V. 24. no. 5. P. 1031–1043.

экспоненциально концентрирующееся в окрестности единственной точки глобального минимума x^* дважды гладкой функции $f(x)$ при $T \to 0+$. Однако при $T \to 0+$ и время выхода на это стационарное распределение неограниченно возрастает, что создает проблемы для практического применения. Более правильно брать $T(t) = c/\ln(2+t)$, где c – достаточно большое число. Покажите, что тогда для любой начальной точки x_0 траектория процесса x_t имеет в пределе $t \to \infty$ (который, фактически, с хорошей точностью проявляется уже на конечных временах) распределение, сосредоточенное в точке x^*.

Замечание. Детали и способы дискретизации можно почерпнуть из работы *Kushner H.* Asymptotic global behavior for stochastic approximation and diffusion with slowly decreasing noise effects: global minimization via Monte Carlo // SIAM J. Appl. Math. 1987. V. 47. no. 1. P. 169–183.

Задача 21 (Multilevel Monte Carlo; M. Giles). Некоторый диффузионный процесс описывается стохастическим дифференциальным уравнением

$$dS(t) = a(S,t)dt + b(S,t)dW(t),\ 0 \le t \le T,\ S(0) = S_0,$$

где $W(t)$ – винеровский процесс. Задана липшицева функция $f(S)$. Требуется предложить численный способ оценивания $Y = E\left[f\left(S(T)\right)\right]$.

а)* Дискретизируем задачу по схеме Эйлера

$$\hat{S}_{n+1} = \hat{S}_n + a\left(\hat{S}_n, t_n\right)h + b\left(\hat{S}_n, t_n\right)\Delta W_n,$$

возьмем N независимых реализаций $\left\{\hat{S}_n^{(i)}\right\}$, и положим

$$\bar{Y} = \frac{1}{N}\sum_{i=1}^{N} f\left(\hat{S}_{T/h}^{(i)}\right).$$

Покажите, что найдутся такие $C_1, C_2 > 0$, что

$$\mathrm{MSE} = E\left[\left(\bar{Y} - Y\right)^2\right] \approx C_1 N^{-1} + C_2 h^2.$$

Покажите, что если от оценки требуется точность ε ($\sqrt{\mathrm{MSE}} = \mathrm{O}(\varepsilon)$), то оптимально (с точки зрения $\mathrm{Total}(\varepsilon)$ – общего числа арифметических операций / генерирования нормальных с.в. ΔW_n) выбирать

$$h = \mathrm{O}(\varepsilon),\ N = \mathrm{O}\left(\varepsilon^{-2}\right),\ \mathrm{Total}(\varepsilon) = \mathrm{O}\left(\varepsilon^{-3}\right).$$

б)** Предложим другой (более эффективный) способ оценивания. Для этого введем константу $M>1$ и положим

$$h_l = M^{-l}T,\ \bar{Y}_l = \frac{1}{N_l}\sum_{i=1}^{N_l}\left(f\left(\hat{S}^{(i)}_{T/h_l}\right) - f\left(\hat{S}^{(i)}_{T/h_{l-1}}\right)\right),\ \bar{Y}_0 = \frac{1}{N_0}\sum_{i=1}^{N_0} f\left(\hat{S}^{(i)}_{T/h_0}\right),\ \bar{Y} = \sum_{l=0}^{L}\bar{Y}_l .$$

Покажите, что

$$\mathrm{Bias} = E\left[\bar{Y} - Y\right] = \mathrm{O}\left(h_L\right),$$

$$V_l = D\left[f\left(\hat{S}^{(i)}_{T/h_l}\right) - f\left(\hat{S}^{(i)}_{T/h_{l-1}}\right)\right] = \mathrm{O}\left(h_l\right),$$

Мы хотим, чтобы

$$\sqrt{\mathrm{MSE}} = \sqrt{\mathrm{Bias}^2 + D\left[\bar{Y}\right]} \le \mathrm{Bias} + \sqrt{D\left[\bar{Y}\right]} \sim \varepsilon,$$

что достигается, если положить

$$L = \log\left(\varepsilon^{-1}\right)\Big/\log M + \mathrm{O}(1),\ \ D\left[\bar{Y}\right] = \sum_{l=0}^{L} N_l^{-1}V_l \sim \sum_{l=0}^{L} N_l^{-1}h_l \sim \varepsilon^2 .$$

Учитывая это, покажите, что решение задачи $\mathrm{Total}\left(\varepsilon\right) = \sum_{l=0}^{L} N_l h_l \to \min\limits_{\{N_l\}\ge 0}$ при ограничении $\sum_{l=0}^{L} N_l^{-1}h_l = \mathrm{O}\left(\varepsilon^2\right)$ имеет вид $N_l = \mathrm{O}\left(\varepsilon^{-2}Lh_l\right)$. Таким образом, $\mathrm{Total}\left(\varepsilon\right) = \mathrm{O}\left(\varepsilon^{-2}\left(\log\varepsilon\right)^2\right)$.

Замечание. Описанный в п. б) метод был предложен относительно недавно в контексте разработки эффективных численных методов оценки финансовых инструментов на рынке, поэтому он попал далеко не во все классические монографии на эту тему: *Glasserman P.* Monte Carlo methods in financial engineering. Springer, 2005; *Graham C., Talay D.* Mathematical foundation of stochastic simulation. Series “Stochastic modelling and applied probability”. V. 68. 2013. Тем не менее, мы рекомендуем эти книги для погружения в область численных методов финансовой математики.

Задача 22 (у.з.б.ч. Арнстейна–Виталя)*. Пусть задана многозначная случайная величина $A\colon \Omega \to \mathrm{M}\left(\mathbb{R}^n\right)$, где $\mathrm{M}\left(\mathbb{R}^n\right)$ – пространство всевозможных измеримых непустых компактных подмножеств $\mathbb{R}^n$. Мы считаем это пространство метрическим (с метрикой Хаусдорфа) с борелевской сигма-алгеброй, порожденной всевозможными замкнутыми множествами пространства $\mathrm{M}\left(\mathbb{R}^n\right)$ (A – многозначная случайная величина, если прообраз любого элемента этой сигма-алгебры измерим). Покажите, что (сумма множеств и, как предел, интегрирование при взятии математического ожидания, понимаются в смысле Минковского; сходимость в метрике Хаусдорфа; а $\mathrm{conv}\left(A\right)$ – выпуклая оболочка множества A):

$$\frac{1}{n}\left(A_1+\ldots+A_n\right)\xrightarrow[n\to\infty]{п.н.}E\left(\operatorname{conv}\left(A\right)\right),$$

где A_i – i.i.d. с распределением A и $E\left(\|A\|\right)<\infty$.

Замечания. Необходимые факты из выпуклого анализа имеются в книге *Магарил-Ильяев Г.Г., Тихомиров В.М.* Выпуклый анализ и его приложения. М.: УРСС, 2011. Эту же книгу можно рекомендовать и к следующим двум задачам. Утверждение этой задачи полезно в негладкой стохастической оптимизации при изучении стохастического субдифференциала (Shapiro–Dentcheva–Ruszczynski).

ВНИМАНИЕ! ПЛОХО НАПИСАННАЯ ЗАДАЧА ☹

Задача 23 (теорема А.А. Ляпунова о векторных мерах и принцип максимума; Поляк–Хауслер–Линденштраусс)*. Теорема Ляпунова о векторных мерах, заключается в том, что если дан конечный набор неатомарных мер $\left(\mu_1,\ldots,\mu_n\right)$ на $\left(\Omega,\Xi\right)$, то тогда множество $\left\{\left(\mu_1(S),\ldots,\mu_n(S)\right):S\in\Xi\right\}$ выпукло и замкнуто в $\mathbb{R}^n$. Используя эту теорему докажите принцип максимума Понтрягина. Нетривиальность принципа максимума в требовании глобального максимума по управлениям функции Понтрягина, в то время как сам принцип максимума гарантирует лишь условие локального максимума. Лианеризация задачи оптимального управления в окрестности оптимального режима и применение теоремы Ляпунова позволяет объяснить эту, на первый взгляд, нестыковку.

Уточнить ссылки у Б.Т. Поляка. Переписать желтый текст!

Задача 24 (лемма Неймана–Пирсона). В результате эксперимента получена простая выборка $\vec{x}$, относительно которой имеются две простые гипотезы:

$$H_0:\ \vec{x}\in L\left(\vec{x}\middle|H_0\right)\ \text{и}\ H_1:\ \vec{x}\in L\left(\vec{x}\middle|H_1\right).$$

С уровнем значимости $\alpha>0$ постройте наиболее мощный критерий проверки гипотезы H_0 против альтернативы H_1, т.е. найдите такую функцию $\varphi\left(\vec{x}\right)$ (решающее правило – вероятность, с которой следует принимать гипотезу H_1, если выпал $\vec{x}$), что

$$P\left(H_0\middle|H_1\right)=\int_\Omega\left(1-\varphi\left(\vec{x}\right)\right)L\left(\vec{x}\middle|H_1\right)d\vec{x}\to\min_{0\le\varphi(\cdot)\le1}$$

$$P\left(H_1\middle|H_0\right)=\int_\Omega\varphi\left(\vec{x}\right)L\left(\vec{x}\middle|H_0\right)d\vec{x}=\alpha.$$

Покажите, что решение этой задачи – наиболее мощное решающее правило (Неймана–Пирсона) имеет вид

$$\varphi(\vec{x}) = \begin{cases} 1, & \Lambda(\vec{x}) > \overline{\Lambda} \\ p(\vec{x}), & \Lambda(\vec{x}) = \overline{\Lambda} \\ 0, & \Lambda(\vec{x}) < \overline{\Lambda} \end{cases}, \text{ где } \Lambda(\vec{x}) = \frac{L(\vec{x}|H_1)}{L(\vec{x}|H_0)},$$

$\overline{\Lambda}$ и $0 \le p(\vec{x}) \le 1$ следует определять из условия (ошибкой первого рода):

$$P(H_1|H_0) = \int\limits_{\{\vec{x}:\ \Lambda(\vec{x})>\overline{\Lambda}\}} L(\vec{x}|H_0)\,d\vec{x} + \int\limits_{\{\vec{x}:\ \Lambda(\vec{x})=\overline{\Lambda}\}} p(\vec{x})L(\vec{x}|H_0)\,d\vec{x} = \alpha\,. \qquad (*)$$

Причем $\overline{\Lambda}$ определяется единственным образом, а от того как выбирать $p(\vec{x})$, удовлетворяющее (*), не зависит ошибка второго рода:

$$P(H_0|H_1) = \int\limits_{\{\vec{x}:\ \Lambda(\vec{x})<\overline{\Lambda}\}} L(\vec{x}|H_1)\,d\vec{x} + \int\limits_{\{\vec{x}:\ \Lambda(\vec{x})=\overline{\Lambda}\}} \left(1-p(\vec{x})\right)L(\vec{x}|H_1)\,d\vec{x} = \beta\,.$$

Замечание. На примере этой задачи хорошо демонстрируется принцип множителей Лагранжа. В книге Магарил-Ильяева–Тихомирова принцип множителей Лагранжа выводится на основе теоремы отделимости: существует такая гиперплоскость, проходящая через заданную граничную точку выпуклого множества, что само это множество лежит в одной полуплоскости. Любой задачи выпуклой оптимизации можно поставить в соответствие выпуклое множество. Решению задачи соответствует “граничная” точка этого множества. Уравнение отделяющий гиперплоскости определяется множителями Лагранжа. Удивительным образом этот результат не требует ни какой топологической структуры у задачи. Вместе с принципом Ферма, принцип множителей Лагранжа составляет основной инструментарий получения всевозможных условий оптимальности.