Практикум на ЭВМ

Практическое задание №1
«Линейные модели для классификации»

Георгий Демин
студент 317 группы ВМК МГУ,

02 декабря 2018 г.
Москва

1 Введение

В данном отчете представлены результаты выполнения практического задания №2 “Линейные модели для классификации” по курсу “Практикум на ЭВМ” кафедры ММП факультета ВМК МГУ. В задании изучались

- бинарная логистическая регрессия
- мультиномиальная регрессия
- методы многоклассовой классификации one-vs-all и all-vs-all
- градиентный и стохастический градиентый спуск
- некоторые методы обработки текстов

на основе датасета 20newsgroups Были проведены эксперименты со сравнением двух градиентных спусков (обычного и стохастического) и по результатам сделаны выводы о о быстроте сходимости и точности градиентного спуска в зависимости от параметров (размер шага, начальное приближение и размера подвыборки для стохастического). Эксперименты с предобработкой текста выполнены не были.

2 Теоретические выкладки

2.1 Формула градиента для функции потерь логистической регрессии

Рассматривается задача бинарной логистической регрессии с регуляризатором $L2$:

$$Q(X,w) = \frac{1}{l}\sum_{i=1}^{l} ln(1+e^{-y_i(w,x_i)}) + \frac{\lambda}{2}\|w\|_2^2 \to \min_w \tag{1}$$

где $x_i = (x^1, x^2 \dots x^D)$ один объект, а $y_i \in \{-1;1\}$ — класс, которому объект принадлежит. Требуется найти градиент функции потерь Q по вектору w. Обозначим

$$ln(1+e^{-y_i(w,x_i)}) = R_i(X,w) \tag{2}$$

По правилу дифференциирования сложной функции имеем

$$\frac{\partial R_i((w,x_i))}{\partial w} = \frac{\partial R_i((w,x_i)}{\partial(w,x_i)}\frac{\partial(w,x_i)}{\partial w} \tag{3}$$

Имеем (для наглядности явно указываем, что x — это вектор):

$$\frac{\partial R_i(t)}{\partial t} ln(1+e^{-y_i t})' = \frac{-y_i e^t}{1+e^{-y_i t}}$$

$$\frac{\partial(w,x_i)}{\partial w} = \overrightarrow{x}$$

$$\frac{\partial(\|w\|_2^2)}{\partial w} = \frac{\partial(w_1^2 + w_2^2 + \dots + w_D^2)}{\partial w} = 2w$$

Подставляя эти производные в 1 и в 3 получим окончательно:

$$\frac{\partial Q(w)}{\partial w} = \frac{1}{l}\sum_{i=1}^{l}\left(\frac{-y_i e^t}{1+e^{-y_i(w,x_i)}}\right) + \lambda w \tag{4}$$

2.2 Градиент функции потерь мультиномиальной логистической регресии

Здесь перед нами стоит такая задача оптимизации.

$$Q(X,w) = -\frac{1}{l}\sum_{i=1}^{l} ln\left(\frac{\exp\left((w_{y_i},x_i)\right)}{\sum_{r=1}^{C}\exp\left((w_{y_r},x_i)\right)}\right) + \frac{\lambda}{2}\sum_{r=1}^{C}\|w_k\|_2^2 \to \min_{w_1,\ldots,w_C} \tag{5}$$

где $w = \{w\}_{i,j=1}^{C\times D}$ - матрица весов. Чтобы решить эту задачу найдем матричную производную

$$\frac{\partial Q(w)}{\partial w} = \left(\frac{\partial Q(w)}{\partial w_1},\ldots,\frac{\partial Q(w)}{\partial w_C}\right)^{\mathrm{T}}$$

каждая компонента которого - это градиент весов соотвествующего класса Для начала преобразуем:

$$\sum_{i=1}^{l} ln\left(\frac{\exp\left((w_{y_i},x_i)\right)}{\sum_{r=1}^{C}\exp\left((w_{y_r},x_i)\right)}\right) = \sum_{i=1}^{l}\left((w_{y_i},x_i) - ln\sum_{r=1}^{C}\exp\left((w_{y_r},x_i)\right)\right) \tag{6}$$

Градиент левой части выражения 6 не будет тождественно равен нулю лишь когда мы берем его по весам класса, которому объект на самом деле и принадлежит. Учитывая это и то, что градиент правой части берется аналогично 3 и 4 с той лишь разницей, что вместо константы 1 будет сумма экспонент от скалярного произведения “неправильных” классов, получим:

$$\frac{\partial Q}{\partial w_1} = \sum_{\{x_t|y_t=1\}}^{X}\overrightarrow{x_t} - \sum_{i=1}^{l}\left(\frac{\exp\left(w_1,x_i\right)\overrightarrow{x_i}}{\sum_{r=1}^{C}\exp\left((w_{y_r},x_i)\right)}\right) + \lambda w_1$$

$$\vdots$$

$$\frac{\partial Q}{\partial w_j} = \sum_{\{x_t|y_t=j\}}^{X}\overrightarrow{x_t} - \sum_{i=1}^{l}\left(\frac{\exp\left(w_j,x_i\right)\overrightarrow{x_i}}{\sum_{r=1}^{C}\exp\left((w_{y_r},x_i)\right)}\right) + \lambda w_j$$

$$\vdots$$

$$\frac{\partial Q}{\partial w_1} = \sum_{\{x_t|y_t=C\}}^{X}\overrightarrow{x_t} - \sum_{i=1}^{l}\left(\frac{\exp\left(w_C,x_i\right)\overrightarrow{x_i}}{\sum_{r=1}^{C}\exp\left((w_{y_r},x_i)\right)}\right) + \lambda w_C$$

Здесь выражение, стоящее в слева от знака равенство — это частная производная функции потерь по целому вектору весов. Поэтому справа от знака равенства также стоят вектора ($\overrightarrow{x_i}$), умноженные на некоторые числовые коэффициенты. В левой части суммирование ведется по всем объектам, принадлежащих определенному классу. Таким образом мы учитываем оба слагаемых стоящих справа в 6. Производные, представленные в таком виде хорошо воспринимаются как вектора, поэтому мы не будем приводить формулу производной каждой компоненты

3 Эксперименты

3.1 Исследование поведения градиентного спуска

В первом эксперименте мы будем изучать градиентный спуск, а точнее то, как меняется время сходимости и его точность (здесь и всюду далее под точностью понимается значение метрики *accuracy*) в зависимости от величины шага, рассчитываемой по формуле $\frac{\alpha}{n^\beta}$ и начального приближения. С начала подберем наилучшие параметры, а затем будем варьировать один при фиксированных других. На рис. 2 для разных λ показана точность для различных α и β. Наилучшая точность 0.93 достигается при $\lambda = 0.03$ $\alpha = 1.5$ и $\beta = 0.3$ в дальнейшем мы и будем использовать эти параметры. Здесь любопытно отметить 2 вещи: во-первых, метод почти не переобучается — это по всей видимости связано с хорошим разбиением исходного датасета на обучающую и тестовую выборки. Во-вторых, мы видим, что есть некая поверхность в пространстве (α, β), на которой метод достигает наилучшей точности. При увеличении λ она немного сдвигается. Мы не будем останавливаться сейчас на поиске уравнения этой поверхности, а вместо этого отметим то, что предпочтительными являются малые значения параметров: при них градиентый спуск начинает с небольших шагов, но и при росте числа итераций, этот шаг не уменьшается сильно, так как знаменатель возводится в степень близкую к нулю,

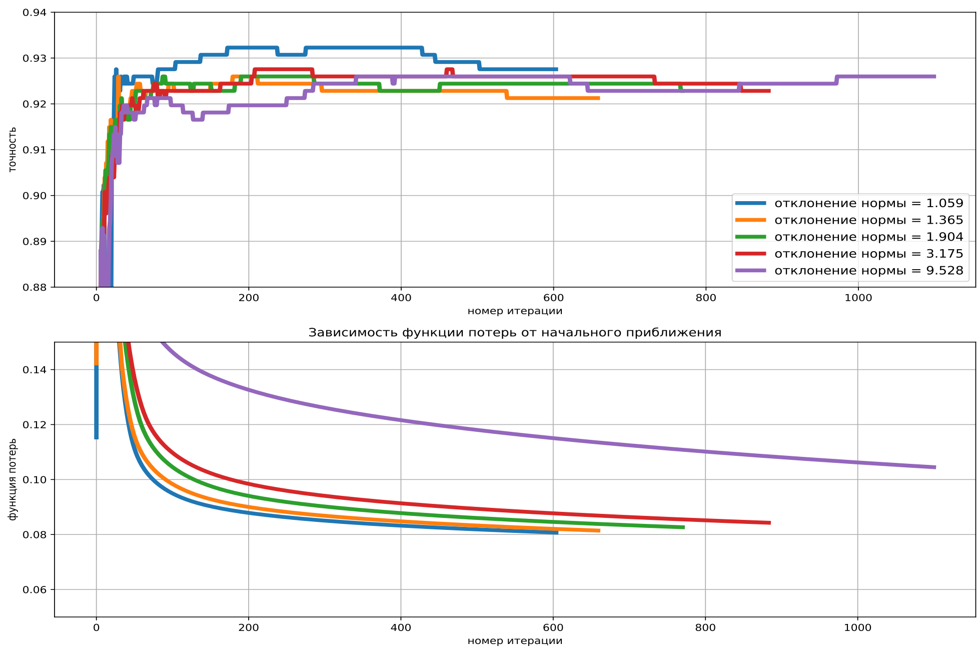

Рис. 1: Зависисимость точности и функции потерь от начального приближения. Отклонение от нормы — это квадрат евклидовой нормы разности вектора начального приближения и вектора, на котором достигается точность 0.93

Рассмотрим теперь, как влияет на точность и скорость сходимости (в этом и следующем эксперименте под скоростью будем понимать число итераций, необходимых для того, чтобы метод сошелся; реальное же время выполнения будем рассматривать в 3ем эксперименте) начальное приближение (см рис. 1. Мы видим, что при большом отклонении происходит сильное недообучение алгоритма, при весах же очень близких к "хорошему" весу мы видим даже некоторое переобучение: с 200 по 300 итерациях точность метода была лучше, чем при последних итерациях. Также отметим то, что точность представляет собой кусочно-постоянную функцию, это несколько странно, однако объяснение может быть такое: тестовая выборка довольно мала и при больших номерах итерациях изменение весов кранйне мало, таким образом на тестовой выборке с большим числом объектов изменение заметно "сразу", а на тестовой только при прохождение некого качественного порога.

Теперь обратимся к зависимостям от β и α. На рис. 3 мы видим довольно ожидаемую картину: при $\beta < 1$ метод неплохо сеюбя показывает к точности > 0.9 при больших же значениях величина шага становится малой слишком рано и метод уже не успевает близко подойти к минимуму, хотя двигается в верном направлении

Аналогичный результат мы наблюдаем и для α на рис. 4. Разница лишь в том, что здесь, наоборот, сходятся быстрее методы с большим α (ведь он стоит в числителе) и даже тот метод, у которого α очень мало сходится к минимуму намного быстрее, чем аналогичный с β (ясно почему – здесь шаг только умножается на малый коэффициент, а там знаменатель, который сам по себе возрастает с каждой итерацией, возводится в большую степень, что еще больше замедляет ход)

3.2 Исследование стохастического градиентного спуска

Во втором эксперименте будем действовать по той же схеме — изменяем один параметр при фиксированных других (мы не будем останавливаться на том, что для стохастического спуска могут быть

Рис. 2: Зависимость точности от параметра регуляризации и величины шага в GD(здесь и далее GD означает градиентный спуск, а SGD стохастический градиентый спуск). Точности ниже 0.5 помечена темным фиолетовым цветом. Этим же цветом помечены те запуски алгоритма, при котором градиентный спуск не сходился, и точность при разных итерациях принимала абсолютно различные значения

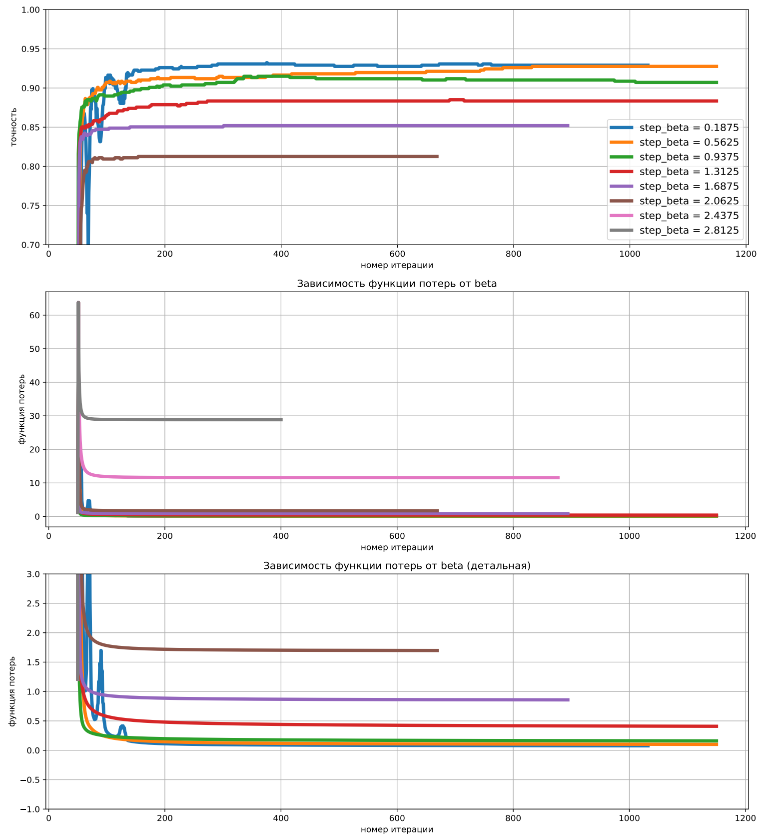

Рис. 3: Зависисимость точности и функции потерь GD от номера итерации На втором графике изображены методы при всех β, на 3ем же более детально рассмотрены значения параметра, при которых получается лучший результат

оптимальны некоторые другие параметры) . Получаем схожие результаты, обратим особое внимание на зависимость от размера подвыборки и α (графики остальных зависимотстей см в следующем эксперименте). На рис. 5 мы можем заметить интересный факт: размер подвыборки несильно влияет на итоговую точность, но довольно сильно на скорость сходимости. Однако мы видим, что связь эта нелинейна и в общем случае при разных коэффициентах будут выигрывать разные размеры подвыборок. На рис. 6 мы опять же видим, что значение параметра не слишком сильно влияет на итоговую точность, но зато может сильно ускорить или замедлить сходимость; при больших α мы получаем выигрыш и в скорости, и в точности. Нельзя также не отметить странные осцилляции при $\alpha \approx 2$: казалось бы при таком значении функция веса должны мало меняться — видимо, метод был на некой границе: при разных итерациях попадали разные подвыборки - в одних были объекты с одними значимыми признаками, с другой - другими, соответственно, метод не мог выбрать между ними. Кроме того, отметим, что по итогам проведенных экспереметов было установлено, что результирующая точность SGD (при параметрах, которые были опредлены оптимальными в первом эксперименте, — повторим еще раз, что при других параметрах, все может быть иначе!), вообще не зависит от начального приближения, также время работы не зависит от начального приближения: при удачном стечении обстоятельств метод может сойтись и за 50 итераций при норме отклонения больше 5, так и не сойтись при отклонении меньше 1 за 150 итераций.

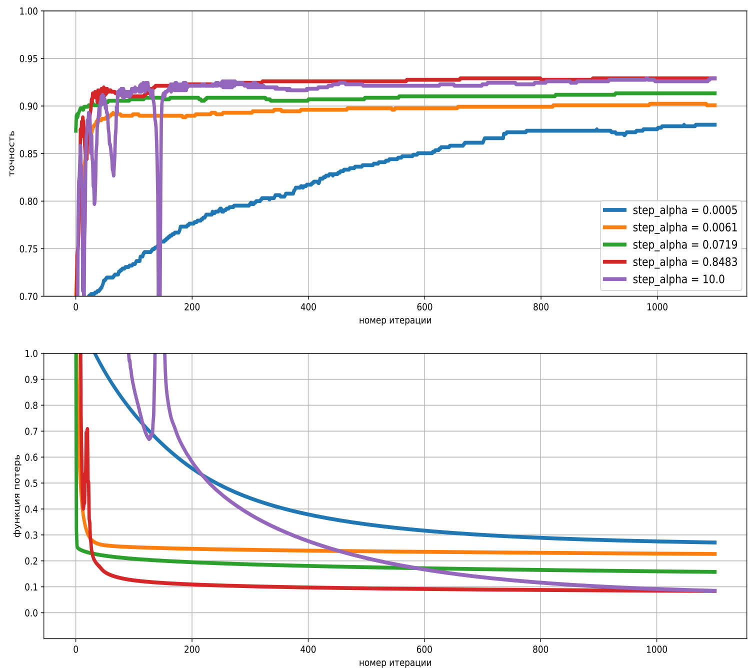

Рис. 4: Зависисимость точности и функции потерь GD от номера итерации при разных α

3.3 Сравнение обычного градиентного спуска и стохастического

В этом эксперименте мы сравним GD и SGD и выьерем оптимальный алгоритм для следующих экспериментов. На рис. 8 мы можем увидеть, что на самом деле стохастический градиентный спуск работает даже медленнее обычного - это не очень согласуется с нашим интуитивным представлением. Скорее всего, дело в том, что значение $tolerance$ для SGD должно быть ниже, чем у GD - мы видим, что стохастический спуск достиг очень большой точностив очень короткий срок, и все остальное время осциллировал вблизи минимума. Уже на 5ой секунде при всех нормальных значениях β SGD имеет точность больше 0.92 и продолжает ее наращивать, в то время как GD при всех β, кроме одного стагнировал.

Кроме того, у стохастического намного выше средняя точность по всем β. Из всего этого мы делаем вывод о том, что стохастический градиентный спуск быстрее находит окрестность минимума, но дольше продолжает в ней осциллировать, не прекращаю работы. Объяснение этому простое - критерий остановки это модуль разности функции потерь на соседних итерациях, но на соседних как раз итерациях она и различается больше всего - ведь берутся разные объекты из одной выборки и при небольшом размере этой подвыборки при каждой новой итерации веса будут стльно изменяться в различные стороны. Значит для улучшения работы стохастического градиентного спуска в данной задаче нужно изменить критерий останова (брать разность со средней величиной функции потерь за некторое количесвто итераций или что-то подобное)

$$M1 = \eta * A_s * R_s * h_0$$

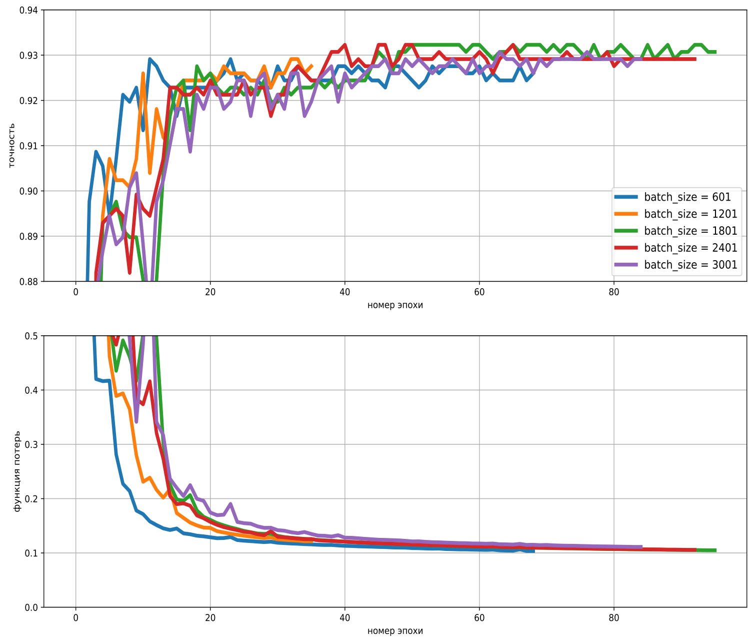

Рис. 5: Зависисимость точности и функции потерь SGD при разных размерах подвыборки

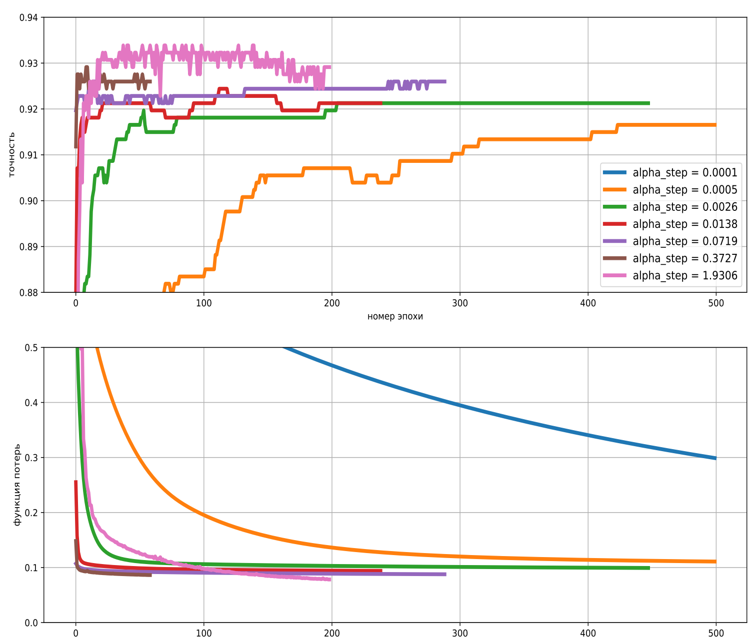

Рис. 6: Зависисимость точности и функции потерь SGD при разных α. При $\alpha = 10^{-4}$ точность на всех итерациях не превосходит 0.88

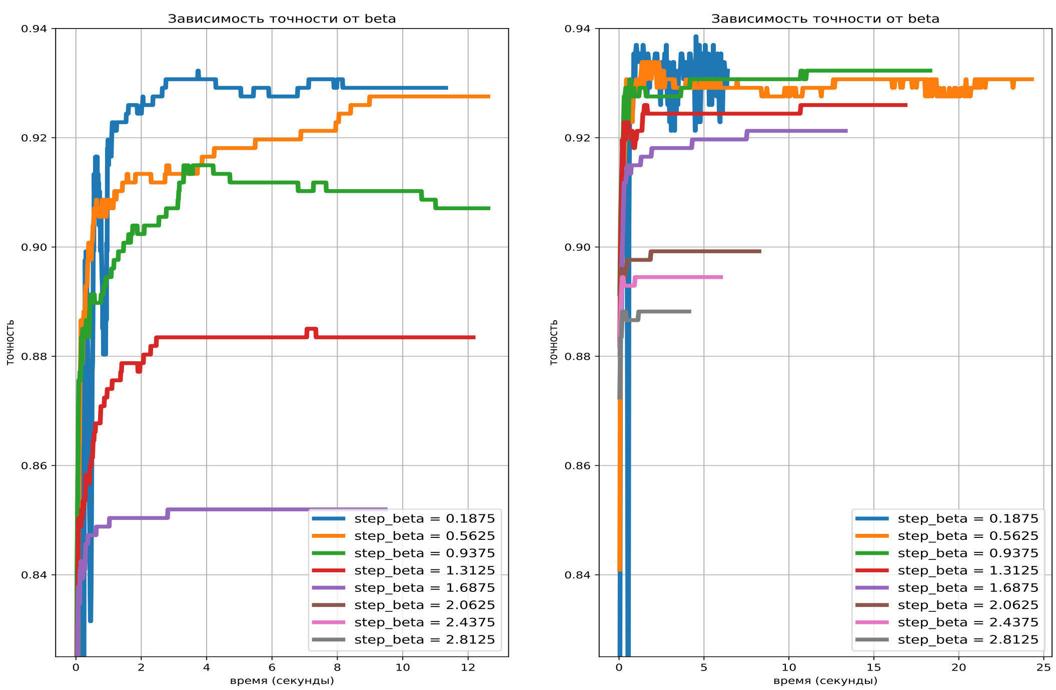

Рис. 7: Сравнение зависимостей точности от реального времени работы при разных β градиентного спуска и стохастического.

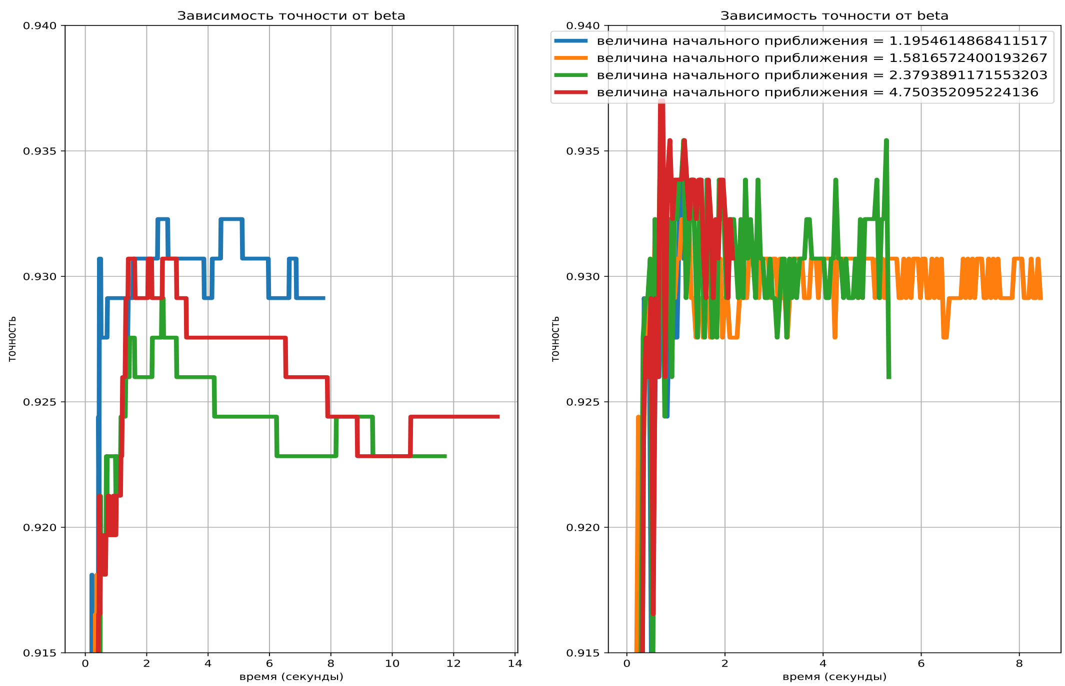

Рис. 8: Сравнение зависимостей точности от реального времени работы при разных β градиентного спуска и стохастического.