



Занятие 5: Подготовка текстовых отчётов в системе \TeX

Практикум на ЭВМ, осень 2021

Находнов Максим Сергеевич

кафедра ММП, ВМК МГУ

27 сентября 2021 г.

Написание научных текстов

Научные тексты: научные статьи, отчеты, курсовые, выпускные работы и т.д.

Есть правила (стандарты) написания научных текстов.

Зачем нужны правила?

Чтобы разные люди без дополнительных усилий понимали друг друга. Чем жёстче требования к терминологии, языку, форме подаче материала, оформлению, тем быстрее читатель сможет понять основную суть работы.

На этой паре разбираем, как писать отчёты.



Типичная структура отчёта

Структура отчёта примерно соответствует пунктам задания:

1. Титульная страница (Заголовок)
2. Введение
3. (опционально) Пояснения к задаче, например выкладки для выведенных в ходе работы формул
4. Список экспериментов
 - 4.1 Дизайн эксперимента
 - 4.2 Результаты эксперимента
 - 4.3 Выводы из эксперимента
5. (опционально) Общие выводы из работы
6. (если есть ссылки) Список использованной литературы



У отчёта должно быть введение

Из введения должно быть понятно:

- В чём заключалось задание?

Введение не повторяет полностью формулировку задания, а лишь кратко описывает её.

Можно ссылаться на формулировку задания, если не получается кратко сформулировать все аспекты.

Введение должно быть конкретным (относится к конкретному заданию, а не к выбранной области анализа данных в целом).



oooooooooooooooooooo

ooo

oooooo

oooooooooooo

Пример неконкретного введения к заданию

Компьютеры все чаще и чаще выступают помощниками человека. Водитель, потеряв дорогу, скорее воспользуется навигатором, чем спросит путь у прохожего или другого водителя. Захотев связаться с кем-то, мы скорее напишем ему письмо, состоящее из байтов, чем из бумаги и чернил. То же самое можно сказать и о распознавании изображений. Возьмем, как пример, ЕГЭ. Множество учеников со всей России заполняют огромное количество бланков, и все эти бланки необходимо проверить. Что же делать в этой ситуации? Ведь вряд ли кто согласится проверять эти бланки. И здесь на помощь приходят компьютеры. Они распознают ответы учеников и заполняют по ним базу данных, откуда берутся данные для подсчета результата экзамена. Теме анализа изображения, а точнее, анализа рукописного текста, и посвящена данная работа.

Описание экспериментов: как не надо делать

Задание 2

Векторизованный вариант:

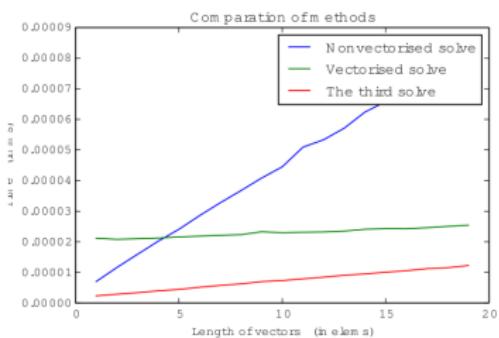
```
def vect_2(x, i, j):  
    return np.vectorize(lambda x, y: x[z, y])(i, j)
```

Не векторизованный вариант:

```
def vect_1(x, i, j):  
    r = np.array([], x.dtype)  
    for k in range(np.size(i)):  
        r = np.append(r, x[i[k], j[k]])  
    return r
```

Третий вариант:

```
def vect_3(x, i, j):  
    return np.array([x[t[0]][t[1]] for t in zip(i, j)])
```

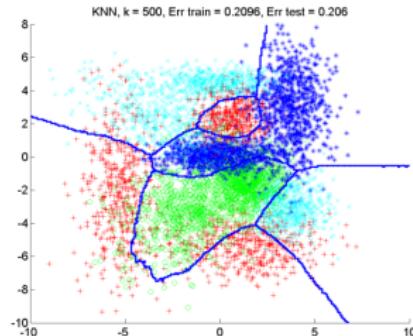
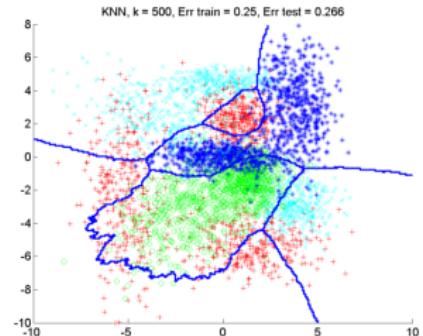


Самое оптимальное третье решение.

Эксперименты не описаны, нет выводов.

Описание экспериментов: как надо делать

Метод 500 ближайших соседей сильно недообучен, границы, выдаваемые им, слишком просты. Он обладает высоким *bias* (близкие кривые обучения на высоком уровне ошибки) при низком *variance*. При размере данных меньше 500 (50%) на тестовой выборке наблюдается плохо на уровне 75%: классификатор выдаёт в качестве ответа максимальный класс. На тестовой выборке при этом ошибка всё же ниже: сказывается случайный порядок объектов в обучающей выборке, соотношение классов не по 25%, и доминирующий класс состаляет чуть более 25%. Однако видна тенденция к падению уровня ошибки. Это происходит потому, что с увеличением объёма данных ослабляется эффект «доминирующего класса», и в этом случае добавление новых данных приведёт к значительному улучшению. Результат для 3000 обучающих объектов будет выглядеть гораздо более приемлемо, а для 5000 он даже приближается к оптимальному:

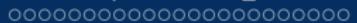


Выводы проведены.

Анализируйте результаты экспериментов

размер данных	numpy	only python	смешанная
10	1.2	50.21	2.1
100	2.52	40.1	10.2
1000	4.5	45.34	30.95

Что в этих результатах подозрительно?



Анализируйте результаты экспериментов

размер данных	numpy	only python	смешанная
10	1.2	50.21	2.1
100	2.52	40.1	10.2
1000	4.5	45.34	30.95

Что в этих результатах подозрительно?

- ▶ Вычисления для размерности 100 и 1000 происходит быстрее чем для 10.





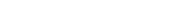


























































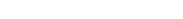
























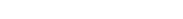






















































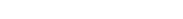








































































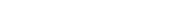




































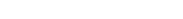








































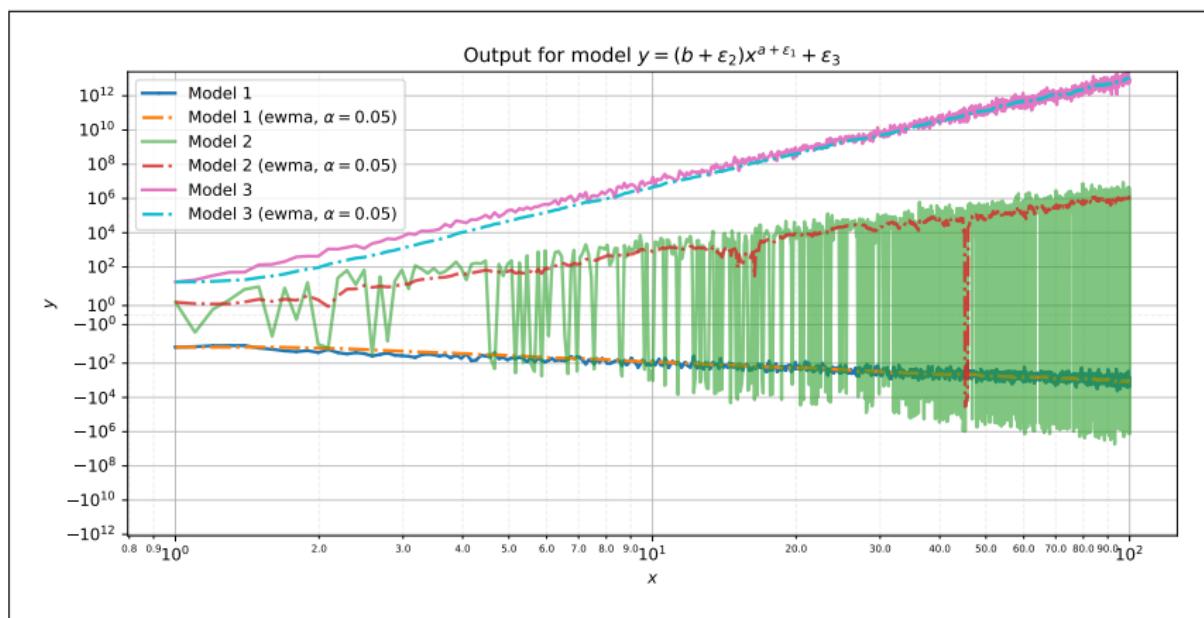






Оформление графиков. Некоторые советы

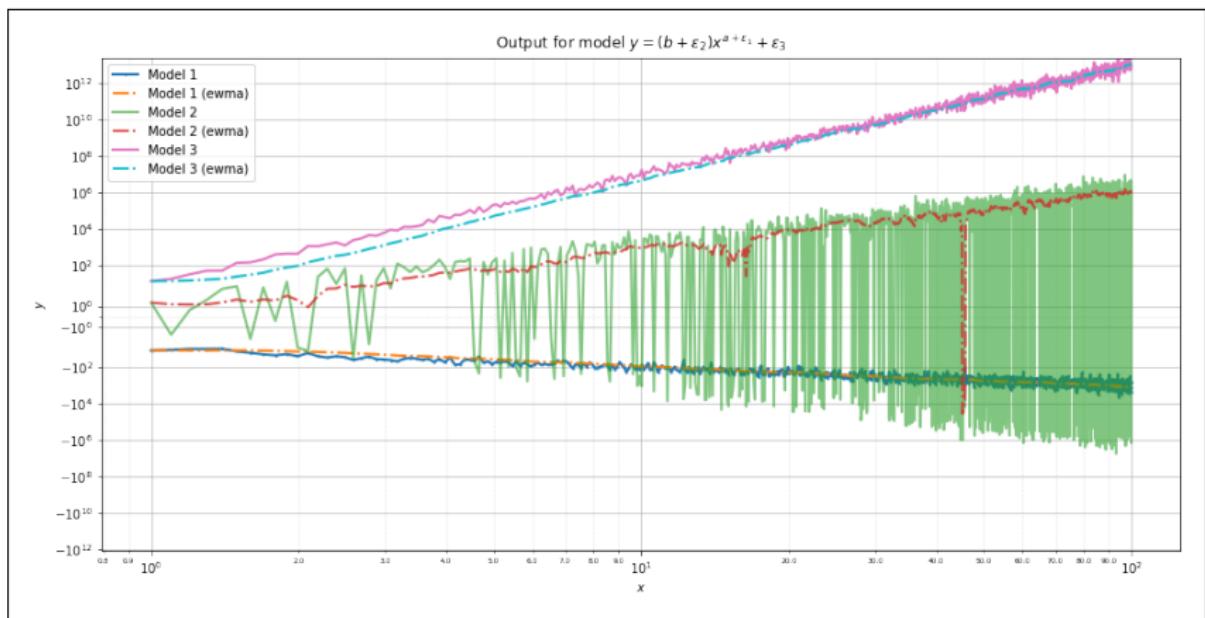
Графики должны быть с одной стороны понятными и информативными, а с другой стороны красивыми.





Оформление графиков. Некоторые советы

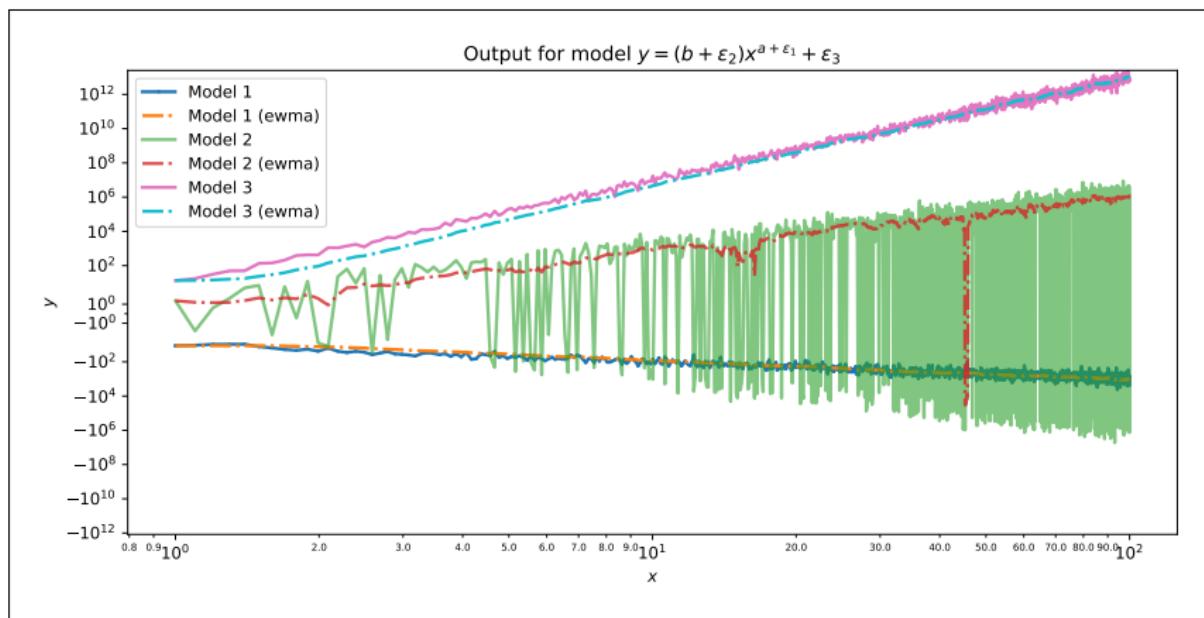
- Все графики должны быть отрисованы в векторном формате





Оформление графиков. Некоторые советы

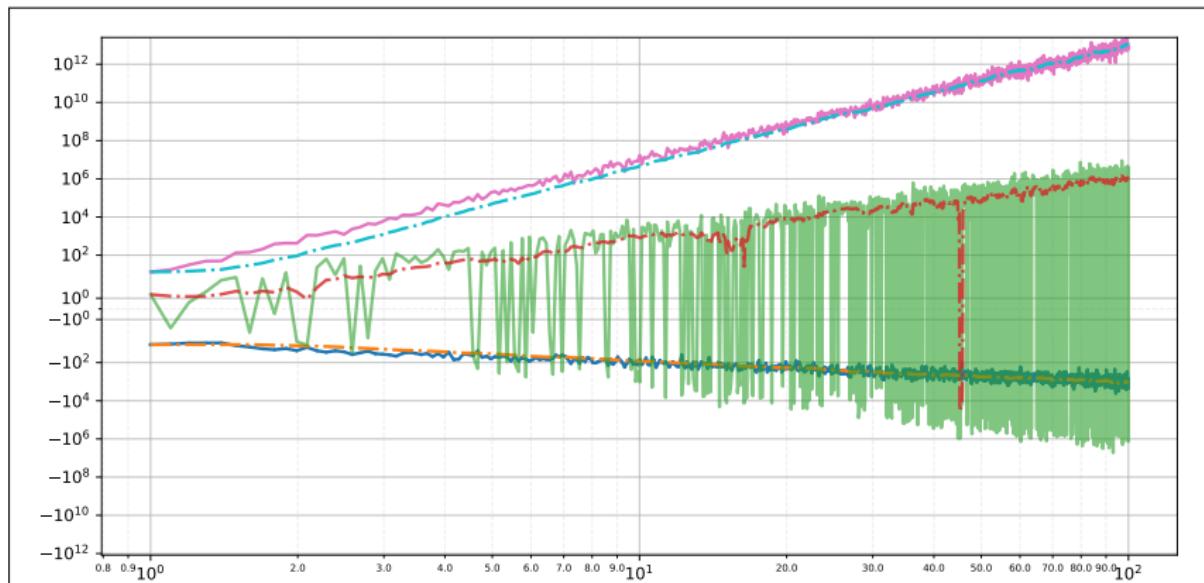
- На всех графиках без исключения должна быть отрисована сетка





Оформление графиков. Некоторые советы

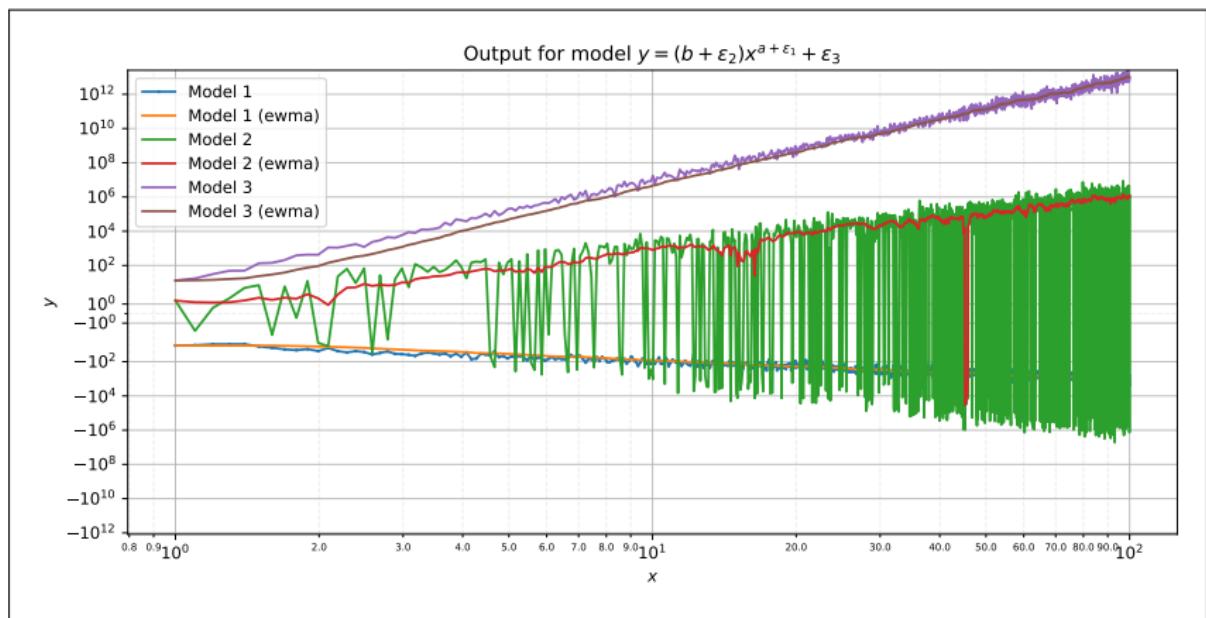
- ▶ Все графики и группы графиков должны иметь заголовок
- ▶ При необходимости оси должны быть подписаны
- ▶ Если на графике отображено несколько сущностей, то необходима исчерпывающая легенда





Оформление графиков. Некоторые советы

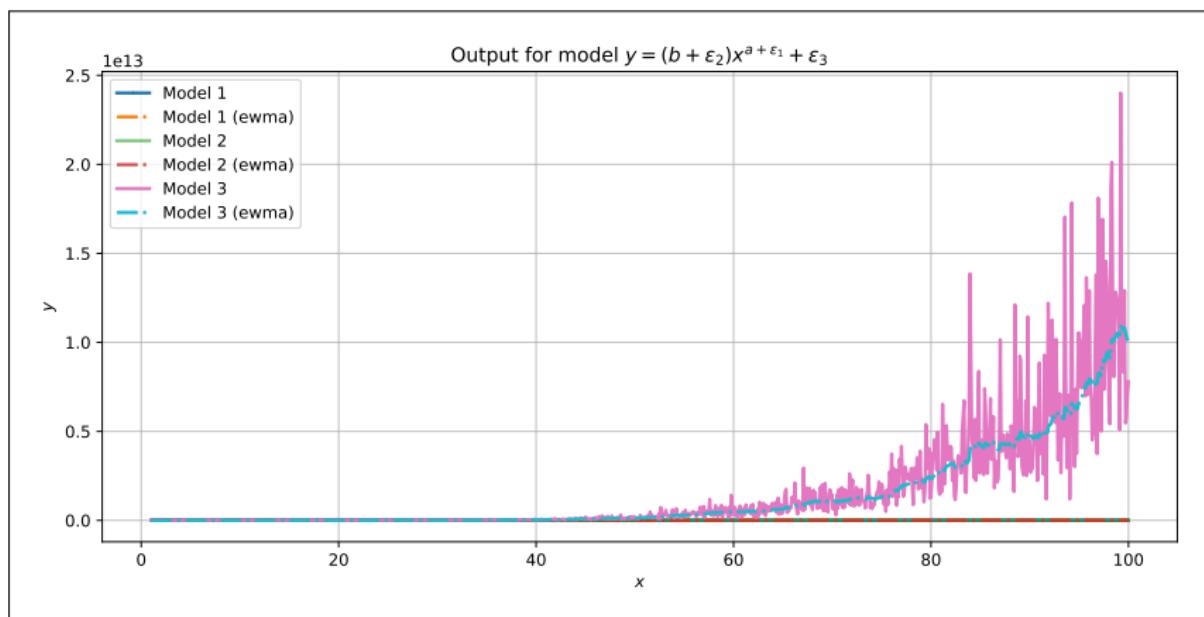
- ▶ Все линии на графиках должны быть чётко видны
- ▶ Используйте красивую цветовую палитру с хорошо различимыми цветами





Оформление графиков. Некоторые советы

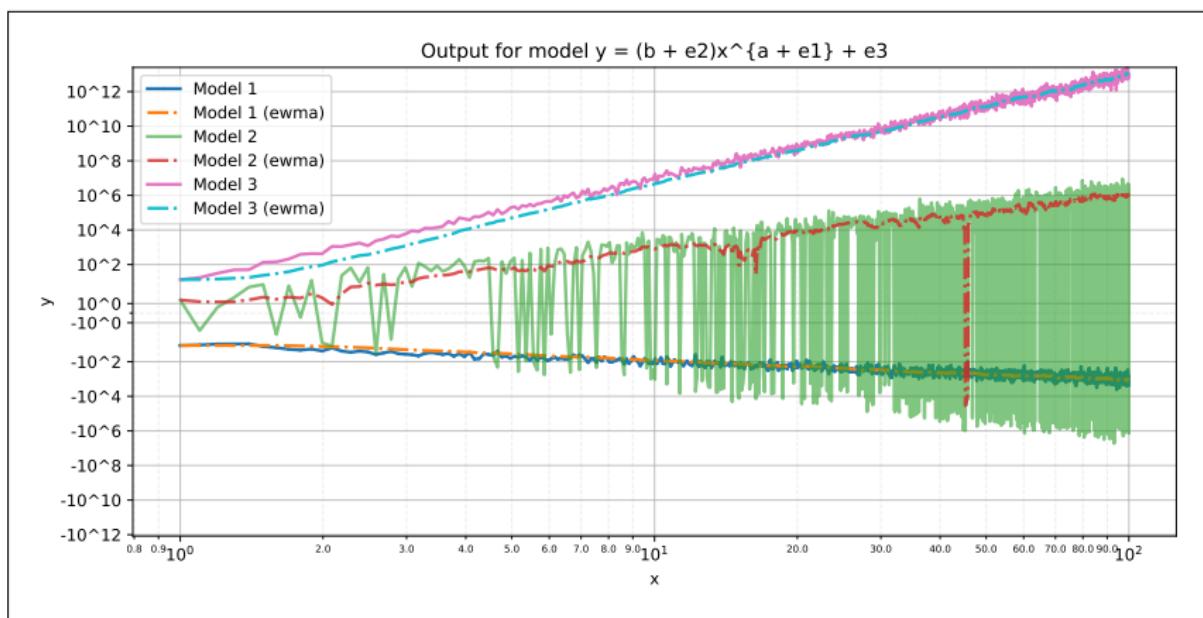
- Масштаб по каждой оси на графике должен быть выбран правильно





Оформление графиков. Некоторые советы

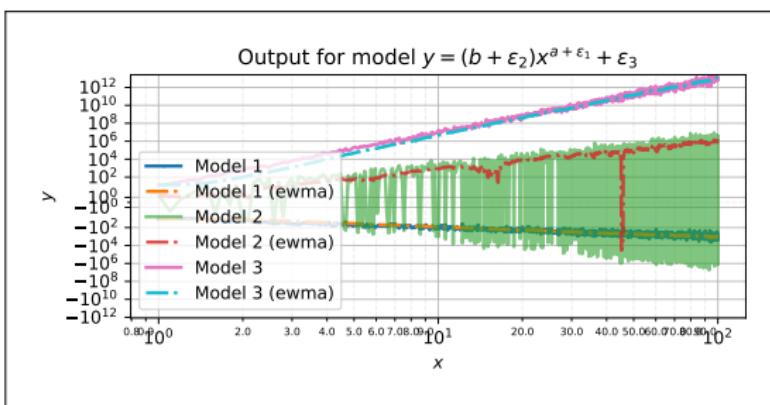
- Написания формул в заголовках, легенде и в подписях осей необходимо выполнять с помощью LaTeX





Оформление графиков. Некоторые советы

- Графики должны быть не супер-микро и не супер-макро по размерам, так, чтобы можно было увидеть все, что нужно

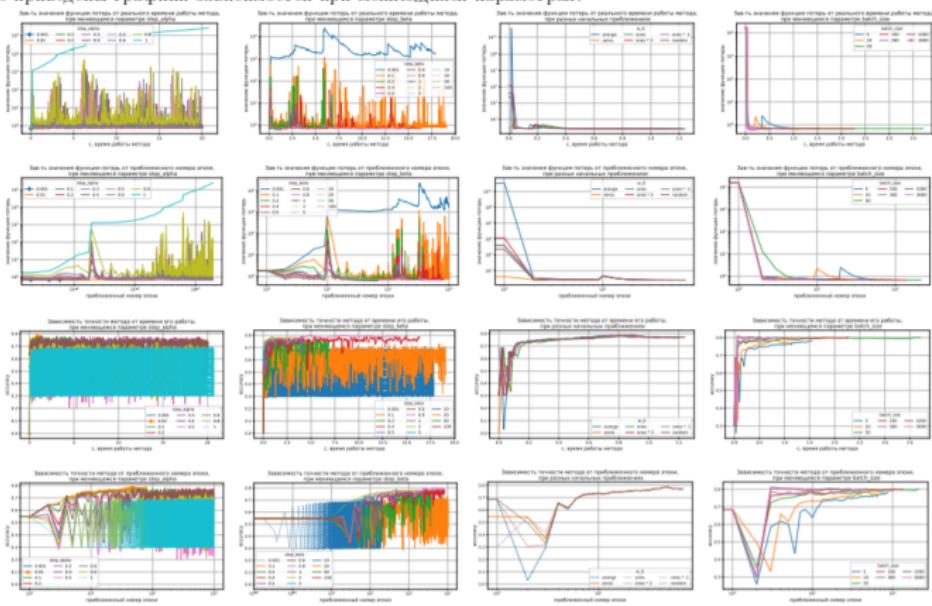




Оформление графиков. Примеры из жизни

3.2.2 Результаты

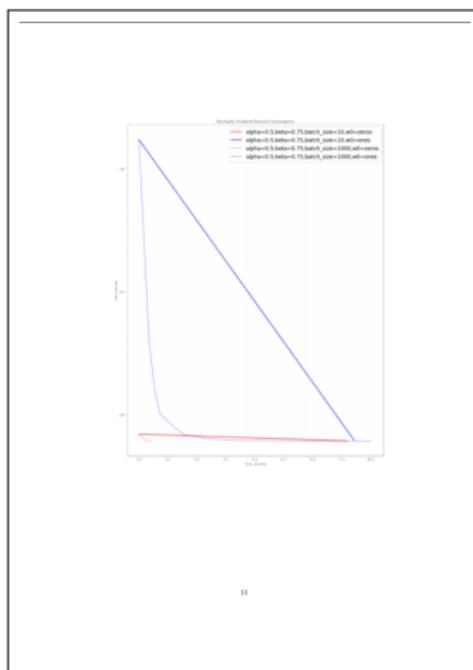
Ниже приведены графики зависимостей при меняющихся параметрах.



Много подробных графиков, но все они нечитаемы!



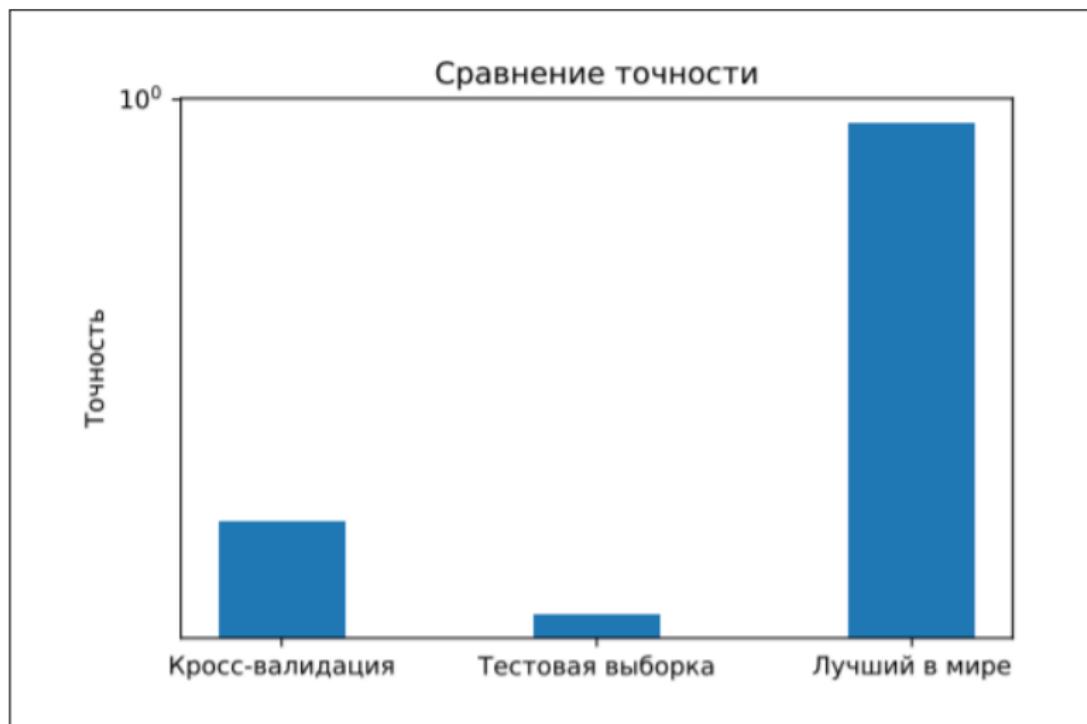
Оформление графиков. Примеры из жизни



Тут – наоборот, большой график, но понять что на нём происходит нельзя.



Оформление графиков. Примеры из жизни



Плохо подобраны отсчёты на вертикальной шкале.



Оформление графиков. Примеры из жизни

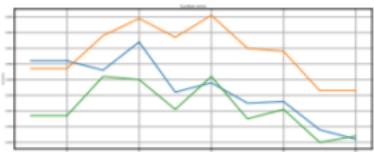


Рис. 1 Зависимость точности от числа соседей
для евклидовой метрики

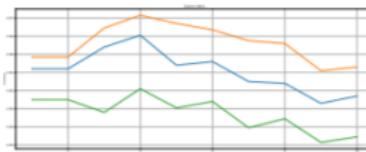


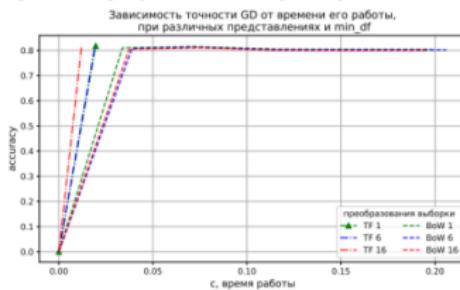
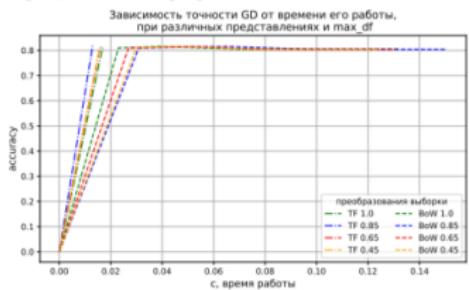
Рис. 2 Зависимость точности от числа
соседей для косинусной метрики

Отсчёты не видно.



Оформление графиков. Примеры из жизни

Ниже представлены графики зависимости качества от времени при различных параметрах.



Число, стоящее рядом с методом преобразования, является max_df

Число, стоящее рядом с методом преобразования, является min_df

Размер признакового пространства уменьшается в зависимости от увеличения min_df : 89055

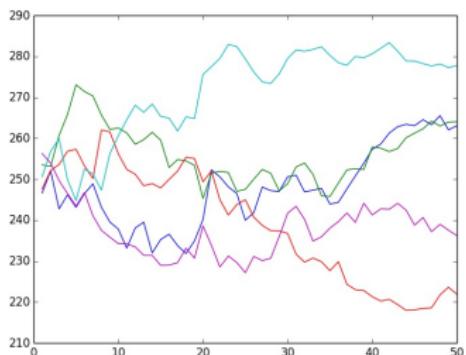
-> 15744 -> 8032. При изменении параметра min_df размерность не меняется: 89055

Подпись графика слишком мелкая.

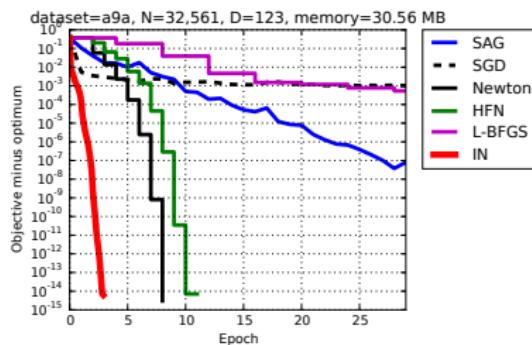


Оформление графиков. Примеры из жизни

Плохой график:



Хороший график:



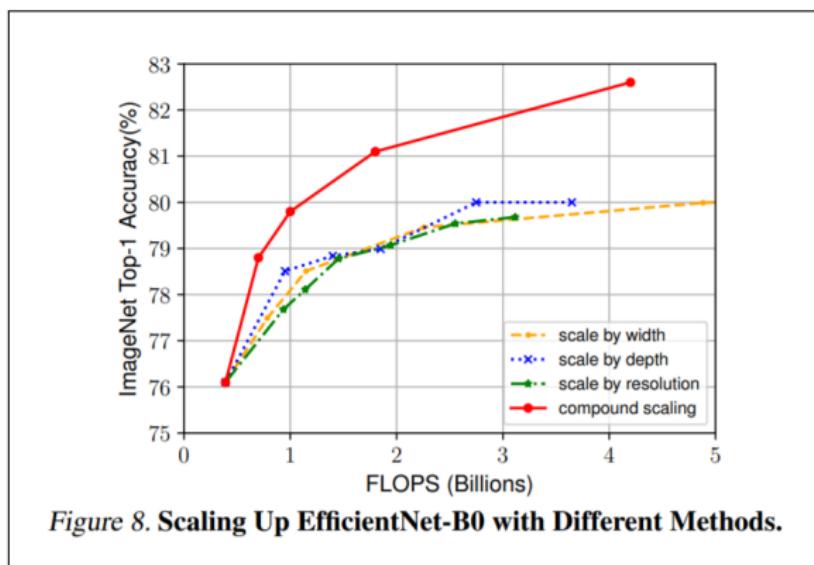
Элементы хорошего графика:

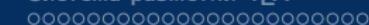
- ▶ Все линии жирные;
- ▶ Есть легенда;
- ▶ По осям указаны значения, сами оси подписаны;
- ▶ По осям выбрана правильная шкала;
- ▶ Сохранён в векторном формате.



Оформление графиков. Примеры из жизни

Хороший график:





Оформление графиков. Примеры из жизни

Хороший график:

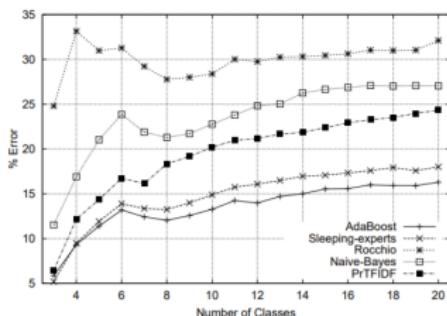
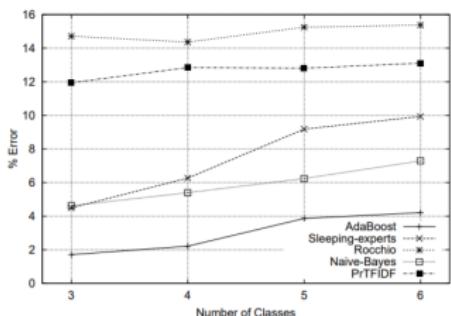
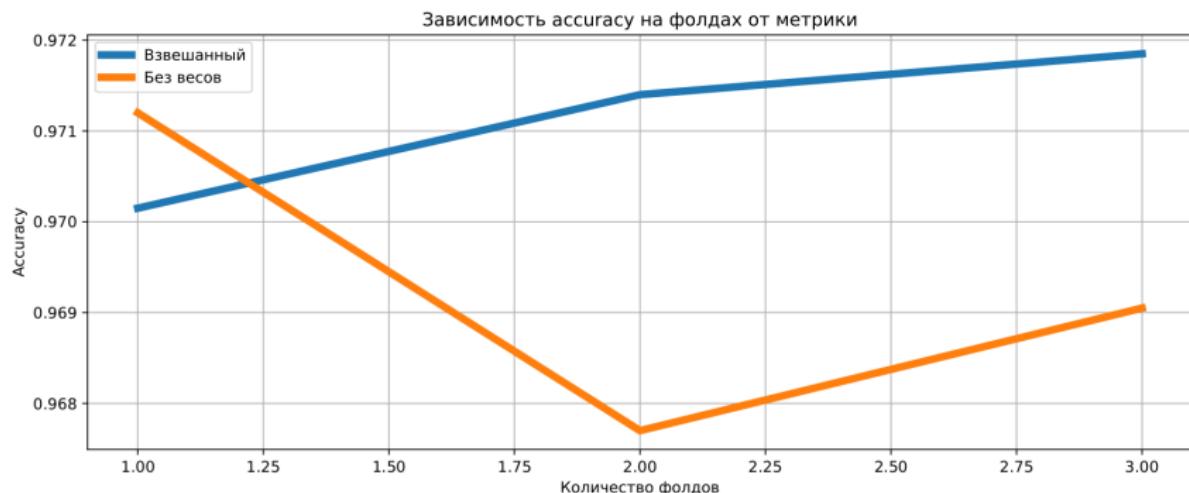


Figure 4: Comparison of error rates for AdaBoost and four other text categorization methods (naive Bayes, probabilistic TF-IDF, Rocchio and sleeping experts) as reported by Schapire and Singer [43]. The algorithms were tested on two text corpora — Reuters newswire articles (left) and AP newswire headlines (right) — and with varying numbers of class labels as indicated on the *x*-axis of each figure.

Стоит задумываться о том, как будет выглядеть график, если он будет отображаться чёрно-белым.



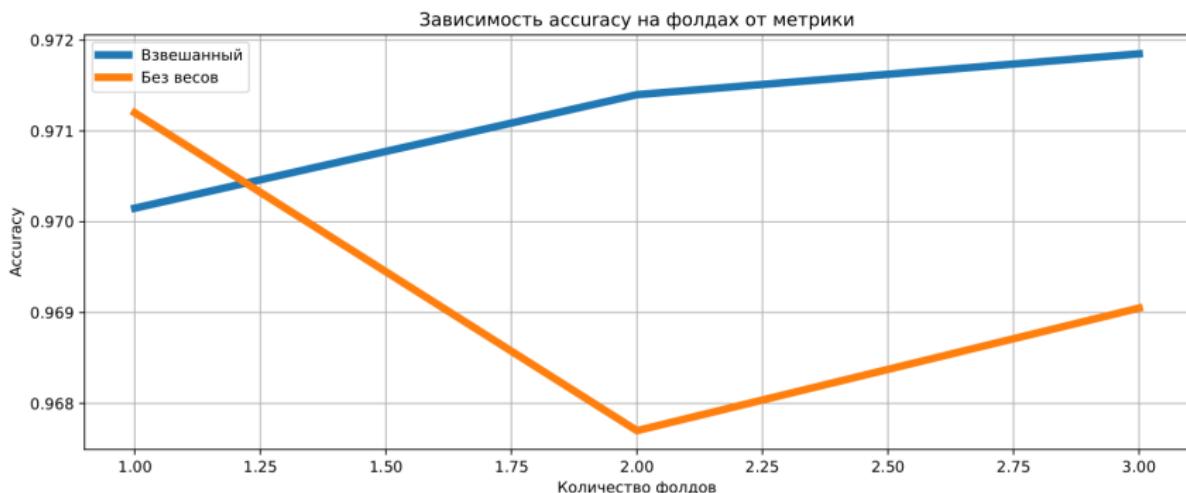
Думайте об оформлении результатов



Что плохо на этой картинке?



Думайте об оформлении результатов



Что плохо на этой картинке? Почти всё:

- ▶ Нет отношения порядка на оси x, опечатки, слишком подробная шкала.



Примеры плохо организованных страниц

2. Теоритическая часть

1.

$$F(w) = \frac{1}{l} \sum_{i=1}^l \log(1 + \exp(-y_i w^T x_i)) + \lambda \frac{\|w\|^2}{2}$$

$$\nabla F(w) = \frac{1}{l} \sum_{i=1}^l \nabla \mathcal{L}(M_i(w)) + \frac{\lambda}{2} \nabla \|w\|^2$$

$$d\mathcal{L} = d(\log(1 + e^{-M})) = \frac{de^{-M}}{1+e^{-M}} = \frac{-e^{-M}}{1+e^{-M}} dM = \frac{1}{e^{M+1}} y_i x_i^T dw \Rightarrow$$

$$\nabla \mathcal{L} = \frac{-y_i x_i}{1+e^{-M}} \Rightarrow \nabla F(w) = -\frac{1}{l} \sum_{i=1}^l \frac{y_i x_i}{1+e^{-M_i}}, \text{ где } M_i = y_i w^T x_i$$

2.

$$\nabla F(X, w)_j = \frac{\partial}{\partial w_j} \left[-\frac{1}{l} \sum_{i=1}^l \log \frac{[y_i=j] e^{(w_j \cdot x_i)}}{\sum_{k=1}^K e^{(w_k \cdot x_i)}} + \frac{\lambda}{2} \sum_{k=1}^K \|w_k\|^2 \right].$$

Преобразуем первую сумму:

$$\frac{\partial}{\partial w_j} \left[\sum_{i=1}^l (w_j \cdot x_i) [y_i = j] - \log \sum_{k=1}^K e^{(w_k \cdot x_i)} \right] =$$

$$\sum_{i=1}^l x_i [y_i = j] - \sum_{i=1}^l \frac{e^{(w_j \cdot x_i)}}{\sum_{k=1}^K e^{(w_k \cdot x_i)}} x_i \Rightarrow$$

$$\nabla F(X, w)_j = \frac{1}{l} \sum_{i=1}^l \left(\frac{e^{(w_j \cdot x_i)}}{\sum_{k=1}^K e^{(w_k \cdot x_i)}} - [y_i = j] \right) x_i + \lambda w_j$$

3.

$$F(X, w) = -\frac{1}{l} \sum_{i=1}^l \left(\log P(y_i = 1 | x_i) + (\log P(y_i = -1 | x_i)) \right) =$$

$$-\frac{1}{l} \sum_{i=1}^l \left(\log \frac{[y_i=1] e^{(w_1 \cdot x_i)}}{e^{(w_1 \cdot x_i)} + e^{(w_2 \cdot x_i)}} + \log \frac{[y_i=-1] e^{(w_2 \cdot x_i)}}{e^{(w_1 \cdot x_i)} + e^{(w_2 \cdot x_i)}} \right) =$$

$$= \begin{cases} w_1 = 0 \end{cases} = -\frac{1}{l} \sum_{i=1}^l \log \frac{[y_i=-1] e^{(w_2 \cdot x_i)}}{1 + e^{(w_2 \cdot x_i)}} =$$

$$\frac{1}{l} \sum_{i=1}^l \log \left(1 + [y_i = -1] e^{(w_2 \cdot x_i)} \right) = \frac{1}{l} \sum_{i=1}^l \log \left(1 + e^{-y_i w^T x_i} \right)$$

Формулы «съехали» и стали плохо читаться.



Примеры плохо организованных страниц

Это все равно, что манипулировать (текст, логотипы и даже узоры).
 $\text{и } m = 1.$

$$-\sum_{n=1}^N \sum_{k=1}^C \mathbb{I}[y_n = k] \ln p(y_n = k | x_n) =$$

$$= \sum_{n=1}^N \sum_{k=1}^C \mathbb{I}[y_n = k] \left(\ln \left(e^{w_1^T x_n} + \dots + e^{w_N^T x_n} \right) - w_k^T x_n \right) = N \cdot L_w(X, y)$$

Это и есть функция потерь для многоштатной логистической регрессии.
 Тогда

$$\nabla_{w_i} L_w = \frac{1}{N} \sum_{n=1}^N \sum_{k=1}^C \mathbb{I}[y_n = k] \left(\frac{e^{w_i^T x_n}}{e^{w_1^T x_n} + \dots + e^{w_N^T x_n}} - \delta_{ki} \right) x_n$$

где $\delta_{ki} = \mathbb{I}[i = k]$ - единичная Кронекера.

Пункт 3

Напомним, что при количестве классов $= 2$ задача многоштатной логистической регрессии сводится к бинарной логистической регрессии.
 Действительно, при $C = 2$ и $y \in \{-1, +1\}$ имеем

$$L_w(X, y) = \frac{1}{N} \sum_{n=1}^N \left(\mathbb{I}[y_n = 1] \ln \left(1 + e^{(w_1 - w_2)^T x_n} \right) + \mathbb{I}[y_n = -1] \ln \left(1 + e^{(w_1 - w_2)^T x_n} \right) \right) =$$

$$= \frac{1}{N} \sum_{n=1}^N \ln \left(1 + e^{-(w_1 - w_2)^T x_n} \right)$$

где $w = w_1 - w_2$. Получаем функцию потерь для бинарной логистической регрессии.

Эксперименты

Тексты были предоброблены:

- Все склонения, не имеющие букв или цифр, заменены на пробела.
- Все буквы приведены к нижнему регистру.

Затем при помощи `sklearn.feature_extraction.text.CountVectorizer` с параметром `ngram_range` = 0-02 текстовый признак был преобразован в несколько частей.



Примеры плохо организованных страниц

Таблица 9: Результаты экспериментов задачи №8. Время работы измерено в микросекундах

	vectorized_8	non_vectorized_8	my_method_8	multivariate_normal
shape = (50, 100)	940.2	636531.5	42079.1	426.7
shape = (100, 200)	3985.8	4968530.7	386522.0	2636.1
shape = (150, 300)	10492.1	16607857.2	1408267.6	8813.9

Таблица 10: Результаты точности вычисления

	vectorized_8	non_vectorized_8	my_method_8
shape = (50, 100)	2.29e-13	1.29e-12	3.23e-12
shape = (100, 200)	2.44e-13	1.33e-12	3.29e-12
shape = (150, 300)	2.32e-13	1.29e-12	3.25e-12

Есть большие пустые пространства на странице.



Примеры плохо организованных страниц

Заметим, что стандартная реализация `scipy.stats.multivariate_normal` отсутствует от `vector_method`:

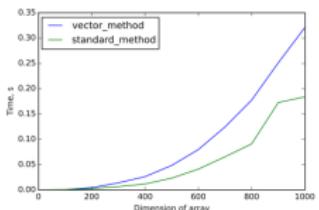


Рис. 6

Погрешность была вычислена как единица нормы от разности двух функций:

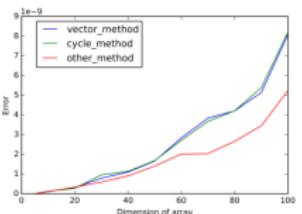


Рис. 7. Погрешность

Графики занимают слишком много места.

Примеры плохо организованных страниц

Измерение времени

Время выполнения функций проверялось на квадратных матрицах X размера $n \times n$ сгенерированных из равномерного распределения и векторах i, j размера n сгенерированных из дискретного равномерного распределения. Результаты (Рис. 2), как и в задаче 1, показывают, что стандартные циклы в Python работают медленнее чем функции и методы, реализованные в библиотеке numpy. Индексация в numpy массивах работает медленнее чем метод take, но разница небольшая. В отличии от первой задачи, разница во времени выполнения между реализациями с ростом размерности данных не изменяется.

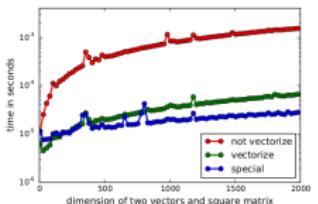


Рис. 2: Зависимость времени выполнения задачи 2 от размерности данных

Есть большие пустые пространства на странице.

Примеры плохо организованных страниц

10 Задача 8

Условие

Реализовать функцию вычисления логарифма плотности многомерного нормального распределения

Входные параметры: точки X , размер (N, D) , мат. ожидание m , вектор длины D , матрица ковариаций C , размер (D, D) . Сравнить с $scipy.stats.multivariate_normal(m, C).logpdf(X)$ как по скорости работы, так и по точности вычислений.

Решение 1. Векторизованное

Формула плотности невырожденного нормального распределения выполнена для всей матрицы X .

```
1 def v1_vector(X, m, C):  
2     n = m.shape[0]  
3     ans = -(n/2.0)*np.log(2*np.pi) - 0.5*np.linalg.slogdet(C)[1]  
4     ans -= 0.5*np.dot(np.dot((X-m), np.linalg.inv(C)), (X-m).T)  
5     return np.diag(ans)
```



Пример хорошо организованных страниц

Layer name	Layer size	18-layer	34-layer	56-layer	101-layer	152-layer
conv1_3x3	1157,112					
conv1_3x3	56 × 56	$[3 \times 3, 64]$ $[3 \times 3, 64]$				
conv2_3x3	28 × 28	$[3 \times 3, 128]$ $[3 \times 3, 128]$				
conv3_3x3	14 × 14	$[3 \times 3, 256]$ $[3 \times 3, 256]$				
conv4_3x3	7 × 7	$[3 \times 3, 512]$ $[3 \times 3, 512]$				
conv5_3x3	1 × 1	$[3 \times 3, 512]$ $[3 \times 3, 512]$				
FC	1000	1.8×10^6	3.6×10^6	3.6×10^6	7.8×10^6	11.1×10^6

Table 1. Architectures for ImageNet. Building blocks are shown in brackets (see also Fig. 5), with the numbers of blocks stacked. Down-sampling is performed by conv3_3, conv4_3, and conv5_3 with a stride of 2.

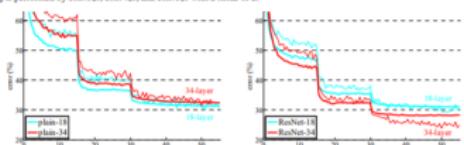


Figure 4. Training on ImageNet. Thin curves denote training error, and bold curves denote validation error of the center crops. Left: plain networks of 18 and 34 layers. Right: ResNets of 18 and 34 layers. In this plot, the residual networks have no extra parameter compared to their plain counterparts.

	plain	ResNet
18 layers	27.94	27.88
34 layers	28.54	25.03

Table 2. Top-1 error (%) (10-crop testing) on ImageNet validation. Here the ResNets have no extra parameter compared to their plain counterparts. Fig. 4 shows the training procedures.

34-layer plain net has higher training error throughout the whole training procedure, even though the solution space of the 18-layer plain network is a subspace of that of the 34-layer one.

We argue that this optimization difficulty is unlikely to be caused by vanishing gradients. These plain networks are trained with BN [16], which ensures forward propagated signals to have non-zero variances. We also verify that the backward propagated gradients exhibit healthy norms with BN. So neither forward nor backward signals vanish. In fact, we can see that the plain network has better generalization accuracy (Table 3), suggesting that the solver works to some extent. We conjecture that the deep plain nets may have exponentially low convergence rates, which impact the

reducing of the training error³. The reason for such optimization difficulties will be studied in the future.

Residual Networks. Next we evaluate 18-layer and 34-layer residual nets (ResNets). The basic architecture matches that of the plain networks. A residual connection is added to each pair of 3 × 3 filters in Fig. 3 (right). In the first comparison (Table 2 and Fig. 4 right), we use identity mapping for all shortcuts and zero-padding for increasing dimensions (option A). So they have no extra parameter compared to the plain counterparts.

We have three major observations from Table 2 and Fig. 4. First, the situation is reversed with residual learning – the 34-layer ResNet is better than the 18-layer ResNet (by 2.8%). More importantly, the 34-layer ResNet exhibits considerably lower training error and is generalizable to the validation data. This indicates that the degradation problem is well addressed in this setting and we manage to obtain accuracy gains from increased depth.

Second, compared to its plain counterpart, the 34-layer

³We have experimented with many training iterations ($\geq 10^4$) and still observed the degradation problem, suggesting that this problem cannot be feasibly addressed by simply using more iteration.

2 THE GUMBEL-SOFTMAX DISTRIBUTION

We begin by defining the Gumbel-Softmax distribution, a continuous distribution over the simplex that can approximate samples from a categorical distribution. Let z be a categorical variable with class probabilities $\pi_1, \pi_2, \dots, \pi_k$. For the remainder of this paper we assume categorical samples are encoded as k -dimensional one-hot vectors lying on the corners of the $(k - 1)$ -dimensional simplex, Δ^{k-1} . This allows us to define quantities such as the element-wise mean $\mathbb{E}_{\pi}[z] = [\pi_1, \dots, \pi_k]$ of these vectors.

The Gumbel-Max trick (Gumbel, 1954; Maddison et al., 2014) provides a simple and efficient way to draw samples z from a categorical distribution with class probabilities π :

$$z = \text{one_hot}\left(\arg\max_i[g_i + \log \pi_i]\right) \quad (1)$$

where g_1, \dots, g_k are i.i.d samples drawn from $\text{Gumbel}(0, 1)^4$. We use the softmax function as a continuous, differentiable approximation to $\arg\max$, and generate k -dimensional sample vectors $y \in \Delta^{k-1}$ where

$$y_i = \frac{\exp((\log(g_i) + g_i)/\tau)}{\sum_{j=1}^k \exp((\log(g_j) + g_j)/\tau)} \quad \text{for } i = 1, \dots, k. \quad (2)$$

The density of the Gumbel-Softmax distribution (derived in Appendix B) is:

$$p_{\pi, \tau}(y_1, \dots, y_k) = \Gamma(k)\tau^{k-1} \left(\sum_{i=1}^k \pi_i/y_i^{\tau} \right)^{-k} \prod_{i=1}^k (\pi_i/y_i^{\tau+1}) \quad (3)$$

This distribution was independently discovered by Maddison et al. (2016), where it is referred to as the concrete distribution. As the softmax temperature τ approaches 0, samples from the Gumbel-Softmax distribution become one-hot and the Gumbel-Softmax distribution becomes identical to the categorical distribution $p(z)$.

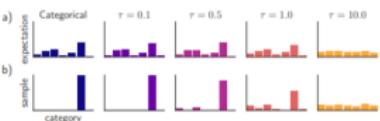


Figure 1: The Gumbel-Softmax distribution interpolates between discrete one-hot-encoded categorical distributions and continuous categorical densities. (a) For low temperatures ($\tau = 0.1, \tau = 0.5$), the expected value of a Gumbel-Softmax random variable approaches the expected value of a categorical random variable with the same logits. As the temperature increases ($\tau = 1.0, \tau = 10.0$), the expected value converges to a uniform distribution over the categories. (b) Samples from Gumbel-Softmax distributions are identical to samples from a categorical distribution as $\tau \rightarrow 0$. At higher temperatures, Gumbel-Softmax samples are no longer one-hot, and become uniform as $\tau \rightarrow \infty$.

2.1 GUMBEL-SOFTMAX ESTIMATOR

The Gumbel-Softmax distribution is smooth for $\tau > 0$, and therefore has a well-defined gradient $\partial y/\partial \pi$ with respect to the parameters π . Thus, by replacing categorical samples with Gumbel-Softmax samples we can use backpropagation to compute gradients (see Section 3.1). We denote

⁴The $\text{Gumbel}(0, 1)$ distribution can be sampled using inverse transform sampling by drawing $u \sim \text{Uniform}(0, 1)$ and computing $y = -\log(-\log(u))$.



Все числа указываются с необходимым числом знаков

угол (градусы)	fold-1	fold-2	fold-3
0	0.97475	0.9734	0.97375131
5	0.9808	0.9807	0.98160092
10	0.9809	0.98095	0.98170091
15	0.97965	0.979	0.97855107

Таблица 2: accuracy для разных значений углов поворота

сдвиг (пиксели)	fold-1	fold-2	fold-3
0	0.96175191	0.95740213	0.96340183
1	0.97090145	0.96520174	0.96895155
2	0.96700165	0.96235188	0.96460177
3	0.96565172	0.96295185	0.96355182

Число указывается с точностью до 2 или 3 знака после запятой.



Избегайте длинных предложений

Далее в опытах, для настройки нашего метода, подбора оптимальных параметров, нам придётся использовать кросс-валидацию. Но эта операция занимает очень много времени, поэтому так как все стратегии, перечисленные выше, являются точными, то есть одинаково классифицируют один объект, имея одинаковую обучающую выборку, нам нужно измерить время их работы, и определить лучшую стратегию. Результат опыта на ниже приведнном рисунке.



Чего ещё следует избегать?

- ▶ Ненаучной лексики («результаты модели получились фиговые»)
- ▶ Орфографических ошибок (установите проверку в среде)
- ▶ Грамматических, синтаксических и других ошибок
- ▶ Повествование от первого лица единственного числа
- ▶ Обращений к читателю («вашему вниманию представлены результаты экспериментов»>)
- ▶ Смешения стиля использования буквы «ё» (либо везде используете «ё», либо везде «е»)



Итог: элементы хорошего отчёта по заданию

- ▶ Отчёт подготовлен в системе \TeX ;
- ▶ Объём отчёта: 5–20 страниц;
- ▶ Текст отчёта не повторяет полной формулировки задания;
- ▶ Структура отчёта соответствует пунктам задания;
- ▶ Используются векторные шрифты;
- ▶ Графики оформлены надлежащим образом;
- ▶ Шкала для графиков выбрана правильно;
- ▶ На разных графиках результаты для одинаковых методов отображаются одним и тем же цветом;



Итог: элементы хорошего отчёта по заданию

- ▶ Между расположением графиков и местами их упоминания в тексте относительно небольшое расстояние (на той же или на соседней странице);
- ▶ На страницах не должно быть много пустого места;
- ▶ В большинстве случаев графики/таблицы/псевдокоды алгоритмов не должны занимать большей части одной страницы отчёта;
- ▶ Все числа в тексте/таблицах указаны с необходимым числом значащих цифр;
- ▶ В большинстве случаев в отчёте не должно быть никакого кода;
- ▶ Для всех экспериментов описан выбранный дизайн экспериментов, а также сделаны выводы из полученных результатов;



Система \TeX

\TeX — система компьютерной вёрстки, построенная по принципу компиляции документа, записанного с помощью специального языка разметки.

Изобретена Д. Кнутом в конце 70х годов.

Является де-факто стандартом для написания научных статей.

Язык разметки \TeX используется для набора формул во многих системах: в вики-разметке, в matplotlib, Microsoft Office и др.

\LaTeX — надстройка поверх \TeX .



Дистрибутивы и редакторы **TeX**

Дистрибутивы:

- ▶ Windows: MiKTeX, TeX Live
- ▶ Linux: TeX Live
- ▶ Mac OS: MacTeX, TeX Live

Редакторы: WinEdt, TeXnicCenter, Kile, TeXmaker, TeXstudio . . .



Дистрибутивы и редакторы \TeX

Рекомендуемые редакторы:

- ▶ VSCode + Latex Workshow. Инструкция по установке
- ▶ TeXstudio
- ▶ Overleaf – облачный редактор. Туториал

Режимы компиляции

- ▶ tex → dvi → ps → pdf
- ▶ tex → dvi → pdf
- ▶ tex → pdf

Структура файла .tex

```
\documentclass{article} % класс документа
%Пreamble документа
\usepackage[T2A]{fontenc} % задаём кодировку шрифтов
\usepackage[utf8x]{inputenc} % задаём кодировку файла
% задаём правила переносов для русского языка
\usepackage[english,russian]{babel}
```

% Текст документа

```
\begin{document}
```

Некоторый текст в первом абзаце.

Несколько пробелов подряд считаются как один.

Конец абзаца задаётся задаётся пустой строкой.

```
\end{document}
```



Разделы и подразделы

```
\section{Раздел 1}
\subsection{Подраздел}
\subsubsection{Подподраздел}

\section{Раздел 2}
```



Окружения списков

```
\begin{itemize}
    \item Пункт
    \item Ещё один пункт
\end{itemize}
```

```
\begin{enumerate}
    \item Пункт 1
    \item Пункт 2
\end{enumerate}
```

- ▶ Пункт
 - ▶ Ещё один пункт
- 1. Пункт 1
 - 2. Пункт 2



Формулы

Формула в тексте: $\sin^2 x + \cos^2 x = 1$.

Формула в тексте: $\sin^2 x + \cos^2 x = 1$.

Выносная формула:

\$\$

$A_{ij} = b_i^2 + c_j^3 \quad \forall i, j = 1, \dots, n.$

\$\$

Выносная формула:

$$A_{ij} = b_i^2 + c_j^3 \quad \forall i, j = 1, \dots, n.$$



Формулы

Выравнивание по левому краю: `\[x^n + y^n = z^n \]`

Выравнивание по левому краю:

$$x^n + y^n = z^n$$

```
\begin{equation}
    \sin^2 x + \cos^2 x = 1 \label{eq:trig_eq}
\end{equation}
```

Ссылка по метке: `\ref{eq:trig_eq}`

$$\sin^2 x + \cos^2 x = 1 \tag{1}$$

Ссылка по метке: 1



Формулы

Выравнивание формул:

```
\begin{align}
\notag p(y|x) = \frac{p(x|y)p(y)}{p(x)} &= \\
\notag \label{eq:bayes} &= \frac{1}{Z}p(x|y)p(y)
\end{align}
```

Выравнивание формул:

$$\begin{aligned} p(y|x) &= \frac{p(x|y)p(y)}{p(x)} = \\ &= \frac{1}{Z}p(x|y)p(y) \end{aligned} \tag{2}$$



Ссылки

```
\begin{equation}
\label{eq::1}
E = mc^2.
\end{equation}
```

Ссылка на формулу~`\eqref{eq::1}`

$$E = mc^2. \tag{3}$$

Ссылка на формулу (3)

```
\section{Ссылки}\label{sec:1}
```

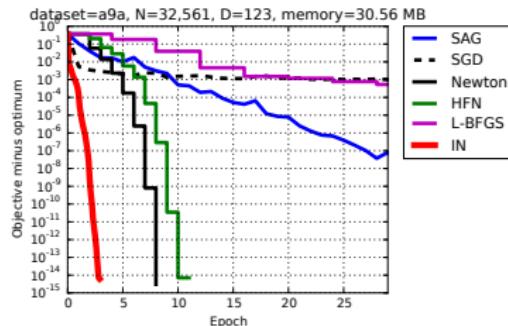
Ссылка на раздел~`\ref{sec:1}` в документе.



Картинки

Картинка в тексте:

```
\includegraphics[width=5cm]{a9a_epoch.pdf}
```



Картинка в тексте:



Таблица

Таблица:

```
\begin{tabular}{|cc|c}
  a & b & c \\
  d & e & f \\
\end{tabular}
```

Таблица:

a	b	c
d	e	f



Расположение картинок на странице

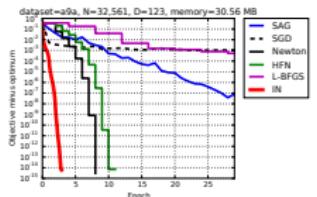
```
\begin{figure}[h] %Разместить таблицу здесь
\begin{center}
\includegraphics[width=5cm]{a9a_epoch}
\end{center}
\caption{Картишка}\label{fig::1}
\end{figure}
```

Ссылка на картинку: рис.[~]\ref{fig::1}

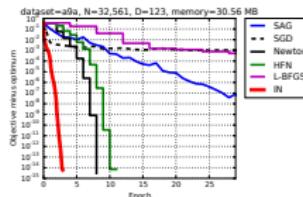


Несколько картинок на странице

```
\tabcolsep = 20pt %длина разделителя между колонками
\begin{tabular}{cc}
\includegraphics[width=5cm]{a9a_epoch}
& \\
\includegraphics[width=5cm]{a9a_epoch} \\
(a) & (b)
\end{tabular}
```



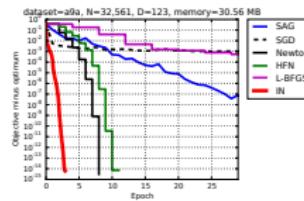
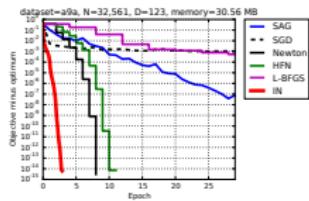
(a)



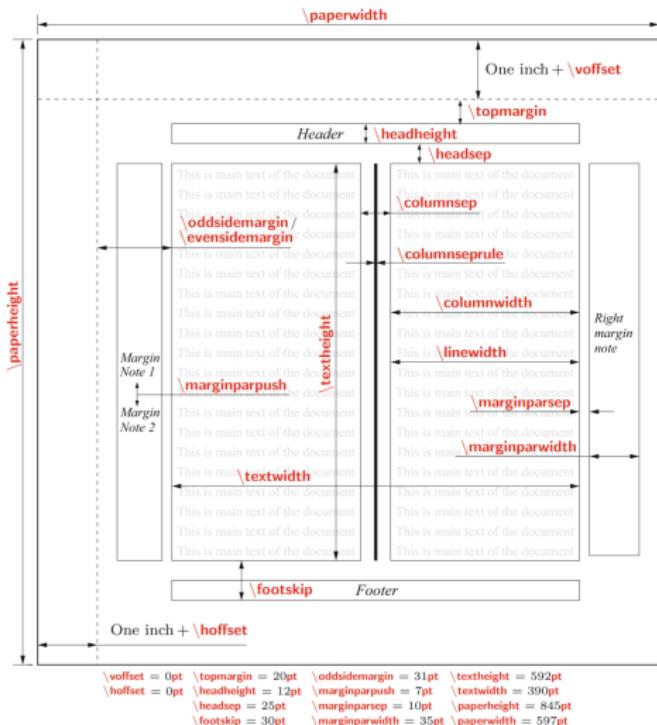
(b)

Несколько картинок на странице

```
\begin{figure}[h]
    \begin{minipage}[b]{0.45\textwidth}\centering
        \includegraphics[height=2cm]{a9a_epoch}
    \end{minipage}
    \begin{minipage}[b]{0.45\textwidth}\centering
        \includegraphics[height=2cm]{a9a_epoch}
    \end{minipage}
\end{figure}
```



Управление отступами на странице



Несколько команд для управлением текстом

SN	Type	Variety	Command	Declaration
1	Family	Serif family (default)	<code>\textrm{atext}</code> or <code>\rm atext</code>	<code>\rmfamily</code>
		Sans serif family	<code>\textsf{atext}</code> or <code>\sf atext</code>	<code>\sffamily</code>
		Typewriter family	<code>\texttt{atext}</code> or <code>\tt atext</code>	<code>\ttfamily</code>
2	Series	Medium series (default)	<code>\textmd{atext}</code>	<code>\mdseries</code>
		Boldface series	<code>\textbf{atext}</code> or <code>\bf atext</code>	<code>\bfseries</code>
3	Shape	Upright shape (default)	<code>\textup{atext}</code>	<code>\upshape</code>
		<i>Italic shape</i>	<code>\textit{atext}</code> or <code>\it atext</code>	<code>\itshape</code>
4	Shape	Slanted shape	<code>\textsl{atext}</code> or <code>\sl atext</code>	<code>\slshape</code>
		CAPS & SMALL CAPS SHAPE	<code>\textsc{atext}</code> or <code>\sc atext</code>	<code>\scshape</code>
4	Size	Emphasized shape	<code>\textemph{atext}</code> or <code>\em atext</code>	—
		Tiny size	<code>\tiny atext</code>	<code>\tiny</code>
		Script size	<code>\scriptsize atext</code>	<code>\scriptsize</code>
		Foot note size	<code>\footnotesize atext</code>	<code>\footnotesize</code>
		Small size	<code>\small atext</code>	<code>\small</code>
		Normal size (default)	—	<code>\normalsize</code>
		Large size	<code>\large atext</code>	<code>\large</code>
		Larger size	<code>\Large atext</code>	<code>\Large</code>
		Largest size	<code>\LARGE atext</code>	<code>\LARGE</code>
		Huge size	<code>\huge atext</code>	<code>\huge</code>
		Hugest size	<code>\Huge atext</code>	<code>\Huge</code>



Несколько команд для управлением текстом

Font type	Command	Package required	Output
Serif family	<code>\mathrm{ABC abc}</code>	—	ABCabc
Italic shape	<code>\mathit{ABC abc}</code>	—	<i>ABCabc</i>
Boldface series	<code>\mathbf{ABC abc}</code>	—	ABCabc
Sans serif family	<code>\mathsf{ABC abc}</code>	—	ABCabc
Typewriter family	<code>\mathtt{ABC abc}</code>	—	ABCabc
Mathematical boldface	<code>\boldmath{ABC abc}</code>	amssymb	ABCabc
Mathematical normal	<code>\mathnormal{ABC abc}</code>	—	ABCabc
Calligraphic	<code>\mathcal{A B C}</code>	—	\mathcal{ABC}
Open	<code>\Bbb{A B C}</code>	amsfonts/ amssymb	ABC
Open	<code>\mathbb{A B C}</code>	amsfonts/ amssymb	ABC
German/ Fraktur	<code>\mathfrak{ABC abc}</code>	eufrak/ amsfonts/ amssymb	\mathfrak{ABCabc}



Особенности типографии: тире и дефис

- ▶ дефисы в словах: из-за δ -функции
дефисы в~словах: из-за $\$ \backslash delta \$$ -функции
- ▶ диапазоны чисел: страницы 3–7
диапазоны чисел: страницы~3--7
- ▶ тире в предложениях: Это — тире.
тире в~предложениях: Это~--- тире.
- ▶ минусы в формулах: $-f(-x) = f(x)$
минусы в~формулах: $\$ -f(-x) = f(x) \$$



Особенности типографии: кавычки

- ▶ Французские «ёлочки»
Французские <<ёлочки>>
- ▶ Немецкие „лапки или 99–66“
Немецкие „, лапки или 99–66“
- ▶ Английские “лапки или 66–99”
Английские “‘лапки или 66–99”
- ▶ Неверно: „нигде так не принято“
Неверно: „, нигде так не принято“
- ▶ Неверно: ”и так тоже никто не делает“
Неверно: ”’и так тоже никто не делает“
- ▶ Неверно: "а это вообще не кавычки"
Неверно: "а это вообще не кавычки"



Список литературы

Ссылка в тексте на публикацию `\cite{vorontsovLX}`.

```
% В конце документа
\section{Список литературы}
\begin{thebibliography}{99}
\bibitem{vorontsovUrl}
```

Воронцов К. В., Полезная информация для
пользователей `\LaTeX`,

```
\url{www.ccas.ru/voron/latex.html}
```

```
\bibitem{vorontsovLX}
```

Воронцов К. В., `\LaTeX` в примерах, 2005,
`\url{www.ccas.ru/voron/download/voron05latex.pdf}`

```
\end{thebibliography}
```



Список литературы

Ссылка в тексте на публикацию `\cite{blei06}`.

% В конце документа

```
\section{Список литературы}
\bibliographystyle{gost780s}
\bibliography{references}
```

В файле `references.bib`:

```
@ARTICLE{blei06,
author = {D. Blei and M. Jordan},
title = {Variational inference for ...},
journal = {Journal of Bayesian Analysis},
year = {2006},
volume = {1},
}
```

BibTeX и русский язык

BibTeX не дружит с кириллицей и utf-8 одновременно!

Способ 1. Сохранить файл с библиографией в cp1251, при запуске предупредить о кодировке
(либо вы счастливый обладатель Windows)

```
\inputencoding{cp1251}
\bibliographystyle{abrv}
\bibliography{liter_cp}
```

BibTeX и русский язык

BibTeX не дружит с кириллицей и utf-8 одновременно!

Способ 2. Использовать при компиляции `bibtexu` вместо `bibtex`
(если вы несчастный обладатель Linux)

Статья[1]

```
\bibliography{liter_utf}
```



. Артём.

Заголовок статьи.

Журнал, 8(3):305–316, Nov. 2014.

Русский язык в списках

Переопределить счётчики списков второго уровня на русские буквы:

```
\renewcommand{\theenumii}{\asbuk{enumii}}
```

1. внешний элемент списка;
2. другой внешний элемент;
 - 2.а внутренний элемент 1
 - 2.б внутренний элемент 2
 - 2.в внутренний элемент 3

verbatim

Окружение `verbatim` — запрещает LaTeX обрабатывать вставленный текст, отображает код как есть

```
def sum(list_of_numbers):  
    my_sum = 0  
    for elem in list_of_numbers:  
        my_sum += elem  
    return my_sum
```

```
\begin{verbatim}  
def sum(list_of_numbers):  
    my_sum = 0  
    for elem in list_of_numbers:  
        my_sum += elem  
    return my_sum  
\end{verbatim}
```

Пакет `listings`

Пакет `listings` — мощный пакет LaTeX , позволяющий настраивать специфическое оформление для кода

```
def sum(list_of_numbers):  
    my_sum = 0  
    for elem in list_of_numbers:  
        my_sum += elem  
    return my_sum
```

```
\begin{lstlisting}  
def sum(list_of_numbers):  
    my_sum = 0  
    ...  
\end{lstlisting}
```

Пакет `listings`

```
\usepackage{listings}
\usepackage{color}

\lstdefinestyle{myLatexStyle}{
basicstyle=\small\ttfamily,
language={python},
numbersep=5mm, numbers=left, numberstyle=\tiny,
breaklines=true, frame=single, framexleftmargin=8mm,
xleftmargin=8mm, backgroundcolor=\color{green!5},
frameround=fttt, escapeinside=??, rulecolor=\color{red},
morekeywords={reduce},
keywordstyle=\color[rgb]{0,0,1},
commentstyle=\color[rgb]{0.133,0.545,0.133},
stringstyle=\color[rgb]{0.627,0.126,0.941}
}
\lstset{style=myLatexStyle}
```

Пакет `minted`

Пакет `minted` — пакет LaTeX, позволяющий настраивать оформление кода

Плюс `minted` — большое число предустановленных тем

```
def sum(list_of_numbers):  
    my_sum = 0  
    for elem in list_of_numbers:  
        my_sum += elem  
    return my_sum
```

```
\begin{minted}[fontsize=\small]{python}  
def sum(list_of_numbers):  
    my_sum = 0  
    ...  
\end{minted}
```

Пакет `minted`

Команда `\mintinline{language}{}{}` для оформления кода внутри связного текста

Команда `\mintinline{latex}{\mintinline{language}{}{}}`
для оформления кода внутри связного текста

Плюсы и минусы различных способов

► `verbatim`

- + Быстро, не требует настройки
- Отсутствие возможностей настройки

► `lstlisting`

- + Огромное количество возможностей
- Красивый результат требует тщательной настройки
- Сложно задавать свои окружения для разных языков

► `minted`

- + Огромное количество возможностей
- + Больше число предустановленных тем
- + Легко задавать окружения для разных языков
- Есть проблемы при установке

При использовании всех этих пакетов объявление слайда
приходится записывать так:

```
\begin{frame}[fragile]\frametitle{Плюсы и минусы  
различных способов}
```

TikZ и PFG

PFG — низкоуровневый пакет для векторной графики в \TeX

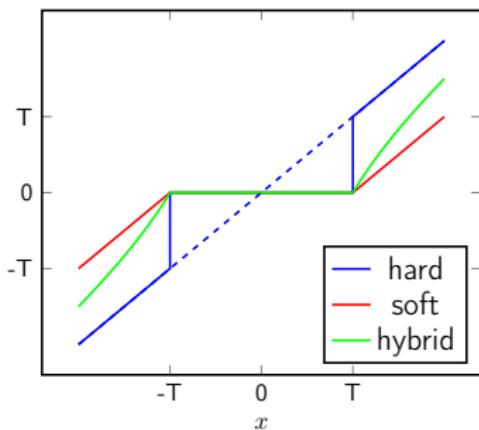
TikZ — высокоуровневое расширение этого пакета

<http://www.texample.net/tikz/> — сайт с примерами работы

<http://www.texample.net/tikz/> — сайт с примерами работы

короткая ссылка — самое подробное руководство

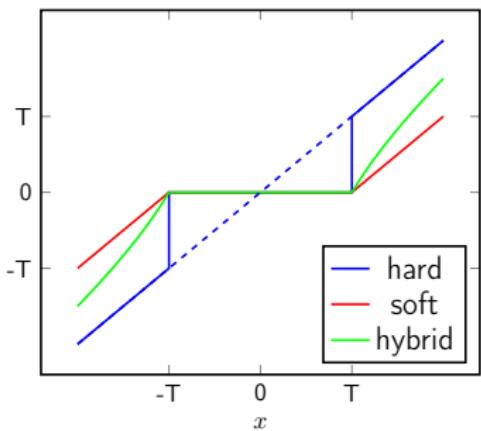
Изображение графиков



```
\begin{tikzpicture}[scale=0.7]
\begin{axis}[
line width = 1pt,
xlabel = {$x$},
xtick={-1, 0, 1},
xticklabels={-T, 0, T},
ytick={-1, 0, 1},
yticklabels={-T, 0, T},
mark=none,
legend entries={hard, soft, hybrid},
legend style={font=\Large,
legend pos=south east}
]
...

```

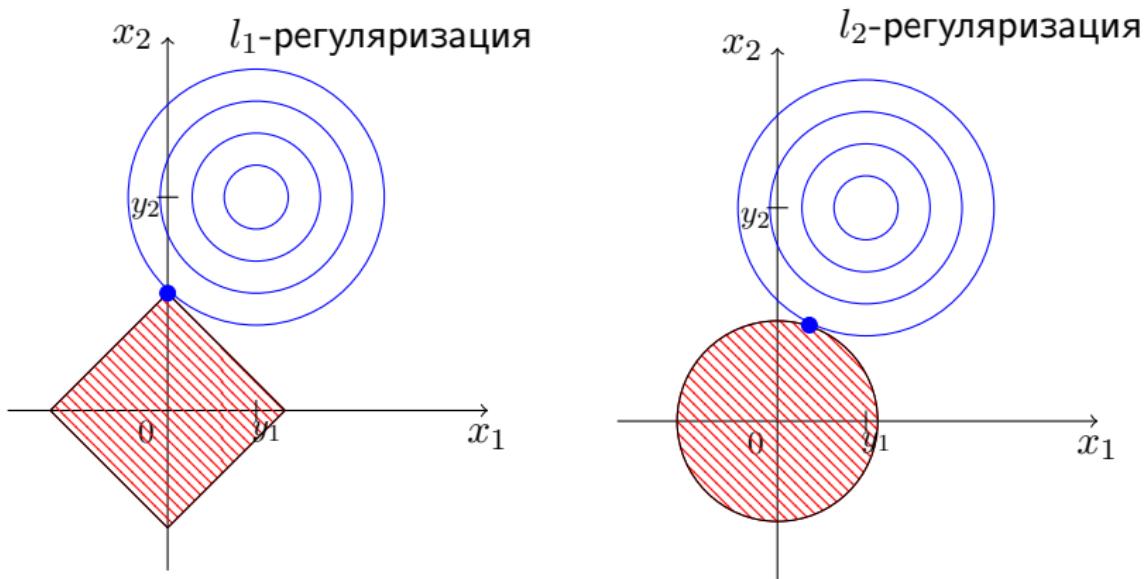
Изображение графиков



```
...
\addplot[blue] coordinates
{(-2, -2) (-1, -1) (-1, 0)
(1, 0) (1, 1) (2, 2)};
\addplot[blue, dashed] coordinates
{(-2, -2) (2, 2)};
\addplot[red] coordinates ...
\addplot[green, domain=-2:-1]
{x - 1/x};
\addplot[green, domain=1:2]
{x - 1/x};
\addplot[green, domain=-1:1]
{0};
\end{axis}
\end{tikzpicture}
```

Изображение линий уровня

Линии уровня для $\|X\vec{w} - \vec{y}\|_2$ и области $\|w\|_1 \leqslant \kappa, \|w\|_2 \leqslant \kappa$:



Изображение линий уровня

```
\begin{tikzpicture }

\draw[->] (-1.5, 0) -- (3, 0) node[anchor=north] {$x_1$};
\draw[->] (0, -1.5) -- (0, 3.5) node[anchor=east] {$x_2$};
\draw (-0.2, -0.2) node[scale=0.8] {$0$};
\draw (2, 3.5 ) node {$l_1$-регуляризация};

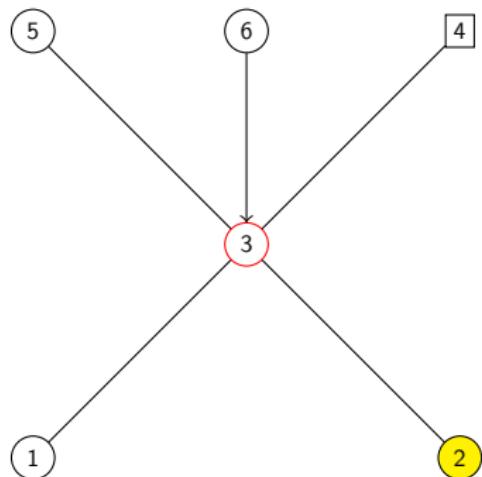
\draw[blue] (0.83, 2) circle (1.2cm);
\draw[blue] (0.83, 2) circle (0.9cm);
\draw[blue] (0.83, 2) circle (0.6cm);
\draw[blue] (0.83, 2) circle (0.3cm);

\draw[red] (0, -1.1) -- (1.1, 0) -- (0, 1.1) --
(-1.1, 0) -- (0, -1.1);
\filldraw[pattern color=red, pattern=north west lines]
(0, -1.1) -- (1.1, 0) -- (0, 1.1) --
(-1.1, 0) -- (0, -1.1);

...
```



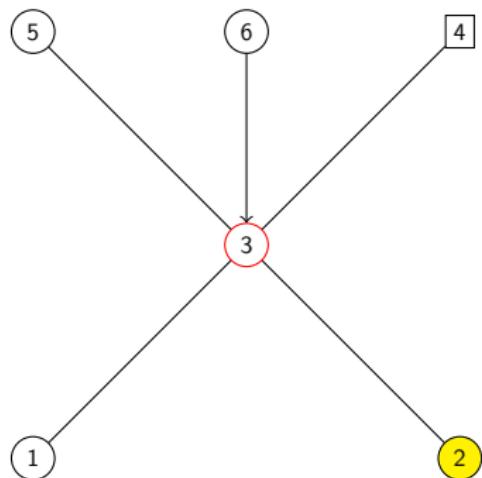
Изображение графа



```
\begin{tikzpicture }[scale=2]
\node (p1) at (0,0)
[scale=0.6,shape=circle,
draw=black,fill=white] {1};
\node (p2) at (2,0) [scale=0.6,
shape=circle,draw=black,
fill=yellow] {2};
\node (p3) at (1,1) [scale=0.6,
shape=circle,draw=red,
fill=white] {3};
\node (p4) at (2,2) [scale=0.6,
shape=rectangle,draw=black,
fill=white] {4};
\node (p5) at (0,2) [scale=0.6,
shape=circle,draw=black,
fill=white] {5};
...

```

Изображение графа

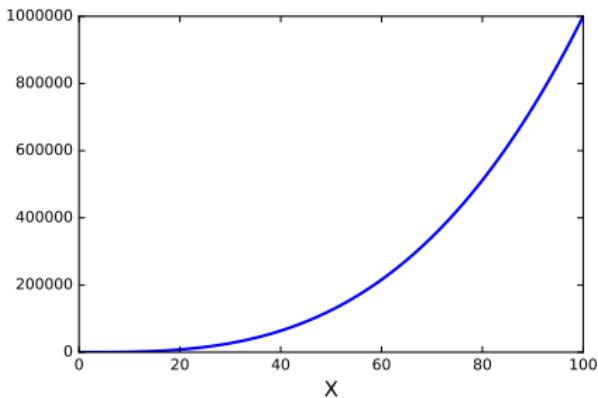


```
...
\node (p6) at (1,2) [scale=0.6,
shape=circle,draw=black,
fill=white] {6};

\draw (p1) -- (p3) -- (p5);
\draw (p2) -- (p3) -- (p4);
\draw[->] (p6) -- (p3);
```

Сохранение графиков экспериментов

Проблема: провели эксперимент, сохранили график, но...



забыли подписать ось y .

Интеграция Python и TikZ

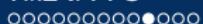
Пакет `matplotlib2tikz` позволяет сохранять графики в формате `TikZ`.

```
import matplotlib.pyplot as plt
from matplotlib2tikz import save as tikz_save

x = [i for i in range(0, 101, 1)]
y = [x_el ** 3 for x_el in x]

plt.plot(x, y, linewidth=2)
plt.xlabel('X', fontsize=14)

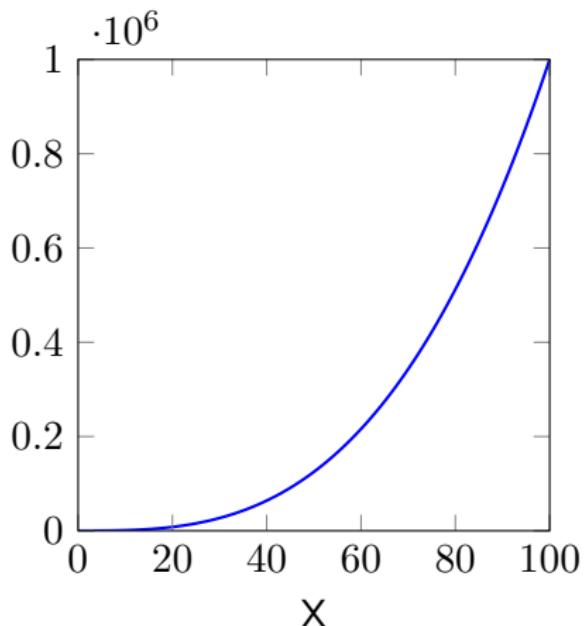
tikz_save('tmp_tikz.txt')
```



Результат сохранения картинки

```
% This file was created by matplotlib2tikz v0.6.13.  
\begin{tikzpicture}  
\begin{axis}[  
 xlabel={X},  
 xmin=0, xmax=100,  
 ymin=0, ymax=1000000,  
 axis on top,  
 tick pos=both  
]  
\addplot [thick, blue, forget plot]  
table {  
 0 0  
 1 1  
 2 8  
 ...  
}
```

Вставка картинки в формате .tikz



```
\newlength\figureheight  
\newlength\figurewidth  
\setlength\figureheight{6cm}  
\setlength\figurewidth{6cm}  
\input{tmp_tikz.txt}
```

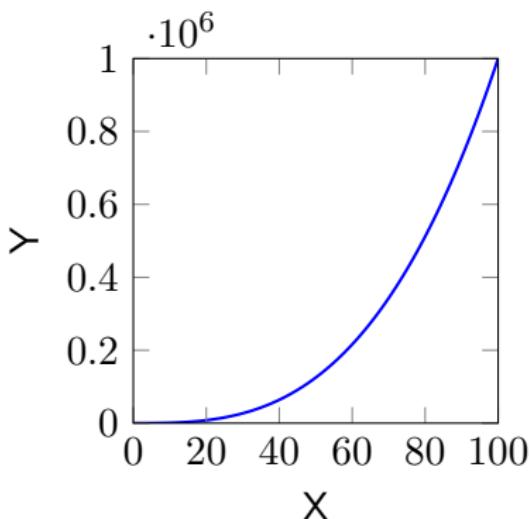
Что это нам даёт?

Вставка картинки в формате .tikz

Картина сохранена в удобном текстовом представлении

Добавим название оси y отредактировав текстовый файл:

```
...
\begin{axis}[
 xlabel={X},
 ylabel={Y},
 ...
```



Полезные ссылки

-  Львовский С. М. Набор и вёрстка в системе \LaTeX . 2003.
<http://www.ptep-online.com/ctan/llang2003.pdf>
-  Воронцов К. В. \LaTeX в примерах, 2005,
<http://www.ccas.ru/voron/download/voron05latex.pdf>
-  Написание отчётов и статей (рекомендации), ссылка.
-  Балдин Е.М. LATEX, GNU/Linux и русский стиль, ссылка.
-  Dilip Datta. LaTeX in 24 Hours, ссылка.
-  Шпаргалка по частоиспользуемым математическим символам в \LaTeX , ссылка.