Линейные модели: жатые чувства, SVM (начнем)

И. Куралёнок, Н. Поваров

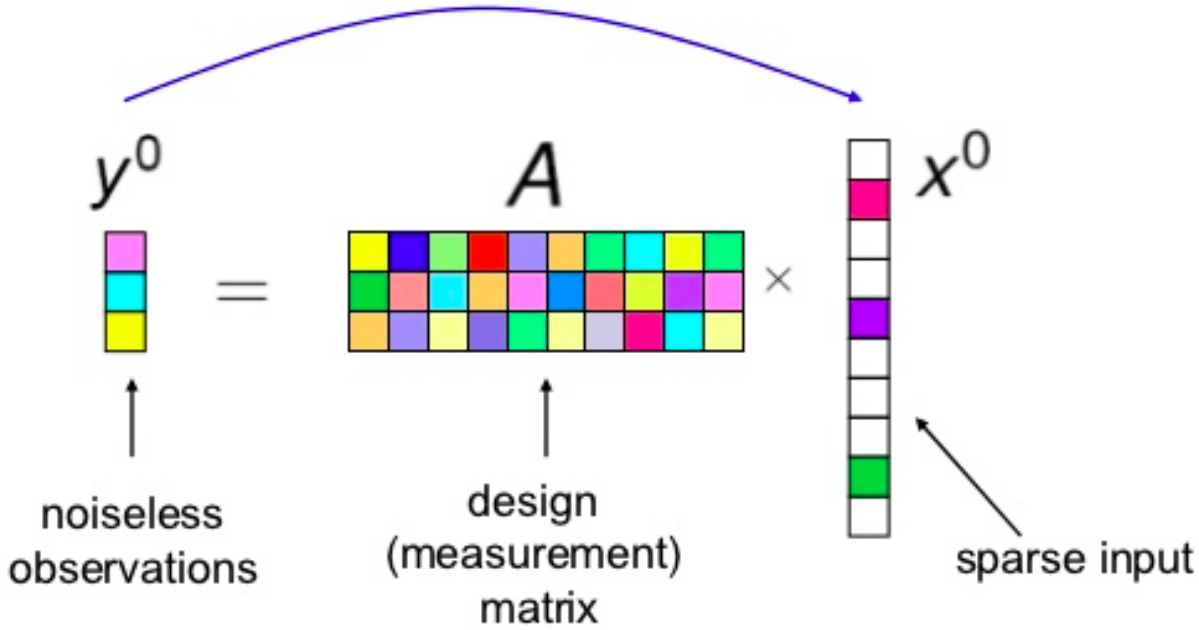
Яндекс

СПб, 2013

Пример

Сергей Юрьевич любит смотреть телевизор и рассуждать. Есть мнение, что в основном по телевизору "льют воду". Надо понять как часто надо обращать внимание на то, что проходит на экране, чтобы не упустить "нить".

Пример: постановка задачи



- Хотим восстановить сигнал x^0
- Хотим как можно меньше “смотреть в телевизор”
- Матрица A в данном случате — то, что происходит в голове Сергея Юрьевича

Картинка из Tutorial ICML2010 by Irina Rish & Genady Grabarnik

Сюрприз compressed sensing

$$y = X\beta + \epsilon$$

Если компоненты матрицы X независимые, одинаково распределенные, нормальные, то β можно восстановить точно с большой вероятностью:

- из $O(klog(\frac{m}{k}))$ измерений;
- решив оптимизацию

$$\begin{aligned} &\arg\min_\beta \|\beta\|_1 \\ &\|y - X\beta\| < \epsilon \end{aligned}$$

$\Rightarrow$ где-то мы уже такое видели

Линейная регрессия vs. восстановление сигнала

- Решают одну и ту же задачу
- Одни и те же алгоритмы
- Учиться сложнее:
  - нету влияния на построение матрицы X;
  - в частности нет гарантий на свойства матрицы X;
  - наличие в β большого количество нулей – лишь наше предположение.

Постановка в терминах RFP I

Нам надо перевести произвольный сигнал (в нашем примере он же во времени!) в линейную комбинацию:

$$\hat{x_\omega} = \sum_{\omega \in \mathbb{Z}_n} x_j e^{\frac{-2\pi i \omega j}{n}}$$
$$\mathcal{F}x = \hat{x}$$

После того как мы все посчитаем в терминах $\hat{x_\omega}$, восстановить телевизор мы сможем по IDFT ($\mathcal{F}^{-1} = \frac{1}{n}\mathcal{F}^*$).

Постановка в терминах RFP II

$$\arg\min_\beta \|\beta\|_1$$
$$y = X\beta$$

В новых обозначениях:

$$\arg\min_\beta \|\beta\|_1$$
$$(\mathcal{F}\beta)_k = (\mathcal{F}x)_k, k \in \Omega$$

Теорема о качестве восстановленного сигнала для RFP

Theorem (Candes et al. (2006))

$x \in \mathbb{C}^n : \{i \in \mathbb{Z}_n | x_i \neq 0\} = T \subset \mathbb{Z}_n$

$\Omega \subset \mathbb{Z}_n$ — *одно из равновероятных множеств размера* n

зафиксируем точность B

$\Rightarrow$ *с вероятностью* $P \geq 1 - O(n^{-B})$ *мы можем точно восстановить* x, *если:*

$$\|\Omega\| \geq C_B' |T| \log m$$

где $C_B' \simeq 23(B+1)$

Выводы из теоремы

- Теорема рассказывает о свойствах случайной DFT проекции
- Загаданный вектор x может быть восстановлен:
  - с высокой вероятностью
  - используя LASSO
  - количество наблюдений пропорционально количеству ненулей в “загаданном” сигнале

Упрощение рандома

В теореме Ω равномерно распределена по всем множествам размера n. Такое сложно генерировать. Значительно проще $\Omega' : \forall j \in \mathbb{Z}_n, P(j \in \Omega) = \tau$.
$\Rightarrow$ Для таких проекций вероятность восстановить сигнал примерно такая же.

Стабильно ли решение?

Интересны два вида “стабильности”:

стабильность: маленькие изменения в решении при малом изменении в наблюдениях (изменения в загаданном);

робастность: устойчивость к шуму в данных (неточно померяли отлик x).

Если мы уже решили проблему построения T, то решение стабильно:

$$\hat{\beta} = (\mathcal{F}^*_{T,\Omega}\mathcal{F}_{T,\Omega})\mathcal{F}^*_{T,\Omega}y$$

Из доказательства теоремы о восстановлении сигнала $\mathcal{F}^*_{T,\Omega}\mathcal{F}_{T,\Omega} > \delta E$ с высокой вероятностью при условии на Ω. А вот с робастностью все сложнее...

А что же с произвольно построенным X?

Пока Сергей Юрьевич получал закодированный в Фурье сигнал и раскодировал его обратным Фурье. А что, если кодировани и раскодирование сигнала происходит как-то иначе. Положим, что так:

$$(\Phi\beta)_\Omega = (\Psi x)_\Omega$$

В данном случае мы верим в то, что $dim(\Phi) = dim(\Psi)$, более того, будем рассматривать ортонормированные Φ, Ψ

Когерентность базисов

Definition

Для пары ортонормальных базисов назовем

$$\mu(\Phi, \Psi) = \sqrt{n} \max_{i,j} |\phi_i, \psi_j|$$

когерентностью.

- Заметим, что $1 \leq \mu(\Phi, \Psi) \leq \sqrt{n}$
- В случае Фурье получается экстремально хороший случай: $\mu(DFT, IDFT) = 1$

Теорема о качестве восстановленного сигнала для произвольных базисов

Theorem (Candes and Romberg (2006))

Для фиксированной $\delta > 0$ *и* $x \in \mathbb{R}^n$, $|\{i|x_i \neq 0\}| < S$. *Выберем* Ω *точек для наблюдения равномерно из* $\mathbb{Z}_n$ *без повторений. Если*

$$|\Omega| \geq C\mu^2(\Phi, \Psi)S \log \frac{n}{\delta}$$

тогда решение LASSO:

$$\begin{aligned} &\arg\min_{\beta \in \mathbb{R}^n} \|\beta\|_1 \\ &(\Phi\beta)_\Omega = (\Psi x)_\Omega \end{aligned}$$

восстановит x *с вероятностью* $1 - \delta$

Что дальше?

1. Вводим ограничение на модельную матрицу (Restricted Isomenry Property), которое позволяет перейти от условий на модуль образа к условиям на модуль исходного вектора
2. В введенных условиях получаем ограничение на робастность в рамках восстановления сигнала
3. Переходим от когерентности к условиям на собственные числа модельной матрицы

В итоге получается, что (LASSO persistency theorem, Bickel et al., 2009):

$$\|\hat{\beta} - \beta^*\| \leq O\left(\sqrt{\frac{\log n}{m}}\right)$$

Что мы узнали про CS

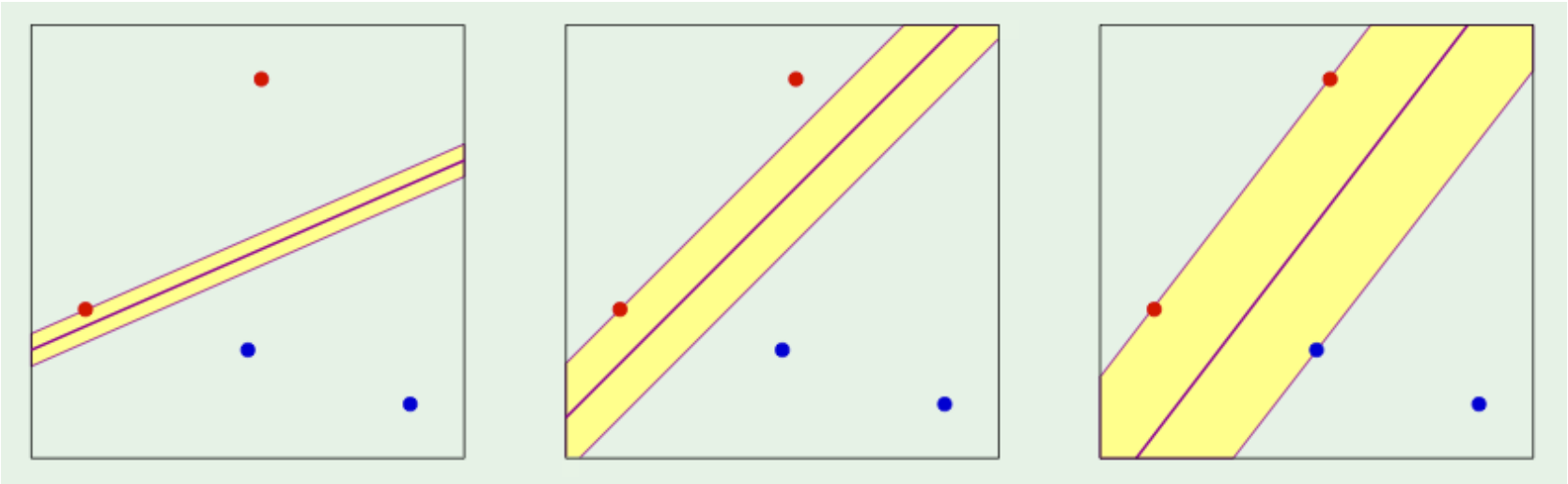
1. Можно ставить задачу по восстановлению сигнала
2. Для решения задачи нам понадобится рандомно выбирать точки наблюдения
3. Оказывается, что решать подобные задачи нужно тем же самым LASSO
4. Эффективность решения зависит от того, как построить “язык передачи информации”
5. Одним из самых хороших универсальных языков (с минимально возможной когерентностью) является DFT/IDFT
6. С помощью механизма CS можно доказать устойчивость решения LASSO

SVM(воспоминания о былом)

- Последний из линейных методов, который мы рассмотрим подробно.
- Rocket science до конца 90-х, по крайней мере в задачах классификации.

SVM на пальцах

- Максимальный зазор.
- Нелинейные преобразования.



Мысли вслух

- Почему большой зазор это хорошо?
- Какая β максимизирует зазор?

Найдем ширину “зазора”: геометрия

Есть две параллельные плоскости:

$$\begin{cases} \beta^T x = a \\ \beta^T x = b \end{cases}$$

проведем прямую, перпендикулярную этой плоскости: $y = \|\beta\| \frac{\beta}{\|\beta\|} t$. Пересечет она наши плоскости вот так:

$$\begin{cases} \beta^T (\|\beta\| \frac{\beta}{\|\beta\|} t_a) = a \\ \beta^T (\|\beta\| \frac{\beta}{\|\beta\|} t_b) = b \end{cases}$$

$$\begin{cases} t_a = \frac{a}{\|\beta\|} \\ t_b = \frac{b}{\|\beta\|} \end{cases}$$

тогда расстояние по полученной прямой: $|t_a - t_b| = \frac{|a-b|}{\|\beta\|}$

Найдем ширину "зазора": мат. анализ

Решим оптимизацией:

$$\min \frac{1}{2}\|x-y\|^2$$

$$\begin{cases} \beta^T x = a \\ \beta^T y = b \end{cases}$$

Перейдем к коэффициентам Лагранжа:

$$\min \frac{1}{2}\|x-y\|^2 + \lambda_1(\beta^T x - a) + \lambda_2(\beta^T y - b)$$

Найдем нули производных по всем переменным:

$$\begin{cases} \beta^T x = a \\ \beta^T y = b \\ x - y + \lambda_1 \beta = 0 \\ x - y + \lambda_2 \beta = 0 \end{cases} \quad \begin{cases} \beta^T (x-y) = a - b \\ \lambda_1 = \lambda_2 \\ \|\beta\|\lambda_1 = b - a \end{cases} \quad \begin{cases} \lambda_1 = \lambda_2 = \frac{b-a}{\|\beta\|^2} \\ x - y = \frac{b-a}{\|\beta\|^2}\|\beta\|\left(\frac{\beta}{\|\beta\|}\right) \end{cases}$$

Возвращаясь к SVM

Теперь мы знаем что оптимизировать. Отнормируем разделяющие плоскости так:

$$\begin{cases} \beta^T x = b - 1 \\ \beta^T x = b + 1 \end{cases}$$

В этих терминах нас $|a - b|$ фиксированы и оптимизировать мы будем только β:

$$\arg\min \frac{\|\beta\|}{2}$$

Вот в таких условиях ($y_i \in \{-1, 1\}$):

$$y_i(\beta^T x_i - b) \geq 1$$

По методу Лагранжа

По теореме Куна-Таккера:

$$\mathcal{L} = \frac{1}{2}\|\beta\|^2 - \sum_{i=1}^{m} \lambda_i (y_i(\beta x_i - \beta_0) - 1), \lambda_i \geq 0$$

$$\begin{cases} -\mathcal{L} = -\sum_{i=1}^{m} \lambda_i + \frac{1}{2}\sum_{i=1}^{m}\sum_{j=1}^{m} \lambda_i \lambda_j y_i y_j (x_i x_j) \\ \lambda_i \geq 0 \\ \sum_{i=1}^{m} \lambda_i y_i = 0 \end{cases}$$

Тогда:

$$\beta = \sum_{i=1}^{m} \lambda_i y_i x_i$$
$$\beta_0 = \beta x_i - y_i, \lambda_i > 0$$

Чем стало легче?

- Адовые условия сменились простым $\lambda_i > 0$
- У нас получился квадрат количества точек
- Интересны только (x_i, x_j) с которыми мы можем играться (kernel trick)!

Результаты ДЗ четвёртой недели

1. ca876
2. 3fc89
3. e46c8
4. 76a61
5. 165f4
6. 729da
7. cd90b
8. cdb5c
9. 23449
10. c8b18
11. 5660e
12. 257d3
13. 2431e
14. 346a9
15. 6f1ba
16. ab851
17. 3ebe0
18. 49dd1
19. 1938b

Интерпретация результатов

Задача была придумать несколько таргетов.

- 1 место - очень круто;
- 2 место - почти очень круто;
- 3-4 места - знают, что такое целевая функция, помнят про "бесконечные потери";
- 5-10 места - знают, что такое целевая функция, но местами забыли про "бесконечные потери";
- 11-12 места - знают, что такое целевая функция, но забыли про "бесконечные потери"совсем;
- 13-19 места - перепутали целевую функцию с решающей, а особо отличившиеся с факторами.

Домашнее задание

- датасет тот же;
- сегодня узнали про новые методы - будем применять;
- howto.txt;
- дедлайн - 29 ноября.