Сборник теоретических задач по машинному обучению

Кантор Виктор, Анастасьев Даниил, Сандрикова Мария, Даниляк Александр

11 января 2016 г.

Оглавление

Зачем нужна эта книга и как ей пользоваться

Чтобы на практике делать меньше глупостей и понимать поведение алгоритмов в тех или иных ситуациях, нужно хорошо разобраться в теоретических вопросах. Для этого и бывают полезны теоретические задачи. Не все из задач в этом сборнике имеют непосредственное отношение к тому, что понадобится на практике (хотя многие все же имеют), но все достаточно познавательны.

Задачи будут полезны и если вы захотите заниматься какими-то теоретическими вопросами машинного обучения, например разрабатывать новые методы или адаптировать существующие под какие-то особые ситуации.

Многие задачи в сборнике включают в себя вывод выражений, используемых в различных алгоритмах (например 8.1, 10.1 и 11.1). Приведенные решения некоторых таких задач в значительной степени повторяют выкладки из лекций К.В. Воронцова по машинному обучению (в частности, 4.1, 4.2, 10.1, одно из решений в 7.2 и первая часть 8.1).

Читатель волен сам выбирать, как и зачем использовать эту книгу, но если основная цель — лучше понять, как устроены и как ведут себя различные методы машинного обучения, то последовательность действий при изучении каждой темы может быть следующей:

1. Изучить материал по теме в неплохом для первого знакомства источнике (лекции курса Воронцова хорошо подойдут, на них часто будут ссылаться задачи этого сборника). Вывод основных утверждений сначала достаточно разобрать в общих чертах.
2. Попробовать решать задачи из сборника, в случае затруднений — обращаться к решениям. Желательно читать решение ровно настолько, чтобы стало понятно, что делать дальше — а затем завершать его самостоятельно.
3. Разобрать в полной мере вывод основных алгоритмов, из рассчета на то, что нужно быть готовым объяснить метод со всеми выкладками другому человеку. Если к теме есть «листок» в — он может быть полезен на этом этапе.
4. Поэкспериментировать с изложением методов «с чистого листа», пока не начнет получаться.
5. Подумать о преимуществах и недостатках алгоритмов, о том какие из них в каких случаях может быть целесообразно применять, о том, как они будут себя вести (и почему так) при изменении различных параметров. Попробовать придумать вопросы или задачи по мотивам изученного теоретического материала и самостоятельно найти на них ответы.

Неточности и опечатки

Пробовать придумывать более изящные решения и находить ошибки в предложенных — неотъемлемая часть эффективной работы со сборником. О найденных ошибках при желании можно писать на почту *victor.cantor@yandex.ru*.

Совсем критичные ошибки будут исправляться, а менее существенные останутся, чтобы не давать повод черезчур терять бдительность.

Новые редакции и пополнение сборника

Если у вас есть интересная задача или вы хотите предложить стилистические правки — пишите на *victor.cantor@yandex.ru*.

Часть I

Задачи и решения

Глава 1

Метрические методы

1.1 2NN

Может ли в методе k ближайших соседей при $k = 2$ получиться лучший результат, чем при $k = 1$? Отказы от классификации тоже считать ошибками.

Решение

При использовании NN ошибка появится, если ближайший объект будет не из того класса, что классифицируемый (зеленая звездочка на рисунке). Но 2NN эту ошибку не исправит, так как из двух ближайших соседей хотя бы один уже не из того класса, значит, получим либо отказ от классификации (если второй — того класса), либо снова ошибку.

Кроме того, 2NN может ухудшить классификацию, если в 2-окрестность влезет объект не того класса (синяя звездочка).

Таким образом, 2NN не будет лучше 1NN.

1.2 Проклятие размерности

Покажите, что с ростом размерности пространства признаков при равномерном распределении точек в кубе $[0; 1]^d$ вероятность попасть в куб $[0; 0,99]^d$ стремится к нулю. Это одна из иллюстраций проклятия размерностей (dimension curse). Попробуйте придумать или найти еще какую-нибудь иллюстрацию к этому явлению и кратко изложить. В чем по-вашему суть проклятия размерности и какое это имеет значение для задач машинного обучения?

Решение

Вероятность попасть в куб $[0, 0.99]^d$ — это вероятность попасть каждой координатой в отрезок $[0, 0.99]$ (так как координаты распределены независимо):

$$\mathsf{P}([0, 0.99]^d) = \prod_{i=1}^{d} \mathsf{P}([0, 0.99]) = 0.99^d \underset{d\to\infty}{\to} 0.$$

Похожий пример: Рассмотрим единичный интервал $[0, 1]$. 100 равномерно разбросанных точек будет достаточно, чтобы покрыть этот интервал с частотой не менее $0, 01$. Теперь рассмотрим 10-мерный куб. Для достижения той же степени покрытия потребуется уже 10^{20} точек. То есть, по сравнению с одномерным пространством, требуется в 10^{18} раз больше точек.

Суть *проклятья размерности* в том, что с увеличением размерности пространства (соответствующего увеличению числа признаков) размер выборки должен расти экспоненциально для сохранения той же плотности покрытия. Это усложняет вычисления и приводит к необходимости хранить большие данные. Значит, нужно уменьшить размерность пространства для полученной выборки.

1.3 Связь ошибки 1NN и оптимального байесовского классификатора

Утверждается, что метод одного ближайшего соседа асимптотически (при стремлении плотности точек из обучающей выборки к бесконечности) имеет матожидание ошибки не более чем вдвое больше по сравнению с оптимальным байесовским классификатором (который это матожидание минимизирует).

Покажите это, рассмотрев задачу бинарной классификации. Достаточно рассмотреть вероятность ошибки на фиксированном объекте x, т.к. матожидание ошибок на выборке размера V будет просто произведением V на эту вероятность. Байесовский классификатор ошибается на объекте x с вероятностью:

$$E_B = \min\{P(1|x), P(0|x)\}$$

Условные вероятности будем считать непрерывными функциями от $x \in R^m$, чтобы иметь возможность делать предельные переходы. Метод ближайшего соседа ошибается с вероятностью:

$$E_N = P(y \neq y_n)$$

Здесь y - настоящий класс x, а y_n - класс ближайшего соседа x_n к объекту x в предположении, что в обучающей выборке n объектов, равномерно заполняющих пространство.

Докажите исходное утверждение, выписав выражение для E_N (принадлежность к классам 0 и 1 для объектов x и x_n считать независимыми событиями) и осуществив предельный переход по n.

Решение

Байесовский классификатор ошибается с вероятностью $E_B = \min\{\mathsf{P}(1|x); \mathsf{P}(0|x)\} = 1 - \max\limits_{y\in\{0,1\}} \mathsf{P}(y|x)$.

Обозначим $r = \max\limits_{y\in\{0,1\}} \mathsf{P}(y|x)$.

Метод ближайшего соседа ошибается с вероятностью $E_{N,n} = \mathsf{P}(y \neq y_n)$. При n, стремящемся к бесконечности, распределение вероятностей классов для ближайшего соседа x — $\{\mathsf{P}(y_n|x_n)\}_{y_n=0}^1$ — стремится к распределению для x: $\{\mathsf{P}(y|x)\}_{y=0}^1$.

Значит, $E_{N,n} = \Big|\text{независимость принадлежностей классов для } x \text{ и } x_n\Big| = \sum\limits_{y\neq y_n\in\{0,1\}} \mathsf{P}(y|x)\cdot\mathsf{P}(y_n|x_n) \underset{n\to\infty}{\to}$

$\sum\limits_{y\in\{0,1\}} \mathsf{P}(y|x) \cdot (1 - \mathsf{P}(y|x)) = 2r(1-r) \leqslant 2(1-r) = 2E_B.$

Глава 2

Решающие деревья

2.1 Асимптотическая эквивалентность критериев информативности

Покажите асимптотическую эквивалентность энтропийного и статистического критериев информативности.

Решение

Пусть P, N — число своих и чужих во всей выборке, $P+N=l$. p, n — число классифицируемых своими и чужими закономерностью R.

Запишем энтропийный критерий информативности:

$$\begin{gathered}\mathrm{IGain}(p,n) = H(y) - H(y|R) = \\ = h\left(\frac{P}{l}\right) - \frac{p+n}{l}h\left(\frac{p}{p+n}\right) + \frac{l-p-n}{l}h\left(\frac{P-p}{l-p-n}\right) = \\ = \frac{P}{l}\log_2\frac{P}{l} - \frac{N}{l}\log_2\frac{N}{l} + \frac{p}{l}\log_2\frac{p}{p+n} + \\ + \frac{n}{l}\log_2\frac{n}{p+n} + \frac{P-p}{l}\log_2\frac{P-p}{l-p-n} + \frac{N-n}{l}\log_2\frac{N-n}{l-p-n}.\end{gathered}$$

Распишем теперь статистический критерий информативности по формуле Стирлинга:

$$\mathrm{IStat}(p,n) = -\frac{1}{l}\log_2 \frac{C_P^p C_N^n}{C_{P+N}^{p+n}} =$$

$$= -\frac{1}{l}\Bigg(P\log_2 P - P\log_2 e + \frac{1}{2}\log_2(2\pi P) + o(1) -$$

$$-(P-p)\log_2(P-p) + (P-p)\log_2 e - \frac{1}{2}\log_2(2\pi(P-p)) + o(1) -$$

$$-p\log_2 p + p\log_2 e - \frac{1}{2}\log_2(2\pi p) + o(1) +$$

$$+N\log_2 N - N\log_2 e + \frac{1}{2}\log_2(2\pi N) + o(1) -$$

$$-(N-n)\log_2(N-n) + (N-n)\log_2 e - \frac{1}{2}\log_2(2\pi(N-n)) + o(1) -$$

$$-n\log_2 n + n\log_2 e - \frac{1}{2}\log_2(2\pi n) + o(1) -$$

$$-(P+N)\log_2(P+N) + (P+N)\log_2 e - \frac{1}{2}\log_2(2\pi(P+N)) + o(1) +$$

$$+(P+N-p-n)\log_2(P+N-p-n) - (P+N-p-n)\log_2 e + \frac{1}{2}\log_2(2\pi(P+N-p-n)) + o(1) -$$

$$+(p+n)\log_2(p+n) - (p+n)\log_2 e + \frac{1}{2}\log_2(2\pi(p+n)) + o(1)\Bigg) =$$

$$= \Big| P+N=l \Big| =$$

$$= -\frac{P}{l}\log_2\frac{P}{l} - \frac{N}{l}\log_2\frac{N}{l} + \frac{p}{l}\log_2\frac{p}{p+n} +$$

$$+\frac{n}{l}\log_2\frac{n}{p+n} + \frac{P-p}{l}\log_2\frac{P-p}{l-p-n} + \frac{N-n}{l}\log_2\frac{N-n}{l-p-n} + o(1).$$

Получаем, что энтропийный и статистический критерии информативности асимптотически эквивалентны.

2.2 Решающее правило в листе

Какая стратегия поведения в листьях решающего дерева приводит к меньшей вероятности ошибки: отвечать тот класс, который преобладает в листе, или отвечать случайно с тем же распределением классов, что и в листе?

Решение

Пусть есть c классов. Пусть в листе за обучение накопилось k объектов: $(k_1, \ldots, k_c)$, где k_i — число объектов i-ого класса, $\sum\limits_i k_i = k$.

Если отвечать случайно с соответствующим распределением классов, то вероятность ошибиться:

$$\mathsf{P}_1(\text{ошибки}) = \sum_{i=1}^{c} \mathsf{P}(\text{ошибки}|i)\,\mathsf{P}(i) = \sum_{i=1}^{c}\left(1 - \frac{k_i}{k}\right)\mathsf{P}(i),$$

где $\mathsf{P}(i)$ — распределение классов.

Если отвечать тем классом, который преобладает, то

$$\mathsf{P}_2(\text{ошибки}) = \sum_{i=1}^{c} \mathsf{P}(\text{ошибки}|i)\,\mathsf{P}(i) = \sum_{i=1}^{c}\left(1 - \frac{k_l}{k}\right)\mathsf{P}(i) = \left(1 - \frac{k_l}{k}\right)\sum_{i=1}^{c}\mathsf{P}(i) = 1 - \frac{k_l}{k},$$

где k_l — самое большое из k_i чисел.

Но $\mathsf{P}_2(\text{ошибки}) = \sum_{i=1}^{c}\left(1-\frac{k_i}{k}\right)\mathsf{P}(i) \geqslant \sum_{i=1}^{c}\left(1-\frac{k_l}{k}\right)\mathsf{P}(i) = 1-\frac{k_l}{k} = \mathsf{P}_1(\text{ошибки})$, значит, если отвечать тем классом, который преобладает, вероятность ошибиться ниже.

2.3 Unsupervised decision tree

Unsupervised решающие деревья можно было бы применить для кластеризации выборки или оценки плотности, но проблема построения таких деревьев заключается в введении меры информативности. В одной статье предлагался следующий подход: давайте использовать тот же Information Gain, а энтропию множества S теперь оценивать по формуле:

$$H(S) = \frac{1}{2}\ln\left((2\pi e)^n|\Sigma|\right)$$

Здесь Σ — оцененная по множеству матрица ковариаций, т.е. не имея других сведений, мы по умолчанию считаем, что скопления точек можно приближенно считать распределенными нормально. Убедитесь, что это выражение в самом деле задает энтропию многомерного нормального распределения.

Решение

Плотность многомерного нормального распределения $f(x) = \frac{1}{(2\pi)^{n/2}|\Sigma|^{1/2}}e^{-\frac{1}{2}(x-\mu)^{\mathrm{T}}\Sigma^{-1}(x-\mu)}$.

Его энтропия:

$$\begin{aligned}
H(f) &= -\int_{R^n}\cdots\int f(x)\ln f(x)dx = \\
&= \int_{R^n}\cdots\int f(x)\left(\frac{1}{2}(x-\mu)^T\Sigma^{-1}(x-\mu) + \ln\left((2\pi)^{n/2}|\Sigma|^{1/2}\right)\right)dx = \\
&= \frac{1}{2}\mathsf{E}\left(\sum_{i,j}(x_i-\mu_i)\left(\Sigma^{-1}\right)_{i,j}(x_j-\mu_j)\right) + \frac{1}{2}\ln\left((2\pi)^n|\Sigma|\right) = \\
&= \frac{1}{2}\sum_{i,j}\left(\mathsf{E}((x_i-\mu_i)(x_j-\mu_j))\left(\Sigma^{-1}\right)_{i,j}\right) + \frac{1}{2}\ln\left((2\pi)^n|\Sigma|\right) = \\
&= \frac{1}{2}\sum_i\sum_j(\Sigma)_{i,j}\left(\Sigma^{-1}\right)_{i,j} + \frac{1}{2}\ln\left((2\pi)^n|\Sigma|\right) = \\
&= \frac{1}{2}\sum_i\left(\Sigma\Sigma^{-1}\right)_{i,i} + \frac{1}{2}\ln\left((2\pi)^n|\Sigma|\right) = \\
&= \frac{1}{2}\sum_i(E)_{i,i} + \frac{1}{2}\ln\left((2\pi)^n|\Sigma|\right) = \\
&= \frac{n}{2} + \frac{1}{2}\ln\left((2\pi)^n|\Sigma|\right) = \\
&= \frac{1}{2}\ln\left((2\pi e)^n|\Sigma|\right).
\end{aligned}$$

Глава 3

Линейные классификаторы

3.1 Знакомство с линейным классификатором

1. Как выглядит бинарный линейный классификатор? (Формула для отображения из множества объектов в множество классов.)
2. Что такое отступ алгоритма на объекте? Какие выводы можно сделать из знака отступа?
3. Как классификаторы вида $a(x) = sign(< w, x > -w_0)$ сводят к классификаторам вида $a(x) = sign(< w, x >)$?
4. Как выглядит запись функционала эмпирического риска через отступы? Какое значение он должен принимать для «наилучшего» алгоритма классификации?
5. Если в функционале эмпирического риска всюду написаны строгие неравенства ($M_i < 0$) можете ли вы сразу придумать параметр w для алгоритма классификации $a(x) = sign(< w, x >)$, минимизирующий такой функционал?
6. Запишите функционал аппроксимированного эмпирического риска, если выбрана функция потерь $L(M)$.
7. Что такое функция потерь, зачем она нужна? Как обычно выглядит ее график?
8. В чем практический смысл квадратичной функции потерь? Почему может быть полезна функция потерь, принимающая большие значения для большого положительного отступа?
9. Приведите пример негладких и немонотонный функций потерь.
10. Что такое регуляризация? Какие регуляризаторы вы знаете?
11. Как связаны переобучение и обобщающая способность алгоритма. Как влияет регуляризация на обобщающую способность?
12. Как связаны острые минимумы функционала аппроксимированного эмпирического риска с проблемой переобучения?
13. Что делает регуляризация с аппроксимированным риском как функцией параметров алгоритма?
14. Для какого алгоритма классификации функционал аппроксимированного риска будет принимать большее значение на обучающей выборке: для построенного с регуляризацией или без нее? Почему?
15. Для какого алгоритма классификации функционал риска будет принимать большее значение на тестовой выборке: для построенного с оправдывающей себя регуляризацией или вообще

без нее? Почему?

16. Что представляют собой метрики качества Accuracy, Precision и Recall?
17. Что такое метрика качества AUC и ROC-кривая?
18. Как построить ROC-кривую (нужен алгоритм), если например, у вас есть правильные ответы к домашнему заданию про фамилии и ваши прогнозы?

Решение

1. Общий вид $a(x) = sign(f(x))$, где $f(x)$ - дискриминантная функция. Частный случай $a(x) = sign(\langle w, x\rangle + w_0)$.
2. Общий вид $M_i = y_i f(x_i)$. Частный случай $M_i = y_i(\langle w, x_i\rangle + w_0)$.
   $a(x_i) \neq y_i \Leftrightarrow M_i \leq 0$, то есть неположительный отступ соответствует ошибки классификатора.
3. К x добавляют фиктивный признак $x_0 = 1$, а к вектору весов координату w_0.
4. Функционал эмпирического риска:
$$Q(a, X^l) = \frac{1}{l}\sum_{i=1}^{l}[a(x_i) \neq y_i] = \frac{1}{l}\sum_{i=1}^{l}[M_i \leq 0]$$
   Для «наилучшего» алгоритма классификации он принимает значение 0.
5. $w = 0$
6. $Q(a, X^l) = \frac{1}{l}\sum_{i=1}^{l} L(M)$
7. Функция потерь $L(a, x)$ соответствует величине ошибки алгоритма a на объекте x. Эта функция неотрицательная. В случае линейной классификации, если смотреть на функцию потерь как на функцию от отступа M, то она будет невозрастающей. В большинстве случаев она также будет выпуклой для удобства решения задачи оптимизации.
8. Если она используется в задачах линейной классификации не в «чистом» виде: $L(M) = \max\{0; (1 - M)^2\}$, то она будет штрафовать за большой отрицательный отступ. Если же рассматривать ее в «чистом» виде и помнить знак отступа, то можно еще поощрять большой положительный отступ.
9. Негладкая: $L(M) = [M \leq 0]$ - пороговая функция потерь, немонотонная: $L(M) = (1 - M)^2$ - квадратичная функция потерь.
10. Регуляризатор играет роль штрафа за сложность. В случае линейной классификации это штраф за большой вес, который позволяет избежать нестабильности дискриминантной функции.

    $l1$-регуляризатор: $\tau V(w) = \tau\sum_{i=1}^{n}|w_i|$

    $l2$-регуляризатор: $\tau V(w) = \tau\sum_{i=1}^{n}w_i^2$

    $l0$-регуляризатор: $\tau V(w) = \tau\sum_{i=1}^{n}[w_i \neq 0]$

    Параметр τ задает степень влияния регуляризатора.
11. Обобщающая способность алгорима a оценивается функционалом качества $Q(a(X^l), X^k)$ на двух непересекающихся представительных выборках X^l и X^k. И если алгоритм сильно пере-

обучен, то значение такого функционала качества будет большим. В случае выхода параметров алгоритма за некоторые рамки (например, сильное увеличение веса одного из признаков), может произойти переобучение, а регуляризация как раз помогает с этим справиться.

12. «Уход» из такого минимума даже на немного дает сильное возрастание значения функционала аппроксимированного эмпирического риска, и не дает другим параметрам практически никакого шанса.

13. Увеличивает при приближении или выходе параметров алгоритма за недопустимые границы.

14. С регуляризацией, потому что она увеличивает значение функционала риска.

15. Неоднозначно. Возможно без регуляризации, если она без нее алгоритм сильно переобучался, и соответсвенно на тестовой выборке показывал плохие результаты. Но возможно и с регуляризацией,если переобучение без регуляризации не превосходило вес, вносенный регуляризацией.

16. Пусть стоит задача классификации на два класса +1 и -1. Обозначим TP - количество объектов, для которых правильный ответ +1 и полученный ответ +1, FN - количество объектов, для которых правильный ответ +1 и полученный ответ -1, TN - количество объектов, для которых правильный ответ -1 и полученный ответ -1, FP - количество объектов, для которых правильный ответ -1 и полученный ответ +1, P - количество объектов класса +1, N - количество объектов класса -1 Тогда:
$$Accuracy = \frac{TP + TN}{P + N}$$
$$Precision = \frac{TP}{TP + FP}$$
$$Recall = \frac{TP}{P}$$

17. Воспользуемся обозначениями пункта 16.
Обозначим $FPR = \frac{FP}{N}, TPR = \frac{TP}{P}$. Тогда ROC-кривая это график зависимости $TPR(FPR)$. AUC-метрика - это площадь под ROC-кривой.

18. В результате работы алгоритма получим множество $\{(FPR_i, TPR_i)\}_{i=0}^{l}$, по которому строится ROC-кривая:
    (a) l_+ - количество фамилий в выборке X^l, l_- - количество фамилий в выборке.
    (b) Сортируем выборку X^l по значениям дискриминантной функции $f(x_i, w)$.
    (c) $(FPR_0, TPR_0) = (0, 0)$
    (d) for i in $\{1, \ldots, l\}$:
    (e) if $y_i == -1$:
    (f) $(FPR_i, TPR_i) = (FPR_{i-1} + \frac{1}{l_-}, TPR_{i-1})$
    (g) else:
    (h) $(FPR_i, TPR_i) = (FPR_{i-1}, TPR_{i-1} + \frac{1}{l_+})$

3.2 Вероятностный смысл регуляризаторов

Покажите, что регуляризатор в задаче линейной классификации имеет вероятностный смысл априорного распределения параметров моделей. Какие распределения задают $l1$-регуляризатор и $l2$-регуляризатор?

Решение

Допустим, множество $X \times Y$ является вероятностным пространством, и задана параметрическая модель совместной плотности распределения объектов и классов $p(x, y|w)$. Введем параметрическое семейство априорных распределений $p(w; \gamma)$, где γ — неизвестная и не случайная величина (гиперпараметр).

Тогда будем считать, что выборка может быть порождена каждой из плотностей $p(x, y|w)$ с параметризованной γ вероятностью $p(w; \gamma)$.

Приходим к принципу максимума совместного правдоподобия данных и модели:

$$L_\gamma(w, X^l) = \ln p(X^l, w; \gamma) = \sum_{i=1}^{l} \ln p(x_i, y_i|w) + \underbrace{\ln p(w; \gamma)}_{\text{регуляризатор}} \to \max_w .$$

Вспомним, что этот принцип эквивалентен принципу минимизации аппроксимированного эмпирического риска

$$Q(w; X^l) = \sum_{i=1}^{l} \mathscr{L}(y_i f(x_i, w)) + \underbrace{\gamma V(w)}_{\text{регуляризатор}} \to \min_w,$$

если положить

$$-\ln p(x_i, y_i|w) = \mathscr{L}(y_i f(x_i, w)),$$
$$\ln p(w; \gamma) = \gamma V(w).$$

Таким образом, получаем, что регуляризатор $V(w)$ соответствует параметрическому семейству априорных распределений плотностей $p(w; \gamma)$ — параметров моделей.

ℓ_1-регуляризатор

Пусть $w \in \mathbb{R}^n$ имеет n-мерное распределение Лапласа:

$$p(w; C) = \frac{1}{(2C)^n} \exp\left(-\frac{\|w\|_1}{C}\right), \quad \|w\|_1 = \sum_{j=1}^{n} |w_j|,$$

т.е. все веса независимы, имеют нулевое матожидание и равные дисперсии; C — гиперпараметр.

Логарифмируя, получаем регуляризатор по ℓ_1-норме:

$$-\ln p(w; C) = \frac{1}{C} \sum_{j=1}^{n} |w_j| + \text{const}(w).$$

ℓ_2-регуляризатор

Пусть $w \in \mathbb{R}^n$ имеет n-мерное гауссовское распределение:

$$p(w; \sigma) = \frac{1}{(2\pi\sigma)^{n/2}} \exp\left(-\frac{\|w\|^2}{2\sigma}\right), \quad \|w\|^2 = \sum_{j=1}^{n} w_j^2,$$

т.е. все веса независимы, имеют нулевое матожидание и равные дисперсии σ; σ — гиперпараметр.

Логарифмируя, получаем регуляризатор по ℓ_2-норме:

$$-\ln p(w; \sigma) = \frac{1}{2\sigma}\|w\|^2 + \text{const}(w).$$

Глава 4

SVM

4.1 SVM и максимизация разделяющей полосы

Покажите, как получается условная оптимизационная задача, решаемая в SVM из соображений максимизации разделяющей полосы между классами. Можно отталкиваться от линейно разделимого случая, но итоговое выражение должно быть для общего.

Как эта задача сводится к безусловной задаче оптимизации?

Решение

Рассмотрим задачу классификации на два непересекающихся класса, в которой объекты описываются n-мерными вещественными векторами: $X = \mathbb{R}^n, Y = \{-1, +1\}$.

Будем строить линейный пороговый классификатор:

$$a(x) = \text{sign}\left(\sum_{j=1}^{n} w_j x^j - w_0\right) = \text{sign}\left(\langle w, x\rangle - w_0\right),$$

где $x = (x^1, \ldots, x^n)$ — признаковое описание объекта x, вектор $w = (w^1, \ldots, w^n) \in \mathbb{R}^n$ и скалярный порог $w_0 \in \mathbb{R}$ — параметры алгоритма.

Для начала предположим, что выборка линейна разделима: найдутся w, w_0, задающие разделяющую гиперплоскость $\langle w, x\rangle = w_0$, при которых функционал числа ошибок

$$Q(w, w_0) = \sum_{i=1}^{l} [y_i(\langle w, x_i\rangle - w_0) \leqslant 0] = 0.$$

Найдем, как оптимальнее расположить разделяющую гиперплоскость. Для простоты выполним нормировку параметров алгоритма: домножим w и w_0 на такую константу, что

$$\min_{i=\overline{1,l}} y_i\left(\langle w, x_i\rangle - w_0\right) = 1.$$

Хочется максимизировать ширину разделяющей полосы. Тогда на границе разделяющей полосы будут лежать точки из обучающей выборки: x_- и x_+, принадлежащие соответственно -1 и $+1$ классам. Ширина полосы

$$\left\langle (x_+ - x_-), \frac{w}{\|w\|} \right\rangle = \frac{\langle w, x_+\rangle - \langle w, x_-\rangle}{\|w\|} = \Big| y_i\left(\langle w, x_i\rangle - w_0\right) = 1, i \in \{+, -\} \Big| =$$

$$= \frac{(w_0 + 1) - (w_0 - 1)}{\|w\|} = \frac{2}{\|w\|}.$$

Получаем, что ширина полосы максимальна, когда норма вектора w минимальна. Значит, можно сформулировать следующую задачу оптимизации:

$$\begin{cases} \frac{1}{2}\|w\|^2 \to \min\limits_{w, w_0}; \\ y_i\left(\langle w, x_i\rangle - w_0\right) \geqslant 1, i = 1, \ldots, l. \end{cases}$$

Если работать с линейно неразделимой выборкой, $y_i(\langle w, x_i\rangle - w_0)$ не обязательно будет не меньше 1. Ослабим эти ограничения и введем в минимизируемый функционал штраф за суммарную ошибку:

$$\begin{cases} \frac{1}{2}\|w\|^2 + C\sum\limits_{i=1}^{l}\xi_i \to \min\limits_{w,w_0,\xi}; \\ M_i(w,w_0) = y_i(\langle w, x_i\rangle - w_0) \geqslant 1 - \xi_i, i = 1,\ldots,l; \\ \xi_i \geqslant 0, i = 1,\ldots,l. \end{cases}$$

ξ_i в этих условиях будет показывать величину ошибки на x_i объекте.

Преобразуем условия на ξ_i:

$$\begin{cases} \xi_i \geqslant 1 - M_i(w,w_0) \\ \xi_i \geqslant 0 \end{cases}$$

Значит, минимум ξ_i будет при $\xi_i = (1 - M_i(w,w_0))_+$.

Получаем эквивалентную задачу безусловной оптимизации:

$$\frac{1}{2}\|w\|^2 + C\sum_{i=1}^{l}(1 - M_i(w,w_0))_+ \to \min_{w,w_0}.$$

4.2 Двойственная задача в SVM

Сформулируйте теорему Куна-Таккера. Выпишете двойственную задачу в SVM. Получите квадратичную задачу на двойственные переменные.

Решение

Теорема (Куна-Таккера)**.** *Пусть дана задача условной оптимизации:*

$$\begin{cases} f(x) \to \min\limits_x; \\ g_i(x) \leqslant 0,\ i = 1,\ldots,m; \\ h_j(x) = 0,\ j = 1,\ldots,k. \end{cases}$$

Тогда если x — точка локального минимума, то существуют множители $\mu_i, i = 1,\ldots,m,\ \lambda_j, j = 1,\ldots,k$, что для функции Лагранжа

$$\mathscr{L}(x;\mu,\lambda) = f(x) + \sum_{i=1}^{m}\mu_i g_i(x) + \sum_{j=1}^{k}\lambda_j h_j(x)$$

выполняются условия:

$$\begin{cases} \frac{\partial\mathscr{L}}{\partial x} = 0; & \\ g_i(x) \leqslant 0;\ h_j(x) = 0; & \textit{(исходные ограничения)} \\ \mu_i \geqslant 0; & \textit{(двойственные ограничения)} \\ \mu_i g_i(x) = 0; & \textit{(условие дополняющей нежесткости)} \end{cases}$$

Запишем функцию Лагранжа задачи SVM:

$$\mathscr{L}(w,w_0,\xi;\lambda,\eta) = \frac{1}{2}\|w\|^2 - \sum_{i=1}^{l}\lambda_i(M_i(w,w_0) - 1) - \sum_{i=1}^{l}\xi_i(\lambda_i + \eta_i - C),$$

где λ_i — переменные, двойственные к ограничениям $M_i \geqslant 1 - \xi_i$;
η_i — переменные, двойственные к ограничениям $\xi_i \geqslant 0$.

Согласно теореме Куна-Таккера задача SVM эквивалентна двойственной задаче:

$$\begin{cases} \frac{\partial\mathscr{L}}{\partial w} = 0,\ \frac{\partial\mathscr{L}}{\partial w_0} = 0,\ \frac{\partial\mathscr{L}}{\partial\xi} = 0; \\ \xi_i \geqslant 0,\ \lambda_i \geqslant 0,\ \eta_i \geqslant 0,\ i = 1,\ldots,l; \\ \lambda_i = 0 \text{ либо } M_i(w,w_0) = 1 - \xi_i, i = 1,\ldots,l; \\ \eta_i = 0 \text{ либо } \xi_i = 0, i = 1,\ldots,l; \end{cases}$$

Условие $\dfrac{\partial \mathscr{L}}{\partial \xi_i} = -\lambda_i - \eta_i + C = 0$ дает $\eta_i + \lambda_i = C,\ i = 1, \ldots, l.$

Условие $\dfrac{\partial \mathscr{L}}{\partial w} = w - \sum\limits_{i=1}^{l} \lambda_i y_i x_i = 0$ дает $w = \sum\limits_{i=1}^{l} \lambda_i y_i x_i.$

Условие $\dfrac{\partial \mathscr{L}}{\partial w_0} = -\sum\limits_{i=1}^{l} \lambda_i y_i = 0$ дает $\sum\limits_{i=1}^{l} \lambda_i y_i = 0.$

Значит, в лагранжиане обнуляются все члены, содержащие переменные ξ_i и η_i, и он выражается только через двойственные переменные λ_i:

$$\begin{cases} -\mathscr{L}(\lambda) = -\sum\limits_{i=1}^{l} \lambda_i + \frac{1}{2} \sum\limits_{i=1}^{l} \sum\limits_{j=1}^{l} \lambda_i \lambda_j y_i y_j \langle x_i, x_j \rangle \to \min\limits_{\lambda}; \\ 0 \leqslant \lambda_i \leqslant C,\ i = 1, \ldots, l; \\ \sum\limits_{i=1}^{l} \lambda_i y_i = 0. \end{cases} \tag{4.1}$$

4.3 Kernel trick

Придумайте ядро, которое позволит линейному классификатору с помощью Kernel Trick построить в исходном пространстве признаков разделяющую поверхность $x_1^2 + 2x_2^2 = 3$. Какой будет размерность спрямляющего пространства?

Решение

Возьмем квадратичное ядро: $K(x, y) = \langle x, y \rangle^2 = (x_1 y_1 + x_2 y_2)^2 = x_1^2 y_1^2 + 2x_1 y_1 x_2 y_2 + x_2^2 y_2^2 = \langle (x_1^2, x_2^2, \sqrt{2} x_1 x_2), (y_1^2, y_2^2, \sqrt{2} y_1 y_2) \rangle$.

Получим отображение в спрямляющее пространство $H = \mathbb{R}^3$:

$$\psi : (x_1, x_2) \to (x_1^2, x_2^2, \sqrt{2} x_1 x_2).$$

Тогда линейная поверхность в H будет иметь вид: $\langle (x_1^2, x_2^2, \sqrt{x}_1 x_2), (w_1, w_2, w_3) \rangle + w_0 = w_1 x_1^2 + w_2 x_2^2 + w_3 \sqrt{x}_1 x_2 + w_0 = \Big| w = (1, 2, 0), w_0 = -3 \Big| = x_1^2 + 2x_2^2 - 3 = 0.$

(Вообще-то говоря, w ищется из условия $\dfrac{\partial \mathscr{L}}{\partial w} = w - \sum\limits_{i=1}^{l} \lambda_i y_i \psi(x_i) = 0$ после того, как найдены λ_i в задаче (4.1) со скалярным произведением $K(x_i, x_j)$ вместо $\langle x_i, x_j \rangle$.

w_0 можно найти, подставив в выражение $w_0 = \langle w, \psi(x_i) \rangle - y_i$ произвольный опорный граничный вектор x_i.)

4.4 Размерность спрямляющего пространства

Чему будет равна размерность минимального спрямляющего пространства для ядра $K(w, x) = (< w, x > +1)^2$?

Решение

$$K(x,y) = (\langle x,y\rangle + 1)^2 = \left(\sum_{i=1}^{n} x_i y_i + 1\right)^2 =$$
$$= 1 + 2\sum_{i=1}^{n} x_i y_i + 2\sum_{i\neq j} x_i y_j + \sum_{i=1}^{n} x_i^2 y_i^2 =$$
$$= \langle (1, \sqrt{2}x_1, \ldots, \sqrt{2}x_n, \sqrt{2}x_1x_2, \ldots, \sqrt{2}x_1x_n, \ldots, \sqrt{2}x_nx_{n-1}, x_1^2, \ldots, x_n^2),$$
$$(1, \sqrt{2}y_1, \ldots, \sqrt{2}y_n, \sqrt{2}y_1y_2, \ldots, \sqrt{2}y_1y_n, \ldots, \sqrt{2}y_ny_{n-1}, y_1^2, \ldots, y_n^2)\rangle.$$

Значит, $\dim H = 1 + n + C_n^2 + n$.

4.5 Гауссовское ядро

Покажите (хотя бы на уровне «создания очевидности»), чему будет равна минимальная размерность прямляющего пространства для радиального (гауссовского) ядра.

Решение

Гауссовское ядро: $K_\sigma(x,y) = \exp\left(-\frac{\|x-y\|^2}{2\sigma^2}\right) = \exp\left(-\frac{\|x\|^2 - 2\langle x,y\rangle + \|y\|^2}{2\sigma^2}\right)$.

Тогда

$$\exp\left(-\frac{\|x\|^2 - 2\langle x,y\rangle + \|y\|^2}{2\sigma^2}\right) = \exp\left(-\frac{1}{2\sigma^2}\|x\|^2\right)\exp\left(-\frac{1}{2\sigma^2}\|y\|^2\right)\sum_{k=0}^{\infty}\frac{\langle x,y\rangle^k}{\sigma^{2k}k!}.$$

Как видно из задач про размерность спрямляющего пространства степенного ядра, скалярное произведение, возведенное в степень, увеличивает размерность пространства признаков. Значит, так как k пробегает значения от нуля до бесконечности, размерность полученного пространства будет бесконечной.

Глава 5

Линейная регрессия

5.1 Аналитический и геометрический вывод линейной регрессии

Выведите нормальное уравнение для задачи линейной регрессии аналитически и геометрически, а также покажите, что идея добавления одинаковых ненулевых слагаемых к матрице $F^T F$ и идея добавления $l2$-регуляризации, также называемая гребневой регрессией (ridge regression), приводят к одному и тому же алгоритму.

Решение

Аналитический вывод

Линейная модель регрессии: $g(x, \alpha) = \sum \alpha_j f_j(x)$.

Пусть $F = \Big(f_j(x_i)\Big)_{l \times n}$, $y = (y_i)_l$ — целевой вектор, $\alpha = (\alpha_j)_n$ — вектор параметров.

$$Q = \sum_{i=1}^{l} \left(\sum_{j=1}^{n} \alpha_j f_j(x_i) - y_i \right)^2 = \|F\alpha - y\|^2 \to \min .$$

Необходимое условие минимума:

$$\frac{\partial Q}{\partial \alpha} = \left(\frac{\partial Q}{\partial \alpha_k} \right)_{k=\overline{1,l}} = 2F^{\mathrm{T}}(F\alpha - y) = 0.$$

Значит, $F^{\mathrm{T}} F\alpha = F^{\mathrm{T}} y$ — нормальная система.

Её решение: $\alpha = (F^{\mathrm{T}} F)^{-1} F^{\mathrm{T}} y$.

Геометрический вывод Имеем $F\alpha$ — вектор прогнозов. $F\alpha \in \langle f_1(X^l), \ldots, f_n(X^l) \rangle$. Хотим минимизировать $\|F\alpha - y\|^2$, т.е. хотим получить ближайшую к y точку из пространства столбцов F. Возьмем тогда $F\alpha^*$ — проекция y на $\langle f_1(X^l), \ldots, f_n(X^l) \rangle$. Заметим, что в этом случае $F\alpha^* - y$ ортогональна любому вектору из $\langle f_1(X^l), \ldots, f_n(X^l) \rangle$ (это проекция на ортогональное дополнение к пространству столбцов F):

$$\forall x \in \langle f_1(X^l), \ldots, f_n(X^l) \rangle \quad \langle x, F\alpha^* - y \rangle = 0.$$

Тогда эта разность будет ортогональна и самим столбцам F:

$$F^{\mathrm{T}}(F\alpha^* - y) = 0,$$
$$F^{\mathrm{T}} F\alpha^* = F^{\mathrm{T}} y,$$
$$\alpha^* = (F^{\mathrm{T}} F)^{-1} F^{\mathrm{T}} y.$$

При этом $F\alpha^* = F(F^{\mathrm{T}}F)^{-1}F^{\mathrm{T}}y$ — проекция y на F. Значит, $F(F^{\mathrm{T}}F)^{-1}F^{\mathrm{T}}$ — проекционная матрица.

Добавление одинаковых ненулевых слагаемых к $F^{\mathrm{T}}F$

$$(F^{\mathrm{T}}F + \tau I_n)\alpha = F^{\mathrm{T}}y.$$

$$\alpha = (F^{\mathrm{T}}F + \tau I_n)^{-1}F^{\mathrm{T}}y.$$

Добавление ℓ_2-регуляризатора

$$Q_\tau(\alpha) = \|F\alpha - y\|^2 + \tau\|\alpha\|^2.$$

$$\frac{\partial Q_\tau(\alpha)}{\alpha} = 2F^{\mathrm{T}}(F\alpha - y) + 2\tau\alpha = 2(F^{\mathrm{T}}F + \tau I_n)\alpha - 2F^{\mathrm{T}}y = 0.$$

$$\alpha = (F^{\mathrm{T}}F + \tau I_n)^{-1}F^{\mathrm{T}}y.$$

5.2 $l1$-регуляризация в линейной регрессии

Покажите с помощью теоремы Куна-Таккера, что LASSO Тибширани и $l1$-регуляризация линейной регрессии приводят к построению одного и того же алгоритма.

Решение

Лассо Тибширани:

$$\begin{cases} Q(\alpha) = \|F\alpha - y\|^2 \to \min\limits_{\alpha}; \\ \sum\limits_{j=1}^{n} |\alpha_j| \leqslant \varkappa \;\Leftrightarrow\; \sum\limits_{j=1}^{n} |\alpha_j| - \varkappa \leqslant 0. \end{cases}$$

По теореме Куна-Таккера:

$$\begin{cases} Q(\alpha) = \|F\alpha - y\|^2 + \lambda\left(\sum\limits_{j=1}^{n} |a_j| - \varkappa\right) = \|F\alpha - y\|^2 + \lambda\left(\sum\limits_{j=1}^{n}\right)|a_j| + \text{const} \to \min\limits_{\alpha}; \\ \lambda \geqslant 0; \\ \lambda\left(\sum\limits_{j=1}^{n} |\alpha_j| - \varkappa\right) = 0 \;\Leftrightarrow\; \lambda = 0 \text{ или } \sum\limits_{j=1}^{n} |\alpha_j| = \varkappa. \end{cases}$$

Глава 6

Математическое дополнение

6.1 Дифференцирование по вектору и по матрице

Покажите справедливость следующих выражений (x и b - столбцы, A - матрица):

1. $\frac{\partial}{\partial x}x^2 = 2x$
2. $\frac{\partial}{\partial x}(Ax+b)^T(Ax+b) = 2A^T(Ax+b)$
3. $\frac{\partial}{\partial A}\ln|A| = A^{-1}$

Решение

1 $\dfrac{\partial}{\partial x_j}\mathbf{x}^2 = \dfrac{\partial}{\partial x_j}\sum\limits_{i=1}^{n}(x_i)^2 = 2x_j.$

Тогда производная по вектору: $\dfrac{\partial}{\partial \mathbf{x}}\mathbf{x}^2 = \left(\dfrac{\partial}{\partial x_1}\mathbf{x}^2, \ldots, \dfrac{\partial}{\partial x_n}\mathbf{x}^2\right) = 2\left(x_1, \ldots, x_n\right) = 2\mathbf{x}.$

2 $(A\mathbf{x}+\mathbf{b})^{\mathrm{T}}(A\mathbf{x}+\mathbf{b}) = \sum\limits_{i=1}^{n}\left(\sum\limits_{j=1}^{n}a_{ij}x_j + b_i\right)\left(\sum\limits_{j=1}^{n}a_{ij}x_j + b_i\right).$

$\dfrac{\partial}{\partial x_k}(A\mathbf{x}+\mathbf{b})^{\mathrm{T}}(A\mathbf{x}+\mathbf{b}) = \sum\limits_{i=1}^{n}2a_{ik}\left(\sum\limits_{j=1}^{n}a_{ij}x_j + b_i\right) = 2\sum\limits_{i=1}^{n}a_{ik}(A\mathbf{x}+\mathbf{b})_i = 2(A^{\mathrm{T}}(A\mathbf{x}+\mathbf{b}))_k.$

Значит, $\dfrac{\partial}{\partial \mathbf{x}}(A\mathbf{x}+\mathbf{b})^{\mathrm{T}}(A\mathbf{x}+\mathbf{b}) = A^{\mathrm{T}}(A\mathbf{x}+\mathbf{b}).$

3 $\det A = |A| = \sum\limits_{i=1}^{n}(-1)^{i+j-1}a_{ij}\det A_{ij} = \sum\limits_{i=1}^{n}a_{ij}M_{ij}$, где A_{ij} — матрица, полученная вычеркиванием i-ой строки и j-ого столбца.

$\dfrac{\partial}{\partial a_{ij}}\ln\left(\sum\limits_{i=1}^{n}a_{ij}M_{ij}\right) = \dfrac{M_{ij}}{\sum\limits_{i=1}^{n}a_{ij}M_{ij}}.$

Каждый элемент матрицы A^{-1} выражается через алгебраические дополнения матрицы A как $a_{ij}^{-1} = \frac{1}{|A|}M_{ji}$.

Значит, $\dfrac{\partial}{\partial a_{ij}}\ln\left(\sum\limits_{i=1}^{n}a_{ij}M_{ij}\right) = \dfrac{1}{|A|}|A|a_{ji}^{-1} = a_{ji}^{-1}.$

Тогда $\dfrac{\partial}{\partial A}\ln|A| = (A^{\mathrm{T}})^{-1}.$

Глава 7

Байесовский подход к классификации и регрессии

7.1 Оценка параметров многомерного нормального распределения

Получите оценку максимального правдоподобия для вектора средних и для матрицы ковариаций многомерного нормального распределения. Будет ли оценка матрицы ковариаций несмещенной, если считать вектор средних известным точно? А если вектор средних неизвестен и нужно подставлять его оценку? Обоснуйте ответы и покажите, как сделать смещенные оценки (если они есть) несмещенными.

Решение

Для многомерного нормального распределения плотность $p(x|\mu,\Sigma) = \frac{1}{(2\pi)^{n/2}|\Sigma|^{1/2}} e^{-\frac{1}{2}(x-\mu)^{\mathrm{T}}\Sigma^{-1}(x-\mu)}$.

Тогда

$$L(X^l|\mu,\Sigma) = \ln p(X^l|\mu,\Sigma) = \sum_{i=1}^{l} \ln \frac{1}{\sqrt{(2\pi)^n}\sqrt{|\Sigma|}} \exp -\frac{1}{2}(x_i-\mu)^{\mathrm{T}}\Sigma^{-1}(x_i-\mu) =$$

$$= \sum_{i=1}^{l} \left(-\frac{n}{2}\ln(2\pi) - \frac{1}{2}\ln|\Sigma| - \frac{1}{2}(x_i-\mu)^{\mathrm{T}}\Sigma^{-1}(x_i-\mu) \right).$$

Производная по μ:

$$\frac{\partial}{\partial\mu} L(X^l|\mu,\Sigma) = \frac{\partial}{\partial\mu} \sum_{i=1}^{l} -\frac{1}{2}(x_i-\mu)^{\mathrm{T}}\Sigma^{-1}(x_i-\mu) = \sum_{i=1}^{l} \Sigma^{-1}(x_i-\mu) = 0.$$

Значит, поскольку Σ невырождена

$$\hat{\mu} = \frac{1}{l}\sum_{i=1}^{l} x_i.$$

Найдем $\hat{\Sigma}$.

Пусть $\Lambda = \hat{\Sigma}^{-1}$. Тогда

$$\ln p(X^l|\hat{\mu},\Lambda) = \sum_{i=1}^{l} \left(-\frac{n}{2}\ln(2\pi) - \frac{1}{2}\ln|\Lambda^{-1}| - \frac{1}{2}(x_i-\hat{\mu})^{\mathrm{T}}\Lambda(x_i-\hat{\mu}) \right).$$

Производная по Λ:

$$\frac{\partial}{\partial\Lambda}\ln p(X^l|\hat{\mu},\Lambda) = -\frac{l}{2}\frac{\partial}{\partial\Lambda}\underbrace{\ln|\Lambda^{-1}|}_{=\ln|\Lambda|^{-1}=-\ln|\Lambda|} - \frac{1}{2}\sum_{i=1}^{l}\frac{\partial}{\partial\Lambda}(x_i-\hat{\mu})^{\mathrm{T}}\Lambda(x_i-\hat{\mu}) =$$

$$= \frac{l}{2}\left(\Lambda^{\mathrm{T}}\right)^{-1} - \frac{1}{2}\sum_{i=1}^{l}(x_i-\hat{\mu})(x_i-\hat{\mu})^{\mathrm{T}} = 0.$$

Значит, $\Lambda = \frac{1}{l}\left(\sum_{i=1}^{l}(x_i-\hat{\mu})(x_i-\hat{\mu})^{\mathrm{T}}\right)^{-1}$.

Тогда

$$\hat{\Sigma} = \frac{1}{l}\sum_{i=1}^{l}(x_i-\hat{\mu})(x_i-\hat{\mu})^{\mathrm{T}}.$$

Если вектор средних μ известен, то $\hat{\mu}=\mu$ и

$$\mathsf{E}\hat{\Sigma} = \frac{1}{l}\sum_{i=1}^{l}\mathsf{E}(x_i-\mu)(x_i-\mu)^{\mathrm{T}} = \mathsf{E}(x_1-\mu)(x_1-\mu)^{\mathrm{T}} = \Sigma,$$

т.е. оценка несмещенная.

Если вектор средних неизвестен, то вместо него подставляют его оценку $\hat{\mu}$ и оценка получается смещенной:

$$\begin{aligned}
\mathsf{E}\hat{\Sigma} &= \mathsf{E}\frac{1}{l}\sum_{i=1}^{l}(x_i-\hat{\mu})(x_i-\hat{\mu})^{\mathrm{T}} = \\
&= \mathsf{E}\left(\frac{1}{l}\sum_{i=1}^{l}x_ix_i^{\mathrm{T}} - 2x_i\hat{\mu}^{\mathrm{T}} + \hat{\mu}\hat{\mu}^{\mathrm{T}}\right) = \\
&= \mathsf{E}\left(\frac{1}{l}\sum_{i=1}^{l}x_ix_i^{\mathrm{T}} - \frac{1}{l^2}\sum_{i=1}^{l}\sum_{j=1}^{l}x_ix_j^{\mathrm{T}}\right) = \\
&= \mathsf{E}\left(\frac{1}{l}\sum_{i=1}^{l}x_ix_i^{\mathrm{T}} - \frac{1}{l^2}\sum_{i=1}^{l}x_ix_i^{\mathrm{T}} - \frac{1}{l^2}\sum_{i=1}^{l}\sum_{j\neq i}x_ix_j^{\mathrm{T}}\right) = \\
&= \mathsf{E}xx^{\mathrm{T}} - \frac{1}{l}\mathsf{E}xx^{\mathrm{T}} - \frac{1}{l^2}l(l-1)\mathsf{E}x\mathsf{E}x^{\mathrm{T}} = \\
&= \frac{l-1}{l}(\mathsf{E}xx^{\mathrm{T}} - \mu\mu^{\mathrm{T}}) = \frac{l-1}{l}\Sigma.
\end{aligned}$$

Для учета этого нужно сделать поправку на смещение:

$$\hat{\Sigma} = \frac{1}{l-1}\sum_{i=1}^{l}\mathsf{E}(x_i-\mu)(x_i-\mu)^{\mathrm{T}}.$$

7.2 Минимизация среднего риска

Докажите теорему о минимизации функционала среднего риска в случае $x \in R^n$ (и существования плотности $p(x)$):

$$a(x) == \arg\min_a R(a)$$

Где:

$$a(x) = \arg\min_s \sum_{y\in Y}\lambda_{y\,s}P(y|x)$$

$$R(a) = \int_X \sum_{y \in Y} \lambda_{y\,a(x)} P(y|x) p(x) dx$$

Как изменится теорема в случае дискретного вектора признаков x?

Решение

Пусть $x \in \mathbb{R}^n$. *Функционал среднего риска*:

$$R(a) = \int\limits_X \sum_{y \in Y} \lambda_{ya(x)}\, \mathsf{P}(y|x) p(x) dx.$$

Теорема. *Минимум функционала среднего риска $R(a)$ достигается алгоритмом*

$$a(x) = \arg\min_{s \in Y} \sum_{y \in Y} \lambda_{ys}\, \mathsf{P}(y|x).$$

Доказательство. Заметим, что $p(x) \geqslant 0\ \forall x$.

Пусть $a(x) \in Y$ такой, что $\sum\limits_{y \in Y} \lambda_{ya(x)}\, \mathsf{P}(y|x) \leqslant \sum\limits_{y \in Y} \lambda_{yt}\, \mathsf{P}(y|x) \quad \forall t \in Y\ \forall x$. Тогда

$$\sum_{y \in Y} \lambda_{ya(x)}\, \mathsf{P}(y|x) p(x) \leqslant \sum_{y \in Y} \lambda_{yt}\, \mathsf{P}(y|x) p(x) \quad \forall x \in X,$$

$$\Rightarrow \int\limits_X \sum_{y \in Y} \lambda_{ya(x)}\, \mathsf{P}(y|x) p(x) dx \leqslant \int\limits_X \sum_{y \in Y} \lambda_{yt(x)}\, \mathsf{P}(y|x) p(x) dx,$$

для любой $t(x) \in Y$. Значит, функционал $R(a)$ достигает минимум на $a(x) = \arg\min\limits_{s \in Y} \sum\limits_{y \in Y} \lambda_{ys}\, \mathsf{P}(y|x)$.

□

Другое решение (из лекций Воронцова)

Доказательство.

$$\begin{aligned}
R(a) &= \int\limits_{\{a(x)=t\}} \sum_{y \in Y} \lambda_{yt}\, \mathsf{P}(y|x) p(x) dx + \int\limits_{X \setminus \{a(x)=t\}} \sum_{y \in Y} \lambda_{ya(x)}\, \mathsf{P}(y|x) p(x) dx = \\
&= \sum_{y \in Y} \lambda_{yt} \int\limits_{\{a(x)=t\}} \mathsf{P}(y|x) p(x) dx + \int\limits_{X \setminus \{a(x)=t\}} \sum_{y \in Y} \lambda_{ya(x)}\, \mathsf{P}(y|x) p(x) dx = \\
&= \sum_{y \in Y} \lambda_{yt}\, \mathsf{P}(y|\{a(x)=t\})\, \mathsf{P}(\{a(x)=t\}) + \int\limits_{X \setminus \{a(x)=t\}} \sum_{y \in Y} \lambda_{ya(x)}\, \mathsf{P}(y|x) p(x) dx = \\
&= \Big| \mathsf{P}(y|\{a(x)=t\})\, \mathsf{P}(\{a(x)=t\}) = \mathsf{P}(y, \{a(x)=t\}) = \mathsf{P}(y) - \mathsf{P}(y, \{a(x) \neq t\}) \Big| = \\
&= \sum_{y \in Y} \lambda_{yt}\, \mathsf{P}(y) + \int\limits_{X \setminus \{a(x)=t\}} \sum_{y \in Y} (\lambda_{ya(x)} - \lambda_{yt})\, \mathsf{P}(y|x) p(x) dx = \\
&= \text{const(a)} + \int\limits_{X \setminus \{a(x)=t\}} \sum_{y \in Y} (\lambda_{ya(x)} - \lambda_{yt})\, \mathsf{P}(y|x) p(x) dx.
\end{aligned}$$

С учетом произвольности выбора t получаем, что минимум $R(a)$ будет достигаться при

$$\sum_{y \in Y} \lambda_{ya(x)}\, \mathsf{P}(y|x) \leqslant \sum_{y \in Y} \lambda_{yt}\, \mathsf{P}(y|x) \quad \forall t \in Y\ \forall x \in X.$$

Значит, $a(x) = \arg\min\limits_{s \in Y} \sum\limits_{y \in Y} \lambda_{ys}\, \mathsf{P}(y|x)$ дает минимум функционала $R(a)$.

□

При дискретном векторе признаков x будет $R(a) = \sum\limits_{x\in X}\sum\limits_{y\in Y} \lambda_{ya(x)}\,\mathsf{P}(y|x)\,\mathsf{P}(x)$.

7.3 Нормальный дискриминантный анализ и линейный дискриминант

Рассмотрим задачу бинарной классификации. Обучающая выборка: $X^l = \{x_1, \cdots, x_l\}$, класс объекта x_i будем обозначать $y_i \in Y = \{+1, -1\}$. Априорные вероятности и значимости классов равны: $P_1 = P_2, \lambda_1 = \lambda_2$.

Выпишите байесовский классификатор, получаемый в предположении, что классы имеют многомерную нормальную плотность. Матрицы ковариаций и векторы средних можно оценить их несмещёнными оценками, гипотетически известным вам из курса матстатистики.

Полученный вами метод носит название нормального дискриминантного анализа. Поверхностью какого порядка будет разделяющая поверхность между классами?

Покажите, что если предположить матрицы ковариаций классов равными (и оценивать эту матрицу по всей выборке), получится линейная разделяющая поверхность. Выпишете явно в этом случае выражение для классификатора. Он носит название линейного дискриминанта Фишера (именно его использовал Фишер для демонстрации возможность классификации Ирисов) и может быть получен не только как байесовский классификатор (разумеется, Фишер объяснял свою идею из других соображений).

Решение

$X^l = \{x_1, \ldots, x_l\}, Y = \{+1, -1\}, \mathsf{P}_{+1} = \mathsf{P}_{-1}, \lambda_{+1} = \lambda_{-1}$. Пусть классы имеют многомерную нормальную плотность. Тогда

$$a(x) = \arg\max_{y\in\{-1,+1\}} \lambda_y\,\mathsf{P}_y\,p_y(x) = \arg\max_{y\in\{-1,+1\}} p_y(x) =$$
$$= \arg\max_{y\in\{-1,+1\}} \frac{1}{(2\pi)^{n/2}|\hat{\Sigma}_y|^{1/2}} e^{-\frac{1}{2}(x-\hat{\mu}_y)^{\mathrm{T}}\hat{\Sigma}_y^{-1}(x-\hat{\mu}_y)}.$$

Поверхность, разделяющая классы $+1$ и -1 (т.е. множество точек, где максимум достигается и при $y = -1$, и при $y = +1$), описывается уравнением $\lambda_{+1}\,\mathsf{P}_{+1}\,p_{+1}(x) = \lambda_{-1}\,\mathsf{P}_{-1}\,p_{-1}(x)$, т.е.

$$\ln p_{+1}(x) = \ln p_{-1}(x).$$

В общем случае разделяющая поверхность квадратична, так как

$$\ln p_y(x) = -\frac{n}{2}\ln 2\pi - \frac{1}{2}\ln|\Sigma_y| - \frac{1}{2}(x-\mu_y)^{\mathrm{T}}\Sigma_y^{-1}(x-\mu_y).$$

Если $\Sigma_{-1} = \Sigma_{+1} = \Sigma$, то квадратичные члены сокращаются и уравнение поверхности вырождается в линейную форму:

$$x^{\mathrm{T}}\Sigma^{-1}(\mu_{+1}-\mu_{-1}) - \frac{1}{2}\mu_{+1}^{\mathrm{T}}\Sigma^{-1}\mu_{+1} + \frac{1}{2}\mu_{-1}^{\mathrm{T}}\Sigma^{-1}\mu_{-1} = 0.$$
$$(x - \frac{1}{2}(\mu_{+1}+\mu_{-1}))^{\mathrm{T}}\Sigma^{-1}(\mu_{+1}-\mu_{-1}) = 0.$$

Тогда и классификатор $a(x)$ можно записывать, используя только линейные члены:

$$\begin{aligned} a(x) &= \arg\max_{y\in Y} \ln \lambda_y \mathsf{P}_y\, p_y(x) = \\ &= \arg\max_{y\in Y} \left(\underbrace{-\frac{1}{2}\hat{\mu}_y^{\mathrm{T}}\hat{\Sigma}^{-1}\hat{\mu}_y}_{\beta_y} + x^{\mathrm{T}} \underbrace{\hat{\Sigma}^{-1}\hat{\mu}_y}_{\alpha_y} \right) = \\ &= \arg\max_{y\in Y} (x^{\mathrm{T}}\alpha_y + \beta_y) = \\ &= \operatorname{sign}(\langle \alpha_{+1} - \alpha_{-1}, x\rangle + \beta_{+1} - \beta_{-1}) = \\ &= \operatorname{sign}(\langle w, x\rangle + w_0). \end{aligned}$$

7.4 Геометрический вывод линейного дискриминанта

Рассмотрим предыдущую задачу и построим классификатор из других соображений. Введем матрицу разброса между классами:

$$S_B = (m_1 - m_2)(m_1 - m_2)^T,$$

m_1, m_2 — выборочные средние для объектов из первого и второго классов соответственно.

А также введем матрицу разброса внутри классов:

$$S_W = \sum_{i:y_i=1} (x_i - m_1)(x_i - m_1)^T + \sum_{i:y_i=2} (x_i - m_2)(x_i - m_2)^T$$

Наша задача — найти прямую с направляющим вектором w, вдоль которой классы (если спроецировать объекты классов на эту прямую) можно разделить наилучшим образом из следующих соображений:

$$J(w) = \frac{w^T S_B w}{w^T S_W w} \to \max_{\|w\|=1}$$

a. Получите направляющий вектор прямой. Матрицу S_W считайте обратимой.

b. Докажите, что предложенный функционал $J(w)$ совпадает с $\tilde{J} = \frac{(\tilde{m}_1 - \tilde{m}_2)^2}{s_1^2 + s_2^2}$, где $\tilde{m}_i$ — проекция выборочного среднего объектов i-того класса на прямую, а $\tilde{s}_i = \sum\limits_{k:y_k=i} (w^T x_k - \tilde{m}_i)^2$.

Решение

Пусть $S_B = (m_{+1} - m_{-1})(m_{+1} - m_{-1})^{\mathrm{T}}$, m_{+1}, m_{-1} — выборочные средние для объектов из первого и второго классов соответственно.

$S_W = \sum\limits_{i:y_i=+1} (x_i - m_{+1})(x_i - m_{+1})^{\mathrm{T}} + \sum\limits_{i:y_i=-1} (x_i - m_{-1})(x_i - m_{-1})^{\mathrm{T}}$.

a) Пусть

$$J(w) = \frac{w^{\mathrm{T}} S_B w}{w^{\mathrm{T}} S_W w} \to \max_{\|w\|=1}.$$

Заметим, что $\dfrac{\partial}{\partial w_k} w^{\mathrm{T}} S w = \dfrac{\partial}{\partial w_k} \sum\limits_i w_i \sum\limits_j w_j s_{ji} = \sum\limits_i w_i s_{ki} + \sum\limits_j w_j s_{jk}$.

Тогда

$$\frac{\partial}{\partial w}J(w) = \frac{\frac{\partial}{\partial w}(w^{\mathrm{T}}S_Bw)w^{\mathrm{T}}S_Ww - \frac{\partial}{\partial w}(w^{\mathrm{T}}S_Ww)w^{\mathrm{T}}S_Bw}{(w^{\mathrm{T}}S_Bw)^2} =$$
$$= \frac{(S_B + S_B^{\mathrm{T}})w(w^{\mathrm{T}}S_Ww) - (S_W + S_W^{\mathrm{T}})w(w^{\mathrm{T}}S_Bw)}{(w^{\mathrm{T}}S_Bw)^2} =$$
$$= 2\frac{S_Bw(w^{\mathrm{T}}S_Ww) - S_Ww(w^{\mathrm{T}}S_Bw)}{(w^{\mathrm{T}}S_Bw)^2} = 0.$$

Получаем, что $S_Bw(w^{\mathrm{T}}S_Ww) = S_Ww(w^{\mathrm{T}}S_Bw)$. Пусть минимум $J(w)$ достигается в w_*. Тогда

$$S_Ww_* = \frac{w_*^{\mathrm{T}}S_Ww_*}{w_*^{\mathrm{T}}S_Bw_*}S_Bw_* =$$
$$= \underbrace{\frac{w_*^{\mathrm{T}}S_Ww_*}{w_*^{\mathrm{T}}S_Bw_*}}_{\in\mathbb{R}}(m_{+1} - m_{-1})\underbrace{(m_{+1} - m_{-1})^{\mathrm{T}}w_*}_{\in\mathbb{R}}.$$

Тогда $w_* = CS_W^{-1}(m_{+1} - m_{-1})$. Поскольку w_* — направляющий вектор, можно забить на скалярные множители, значит, $w_* = S_W^{-1}(m_{+1} - m_{-1})$.

b)

$$J(w) = \frac{w^{\mathrm{T}}S_Bw}{w^{\mathrm{T}}S_Ww} = \frac{w^{\mathrm{T}}(m_{+1} - m_{-1})(m_{+1} - m_{-1})^{\mathrm{T}}w}{w^T\left(\sum\limits_{i:y_i=+1}(x_i - m_{+1})(x_i - m_{+1})^{\mathrm{T}} + \sum\limits_{i:y_i=-1}(x_i - m_{-1})(x_i - m_{-1})^{\mathrm{T}}\right)w} =$$
$$= \frac{\left(w^{\mathrm{T}}(m_{+1} - m_{-1})\right)^2}{\sum\limits_{i:y_i=+1}(w^{\mathrm{T}}(x_i - m_{+1}))^2 + \sum\limits_{i:y_i=-1}(w^{\mathrm{T}}(x_i - m_{-1}))^2} =$$
$$= \frac{\left(w^{\mathrm{T}}m_{+1} - w^{\mathrm{T}}m_{-1}\right)^2}{\sum\limits_{i:y_i=+1}(w^{\mathrm{T}}x_i - w^{\mathrm{T}}m_{+1})^2 + \sum\limits_{i:y_i=-1}(w^{\mathrm{T}}x_i - w^{\mathrm{T}}m_{-1})^2} =$$
$$= \frac{\left(\tilde{m}_{+1} - \tilde{m}_{-1}\right)^2}{\sum\limits_{i:y_i=+1}(w^{\mathrm{T}}x_i - \tilde{m}_{+1})^2 + \sum\limits_{i:y_i=-1}(w^{\mathrm{T}}x_i - \tilde{m}_{-1})^2} =$$
$$= \frac{(\tilde{m}_{+1} - \tilde{m}_{-1})^2}{s_{+1}^2 + s_{-1}^2},$$

где $s_j^2 = \sum\limits_{i:y_i=j}(w^{\mathrm{T}}x_i - \tilde{m}_j)$.

7.5 Дискриминант Фишера как линейный классификатор

Рассмотрим всё ту же задачу. Количество объектов обучающей выборки из класса 1 обозначим l_1, из класса 2 — l_2. При построении классификатора $a(x) = sign\{f(x)\}$, где $f(x) = w^Tx + w_0$, будем руководствоваться соображением:

$$\sum_{i=1}^{l}(f(x_i) - y_i)^2 \to \min_{w,w_0}.$$

a. Чему равна сумма $\sum\limits_{i=1}^{l} f(x_i)$ при оптимальном значении w_0? Если переобозначить классы и

положить $Y = \{+\frac{l}{l_1}, -\frac{l}{l_2}\}$, как параметр w_0 оптимального классификатора $a(x)$ будет связан с вектором w?

b. Получите параметры w и w_0 классификатора. Матрицу S_W считайте обратимой. Разрешается пользоваться тем, что в условиях предыдущего пункта (после переобозначения) для оптимального w_0:

$$\sum_{i=1}^{l}\left(w^T x_i + w_0 - y_i\frac{l}{l_{y_i}}\right)x_i = 0 \Rightarrow \left(S_W + \frac{l_1 l_2}{l}S_B\right)w = l(m_1 - m_2).$$

Здесь m_1, m_2 — выборочные средние для объектов из первого и второго классов соответственно,

$$S_B = (m_1 - m_2)(m_1 - m_2)^T,$$
$$S_W = \sum_{i:y_i=1}(x_i - m_1)(x_i - m_1)^T + \sum_{i:y_i=2}(x_i - m_2)(x_i - m_2)^T.$$

Решение

$a(x) = \operatorname{sign} f(x), f(x) = \langle w, x\rangle + w_0$.

$$\sum_{i=1}^{l}(\langle w, x_i\rangle + w_0 - y_i)^2 \to \min_{w,w_0}.$$

a) При оптимальном w_0:

$$\frac{\partial}{\partial w_0}\left(\sum_{i=1}^{l}(\langle w, x_i\rangle + w_0 - y_i)^2\right) = \sum_{i=1}^{l}2(\langle w, x_i\rangle + w_0 - y_i) = 0.$$

$$w_0 = \frac{1}{l}\sum_{i=1}^{l}(y_i - \langle w, x_i\rangle).$$

Значит,

$$\sum_{i=1}^{l}f(x_i) = \sum_{i=1}^{l}\langle w, x_i\rangle + lw_0 = \sum_{i=1}^{l}y_i = l_1 - l_2.$$

Если $Y = \{+\frac{1}{l_1}, -\frac{1}{l_2}\}$, то

$$w_0 = \frac{1}{l}\sum_{i=1}^{l}(y_i - \langle w, x_i\rangle) = -\frac{1}{l}\sum_{i=1}^{l}\langle w, x_i\rangle.$$

b) $\dfrac{\partial}{\partial w^j}\left(\sum_{i=1}^{l}(w^{\mathrm{T}}x_i + w_0 - y_i)^2\right) = 2\sum_{i=1}^{l}\left(w^{\mathrm{T}}x_i + w_0 - y_i\right)x_i^j.$

Тогда $\dfrac{\partial}{\partial w}\left(\sum_{i=1}^{l}(w^{\mathrm{T}}x_i + w_0 - y_i)^2\right) = 2\sum_{i=1}^{l}\left(w^{\mathrm{T}}x_i + w_0 - y_i\right)x_i = 0$. По условию отсюда следует, что $\left(S_W + \dfrac{l_{+1}l_{-1}}{l}S_B\right)w = l(m_{+1} - m_{-1})$.

Тогда

$$S_W w + \frac{l_{+1}l_{-1}}{l}(m_{+1} - m_{-1})(m_{+1} - m_{-1})^{\mathrm{T}}w = l(m_{+1} - m_{-1}),$$
$$w = S_W^{-1}\underbrace{\left(l - \frac{l_{+1}l_{-1}}{l}\right)}_{\in\mathbb{R}}(m_{+1} - m_{-1})\underbrace{(m_{+1} - m_{-1})^{\mathrm{T}}w}_{\in\mathbb{R}}.$$

В задаче оптимизации важно направление w, а не его масштаб, поэтому:

$$w = S_W^{-1}(m_{+1} - m_{-1}),$$

$$w_0 = -\frac{1}{l}\sum_{i=1}^{l}\langle w, x_i\rangle.$$

7.6 Геометрия многомерного нормального распределения и нормального дискриминантного анализа

Решение

a) Пусть $\Sigma = \mathrm{diag}(\sigma_1^2, \ldots, \sigma_n^2), \mu = (\mu_1, \ldots, \mu_n)^{\mathrm{T}}$. Тогда линии уровня плотности распределения:

$$(2\pi)^{-n/2}|\Sigma|^{-1/2}\exp\left(-\frac{1}{2}(x-\mu)^{\mathrm{T}}\Sigma^{-1}(x-\mu)\right) = C,$$

$$(x-\mu)^{\mathrm{T}}\Sigma^{-1}(x-\mu) = C',$$

$$x^{\mathrm{T}}\Sigma^{-1}x - x^{\mathrm{T}}\Sigma^{-1}\mu - \mu^{\mathrm{T}}\Sigma^{-1}x + \mu^{\mathrm{T}}\Sigma^{-1}\mu = C',$$

$$\sum_{i=1}^{n}\left(x_i^2\sigma_i^{-2} - 2x_i\mu_i\sigma_i^{-2} + \mu_i^2\sigma_i^{-2}\right) = C',$$

$$\sum_{i=1}^{n}\frac{(x_i-\mu_i)^2}{\sigma_i^2} = C',$$

— многомерный эллипсоид при замене $x_i' = x_i - \mu_i$. $a_i^2 = \frac{\sigma_i^2}{C'}$.

Если матрица Σ не диагональна, то для нее можно взять её спектральное разложение: $\Sigma = VSV^{\mathrm{T}}$, где $V = (v_1, \ldots, v_n)$ — ортогональные собственные векторы матрицы Σ, соответствующие собственным значениям $\lambda_1, \ldots, \lambda_n$, матрица S диагональна, $S = \mathrm{diag}(\lambda_1, \ldots, \lambda_n)$. Тогда $\Sigma^{-1} = VS^{-1}V^{\mathrm{T}}$, следовательно,

$$(x-\mu)^{\mathrm{T}}\Sigma^{-1}(x-\mu) = (x-\mu)^{\mathrm{T}}VS^{-1}V^{T}(x-\mu) = (x'-\mu')^{\mathrm{T}}S^{-1}(x'-\mu').$$

Значит, в новых координатах $x' = V^{\mathrm{T}}x$ опять же получим эллипсоид.

b) Из задачи про нормальный дискриминантный анализ разделяющая поверхность имеет вид:

$$(x - \frac{1}{2}(\mu_{+1} + \mu_{-1}))^{\mathrm{T}}\Sigma^{-1}(\mu_{+1} - \mu_{-1}) = 0.$$

Значит, она пройдет через точку $\frac{1}{2}(\mu_{+1} + \mu_{-1})$.

c) Касательная плоскость к поверхности $\sum_{i=1}^{n}\frac{(x_i - (\mu_{+1})_i)^2}{\sigma_i^2} - C' = 0$ в точке $\frac{1}{2}(\mu_{+1} + \mu_{-1})$ имеет вид $\sum_{i=1}^{n} 2\left(\frac{1}{2}(\mu_{+1} + \mu_{-1}) - (\mu_{+1})_i\right)\sigma_i^{-2}\left(x - \frac{\mu_{+1}+\mu_{-1}}{2}\right) = (\mu_{-1} - \mu_{+1})^{\mathrm{T}}\Sigma^{-1}\left(x - \frac{\mu_{+1}+\mu_{-1}}{2}\right) = 0$, то есть является разделяющей поверхностью.

7.7 Квадратичная функция потерь и матожидание

Получите выражение для байесовской регрессии ($y \in R$) в случае квадратичной функции потерь. Обратите внимание: это показывает, что если вам нужно получить в качестве оценки матожидание истинного значения, следует использовать квадратичную функцию потерь.

Решение

Пусть $Y = \mathbb{R}$ и функция потерь $L(a(x), y) = (a(x) - y)^2$.

Функционал среднего риска имеет вид:

$$R(a) = \int\limits_X \int\limits_Y (a(x) - y)^2 p(y|x)p(x)dydx = \int\limits_X p(x) \left(\int\limits_Y (a(x) - y)^2 p(y|x)dy \right) dx.$$

Минимизируем внутренний интеграл (тогда, т.к. $p(x) \geqslant 0$, минимизируется и весь функционал): для фиксированного x найдем значение t, при котором $\int\limits_Y (t - y)^2 p(y|x)dy$ минимален:

$$\frac{\partial}{\partial t} \int\limits_Y (t - y)^2 p(y|x)dy = 2 \int\limits_Y (t - y)p(y|x)dy =$$

$$= 2 \left(\int\limits_Y tp(y|x)dy - \int\limits_Y yp(y|x)dy \right) = 2 \left(t \underbrace{\mathsf{P}(y \in Y|x)}_{=1} - \int\limits_Y yp(y|x)dy \right) = 0.$$

Значит, $a(x) = t = \int\limits_Y yp(y|x)dy = \mathsf{E}(y|x)$ минимизирует функционал $R(a)$.

7.8 Модуль отклонения и медиана

Покажите, что для оценки медианы распределения y при условии x следует использовать в качестве функции потерь модуль отклонения.

Решение

Пусть $Y = \mathbb{R}$ и функция потерь $L(a(x), y) = |a(x) - y|$.

Функционал среднего риска имеет вид:

$$R(a) = \int\limits_X \int\limits_Y |a(x) - y|p(y|x)p(x)dydx = \int\limits_X p(x) \left(\int\limits_Y |a(x) - y|p(y|x)dy \right) dx.$$

Минимизируем внутренний интеграл: для фиксированного x найдем значение t, при котором $\int\limits_Y |t - y|p(y|x)dy$ минимален:

$$\frac{\partial}{\partial t} \int\limits_Y |t - y|p(y|x)dy = \frac{\partial}{\partial t} \int\limits_{Y \setminus \{t\}} |t - y|p(y|x)dy =$$

$$= \int\limits_{Y \setminus \{t\}} \operatorname{sign}(t - y)p(y|x)dy =$$

$$= \int\limits_{\{t>y\}} p(y|x)dy - \int\limits_{\{t<y\}} p(y|x)dy =$$

$$= \mathsf{P}(\{t > y\}|x) - \mathsf{P}(\{t < y\}|x) = 0.$$

Значит, $\mathsf{P}(\{t > y\}|x) = \mathsf{P}(\{t < y\}|x)$.

Если считать, что распределение $p(y|x)$ непрерывно, то $\mathsf{P}(\{t = y\}|x) = 0$ и

$$\mathsf{P}(\{t \geqslant y\}|x) = \frac{1}{2},$$

$$\mathsf{P}(\{t < y\}|x) = \frac{1}{2},$$

т.е. $a(x) = t$ оценивает $\frac{1}{2}$-квантиль.

7.9 Неудачный выбор функции потерь

В одном проекте заказчик очень хотел, чтобы исследователь решать не задачу классификации на классы 0 и 1 (которая и стояла), а задачу регрессии на тех же метках с модулем отклонения в качестве функции потерь. Замысел заказчика был в том, что оцененные числа получатся дробными и это будет приближением для вероятности класса 1. Считая, что любой разумный алгоритм старается минимизировать матожидание потерь на объекте при условии известного объекта x, покажите, что затея приведет к тому, что в ответах будут только 0 и 1.

Решение

Рассмотрим матожидание функции потерь:

$$\mathsf{E}(L(y,a(x))) = \int\limits_X \left(\sum_{i=1}^{2} |y_i - a(x)| p(x,y_i)\right) dx =$$

$$= \int\limits_X p(x) \left(\sum_{i=1}^{2} |y_i - a(x)|\, \mathsf{P}(y_i|x)\right) dx \to \min_a \Leftrightarrow$$

$$\Leftrightarrow \sum_{i=1}^{2} |y_i - a(x)|\, \mathsf{P}(y_i|x) \to \min_a$$

Поскольку планировалось прогнозировать вероятность будем считать что $\forall x \in X \hookrightarrow a(x) \in [0,1]$, и обозначим $\mathsf{P}(1|x) = p$. Тогда:

$$a(x)(1-p) + (1-a(x))p = p + a(x)(1-2p) \to \min_a \Rightarrow$$

$$\Rightarrow a(x) = \begin{cases} 0, \text{ если} & p \le 0.5 \\ 1, \text{ если} & p > 0.5 \end{cases}$$

Откуда следует, что все ответы будут 0 или 1.

7.10 Функция потерь и оценка вероятности

Какой должна была быть функция потерь в предыдущей задаче, чтобы действительно оценивать вероятности?

Подсказка: поразмышляйте насчет энтропии или/и почитайте о разных способах записи оптимизируемого функционала в логистической регрессии)

Решение

Возьмем в качестве функции потерь
$L(y,a(x)) = -y\ln(a(x)) - (1-y)\ln(1-a(x))$ (также известную как log loss). Тогда:

$$\mathsf{E}(L(y,a(x))) =$$

$$= \int\limits_X p(x) \left(\sum_{i=1}^{2} \mathsf{P}(y_i|x)(-y_i\ln(a(x)) - (1-y_i)\ln(1-a(x)))\right) dx \to \min_a \Leftrightarrow$$

$$\Leftrightarrow \sum_{i=1}^{2} \mathsf{P}(y_i|x)(-y_i\ln(a(x)) - (1-y_i)\ln(1-a(x))) \to \min_a$$

Обозначим $\mathsf{P}(1|x) = p$. Тогда:

$$\sum_{i=1}^{2} p(y_i|x)(-y_i \ln(a(x)) - (1-y_i)\ln(1-a(x))) =$$
$$= -(1-p)\ln(1-a(x)) - p\ln(a(x)) \to \min_a$$
$$\frac{\partial}{\partial a}\left(-(1-p)\ln(1-a) - p\ln(a)\right) = \frac{1-p}{1-a} - \frac{p}{a} =$$
$$= \frac{(1-p)a - p(1-a)}{(1-a)a} = \frac{a-p}{(1-a)a} = 0 \Rightarrow$$
$$\Rightarrow a(x) = p$$

Значит, такая функция потерь подойдет.

Решение 2

Можно проигнорировать подсказку и использовать уже полученный результат для квадратичной функции потерь. Если целевое значение y для одного класса обозначать 0, а для другого 1, то матожидание y при условии x будет в точности нужной нам вероятностью. Т.к. квадратичная функция потерь приводит к оценке матожидания, то при такой постановке задачи она как раз нам подойдет.

Глава 8

Логистическая регрессия

8.1 Экспонентное семейство и эвристический вывод

Выведите логистическую регрессию (придите к правильной функции потерь и покажите, как оценивать вероятности классов) двумя способами: через экспонентное семейство распределений и из эвристических соображений, взятых из секции 2 листка по логистической регрессии (см. приложение).

Решение

Логистическая регрессия.

Через экспонентное семейство

Пусть $Y = \{-1, +1\}$, объекты описываются n числовыми признаками $f_j : X \to \mathbb{R}, j = 1, \ldots, n$. $X = \mathbb{R}^n : x \equiv (f_1(x), \ldots, f_n(x))$.

Будем считать, что множество прецедентов $X \times Y$ является вероятностным пространством. Выборка прецедентов $X^l = (x_i, y_i)_{i=1}^l$ получена случайно и независимо согласно вероятностному распределению с плотностью $p(x, y) = \mathsf{P}_y\, p_y(x) = \mathsf{P}(y|x)p(x)$, где P_y — априорные вероятности, $p_y(x)$ — функции правдоподобия, $\mathsf{P}(y|x)$ — апостериорные вероятности классов $y \in Y$.

Предположим также, что функции правдоподобия $p_y(x)$ принадлежат экспонентному семейству плотностей с равными значениями параметров d и δ, т.е.

$$p_y(x) = \exp\left(c_y(\delta)\langle\theta, x\rangle + b_y(\delta, \theta_y) + d(x, \delta)\right).$$

Теорема. *Если среди признаков $f_1(x), \ldots, f_n(x)$ есть константа, то апостериорная вероятность принадлежности произвольного объекта $x \in X$ классу $y \in \{-1, +1\}$ может быть вычислена по значению дискриминантной функции: $\mathsf{P}(y|x) = \sigma(\langle w, x\rangle y)$, где $\sigma(z) = \frac{1}{1+e^{-z}}$ — сигмоидная функция.*

Доказательство. Байесовский классификатор в случае двух классов имеет вид

$$a(x) = \operatorname{sign}\left(\lambda_{+1}\,\mathsf{P}(+1|x) - \lambda_{-1}\,\mathsf{P}(-1|x)\right) = \operatorname{sign}\left(\frac{\mathsf{P}(+1|x)}{\mathsf{P}(-1|x)} - \frac{\lambda_{-1}}{\lambda_{-1}}\right).$$

$$\frac{\mathsf{P}(+1|x)}{\mathsf{P}(-1|x)} = \frac{\mathsf{P}_{+1}\,p_{+1}(x)}{\mathsf{P}_{-1}\,p_{-1}(x)} =$$

$$= \exp\left(\langle\underbrace{(c_{+1}(\delta)\theta_{+1} - c_{-1}(\delta)\theta_{-1})}_{w=\text{const(x)}}, x\rangle + \underbrace{b_{+1}(\delta, \theta_{+1}) - b_{-1}(\delta, \theta_{-1}) + \ln\frac{\mathsf{P}_{+1}}{\mathsf{P}_{-1}}}_{\text{const(x)}}\right) = e^{\langle w, x\rangle},$$

т.к. все слагаемые под экспонентой, не зависящие от x, можно считать аддитивной добавкой к коэффициенту при константном признаке и их можно включить в $\langle w, x\rangle$.

По формуле полной вероятности $\mathsf{P}(-1|x) + \mathsf{P}(+1|x) = 1$, значит,

$$\mathsf{P}(+1|x) = \sigma(+\langle w, x\rangle),\ \mathsf{P}(-1|x) = \sigma(-\langle w, x\rangle).$$

Тогда получаем, что $\mathsf{P}(y|x) = \sigma(\langle w, x\rangle y)$. □

Для настройки весов w будем максимизировать логарифм правдоподобия выборки:

$$L(w, X^l) = \ln \prod_{i=1}^{l} p(x_i, y_i) \to \max_w.$$

Так как $p(x, y) = \mathsf{P}(y|x)p(x)$, где $p(x)$ не зависит от w, а $\mathsf{P}(y|x) = \sigma(\langle w, x\rangle y)$, то

$$L(w, X^l) = \sum_{i=1}^{l} \log_2 \sigma(\langle w, x_i\rangle y_i) + \text{const}(w) \to \max_w,$$

что эквивалентно минимизации функционала с логистической функцией потерь

$$\tilde{Q}(w, X^l) = \sum_{i=1}^{l} \log_2 (1 + \exp(-\langle w, x_i\rangle y_i)) \to \min_w.$$

Через эвристические соображения

Будем брать $\mathsf{P}(+1|x) = \sigma(\langle w, x\rangle) = \dfrac{1}{1 + e^{-\langle w,x\rangle}}$. Тогда $\mathsf{P}(-1|x) = 1 - \dfrac{1}{1 + e^{-\langle w,x\rangle}} = \dfrac{1}{1 + e^{\langle w,x\rangle}} = \sigma(-\langle w, x\rangle)$.

Получаем, что $\mathsf{P}(y|x) = \sigma(y\langle w, x\rangle)$.

Тогда

$$\prod_{i=1}^{l} \mathsf{P}(y_i|x_i) = \prod_{i=1}^{l} \frac{1}{1 + e^{-M_i}} \to \max_w \ \Leftrightarrow$$

$$\Leftrightarrow\ \ln \prod_{i=1}^{l} \mathsf{P}(y_i|x_i) = -\sum_{i=1}^{l} \ln(1 + e^{-M_i}) \to \max_w \ \Leftrightarrow$$

$$\Leftrightarrow\ \sum_{i=1}^{l} \ln(1 + e^{-M_i}) \to \min_w.$$

Таким образом, опять получаем минимизацию функционала с логистической функцией потерь.

Глава 9

Expectation-Maximization

9.1 Различия в выводе EM-алгоритма и применение в кластеризации и классификации

Сопоставьте вывод EM-алгоритма из лекций Воронцова и объяснение в английской Википедии. В чем отличия? Как использовать EM-алгоритм для кластеризации? А для классификации? Что общего и различного у кластеризации EM-алгоритмом и методом k средних?

Решение

В английской Википедии на E-шаге предлагается считать условное матожидание логарифма функции правдоподобия по Z (скрытым переменным) относительно $X, \theta^{(t)}$. Это дает функционал $Q(\theta|\theta^{(t)}) = \sum\limits_{Z} p(Z|X,\theta^{(t)}) \ln p(X,Z|\theta)$, который и максимизируют на M-шаге.

Воронцов на E-шаге предлагает считать $g_{ij} = \mathsf{P}(\theta_j|x_i)$ — скрытые переменные — по формуле Байеса через веса $w_j = \mathsf{P}(\theta_j)$ и параметры θ_j. На M-шаге ищется максимум взвешенного правдоподобия $\theta_j = \arg\max\limits_{\theta} \sum\limits_{i=1}^{m} g_{ij} \ln \mathsf{P}(x_i|\theta), j = 1,\ldots,k$, а также пересчитываются веса $w_j = \frac{1}{m}\sum\limits_{i=1}^{m} g_{ij}$.

Кластеризация с помощью EM-алгоритма

Предположим, что кластеры имеют вид эллипсоидов с осями, направленными вдоль осей координат. Тогда каждый кластер $y \in Y$ описывается n-мерной гауссовской плотностью $p_y(x)$ с центром $\mu_y = (\mu_{y_1},\ldots,\mu_{y_n})$ и диагональной матрицей ковариаций $\Sigma_y = \operatorname{diag}(\sigma^2_{y_1},\ldots,\sigma^2_{y_n})$:

$$p_y(x) = (2\pi)^{-n/2}(\sigma_{y_1}\cdot\ldots\cdot\sigma_{y_n})^{-1}\exp\left(-\frac{1}{2}\rho^2_y(x,\mu_y)\right),$$

где $\rho^2_y(x,x') = \sum\limits_{j=1}^{n} \sigma^{-2}_{yj}(f_j(x) - f_j(x'))^2$ — взвешенное евклидово расстояние с весами σ^{-2}_{yj}.

Тогда задача кластеризации совпадает с задачей разделения смеси вероятностных распределений и можно применить EM-алгоритм с последовательным добавлением компонент.

Классификация с помощью EM-алгоритма

x_i приписывается к классу с номером, соответствующим номеру наибольшего из чисел $g_{i_1},\ldots,g_{i_n}$. $\theta_j = \arg\max\limits_{\theta} \sum\limits_{x_i \in K_i} \mathsf{P}(x_i|\theta)$. Остальное совпадает с обычным EM-алгоритмом.

Отличия от k-means

Метод k-means является упрощением EM-алгоритма. В отличие от него, в k-means каждый объект приписан только к одному кластеру (в EM-алгоритме для объекта задается вероятностное

распределение $g_{ij} = \mathsf{P}(y_i = \theta_j)$). Кроме того, в EM-алгоритме на M-шаге настраивается форма кластеров (σ^2_{yj}), а в k-means — нет.

Глава 10

Понижение размерности пространства признаков

10.1 Метод главных компонент (PCA, Principal Component Analysis)

Разберите вывод метода главных компонент из лекций Воронцова и изложите кратко. Покажите справедливость формул, использованных в выводе при дифференцировании по матрицам (можно просто проверить поэлементно).

Решение

Есть l объектов обучающей выборки и n признаков. Запишем их в матрицу F:

$$F_{l\times n} = \begin{pmatrix} f_1(x_1) & \cdots & f_n(x_1) \\ \vdots & \ddots & \vdots \\ f_1(x_l) & \cdots & f_n(x_l) \end{pmatrix} = \begin{pmatrix} x_1 \\ \vdots \\ x_l \end{pmatrix}.$$

Хотим построить матрицу G в пространстве меньшей размерности:

$$G_{l\times m} = \begin{pmatrix} g_1(x_1) & \cdots & g_m(x_1) \\ \vdots & \ddots & \vdots \\ g_1(x_l) & \cdots & g_m(x_l) \end{pmatrix} = \begin{pmatrix} z_1 \\ \vdots \\ z_l \end{pmatrix},$$

из которой линейным преобразованием U получалось бы хорошее приближение исходной матрицы F:

$$\Delta^2(G,U) = \|GU^{\mathrm{T}} - F\|^2 = \mathrm{tr}(GU^{\mathrm{T}} - F)(GU^{\mathrm{T}} - F)^{\mathrm{T}} \to \min_{G,U}.$$

Теорема. *Если $m \leqslant \mathrm{rk}F$, то минимум $\Delta^2(G,U)$ достигается, когда столбцы матрицы U есть собственные векторы $F^{\mathrm{T}}F$, соответствующие m максимальным собственным значениям. При этом $G = FU$, матрицы U и G ортогональны.*

Доказательство. Обозначим A^i — i-й столбец матрицы.

Тогда по задаче 16.2:

$$\frac{\Delta^2}{\partial(G^{\mathrm{T}})^i} = 2U^{\mathrm{T}}(U(G^{\mathrm{T}})^i - (F^{\mathrm{T}})^i) = 0.$$

Значит, $(GU^{\mathrm{T}} - F)U = 0$.

$$\frac{\Delta^2}{\partial(U^{\mathrm{T}})^i} = 2G^{\mathrm{T}}(G(U^{\mathrm{T}})^i - F^i) = 0.$$

Значит, $G^{\mathrm{T}}(GU^{\mathrm{T}} - F) = 0$.

Минимум $\Delta^2(G,U)$ достигается при

$$\begin{cases} \frac{\partial}{\partial G}\Delta^2(G,U) = (GU^{\mathrm{T}} - F)U = 0; \\ \frac{\partial}{\partial U}\Delta^2(G,U) = G^{\mathrm{T}}(GU^{\mathrm{T}} - F) = 0. \end{cases}$$

G и U невырождены, значит,

$$\begin{cases} G = FU(U^{\mathrm{T}}U)^{-1}; \\ U = F^{\mathrm{T}}G(G^{\mathrm{T}}G)^{-1}. \end{cases}$$

Если $G^{\mathrm{T}}G$ и $U^{\mathrm{T}}U$ — диагональные матрицы, то это выражение сильно упростится. Так как минимизируемый функционал зависит только от GU^{T}, матрицы можно домножить на невырожденное преобразование R: $GU^{\mathrm{T}} = (GR)(R^{-1}U^{\mathrm{T}})$.

Пусть $\tilde{G}\tilde{U}^{\mathrm{T}}$ — произвольное решение.

Матрица $\tilde{U}^{\mathrm{T}}$ — невырожденная, значит, $\tilde{U}^{\mathrm{T}}\tilde{U}$ — симметричная ($(\tilde{U}^{\mathrm{T}}\tilde{U})^{\mathrm{T}} = \tilde{U}^{\mathrm{T}}\tilde{U}$), невырожденная и положительно определенная ($x^{\mathrm{T}}\tilde{U}^{\mathrm{T}}\tilde{U}x = \|\tilde{U}x\| \geqslant 0$). Тогда найдется такая $S_{m\times m}$, что $S^{-1}\tilde{U}^{\mathrm{T}}\tilde{U}(S^{-1})^{\mathrm{T}} = I_m$ (ортогональная матрица).

$S^{\mathrm{T}}\tilde{G}^{\mathrm{T}}\tilde{G}S$ — симметричная и невырожденная, поэтому существует ортогональная матрица $T_{m\times m}$ такая, что

$$T^{\mathrm{T}}(S^{\mathrm{T}}\tilde{G}^{\mathrm{T}}\tilde{G}S)T = \mathrm{diag}(\lambda_1,\ldots,\lambda_m) = \Lambda.$$

Преобразование $R = ST$ невырождено. Значит, можно взять $G = \tilde{G}R, U^{\mathrm{T}} = R^{-1}\tilde{U}^{-1}$. Тогда

$$G^{\mathrm{T}}G = T^{\mathrm{T}}(S^{\mathrm{T}}\tilde{G}^{\mathrm{T}}\tilde{G}S)T = \Lambda,$$

$$U^{\mathrm{T}}U = T^{-1}(S^{-1}\tilde{U}^{T}\tilde{U}(S^{-1})^{\mathrm{T}})(T^{-1})^{\mathrm{T}} = (T^{\mathrm{T}}T)^{-1} = I_m.$$

Подставим G и U в систему:

$$\begin{cases} G = FU; \\ U\Lambda = F^{\mathrm{T}}G. \end{cases}$$

Значит, $U\Lambda = F^{\mathrm{T}}FU$, т.е. столбцы матрицы U являются собственными векторами матрицы $F^{\mathrm{T}}F$, а диагональные элементы $\lambda_1,\ldots,\lambda_m$ — соответствующие собственные значения.

Аналогично, $G\Lambda = FF^{\mathrm{T}}U$, т.е. столбцы матрицы G — собственные векторы матрицы FF^{T}, соответствующие тем же самым собственным значениям.

Тогда

$$\begin{aligned} \Delta^2(G,U) &= \|F - GU^{\mathrm{T}}\|^2 = \mathrm{tr}(F^{\mathrm{T}} - UG^{\mathrm{T}})(F - GU^{\mathrm{T}}) = \mathrm{tr}F^{\mathrm{T}}(F - GU^{\mathrm{T}}) = \\ &= \mathrm{tr}F^{\mathrm{T}}F - \mathrm{tr}F^{\mathrm{T}}GU^{\mathrm{T}} = \|F\|^2 - \mathrm{tr}U\Lambda U^{\mathrm{T}} = \\ &= \|F\|^2 - \mathrm{tr}\Lambda = \sum_{j=1}^{n}\lambda_j - \sum_{j=1}^{m}\lambda_j = \sum_{j=m+1}^{n}\lambda_j, \end{aligned}$$

где $\lambda_1,\ldots,\lambda_n$ — все собственные значения матрицы $F^{\mathrm{T}}F$. Минимум Δ^2 достигается, когда $\lambda_1,\ldots,\lambda_m$ — наибольшие m из n собственных значений. □

Глава 11

Нейронные сети

11.1 Backpropagation для произвольной функции потерь

Выведите алгоритм обратного распространения ошибки для случая произвольной функции потерь.

Решение

Пусть выходной слой состоит из M нейронов с функциями активации σ_m и выходами $a^m,\ m = 1,\ldots,M$. Перед ними находится скрытый слой из H нейронов с функциями активации σ_h и выходами $u^h,\ h = 1,\ldots,H$.

Веса синаптических связей между h-м нейроном скрытого слоя и m-м нейроном выходного слоя — w_{hm}. Перед этим слоем лежит входной слой с выходами v^j и весами w_{jh}.

Тогда выходные значения сети на объекте x_i вычисляются как:

$$a^m(x_i) = \sigma_m\left(\sum_{h=0}^{H} w_{hm}u^h(x_i)\right); \quad u^h(x_i) = \sigma_h\left(\sum_{j=0}^{J} w_{jh}v^j(x_i)\right).$$

Функционал потерь для x_i: $Q(w) = \sum_{m=1}^{M} \mathcal{L}(a^m(x_i), y_i^m)$.

$$\frac{\partial Q(w)}{\partial a^m} = \frac{\partial \mathcal{L}(a^m(x_i), y_i^m)}{\partial a^m} = \varepsilon_i^m.$$

$$\frac{\partial Q(w)}{\partial u^h} = \sum_{m=1}^{M} \frac{\partial \mathcal{L}(a^m(x_i), y_i^m)}{\partial a^m}\sigma'_m w_{hm} = \sum_{m=1}^{M} \varepsilon_i^m \sigma'_m w_{hm} = \varepsilon_i^h.$$

Тогда

$$\frac{\partial Q(w)}{\partial w_{hm}} = \frac{\partial Q(w)}{\partial a^m}\frac{\partial a^m}{\partial w_{hm}} = \varepsilon_i^m \sigma'_m u^h(x_i),$$

$$\frac{\partial Q(w)}{\partial w_{jh}} = \frac{\partial Q(w)}{\partial u^h}\frac{\partial u^h}{\partial w_{jh}} = \varepsilon_i^h \sigma'_h u^j(x_i).$$

Глава 12

Композиции алгоритмов

12.1 AdaBoost без нормировки весов объектов и с отказами от классификации

Воспроизведите вывод AdaBoost (см. лекции Воронцова), но без нормировки весов объектов и в предположении возможности отказов от классификации у базовых алгоритмов. Объясните, какой смысл имеет сумма весов объектов до обновления весов. Получите готовое описание алгоритма, и выясните, при каких условиях в шаге выбора нового базового алгоритма выражение будет переходить в $b(x) = \arg\min N$ (функционал N определяется, как и ранее, как сумма весов объектов, относимых алгоритмом b к неправильному классу).

Не забывайте, что семейство алгоритмов может быть несимметричным - например только решающие пни вида $a(x) = sign(x_i \geq t)$, без пней $a(x) = sign(x_i > t)$. Как позволит модифицировать шаг выбора алгоритма симметричность семейства базовых алгоритмов?

Решение

Будем решать задачу бинарной классификации с экспоненциальной функцией потерь:

$$Q(a, X^l) = \sum_{i=1}^{l} exp(-y_i a(x_i)) \to \min_a$$

Пусть нам дан некоторый базовый набор классификаторов A, удовлетворяющий условию: для любого набора весов $w_1, \ldots, w_l$ найдется алгоритм из A, взвешенная ошибка которого меньше 0.5:

$$\exists a \in A : \sum_{i=1}^{l} w_i [a(x_i) \neq y_i] < \frac{1}{2}$$

Классификатор будем искать в виде взвешенной суммы базовых:

$$a(x) = sign \sum_{n=1}^{N} \mu_n a_n(x), \qquad \mu_n > 0$$

Метод итерационный, каждый следующий алгоритм выбирается так, чтобы минимизировать функцию потерь $Q(a, X^l)$.

Пусть у нас есть некоторый набор весов $U^l = (u_1, \ldots, u_l)$, тогда обозначим соответсвующие этому набору и алгоритму $a(x)$ суммарный вес ошибочных классификаций $N(a, U^l)$, суммарный

вес правильных классификаций $P(a, U^l)$ и суммарный вес отказов от классификации $R(a, U^l)$:

$$N(a, U^l) = \sum_{i=1}^{l} u_i[a(x_i) = -y_i]$$

$$P(a, U^l) = \sum_{i=1}^{l} u_i[a(x_i) = y_i]$$

$$R(a, U^l) = \sum_{i=1}^{l} u_i[a(x_i) = 0]$$

Заметим, что $R + N + P = \sum\limits_{i=1}^{l} u_i$.

Будем пользоваться тождеством $e^{-\alpha b} = e^{-\alpha}[b = 1] + e^{\alpha}[b = -1] + [b = 0]$.

Пусть мы уже выбрали первые $(N-1)$ базовых алгоритмов $a_1, \ldots, a_{N-1}$. Будем искать $a_N(x)$ и μ_N, для этого рассмотрим соответсвующую фунцию потерь:

$$Q(a, X^l) = \sum_{i=1}^{l} exp(-y_i a(x_i)) = \sum_{i=1}^{l} exp(-y_i \sum_{n=1}^{N} \mu_n a_n(x_i)) =$$

$$= \sum_{i=1}^{l} exp(-y_i \sum_{n=1}^{N-1} \mu_n a_n(x_i)) exp(-y_i \mu_N a_N(x_i)) = |w_i = exp(-y_i \sum_{n=1}^{N-1} \mu_n a_n(x_i))| =$$

$$= \sum_{i=1}^{l} w_i exp(-y_i \mu_N a_N(x_i)) = e^{-\mu_N} \sum_{i=1}^{l} w_i[a_N(x_i) = y_i] + e^{\mu_N} \sum_{i=1}^{l} w_i[a_N(x_i) = -y_i] +$$

$$+ \sum_{i=1}^{l} w_i[a_N(x_i) = 0] = e^{-\mu_N} P(a_N, W^l) + e^{\mu_N} N(a_N, W^l) + R(a_N, W^l) =$$

$$= e^{-\mu_N} P(a_N, W^l) + e^{\mu_N} N(a_N, W^l) + R(a_N, W^l) =$$

$$= e^{-\mu_N} P(a_N, W^l) + e^{\mu_N} N(a_N, W^l) + \left(\sum_{i=1}^{l} w_i - P(a_N, W^l) - N(a_N, W^l) \right)$$

Продифференцируем по μ_N и приравняем производную к нулю:

$$Q'(a, X^l) = -\mu_N e^{-\mu_N} P + \mu_N e^{\mu_N} N = 0$$

$$-\mu_N + \ln P = \mu_N + \ln N$$

$$\mu_N = \frac{1}{2} \ln \frac{P}{N}$$

Сумма весов с прошлого раза, это значение функции потерь после предыдущей итерации.

Теперь поставим его обратно в Q:

$$Q(a, X^l) = \frac{\sqrt{N}}{\sqrt{P}} P + \frac{\sqrt{P}}{\sqrt{N}} N + \left(\sum_{i=1}^{l} w_i - P - N \right) = 2\sqrt{NP} + \left(\sum_{i=1}^{l} w_i - P - N \right) =$$

$$= \sum_{i=1}^{l} w_i - (\sqrt{P} - \sqrt{N})^2$$

Поскольскольку семейство обладает свойством слабой обучаемости, то $P > N$. Поэтому алгоритм пожно искать как:

$$a_N = \arg\max_a \sqrt{P} - \sqrt{N}$$

Если убрать условие отказов от классификации, то:

$$Q(a, X^l) = e^{-\mu_N} P(a_N, W^l) + e^{\mu_N} N(a_N, W^l) + \left(\sum_{i=1}^{l} w_i - P(a_N, W^l) - N(a_N, W^l) \right) =$$
$$= e^{-\mu_N} P(a_N, W^l) + e^{\mu_N} N(a_N, W^l)$$

Производная не изменится и μ_N будет таким же. Подставим его:

$$Q(a, X^l) = 2\sqrt{NP} = 2\sqrt{N\left(\sum_{i=1}^{l} w_i - N\right)} = 2\sqrt{N\left(Q_{N-1} - N\right)} \to \min a_N \Leftrightarrow$$
$$\Leftrightarrow |\text{если множество алгоритмов симметрично}| \Leftrightarrow$$
$$\Leftrightarrow a_N = \arg\min_a N$$

12.2 AdaBoost с нормировкой и отказами от классификации

Выведите AdaBoost с нормировкой весов объектов (как в лекциях Воронцова), но в предположении возможности отказов от классификации у базовых алгоритмов. Чем отличается вывод AdaBoost с нормировкой весов и без нее?

Решение

Случай нормированных весов $\hat{w}_i = \frac{w_i}{\sum\limits_{i=1}^{l} w_i}$ сильно не изменит вывод. Просто сразу вынесется за скобки $Q_{N-1} = \sum\limits_{i=1}^{l} w_i$, а R, N, P будут зависеть от $\hat{w}_i$ и в сумме давать единицу. Тогда функция потерь примет вид:

$$Q(a, X^l) = \left(e^{-\mu_N} P(a_N, \hat{W}^l) + e^{\mu_N} N(a_N, \hat{W}^l) + \left(1 - P(a_N, \hat{W}^l) - N(a_N, \hat{W}^l)\right) \right) Q_{N-1}$$

На решение оптимизационной задачи это не повлияет, поскольку просто будет вынесен за скобки множитель Q_{N-1}, не зависящий от μ_N и a_N.

12.3 Классификация в GBM

Как решать задачу бинарной классификации с помощью градиентного бустинга (что нужно поменять по сравнению с регрессией)?

Решение

Переформулируем задачу классификации в задачу регрессии: целевое значение – вероятность $p(x)$ отнесения объекта x к классу $+1$.

Будем максимизировать правдоподобие обучающей выборки:

$$\prod_{i=1}^{l} p(x_i)^{[y_i=1]} (1 - p(x_i))^{[y_i=-1]} \to \max_{p(x)} \Leftrightarrow$$
$$\Leftrightarrow Q(p) = -\sum_{i=1}^{l} \left([y_i = 1] \ln p(x_i) + [y_i = -1] \ln(1 - p(x_i)) \right) \to \min_{p(x)}$$

При этом есть ограничение на искомую функцию $p(x_i) \in [0, 1]$, избавимся от него перейдя к поиску функции $a(x)$:

$$p(x) = \frac{1}{1 + \exp(-a(x))}$$

Теперь подставим выражение в логарифм правдоподобия:

$$Q(p) = -\sum_{i=1}^{l} (-[y_i = 1] \ln(1 + \exp(-a(x_i))) - [y_i = -1] \ln(1 + \exp(a(x_i)))) =$$

$$= \sum_{i=1}^{l} \ln(1 + \exp(-y_i a(x_i)))$$

То есть будем использовать градиентный бустинг с логистической функцией потерь.

12.4 О частных случаях GBM

В лекциях Воронцова по композициям алгоритмов сразу после AdaBoost рассказывается алгоритм AnyBoost, позволяющий использовать в бустинге произвольную функцию потерь. Вряд ли вы найдете этот алгоритм в какой-то современной библиотеке, однако бывает полезно понимать, что один алгоритм является лишь частным случаем другого. Попробуйте выяснить, как AnyBoost связан с градиентным бустингом.

Подсказка: приближать алгоритмом антиградиент ошибки в GBM можно по разным метрикам, например можно минимизировать скалярное произведение градиента и ответов алгоритма.

Решение:

Рассмотрим AnyBoost с некоторой функцией потерь L:

$$Q(a, X^l) = \sum_{i=1}^{l} L\left(y_i a_N(x_i)\right) = \sum_{i=1}^{l} L\left(y_i \sum_{n=1}^{N} \alpha_n b_n(x_i)\right)$$

Тогда компоненты ее антиградиента после $(N-1)$-й итерации:

$$-g_i^{(N)} = -\left.\frac{\partial L(y_i, a)}{\partial a}\right|_{a=a_{N-1}(x_i)} = -y_i L'\left(y_i \sum_{n=1}^{N} \alpha_n b_n(x_i)\right)$$

При этом в AnyBoost $w_i = -L'\left(y_i \sum\limits_{n=1}^{N} \alpha_n b_n(x_i)\right)$, откуда:

$$-g_i^{(N)} = y_i w_i$$

Теперь воспользуемся подсказкой и будем минимизировать скалярное произведение градиента и ответов алгоритма:

$$\sum_{i=1}^{l} g_i^{(N)} b(x_i) = -\sum_{i=1}^{l} y_i w_i b(x_i) \to \min_b \Leftrightarrow$$

$$\Leftrightarrow \sum_{i=1}^{l} y_i w_i b(x_i) \to \max_b,$$

что соответствует шагу AnyBoost.

Часть II

Приложение: «Листки» по теории

Система листков

Лекцию можно просто прослушать, не вникая в детали, но если возникнет необходимость воспроизвести все выкладки — разобраться получится намного лучше. Из этой идеи происходит один из форматов знакомства с теорией — «листки» с небольшими задачами, каждая из которых это небольшой шаг к достижению некоторого значительного результата (к выводу метода или доказательству теоремы). Система листков хорошо знакома ученикам матшкол и позволяет неплохо повысить вовлеченность в процесс получения теоретических фактов.

Сейчас в разделе приведены первые два «листка», в новых версиях сборника будут появляться новые листки.

Глава 13

Логистическая регрессия

13.1 Формулировка задачи

Даны векторы признаков $x_1, \cdots, x_l$ для l объектов. На них же известны классы $y_1, \cdots, y_l \in Y = \{-1; +1\}$. Будем пытаться построить линейный классификатор: $a(x) = sign < w, x >= w^T x$, где w - вектор параметров, который мы хотим настроить на обучающей выборке. Здесь мы также имеем ввиду, что могли бы рассматривать более широкий класс решающих функций $a(x) = sign\,(< w, x > +w_0)$, но мысленно добавив к x фиктивный признак $x_0 = 1$ (равный единице для всех объектов), а к вектору весов добавив координату w_0, мы всегда сведем задачу к предыдущей.

Вектор w подбирается из соображений минимизации суммарных потерь на обучающей выборке:

$$Q(w) = \sum_{i=1}^{l} L(M_i) \to \min_w$$

$M_i = y_i < w, x_i >$ — отступ *(margin)* алгоритма на i-том объекте, $L(M_i)$ — ошибка на i-том объекте.

Логистическая регрессия получается в случае использования логистической функции потерь: $L(M) = \ln\left(1 + e^{-M}\right)$.

При этом вероятность принадлежности объекта x к классу $+1$ можно получить следующим образом:

$$P(+1|x) = \sigma(< w, x >) = \frac{1}{1 + e^{-M}}$$

1. Покажите, что при стремлении M к бесконечности, в зависимости от знака, функция L(M) ведет себя как линейная или как экспоненциальная.
2. Покажите, что для производной сигмоидной функции справедлива формула: $\sigma'(z) = \sigma(z)\sigma(-z)$
3. Выпишите явно шаг градиентного спуска и стохастического градиента для задачи минимизации выше.

13.2 От вероятности к функции потерь

Попробуем связать формулу для $P(+1|x)$ с функцией потерь.

1. Получите выражение для $P(-1|x)$, а затем попробуйте обобщить результаты до формулы для $P(y|x)$
2. Запишите произведение вероятностей $P(y_i|x_i)$ и из соображений максимизации этого произведения получите уже знакомую вам задачу минимизации $Q(w)$
3. В качестве функционала правдоподобия правильней было бы брать произведение $P(x_i, y_i)$.

Объясните, будут ли отличия и почему.

13.3 Экспонентное семейство

1. Откройте параграф "логистическая регрессия"в лекциях Воронцова, выясните, что представляет собой экспонентное (exponential, в др. источниках "экспоненциальное") семейство распределений. Принадлежит ли распределение Пуассона этому семейству? (показать)
2. Разберите в лекциях Воронцова теорему о линейности разделяющей поверхности в логистической регрессии. Ответьте на вопрос: оцениваются ли в этом методе априорные вероятности классов, и как они фигурируют в оптимизационной задаче?

13.4 Регуляризация

3. В терминах перехода от правдоподобия к функциям потерь, какой вероятностный смысл будут иметь регуляризаторы? $l1$-регуляризатор? $l2$?
4. Получите формулы для шага метода стохастического градиента в случае $l2$-регуляризованной логистической регрессии.

13.5 Дополнительные вопросы для изучения

5. Изучите вопрос применения для решения оптимизационных задач покоординатного спуска и метода Ньютона. Какие есть преимущества у каждого из них? Какие недостатки?
6. Изучите методы оптимизации, используемые в логистической регрессии в библиотеке liblinear (см. документацию).
7. Попробуйте сравнить покоординатный спуск и метод Ньютона в серии экспериментов на нескольких выборках, чтобы проверить утверждения из первого пункта (что-то из них может оказаться правдой, а что-то — домыслами)
8. Логистическую регрессию в многоклассовом случае называют MaxEnt Classifier. Попробуйте найти объяснения, почему, и разобраться с тем, как устроена многоклассовая версия.
9. Выясните, что такое дивергенция Кульбака-Лейблера и как она связана с энтропией.
10. Получите логистическую регрессию не максимизируя энтропию, а минимизируя или максимизируя дивергенцию Кульбака-Лейблера для некоторых распределений. Какой смысл в этой минимизации/максимизации?

Глава 14

Линейная регрессия

14.1 Формулировка задачи

Пусть, для начала, есть векторы признаков $x_1, \cdots, x_l$ для l объектов. На них же известны значения прогнозируемой величины $y_1, \cdots, y_l$. Будем пытаться аппроксимировать зависимость этой величины от признаков линейной: $y \approx \hat{y} =< w, x >= w^T x$, где w - вектор параметров, который мы хотим настроить на обучающей выборке. Здесь мы также имеем ввиду, что могли бы рассматривать более широкий класс зависимостей $\hat{y} =< w, x > +w_0$, но мысленно добавив к x фиктивный признак $x_0 = 1$ (равный единице для всех объектов), мы всегда сведем задачу к предыдущей.

Пусть, далее, мы хотим минимизировать суммарную потерю на обучающей выборке:

$$Q(w) = \sum_{i=1}^{l} L(y_i, \hat{y}_i) \to \min_w$$

И пусть, кроме того, функция потерь квадратичная: $L(y_i, \hat{y}_i) = (y_i - \hat{y}_i)^2$.

Наконец, пусть X – матрица признаков, строки которой соответствуют признаковым описаниям объектов $x_1, \cdots, x_l$ (т.е. в матрице l строк), вектором y будем обозначать вектор длины l с координатами y_i, вектором $\hat{y}$ – вектор длины l с координатами $\hat{y}_i$.

1. Как тогда выразить вектор $\hat{y}$ через w и X?
2. Как записать функционал $Q(w)$ через $\hat{y}$ и y? Что получится, если подставить выражение для $\hat{y}$?
3. Как выражается $\hat{y}$ через столбцы матрицы X? В каком линейном пространстве лежит $\hat{y}$ (в терминах линейных обочек набора векторов)?

14.2 Нормальное уравнение (normal equation)

1. Покажите справедливость следующего выражения:
$$\frac{\partial}{\partial x}(Ax + b)^T(Ax + b) = 2A^T(Ax + b)$$
2. Приравняв производную $Q(w)$ по вектору w к нулю, выразите вектор весов w в точке минимума через X и y. Полученное выражение носит название $NormalEquation$. Запомните, как оно получается — пригодится на экзамене.

14.3 Геометрическая интерпретация

1. Вспомните, что у вас получалось в последнем вопросе из раздела "формулировка задачи". Мы хотим минимизировать сумму квадратов отклонений, значит мы хотим для заданного вектора y найти в этом линейном пространстве точку $\hat{y}$, которая будет самой близкой к вектору y. Нетрудно сообразить, что такой точкой будет проекция y на линейное пространство, в котором должен жить $\hat{y}$. Чему в этом случае равно скалярное произведение $< x^{(j)}, \hat{y} - y >$, где $x^{(j)}$ - j-ый столбец матрицы X?
2. Исходя из предыдущего пункта, чему будет равно произведение
$$X^T(\hat{y} - y) \ ?$$
3. Закончите геометрический вывод нормального уравнения, подставив сюда выражение для $\hat{y}$ и выразив w.

14.4 Вероятностная интерпретация

4. Вспомните формулу плотности одномерного нормального распределения и как оцениваются параметры распределения методом максимального правдоподобия.
5. Предположим, что для каждого объекта x_i наблюдаемое значение величины y_i распределено нормально с матожиданием $\hat{y}_i$ (некоторым "истинным значением"в рамках линейной модели, которое мы хотим оценить) и дисперсией σ^2, одинаковой для всех i. Напишите минимизируемый функционал в задаче оценки $\hat{y}_i$ по методу максимального правдоподобия.
6. Покажите, что метод максимумального правдоподобия в этом случае приводит к минимизации суммы квадратов отклонений.
7. Теперь вы видите, что есть связь между функцией потерь и нашим преставлением о распределении "правильных ответов". А какое распределение привело бы к минимизации суммы модулей отклонений?

14.5 $l2$-регуляризация: гребневая регрессия (ridge regression)

8. Давайте теперь добавим к функционалу $Q(w)$ штрафное слагаемое $\tau||w||^2$. Дифференцируя по вектору w получите новое выражение для его оптимального значения.
9. Покажите, что это это дает тот же результат, что и добавление к матрице X^TX единичной матрицы I, умноженной на τ (с последующим применением стандартной формулы).
10. Модификация матрицы из предыдущего пункта изменяет собственные числа матрицы, меняя число обусловленности и позволяя обратить X^TX, если до модификации обращение было неустойчиво. Но при этом сохраняются собственные векторы матрицы. Покажите, как меняются минимальное и максимальное собственное число и число обусловленности матрицы. А затем, — что собственные векторы остаются теми же.

14.6 $l1$-регуляризация: лассо Тибширани (LASSO)

11. Применив теорему Куна-Таккера, покажите, что добавление ограничения $||w||_{l1} < r$ равносильно добавлению штрафного слагаемого $\tau||w||_{l1}$ в функционал $Q(w)$ для некоторого τ.

12. Попробуйте с помощью теоремы или из геометрических соображений объяснить, почему минимум попадает в "угловые точки"шаров $l1$-нормы (т.е. почему зануляются некоторые коэффициенты w), и почему с увеличением τ нулей будет становиться больше.

14.7 Дополнительные вопросы

13. Покажите, что если предварительно центрировать выборку, параметр сдвига w_0 получится равным нулю.
14. Представьте теперь, что вы пытаетесь восстановить прямую по известным ее точкам на изображении. Здесь вам уже захочется минимизировать не сумму квадратов отклонений по y, а сумму квадратов расстояний от известных точек до прямой. Как в этом случае будут выглядеть формулы для коэффициентов искомой прямой $w_x x + w_y y + w_0 = 0$?
15. Придумайте изящный ответ на предыдущий вопрос с помощью метода главных компонент.

Благодарности

Этого сборника не возникло бы, если бы не прекрасные преподаватели, которые учили меня, и если бы не замечательные студенты, у которых мне уже повезло вести занятия и, кстати, тоже многому научиться. И тем и другим я обязан своей любовью к преподаванию. Также я глубоко благодарен коллегам из ABBYY и Яндекса за тот опыт в машинном обучении и преподавании, который мне посчастливилось перенимать у них.

За написание решений первых задач сборника хотелось бы поблагодарить Анастасьева Даниила, Сандрикову Марию и Даниляка Александра, подготовивших эти решения при прохождении курса машинного обучения в МФТИ и согласившихся предоставить исходники для включения решений в сборник. Кроме того, я очень признателен Фонареву Александру, проводящему прекрасные семинары по курсу машинного обучения в ШАДе, за проявленный интерес к подобному сборнику задач, который подстегнул меня не затягивать подготовку первой версии бесконечно.

В. Кантор

Литература

[1] Воронцов К.В. — Лекции по машинному обучению [ссылка]

[2] Hastie, Tibshirani, Friedman — Elements of statistical learning [ссылка]