Урок 3

Понижение размерности и отбор признаков

3.1. Зачем отбирать признаки?

3.1.1. Задача отбора признаков

Эта неделя посвящена отбору признаков и понижению размерности. В первую очередь необходимо разобраться, зачем работать с признаками и отбирать их.

Во-первых, признаков может быть слишком много, больше чем нужно. Это может возникнуть в ситуациях, когда используется вся имеющаяся на данный момент информация, потому что неизвестно, какая её часть может понадобиться, а какая — нет. В таких случаях можно повысить качество решения задачи, выбирая только действительно важные признаки. Существует другой подход: можно сформировать новые признаки на основе старых, таким образом признаков станет меньше, но их информативность сохранится.

Во-вторых, существуют признаки, из-за которых при решении задачи возникает много проблем. Это шумовые признаки — признаки, которые не связаны с целевой переменной и никак не относятся к решаемой задаче. К сожалению, не всегда можно понять по обучающей выборке, что в ней присутствуют такие признаки.

Полезно рассмотреть несколько примеров присутствия шумовых признаков в данных. Пусть в выборку добавляют 1000 признаков. Значения каждого признака генерируется из стандартного нормального распределения. Понятно, что эти признаки бесполезны, они никак не помогут решить задачу. Но, поскольку их много, может так оказаться (из соображений теории вероятностей), что один из них коррелирует с целевой переменной. При этом он будет коррелировать только на обучающей выборке, а на контрольной выборке корреляции наблюдаться не будет, поскольку признак абсолютно случайный. Однако внутри модели этот признак может быть учтён как важный и иметь какой-то вес. Получается, что модель зависит от признака, который никак не помогает решить задачу. Из-за этого качество модели и ее обобщающая способность окажутся ниже, чем хотелось бы.

Другой пример. Пусть для решения задачи используется решающее дерево. В выборке присутствуют 1000 признаков, все они информативные. Но при этом, чтобы учесть каждый из них хотя бы один раз, понадобится дерево глубины как минимум 10. У этого дерева будет около 1000 листьев. Для того чтобы построить хорошие прогнозы и избежать переобучения, необходимо, чтобы в каждый лист попало большое количество примеров, а значит обучающая выборка должна быть очень большой. При этом нужно обратить внимание на следующее: такое дерево учтет каждый признак один раз. Для того чтобы учесть признаки большее число раз, понадобится более глубокое дерево, и необходимый для обучения такого алгоритма размер выборки увеличится ещё сильнее.

Еще одна причина, по которой может понадобится отбирать признаки — это ускорение модели. Дело в том, что чем больше признаков, тем более сложная модель получается, и тем больше времени необходимо, чтобы построить прогноз. Существуют задачи, в которых прогнозы нужно строить очень быстро, например, выдача рекомендаций товаров на сайте интернет-магазина. Пользователь что-то ищет, нажимает на ссылку в поисковой выдаче и переходит на страницу интересующего его товара. На этой странице есть поле, в котором показываются рекомендации к этому товару, например похожие товары, которые должна выдавать модель. Важно, чтобы она выдавала рекомендации очень быстро, страница не должна долго загружаться, чтобы пользователь не подумал, что с сайтом что-то не так, и не ушел к конкуренту. В этом случае необходимо,

чтобы модель была очень быстрой, и один из подходов к ускорению модели — это отбор признаков, которых достаточно, чтобы прогнозы были хорошими.

3.1.2. Метода отбора признаков

Опишем некоторые подходы к отбору признаков. Самый простой — это одномерный подход. В нём оценивается связь каждого признака с целевой переменной, например, измеряется корреляция. Такой подход — довольно простой, он не учитывает сложные закономерности, в нём все признаки считаются независимыми, тогда как в машинном обучении модели учитывают взаимное влияние признаков, их пар или даже более сложные действия на целевую переменную. Этот подход не всегда хорош, но иногда его можно использовать, чтобы ранжировать признаки, найти наиболее мощные среди них.

Более сложные подходы к отбору признаков могут быть устроены следующим образом: некоторым способом перебираются различные подмножества признаков, для каждого такого подмножества обучается модель, которая использует только эти признаки, и оценивается ее качество. В итоге выбирается то подмножество, которое дает наилучшее качество. Такие подходы оказываются сложными из-за того, что нужно перебирать подмножества признаков. Понятно, что всех подмножеств очень много, поэтому поиск должен быть более направленным.

Можно использовать саму модель, чтобы оценивать важность признаков и отбирать их. Например, в курсе уже шла речь о методе Lasso, L_1-регуляризации, которая позволяет отбирать признаки с помощью линейных моделей. Похожие методы используются и в решающих деревьях, случайном лесе и градиентном бустинге — об этом будет рассказано далее.

Наконец, может сложиться ситуация, когда все признаки важны, но их слишком много. В этом случае используются методы понижения размерности — такое построение новых признаков на основе старых, чтобы при этом сохранялось максимальное количество информации из исходных признаков.

3.2. Одномерный отбор признаков

Самые простые и наивные методы отбора признаков — это одномерные методы, о них пойдёт речь в этой части.

3.2.1. Постановка задачи

Необходимо ввести следующие обозначения:

- x_{ij} — значение признака j на объекте i,
- $\overline{x_j}$ — среднее значение признака j по всей выборке,
- y_i — значение целевой переменной или ответа на объекте i,
- $\overline{y}$ — среднее значение целевой переменной на всей выборке.

Задача — оценить предсказательную силу (информативность) каждого признака, то есть насколько хорошо по данному признаку можно предсказывать целевую переменную. Данные оцененной информативности можно использовать, чтобы отобрать k лучших признаков или признаки, у которых значение информативности больше порога (например, некоторой квантили распределения информативности).

3.2.2. Отбор признаков с использованием корреляции

Один из самых простых методов измерения связь между признаком и ответами — это корреляция:

$$R_j = \frac{\sum\limits_{i=1}^{\ell}(x_{ij} - \bar{x}_j)(y_i - \bar{y})}{\sqrt{\sum\limits_{i=1}^{\ell}(x_{ij} - \bar{x}_j)^2 \sum\limits_{i=1}^{\ell}(y_i - \bar{y})^2}}$$

Чем больше по модулю корреляция между признаком и целевой переменной, тем более информативным является данный признак. При этом она максимальна по модулю ($R_j = \pm 1$), если между признаком и целевой

переменной есть линейная связь, то есть если целевую переменную можно строго линейно выразить через значение признака. Это означает, что корреляция измеряет только линейную информативность, то есть способность признака линейно предсказывать целевую переменную. Вообще говоря, корреляция рассчитана на вещественные признаки и вещественные ответы. Тем не менее, её можно использовать в случае, если признаки и ответы бинарные (имеет смысл кодировать бинарный признак с помощью значений ± 1).

3.2.3. Использование бинарного классификатора для отбора признаков

Пусть решается задача бинарной классификации, и необходимо оценить важность признака j для решения именно этой задачи. В этом случае можно попробовать построить классификатор, который использует лишь этот один признак j, и оценить его качество. Например, можно рассмотреть очень простой классификатор, который берёт значение признака j на объекте, сравнивает его с порогом t, и если значение меньше этого порога, то он относит объект к первому классу, если же меньше порога — то к другому, нулевому или минус первому, в зависимости от того, как мы его обозначили. Далее, поскольку этот классификатор зависит от порога t, то его качество можно измерить с помощью таких метрик как площадь под ROC-кривой или Precision-Recall кривой, а затем по данной площади отсортировать все признаки и выбирать лучшие.

3.2.4. Использование метрик теории информации для отбора признаков

Существует ещё один подход в одномерном оценивании качества признаков, который основан на метриках, используемых в теории информации. Примером такой метрики является взаимная информация, или mutual information. Она рассчитана на случай, когда и признак, и целевая переменная — дискретные.

Пусть необходимо решить задачу многоклассовой классификации. В этом случае целевая переменная принимает m различных значений:

$$1, 2, \ldots, m,$$

а признак — n значений:

$$1, 2, ..., n.$$

Поскольку для метрики взаимной информации не важны конкретные значения признаков, можно обозначать их с помощью натуральных чисел. Вероятностью некоторого события будем обозначать долю объектов, для которых это событие выполнено. Например, вероятность того, что признак принимает значение v, а целевая переменная — k, вычисляется следующим образом:

$$P(x_j = v, y = k) = \frac{1}{\ell} \sum_{i=1}^{\ell} [x_{ij} = v][y_i = k]$$

Или, например, вероятность того, что признак равен v:

$$P(x_j = v) = \frac{1}{\ell} \sum_{i=1}^{\ell} [x_{ij} = v]$$

Взаимная информация между признаком j и целевой переменой вычисляется по следующей формуле:

$$MI_j = \sum_{v=1}^{n} \sum_{k=1}^{m} P(x_j = v, y = k) \log \frac{P(x_j = v, y = k)}{P(x_j = v)P(y = k)}$$

Главная особенность взаимной информации состоит в следующем: она равна нулю, если признак и целевая переменная независимы. Если же между ними есть какая-то связь, то взаимная информация будет отличаться от нуля, причём она может быть как больше, так и меньше нуля. Это означает, что информативность признаков нужно оценивать по модулю взаимной информации.

3.2.5. Проблемы одномерного отбора признаков

У подхода, при котором важности всех признаков оцениваются по отдельности, есть свои недостатки. На рисунке 3.1 изображена двумерная выборка, для которой необходимо решить задачу классификации. Если спроецировать данную выборку на ось абсцисс, то она будет разделима, хотя и будут присутствовать ошибки. Если же спроецировать данную выборку на ось ординат, то все объекты разных классов перемешаются,

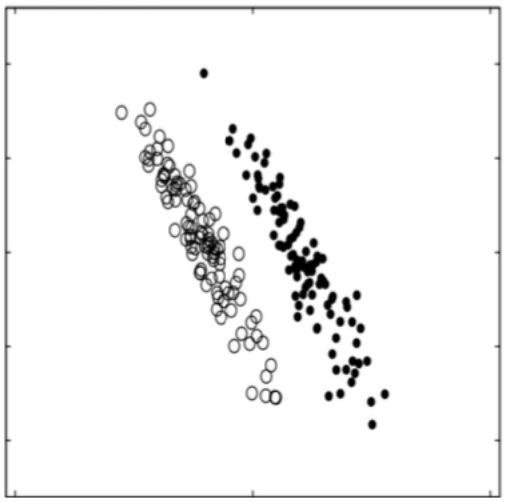

Рис. 3.1: Пример выборки, для которой необходимо решить задачу классификации

и выборка будет неразделима. В этом случае при использовании любого метода одномерного оценивания информативности первый признак будет информативен, а второй — совершенно неинформативен. Тем не менее, видно, что если использовать эти признаки одновременно, то классы будут разделимы идеально. На самом деле, второй признак важен, но он важен только в совокупности с первым, и методы одномерного оценивания информативности не способны это определить.

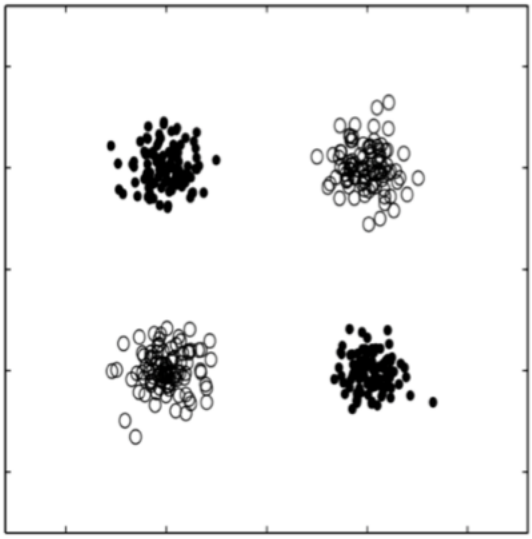

Рис. 3.2: Пример выборки, для которой необходимо решить задачу классификации

На рисунке 3.2 показана выборка, на которой одномерные методы оценки информативности работают ещё хуже. В этом случае, если спроецировать выборку на ось абсцисс или ординат, то объекты классов перемешаются, и в обоих случаях данные будут совершенно неразделимы. И согласно любому из описанных методов, оба признака неинформативны. Тем не менее, если использовать их одновременно, то, например,

решающее дерево может идеально решить данную задачу классификации.

3.3. Жадные методы отбора признаков

3.3.1. Принцип работы жадных методов

Жадные методы отбора признаков, по сути своей, являются надстройками над методами обучения моделей. Они перебирают различные подмножества признаков и выбирают то из них, которое дает наилучшее качество определённой модели машинного обучения.

Данный процесс устроен следующим образом. Обучение модели считается черным ящиком, который на вход принимает информацию о том, какие из его признаков можно использовать при обучении модели, обучает модель, и дальше каким-то методом оценивается качество такой модели, например, по отложенной выборке или кросс-валидации. Таким образом, задача, которую необходимо решить, — это оптимизация функционала качества модели по подмножеству признаков.

3.3.2. Переборные методы

Самый простой способ решения данной задачи — это полный перебор всех подмножеств признаков и оценивание качества на каждом подмножестве. Итоговое подмножество — то, на котором качество модели наилучшее. Этот перебор можно структурировать и перебирать подмножества последовательно: сначала те, которые имеют мощность 1 (наборы из 1 признака), потом наборы мощности 2, и так далее. Это подход очень хороший, он найдет оптимальное подмножество признаков, но при этом очень сложный, поскольку всего таких подмножеств 2^d, где d — число признаков. Если признаков много, сотни или тысячи, то такой перебор невозможен: он займет слишком много времени, возможно, сотни лет или больше. Поэтому такой метод подходит либо при небольшом количестве признаков, либо если известно, что информативных признаков очень мало, единицы.

3.3.3. Метод жадного добавления

Если же признаков много и известно, что многие из них информативны, то нужно применять жадную стратегию. Жадная стратегия используется всегда, когда полный перебор не подходит для решения задачи. Например, может оказаться неплохой стратегия жадного наращивания (жадного добавления). Сначала находится один признак, который дает наилучшее качество модели (наименьшую ошибку Q):

$$i_1 = \operatorname{argmin} Q(i).$$

Тогда множество, состоящее из этого признака:

$$J_1 = \{i_1\}$$

Дальше к этому множеству добавляется еще один признак так, чтобы как можно сильнее уменьшить ошибку модели:

$$i_2 = \operatorname{argmin} Q(i_1, i), \qquad J_2 = \{i_1, i_2\}.$$

Далее каждый раз добавляется по одному признаку, образуются множества $J_3, J_4, \ldots$. Если в какой-то момент невозможно добавить новый признак так, чтобы уменьшить ошибку, процедура останавливается. Жадность процедуры заключается в том, что как только какой-то признак попадает в оптимальное множество, его нельзя оттуда удалить.

3.3.4. Алгоритм ADD-DEL

Описанный выше подход довольно быстрый: в нем столько итераций, сколько признаков в выборке. Но при этом он слишком жадный, перебирается слишком мало вариантов. Процедуру можно усложнить. Один из подходов к усложнению — это алгоритм ADD-DEL, который не только добавляет, но и удаляет признаки из оптимального множества. Алгоритм начинается с процедуры жадного добавления. Множество признаков наращивается до тех пор, пока получается уменьшить ошибку, затем признаки жадно удаляются из подмножества, то есть перебираются все возможные варианты удаления признака, оценивается ошибка и удаляется тот признак, который приводит к наибольшему уменьшению ошибки на выборке. Эта процедура повторяет

добавление и удаление признаков до тех пор, пока уменьшается ошибка. Алгоритм ADD-DEL всё еще жадный, но при этом он менее жадный, чем предыдущий, поскольку может исправлять ошибки, сделанные в начале перебора: если вначале был добавлен неинформативный признак, то на этапе удаления от него можно избавиться.

3.4. Отбор признаков на основе моделей

3.4.1. Использование линейных моделей для отбора признаков

Для оценки информативности признаков и их отбора можно использовать обученные модели, например, линейные.

Линейная модель вычисляет взвешенную сумму значений признаков на объекте:

$$a(x) = \sum_{j=1}^{d} w_j x^j$$

При этом возвращается сама взвешенная сумма, если это задача регрессии, и знак этой суммы, если это задача классификации. Если признаки масштабированы, то веса при признаках можно интерпретировать как информативности: чем больше по модулю вес при признаке j, тем больший вклад этот признак вносит в ответ модели. Однако если признаки не масштабированы, то так использовать веса уже нельзя. Например, если есть два признака, и один по масштабу в 1000 раз меньше другого, то вес первого признака может быть очень большим, только чтобы признаки были одинаковыми по масштабу.

Если необходимо обнулить как можно больше весов, чтобы линейная модель учитывала только те признаки, которые наиболее важны для нее, можно использовать L1-регуляризацию. Чем больше коэффициент при L1-регуляризаторе, тем меньше признаков будет использовать линейная модель.

3.4.2. Применение решающих деревьев для отбора признаков

Еще один вид моделей, обсуждаемых в предыдущем курсе, — это решающие деревья. Решающие деревья строятся «жадно»: они растут от корня к листьям, и на каждом этапе происходит попытка разбить вершину на две. Чтобы разбить вершину, нужно выбрать признак, по которому будет происходить разбиение, и порог, с которым будет сравниваться значение данного признака. Если значение признака меньше этого порога, то объект отправляется в левое поддерево, если больше — в правое поддерево. Выбор признака и порога осуществляется по следующему критерию:

$$Q(X_m, j, t) = \frac{|X_l|}{|X_m|} H(X_l) + \frac{|X_r|}{|X_m|} H(X_r) \to \min_{j,t},$$

где $H(X)$ — это критерий информативности. Ранее в курсе были рассмотрены различные критерии информативности. Например, в задаче регрессии используется функционал среднеквадратичной ошибки, а в классификации — критерий Джини или энтропийный критерий.

Критерий Q вычисляет взвешенную сумму критериев информативности $H(X)$ в обеих дочерних вершинах — левой и правой. Чем меньше данная взвешенная сумма, тем больше признак j и порог t подходят для разбиения.

Если в данной вершине происходит разбиение по признаку j, то чем сильнее уменьшается значение критерия информативности, тем важнее этот признак оказался при построении дерева. Таким образом, можно оценивать важность признака на основе того, насколько сильно он смог уменьшить значение критерия информативности. Пусть, в вершине m произведено разбиение по признаку j. Тогда можно вычислить уменьшение критерия информативности в ней по следующей формуле:

$$H(X_m) - \frac{|X_l|}{|X_m|} H(X_l) - \frac{|X_r|}{|X_m|} H(X_r)$$

R_j - сумма данного уменьшения по всем вершинам дерева, в которых происходило разбиение по признаку j. Чем больше R_j, тем важнее данный признак был при построении дерева.

3.4.3. Использование композиций алгоритмов для отбора признаков

Сами по себе решающие деревья не очень полезны, но они очень активно используются при построении композиций, в частности, в случайных лесах и в градиентном бустинге над деревьями. В данных композициях измерить важность признака можно аналогичным образом: суммируется уменьшение критерия информативности R_j по всем деревьям композиции, и чем больше данная сумма, тем важнее признак j для композиции. То есть признаки оцениваются с помощью того, насколько сильно они смогли уменьшить значение критерия информативности в совокупности по всем деревьям композиции.

Для случайного леса можно предложить еще один интересный способ оценивания информативности признаков. В этой композиции каждое базовое дерево b_n обучается по подмножеству объектов обучающей выборки. Таким образом, есть объекты, на которых дерево не обучалось, и набор этих объектов является валидационной выборкой для дерева n. Такая выборка называется out-of-bag. Итак, метод заключается в следующем: ошибку Q_n базового дерева b_n оценивают по out-of-bag-выборке и запоминают. После этого признак j превращают в абсолютно бесполезный, шумовой: в матрицу «объекты-признаки» все значения в столбце j перемешивают. Затем то же самое дерево b_n применяют к данной выборке с перемешанным признаком j и оценивают качество дерева на out-of-bag-подвыборке. Q'_n — ошибка out-of-bag-подвыборке, она будет тем больше, чем сильнее дерево использует признак j. Если он активно используется в дереве, то ошибка сильно уменьшится, поскольку значение данного признака испорчено. Если же данный признак совершенно не важен для дерева и не используется в нем, то ошибка практически не изменится. Таким образом, информативность признака j оценивают как разность

$$Q'_n - Q_n.$$

Далее эти информативности усредняют по всем деревьям случайного леса, и чем больше будет среднее значение, тем важнее признак. На практике оказывается, что информативности, вычисленные описанным образом, и информативности, вычисленные как сумма уменьшения критерия информативности, оказываются очень связаны между собой.

3.5. Понижение размерности

3.5.1. Примеры использования методов понижения размерности

Зачем нужно понижать размерность и чем этот подход отличается от отбора признаков? Для ответа на этот вопрос полезно рассмотреть несколько примеров.

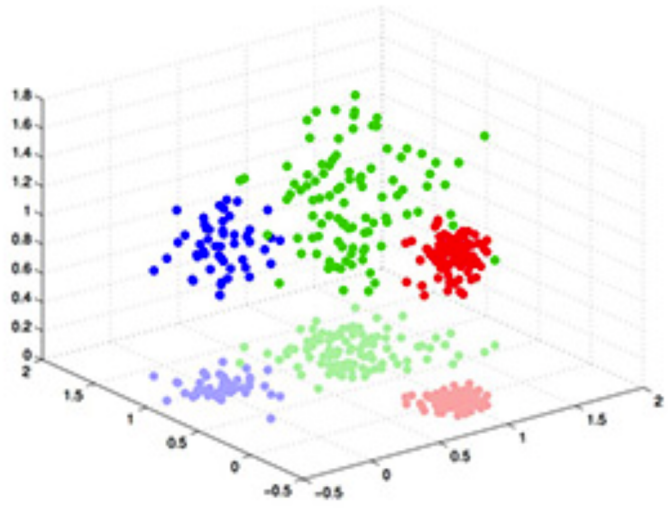

Рис. 3.3: Пример выборки, для которой необходимо решить задачу классификации

На рисунке 3.3 изображена выборка с тремя размерностями. Видно, что если убрать из нее признак, отложенный по оси Z, получится двумерная выборка, в которой синий, зеленый и красный кластеры будут разделимы даже линейными методами.

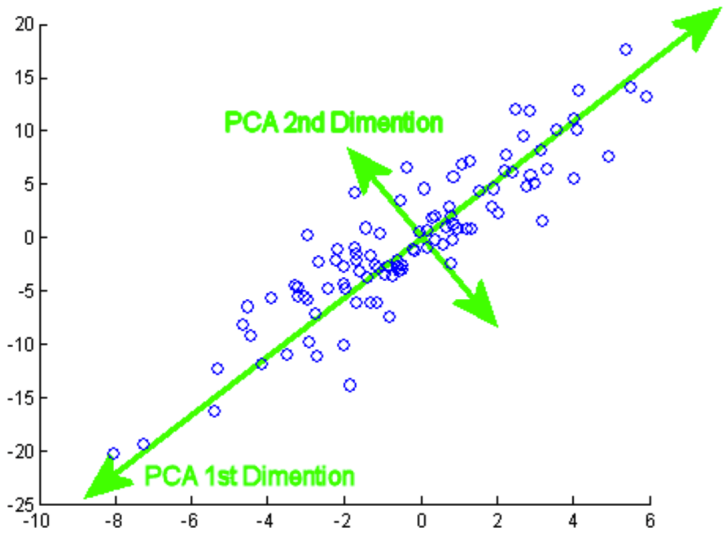

Рис. 3.4: Пример выборки с линейно зависимыми признаками

На рисунке 3.4 показан более сложный случай. В данной выборке оба признака значимые, но при этом они линейно зависимые, и этим можно воспользоваться, чтобы устранить избыточность в данных. Однако отбора признаков для этого не хватит: необходимо сформировать новый признак на основе двух исходных.

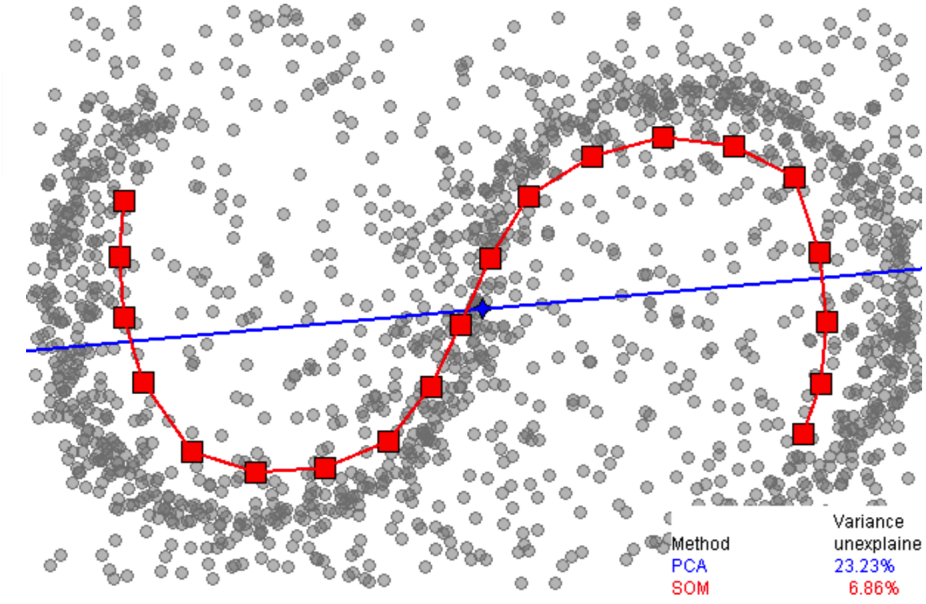

Рис. 3.5: Пример данных, которые необходимо спроецировать на кривую

Бывают и еще более сложные случаи, как, например, на рисунке 3.5. Здесь также можно спроецировать выборку на некоторую кривую, но при этом кривая очень нелинейная, и её, скорее всего, будет сложно найти.

Эти примеры подводят к задаче понижения размерности, которая состоит в формировании новых признаков на основе исходных. При этом количество признаков становится меньше, но они должны сохранять в себе как можно больше информации, присутствующей в исходных признаках.

3.5.2. Метод случайных проекций

Один из основных подходов к понижению размерности — это линейный подход, в котором каждый новый признак представляет собой линейную комбинацию исходных признаков. Необходимо ввести обозначения:

- D — количество исходных признаков;
- x_{ij} — это значение исходного признака j на объекте выборки i;
- d — число новых признаков;
- z_{ij} — это значение нового признака j на объекте выборки i

Новый признак j на объекте i линейно выражается через исходные признаки на этом же объекте:

$$z_{ij} = \sum_{k=1}^{D} w_{jk} x_{ik}$$

где w_{jk} — вес, который показывает, какой именно вклад исходный признак k даёт в новый признак j на каждом объекте.

Один из простейших подходов к такому понижению размерности — это метод случайных проекций, в котором веса генерируются случайно, например, из нормального распределения:

$$w_{jk} \sim \mathcal{N}(0, \frac{1}{d}),$$

где d — это количество новых признаков. Данный подход обосновывает лемма Джонсона-Линденштраусса.

Лемма (Джонсона-Линденштраусса) *Если в выборке мало объектов, но много признаков, то её можно спроецировать в пространство меньшей размерности так, что расстояния между объектами практически не изменятся (то есть топология в новом признаковом пространстве сохранится).*

При этом, если необходимо, чтобы расстояния изменились не больше, чем на ϵ, то размерность нового признакового пространство должна удовлетворять условию

$$d > \frac{8 \ln \ell}{\varepsilon^2},$$

где l — количество объектов в выборке.

В лемме утверждается только, что существует такая проекция и что она будет сохранять расстояния между парами признаков. Однако на практике оказывается, что если использовать d, соответствующее описанной выше формуле, то даже метод случайных проекций так понижает размерность выборки, что расстояния сохраняются неплохо, и это понижение оказывается качественным.

Такой подход хорошо работает для текстов, когда пространства оказываются очень большими (обычно тексты кодируются при помощи мешка слов, и количество признаков — это число различных слов в тексте).

3.6. Метод главных компонент: постановка задачи

3.6.1. Метод главных компонент как линейный метод понижения размерности

Метод главных компонент — широко используемый способ понижения размерности. Задачу метода главных компонент можно вывести из линейного подхода к понижению размерности, где каждый новый признак линейно выражается через исходные:

$$z_{ij} = \sum_{k=1}^{D} w_{jk} x_{ik}$$

Если произвести небольшие преобразования:

$$z_{ij} = \sum_{k=1}^{D} w_{jk} x_{ik} = \sum_{k=1}^{D} x_{ik} w_{kj}^{T},$$

то это выражение можно записать в матричном виде:

$$Z = XW^T.$$

Чтобы у этого уравнения существовало единственное решение, необходимо ввести ограничение на матрицу весов, она должна быть ортогональной:

$$W^TW = I.$$

Итак, если данное требование будет выполнено, то можно получить формулу для матрицы X:

$$X = ZW$$

Поскольку в матрице Z будет меньше признаков, чем в X, то если ранг матрицы X больше, чем число новых признаков d, то данное равенство точно нельзя будет выполнить строго. В этом случае требуется, чтобы отклонение исходной матрицы признаков X от ZW было как можно меньше, чтобы эти матрицы были как можно сильнее похожи друг на друга. Размер этого отклонения будем вычислять с помощью нормы Фробениуса:

$$\|X - ZW\|^2 \to \min_{Z,W}$$

Норма Фробениуса матрицы — это сумма квадратов ее значений, аналог векторной нормы $l2$.

Получившаяся задача — это задача матричного разложения. Необходимо представить матрицу X в виде произведения двух матриц Z и W, которые будут иметь меньший ранг. То есть нужно уменьшить ранг матрицы, при этом потеряв как можно меньше информации в ней.

3.6.2. Метод главных компонент как способ проекции данных на гиперплоскость

Существует иной подход к постановке задачи метода главных компонент.

Пусть имеется выборка, изображенная на рисунке 3.6, и её необходимо спроецировать на некоторую прямую. В этом случае прямая будет тем лучше, чем меньше будет ошибка проецирования сумма по всей выборке расстояний от объекта до его проекции на эту прямую. Чем меньше эти расстояния, тем лучше прямая приближает данные, тем меньше будет ошибка и тем больше информации сохраняется. В идеальном случае прямая должна проходить через все объекты выборки, но в рассматриваемой ситуации это невозможно.

В общем случае, когда признаков много, выборка проецируется на гиперплоскость. Из аналитической геометрии известно, что есть два способа задания гиперплоскости. Первый — с помощью вектора нормали, он использовался в линейных методах. Второй — с помощью направляющих векторов. Пусть в исходном пространстве размерности D строится гиперплоскость размерности $D-1$, тогда если выбрать на этой плоскости D линейно независимых векторов, то они будут однозначно задавать эту гиперплоскость. Если направляющие векторы составить в матрицу W, так что каждый столбец этой матрицы — это один направляющий вектор, то проекция точки x_i на данную гиперплоскость будет вычисляться по формуле $x_i * W$. Тогда для того чтобы уменьшить ошибку проецирования на гиперплоскость, необходимо минимизировать следующее выражение:

$$\sum_{i=1}^{\ell} \|x_i - x_iW\|^2 \to \min_W.$$

Задача ставится аналогично при проецировании объектов на гиперплоскость любой размерности $d \leqslant D-1$.

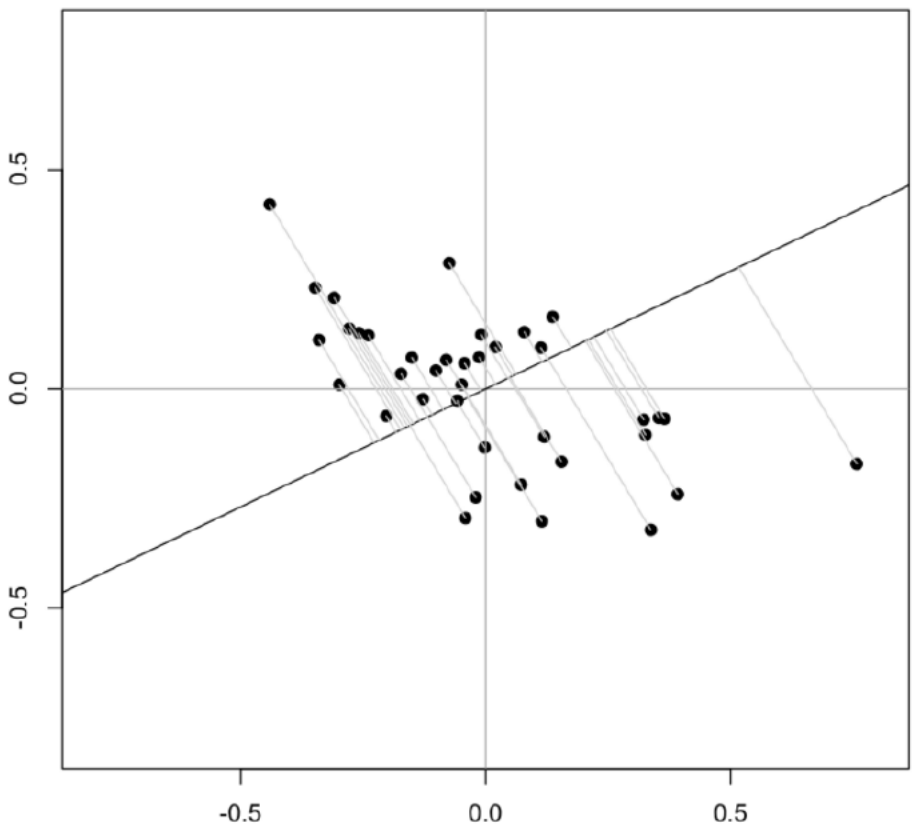

Рис. 3.6: Пример выборки, которую необходимо спроецировать на прямую

3.6.3. Максимизация дисперсии выборки после понижения размерности

Есть и третий взгляд на метод главных компонент. Пусть имеется выборка, показанная на рисунке 3.7, и требуется выбрать прямую, на которую можно будет оптимально спроецировать эту выборку. Синяя прямая лучше подходит для решения данной задачи, поскольку при проецировании на нее сохраняется гораздо больше информации выборки.

Формализовать понятие информации можно с помощью дисперсии: чем больше дисперсия выборки после проецирования на прямую, тем больше сохраняется информации. Для данного случая этот критерий хорошо подходит: дисперсия выборки после проецирования на синюю прямую будет гораздо больше, чем после проецирования на красную прямую.

Формально дисперсию выборки после проецирования можно записать следующим образом:

$$\sum_{j=1}^{d} \mathrm{w}_j^T X^T X \mathrm{w}_j \to \max_W$$

Чем больше значение этой суммы, тем больше оказывается дисперсия выборки после проецирования на гиперплоскость, которая задается матрицей весов W. Таким образом, это выражение нужно максимизировать, чтобы сохранить как можно больше информации.

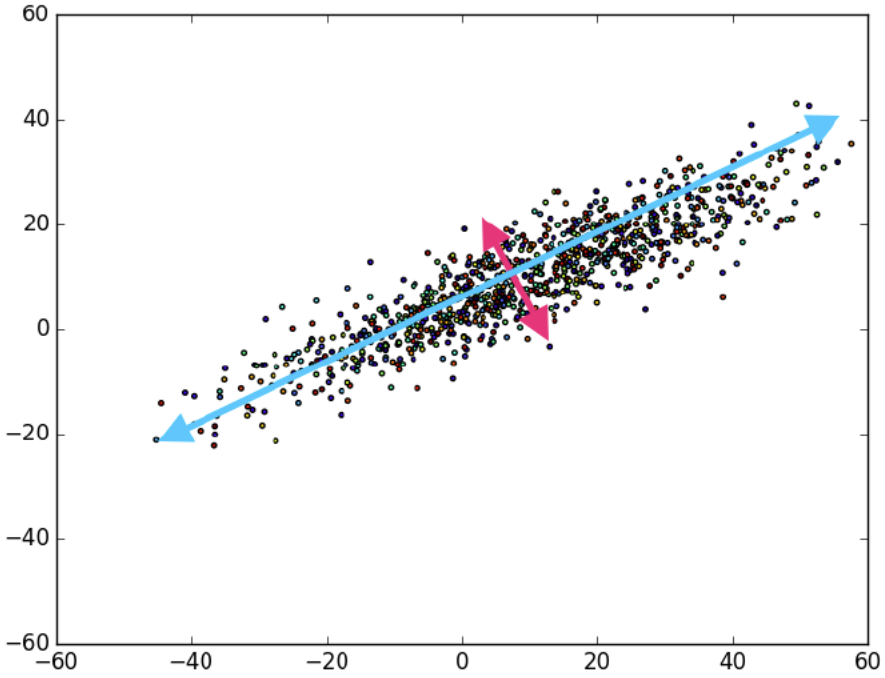

Рис. 3.7: Пример выборки, для которой необходимо решить задачу понижения размерности

3.7. Метод главных компонент: решение

3.7.1. Вывод решения задачи метода главных компонент

Ранее была описана формулировка задачи метода главных компонент, теперь необходимо её решить. Одна из постановок задачи метода главных компонент — это максимизация дисперсии:

$$\begin{cases} \sum\limits_{j=1}^{d} \mathrm{w}_j^T X^T X \mathrm{w}_j \to \max\limits_{W} \\ W^T W = I \end{cases}$$

В первой строке записана дисперсия после проецирования, а во второй — ограничение, обеспечивающее наличие единственного решения.

В методе главных компонент есть один нюанс: выражение, через которое записана дисперсия, будет означать именно дисперсию выборки только в том случае, если матрица объекты-признаки центрирована (среднее каждого признака равно нулю). Далее считается, что, выборка центрирована, и среднее из каждого столбца в матрице объекты-признаки уже вычли.

Итак, чтобы разобраться, как устроено решение этой задачи, необходимо сначала рассмотреть простой частный случай: требуется найти ровно одну компоненту, на которую проецируется вся выборка, так, чтобы дисперсия после проецирования была максимальной:

$$\begin{cases} \mathrm{w}_1^T X^T X \mathrm{w}_1 \to \max\limits_{\mathrm{w}_1} \\ \mathrm{w}_1^T \mathrm{w}_1 = 1 \end{cases}$$

Для решения подобных задач условной оптимизации необходимо выписать лагранжиан:

$$L(\underset{1}{\mathrm{w}}, \lambda) = \overset{T}{\underset{1}{\mathrm{w}}} X^T X \underset{1}{\mathrm{w}} - \lambda(\overset{T}{\underset{1}{\mathrm{w}}} \underset{1}{\mathrm{w}} - 1).$$

Далее этот лагранжиан необходимо продифференцировать по искомой величине:

$$\frac{\partial L}{\partial \mathrm{w}_1} = 2X^T X \underset{1}{\mathrm{w}} - 2\lambda \underset{1}{\mathrm{w}} = 0$$

После преобразований получается следующее выражение:

$$X^T X \underset{1}{\mathrm{w}} = \lambda \underset{1}{\mathrm{w}}.$$

Из него следует, что w_1 — это собственный вектор матрицы $X^T X$, и число λ является собственным значением, соответствующим этому вектору. После подстановки полученного выражения в функционал задачи, оказывается, что дисперсия выборки после проецирования будет равна собственному значению, соответствующему выбранному собственному вектору:

$$\overset{T}{\underset{1}{\mathrm{w}}} X^T X \underset{1}{\mathrm{w}} = \lambda$$

Таким образом, поскольку требуется максимизировать дисперсию, необходимо выбирать максимальное собственное значение и собственный вектор, который соответствует этому значению.

Итак, в методе главных компонент первая компонента — это собственный вектор матрицы $X^T X$, который соответствует максимальному собственному значению этой матрицы. Стоит обратить внимание, что $X^T X$ — это матрица ковариации, то есть именно та матрица, которая характеризует дисперсию выборки.

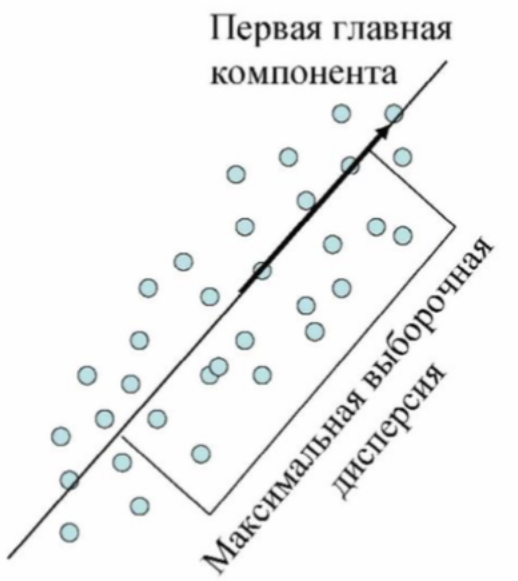

Рис. 3.8: Пример выборки, для которой необходимо решить задачу метода главных компонент

Визуально это выглядит следующим образом: есть облако точек (рисунок 3.8), и необходимо выбрать именно то направление, при проецировании на которое сохраняется как можно больше дисперсии. Это направление и будет задаваться первым собственным вектором матрицы ковариации.

Если продолжить выкладки и дальше искать оптимальные направления, то обнаружится, что, $w_2, w_3, \ldots, w_d$ — собственные векторы матрицы $X^T X$, соответствующие наибольшим собственным значениям $\lambda_2, \lambda_3, \ldots, \lambda_d$. Отсюда же можно получить, какая доля дисперсии сохранилась после проецирования выборки на главные компоненты:

$$\frac{\sum_{i=1}^{d} \lambda_i}{\sum_{i=1}^{D} \lambda_i}$$

3.7.2. Использование сингулярного разложения

При решении задачи метода главных компонент оказывается полезным сингулярное разложение:

$$X = UDV^T,$$

где U и V — это ортогональные матрицы, а D — диагональная матрица. Столбцы матрицы U — это собственные векторы матрицы XX^T, столбцы матрицы V — это собственные векторы матрицы X^TX, а на диагонали матрицы D стоят собственные значения этих матриц, которые, оказывается, совпадают (с точностью до некоторого количества нулевых собственных значений у той матрицы, которая имеет большую размерность). Собственные значения матриц X^TX и XX^T называются сингулярными числами.

Итак, для решения задачи главных компоненты необходимо совершить следующие действия: найти сингулярное разложение матрицы X, сформировать матрицу весов W из собственных векторов (из столбцов матрицы V, соответствующих максимальным сингулярным числам), и после этого произвести преобразование

$$Z = XW,$$

где Z является матрицей объекты-признаки для нового сокращенного признакового описания.