Машинное обучение

Профессор Адил Хан

Цели

1. Краткое описание прошедшей недели
2. Обратная связь по последней теме из рекомендованных исследований прошлой недели
   - Оптимизация с ограничениями
   - L1 и L2 как проблемы оптимизации с ограничениями
3. Данные высокой размерности
4. Что такое анализ главных компонент? Как он помогает работать с данными высокой размерности? Какова его объективная функция? Как она мотивируется?

Отражение

1. Как выбор "k" в kNN влияет на предсказания модели? Какие потенциальные последствия может иметь выбор очень маленького или очень большого "k"?
2. Помимо общепринятого евклидова расстояния, какие еще метрики расстояний можно использовать в kNN?
3. Почему кросс-валидация важна для оценки эффективности модели? Как она помогает снизить риски перекорректировки по сравнению с единственной тренировочно-тестовой разбивкой?
4. Каковы основные цели использования методов регуляризации L1 (Lasso) и L2 (Ridge) в линейной регрессии? Чем они отличаются по своему влиянию на коэффициенты модели?
5. Размышляя о регуляризации и выборе "k" в kNN, как эти методы связаны с компромиссом между смещением и дисперсией в машинном обучении?

Повторн ый пр р (1)

Классификатор Наивного Байеса

$$p(y_{new} = c | \boldsymbol{x}_{new}, \boldsymbol{X}, \boldsymbol{y}) = \frac{p(\boldsymbol{x}_{new}|c)p(c)}{\sum_{c=1}^{C} p(\boldsymbol{x}_{new}|\boldsymbol{c})\ p(c)}$$

$$p\left((x_1, \cdots, x_p)_{new} | c\right) = \prod_{i=1}^{p} p(x_i|c)$$

$$c_{NB} = \underset{c_j \in C}{\arg\max}\, P(c_j) \prod_i P(x_i \mid c_j)$$

Повторн

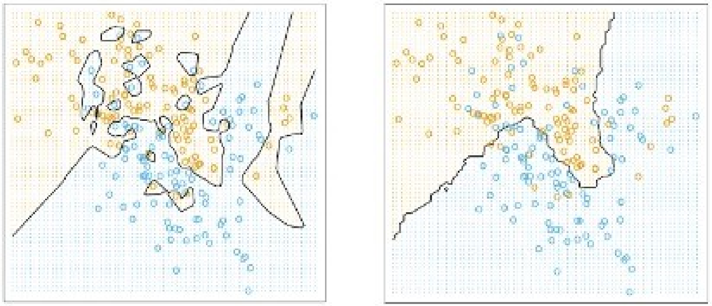
k-nearest Neighbor (KNN)

- Step 1: choose a value for k
- Step 2: Take the k neighbors of the new data point according to Euclidean distance
- Step 3: Among these k neighbor data points, count the number of points in each category
- Step 4: Assign the new data points to the category where you counted the most neighbors

Things to Remember about KNN

- Non parametric model
- Data is the model
- Curse of dimensionality
- Computational cost
- Feature scaling
- Handling missing data

$k = 1$ $\qquad$ $k = 15$



Data

Training Data | Val Data | Test Data

And we train model as follows

Training with $k = 1$ → Err = 5.2

Training with $k = 5$ → Err = 2.1 → Err = 2.4

Training with $k = 15$ → Err = 3.7

Повторный

Regularization

- Helps to reduce *overfitting*

$$Success = Goodness\ of\ the\ Fit + Simplicity\ of\ the\ model$$

$$\frac{1}{n}\sum_{i=1}^{n}(y_i - \boldsymbol{w}^T.\boldsymbol{x}_i)^2 \qquad\qquad \sum_{j=1}^{p} w_j^2$$

L_2 Regularization

$$\mathcal{L} = \frac{1}{n}\sum_{i=1}^{n}(y_i - \boldsymbol{w}^T.\boldsymbol{x}_i)^2 + \lambda\sum_{j=1}^{p} w_j^2$$

$$\lambda \geq 0\ is\ a\ tuning\ parameter$$

Ridge Regression

L_1 Regularization

$$\mathcal{L} = \frac{1}{n}\sum_{i=1}^{n}(y_i - \boldsymbol{w}^T.\boldsymbol{x}_i)^2 + \lambda\sum_{j=1}^{p} |w_j|$$

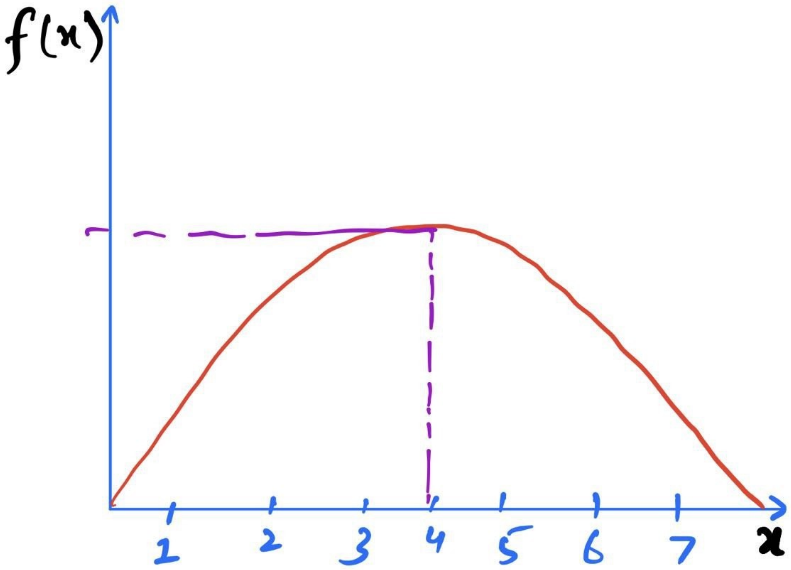
$$\lambda \geq 0\ is\ a\ tuning\ parameter$$

Lasso Regression

Почему регуляризация L_1 дает разреженные модели, а L_2 - нет?

Проблема оптимизации

- Математическая задача, в которой мы хотим МАКСИМИЗИРОВАТЬ или МИНИМИЗИРОВАТЬ заданную функцию

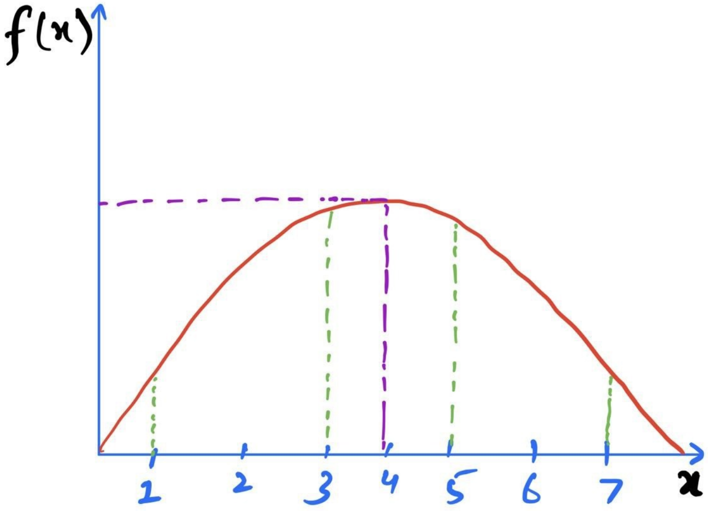


Решение: $x = 4$

Но что, если нам скажут, что x должно быть, странно?

Проблема оптимизации (2)

- Математическая задача, в которой мы хотим МАКСИМИЗИРОВАТЬ или МИНИМИЗИРОВАТЬ заданную функцию



Но что, если нам скажут, что x должно быть, странно?

Решение: x = 3

Ограниченные задачи оптимизации

- Оптимизационные задачи, в которых функция $f(x)$ должна быть максимизирована или минимизирована при условии (s.t.) некоторых ограничений (ограничений) $\phi(x)$

$$\min_{x}(f)$$

$$s.t.\ \phi(x)$$

$$\max(f\,x)$$

$$s.t.\ \phi(x)$$

Решение задач оптимизации с ограничениями

- Такие оптимизационные задачи решаются с помощью **метода множителей Лагранжа**

- В частности, мы берем нашу целевую функцию и ограничения (ограничения) и делаем следующее

  - Мы составляем новую целевую функцию
  - Эта новая целевая функция содержит как исходную цель, так и дополнительный член(ы)
  - Дополнительный термин(ы) представляет(ют) наше(их) ограничение(я)

Решение задач оптимизации с ограничениями (2)

В частности, мы берем нашу целевую функцию и ограничения (ограничения) и делаем следующее

- Мы составляем новую целевую функцию
- Эта новая целевая функция содержит как исходную цель, так и дополнительный член(ы)
- Дополнительный термин(ы) представляет(ют) наше(их) ограничение(я)

$$\underset{w}{\operatorname{argmin}}\ f(w) \quad \longrightarrow \quad \underset{w}{\operatorname{argmin}}\ f(w) - \alpha\,(g(w) - a)$$

в зависимости от $(w) < a$

при условии, что $\alpha > 0$

α называются множителями Лагранжа

Вот все подробности, которые вам нужно знать о них для этого курса

Альтернативные формы целей регуляризации

$$\mathcal{L} = \frac{1}{n}\sum_{i=1}^{n}(y_i - \mathbf{w}^T.\mathbf{x}_i)^2 + \lambda\sum_{j=1}^{p} w_j^2$$

$$\mathcal{L} = \frac{1}{n}\sum_{i=1}^{n}(y_i - \mathbf{w}^T.\mathbf{x}_i)^2 + \lambda\sum_{j=1}^{p} |w_j|$$

$$\text{мин}_{\mathbf{w}}\left\{\frac{1}{n}\sum_{i=1}^{n}(y_i - \mathbf{w}^T.\mathbf{x}_i)^2\right\} \; subject\ to \sum_{j=1}^{p} w_j^2 \le s$$

$$\text{мин}_{\mathbf{w}}\left\{\frac{1}{n}\sum_{i=1}^{n}(y_i - \mathbf{w}^T.\mathbf{x}_i)^2\right\} \; subject\ to \sum_{j=1}^{p} |w_j| \le s$$

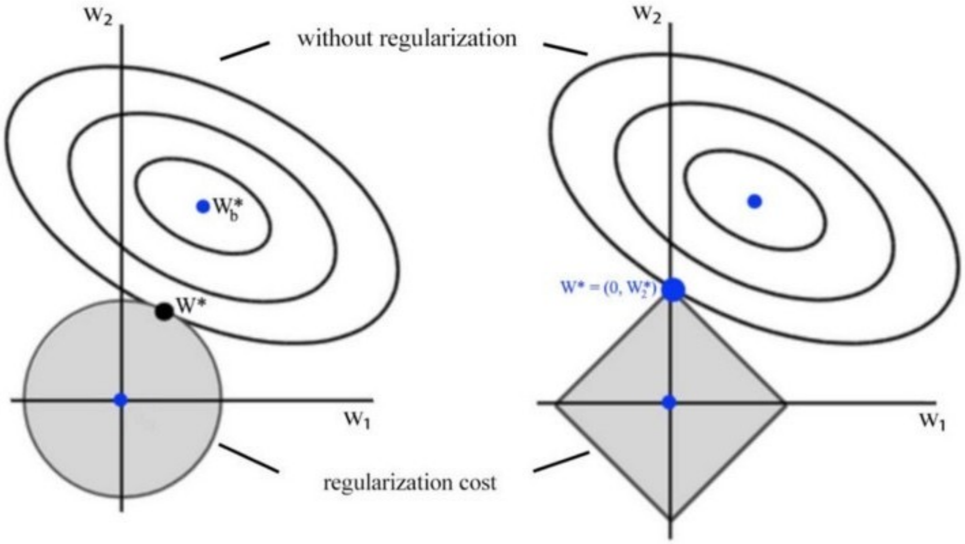
Таким образом, это ограниченные оптимизационные задачи

Ограничения в двумерном пространстве

$$w_1^2 + w_2^2 \leq s$$

$$|w_1| + |w_2| \leq s$$

Оптимизационная задача в двумерном пространстве



L2 regularization promotes small parameters   L1 regularization promotes sparse parameters

Данные высокой размерности

Проклятие размерности

1. В наши дни все сходят с ума по большим данным.

2. Большие данные - наш друг

3. Мы можем использовать его в различных творческих целях.

4. Но для начала давайте спросим себя, как данные могут стать БОЛЬШИМИ?

Проклятие размерности (2)

- Делаем данные большими

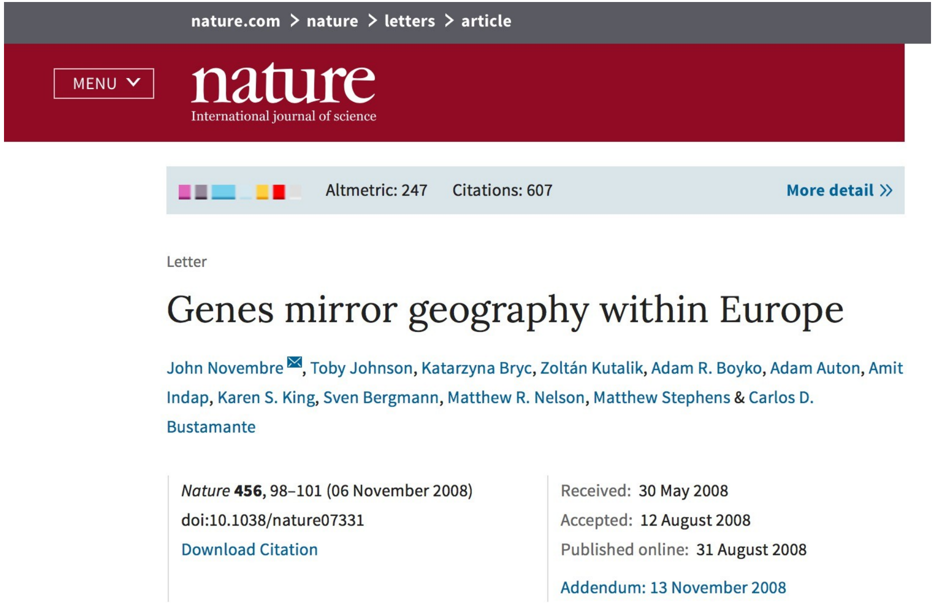
1. Возьмите огромное количество образцов
2. Измерение огромного количества измерений **ДЛЯ** каждого образца

Проклятие размерности (3)

- Пример данных высокой размерности

  - Предметы: 3192

  - Измерения: 500,000


nature.com > nature > letters > article

MENU

nature
International journal of science

Altmetric: 247 Citations: 607 More detail

Letter

Genes mirror geography within Europe

John Novembre, Toby Johnson, Katarzyna Bryc, Zoltán Kutalik, Adam R. Boyko, Adam Auton, Amit Indap, Karen S. King, Sven Bergmann, Matthew R. Nelson, Matthew Stephens & Carlos D. Bustamante

Nature **456**, 98–101 (06 November 2008)
doi:10.1038/nature07331
Download Citation

Received: 30 May 2008
Accepted: 12 August 2008
Published online: 31 August 2008
Addendum: 13 November 2008

Проклятие размерности (4)

- Данные высокой размерности - это
  - Сложно визуализировать
  - Сложно анализировать
  - Сложно понять - получить представление о данных (корреляция и прогнозы)

Общая проблема

- У нас есть набор данных из n точек данных, где каждая точка является p-мерной

$$X = \{(\boldsymbol{x}_i) | \boldsymbol{x}_i \in \mathbb{R}^p\}_{i=1}^{n}$$

- Количество параметров в модели машинного обучения обычно зависит от параметра p.
- Таким образом, если p очень велик, это может сделать оценку параметров сложной.
- Это также может затруднить визуализацию данных.

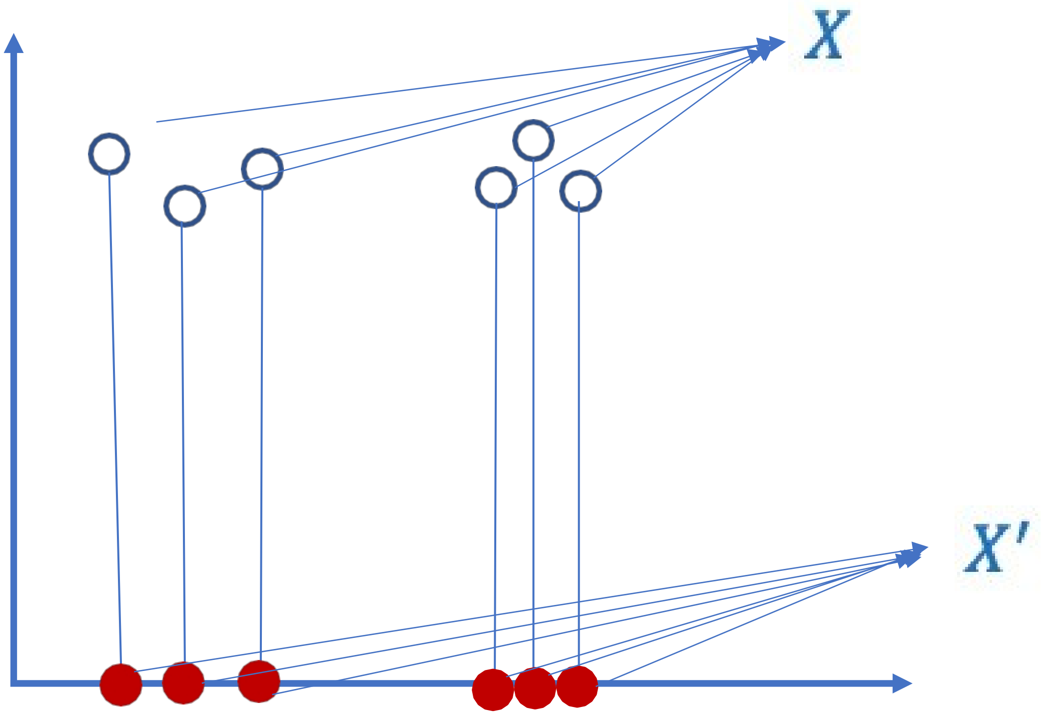
Решение

- Для решения этой задачи мы обычно преобразуем каждую p-мерную точку $\boldsymbol{x}_i$ в новую d-мерную точку $\boldsymbol{x}'_i$.
- Такой, что $d < p$

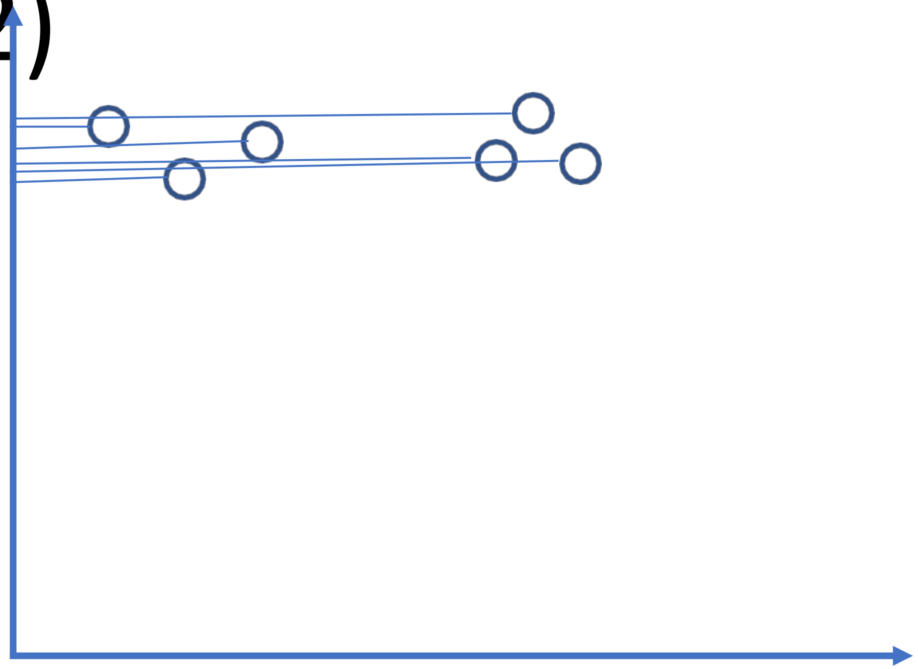
$$X' = \{(\boldsymbol{x}'_i) \mid \boldsymbol{x}'_i \in \mathbb{R}^d\}_{i=1}^{n}$$

- Этот процесс называется ***проекцией***

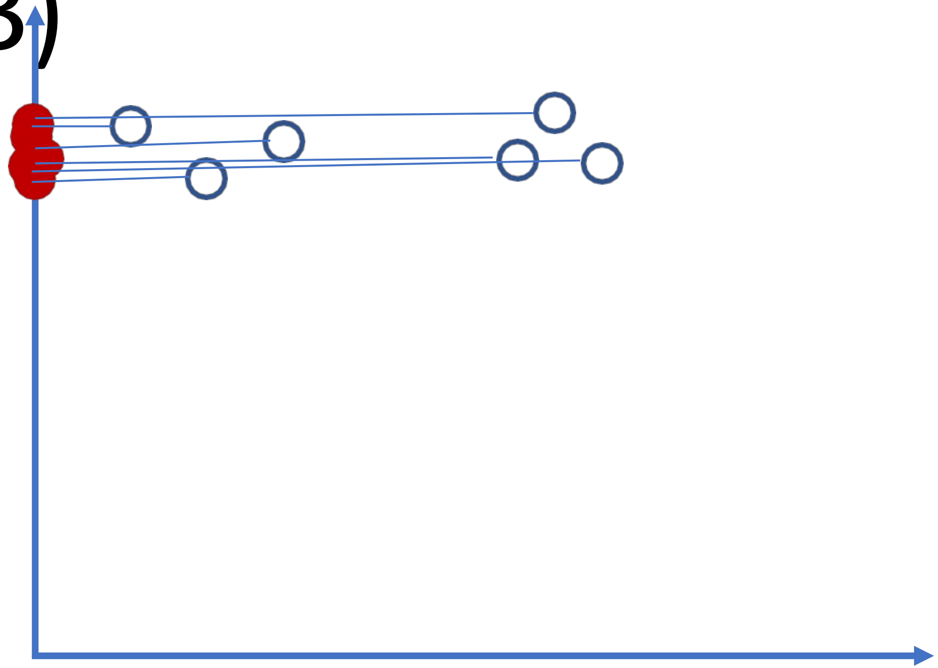
Прогноз данных



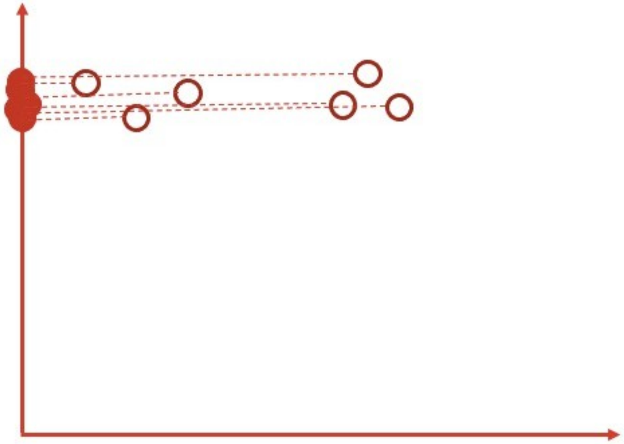
Проекция данных (2)

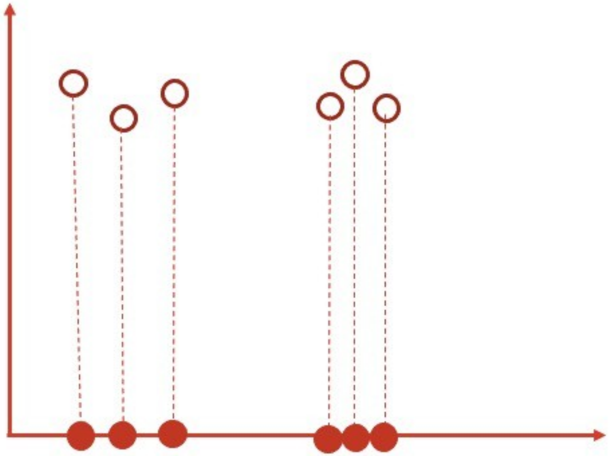


Проекция данных (3)



Проекция данных (4)





Что выбрать из этих двух?

Таким образом,

- При проецировании данных в пространство более низкой размерности мы хотели бы сохранить как можно больше структурной информации о наших данных

- И здесь нам может помочь **дисперсия.**

- Мы можем вычислить дисперсию в каждом одномерном пространстве как

$$\sigma^2 = \frac{1}{n}\sum_{i=1}^{n}(x_i - \mu_x)^2 \qquad \mu_x = \frac{1}{n}\sum_{i=1}^{n} x_i ,$$

- Что мы и постараемся сделать по максимуму при выборе направления проецирования.

Анализ главных компонент (PCA)

PCA

- Один из наиболее широко используемых методов проецирования данных в нижние измерения

PCA (2)

- При проецировании данных из p-мерного пространства в d-мерное пространство PCA определяет d векторов, каждый из которых представлен в виде $\boldsymbol{w}_j$, где $j = 1, \cdots, d$
- Каждый вектор является p-мерным - то есть $\boldsymbol{w} \in \mathbb{R}^p$.
- Проецируемая точка i - th представлена в виде $\boldsymbol{x}'_i = [x'_{i1}, x'_{i2}, \cdots, x'_{id}]^T$.

$$x'_{id} = \boldsymbol{w}_d^T \boldsymbol{x}_i$$

PCA (3)

- PCA использует дисперсию в проектируемом пространстве в качестве критерия для выбора $\boldsymbol{w}_{\cdot d}$

- В частности, $\boldsymbol{w}_1$ будет вектором, который сохранит дисперсию в x'_{i1} такой высокой, как возможно

- $_2\boldsymbol{w}$ будут также выбраны варианты, максимизирующие дисперсию, но с дополнительным ограничением

- $_2\boldsymbol{w}$ должна быть ортогональна к $\boldsymbol{w}_{\cdot 1}$

В общем,

$$\boldsymbol{w}_i^T \boldsymbol{w}_j = \qquad \forall i \neq j$$

PCA (4)

- В дополнение к предыдущему ограничению, PCA также требует, чтобы

$$\mathbf{w}_d^T \mathbf{w}_d = 1$$

- Это означает, что каждый вектор должен иметь длину 1

PCA (5)

- Наконец, PCA требует, чтобы каждое исходное измерение имело нулевое среднее значение

$$\mu_x = \frac{1}{n} \sum_{i=1}^{n} \boldsymbol{x}_i = 0$$

Что мы знаем на данный момент?

1. PCA уменьшает размерность, проецируя данные из пространства высокой размерности в пространство более низкой размерности
2. Для этого используется набор векторов
3. Векторы будут выбраны таким образом, чтобы они максимизировали дисперсию в пространстве проекта
4. Кроме того, векторы
   - Должны быть ортогональными
   - Имеют единичную длину
5. Наконец, данные должны иметь нулевое среднее значение

Как работает PCA?

Как работает PCA?

- Чтобы понять, как работает PCA, начнем с проекции в 1-мерное пространство, то есть $d = 1$
- В этом случае для каждого $\boldsymbol{x}_i$ результатом проецирования будет скалярное значение

$$x_i' = \boldsymbol{w}^T \boldsymbol{x}_i$$

Как работает PCA? (2)

- Дисперсия в проецируемом пространстве задается

$$\sigma^2 = \frac{1}{n}\sum_{i=1}^{n}(x'_i - \mu'_x)^2 \qquad \mu'_x = \frac{1}{n}\sum_{i=1}^{n} x'_i$$

$$= \frac{1}{n}\sum_{i=1}^{n}(x'_i - 0)^2 \qquad = \frac{1}{n}\sum_{i=1}^{n} \boldsymbol{w}^T \boldsymbol{x}_i$$

$$= \frac{1}{n}\sum_{i=1}^{n}(x'_i)^2 \qquad = \boldsymbol{w}^T\left(\frac{1}{n}\sum_{i=1}^{n} \boldsymbol{x}_i\right) = \boldsymbol{w}^T \boldsymbol{\mu}_x$$

$$\mu'_x = 0$$

Как работает PCA? (3)

$$\sigma^2 = \frac{1}{n}\sum_{i=1}^{n}(x'_i)^2$$

- Мы также знаем, что

$$x'_i = \boldsymbol{w}^T \boldsymbol{x}_i$$

- Подставив его значение в приведенные выше уравнения, мы получим

$$\sigma^2 = \frac{1}{n}\sum_{i=1}^{n}(\boldsymbol{w}^T \boldsymbol{x}_i)^2$$

Как работает PCA? (4)

$$\sigma^2 = \frac{1}{n}\sum_{i=1}^{n}\left(\boldsymbol{w}^T \boldsymbol{x}_i\right)^2 = \frac{1}{n}\sum_{i=1}^{n} \boldsymbol{w}^T \boldsymbol{x}_i \boldsymbol{x}_i^T \boldsymbol{w}$$

$$= \boldsymbol{w}^T \left(\frac{1}{n}\sum_{i=1}^{n} \boldsymbol{x}_i \boldsymbol{x}_i^T\right) \boldsymbol{w}$$

- Где C - ковариационная матрица выборки.

$$\boxed{\sigma^2 = \boldsymbol{w}^T C \boldsymbol{w}}$$

Напомним, что PCA хочет

- Максимизируйте дисперсию σ^2
- И мы только что вывели, что

$$\boxed{{}^2\sigma = \boldsymbol{w}^T C \boldsymbol{w}}$$

- Поэтому проекция, которая максимизирует σ^2 , также максимизирует C.

Максимизация σ^2 : Тривиальное решение

- Увеличьте значение элементов в $\boldsymbol{w}$.
- И поэтому мы уже установили ограничение

$$\boxed{{}^{T}\boldsymbol{w}\ \boldsymbol{w} = 1}$$

- Таким образом, мы имеем дело с оптимизационной задачей с ограничениями

Цель PCA

- Найдите $\boldsymbol{w}$, при котором максимизируется следующее значение

$$\boxed{\mathscr{L} = \boldsymbol{w}^T \boldsymbol{C} \boldsymbol{w} - \lambda(\boldsymbol{w}^T \boldsymbol{w} - 1)}$$

- Где λ - множитель Лагранжа

Поиск оптимального $\boldsymbol{w}$

$$\boxed{\mathscr{L} = \boldsymbol{w}^T C \boldsymbol{w} - \lambda(\boldsymbol{w}^T \boldsymbol{w} - 1)}$$

- Возьмите частную производную по отношению к $\boldsymbol{w}$

$$\frac{\partial \mathscr{L}}{\partial \boldsymbol{w}} = 2C\boldsymbol{w} - \lambda(2\boldsymbol{w} - \mathbf{0})$$

$$= 2C\boldsymbol{w} - \lambda(\boldsymbol{w})$$

- Установив значение 0, мы получим очень важный результат

$$\boxed{\lambda \boldsymbol{w} = C\boldsymbol{w}}$$

Давайте проанализируем, что мы получили

$$\boxed{\lambda \boldsymbol{w} = C\, \boldsymbol{w}}$$

- λ - скаляр
- $\boldsymbol{w}$ - вектор
- C - матрица

1. Таким образом, умножение $\boldsymbol{w}$ на C только масштабирует его (только изменяет его длину)

2. Таким образом, $\boldsymbol{w}$, максимизирующий дисперсию, является одним из

собственных векторов C и λ.

собственное значение $\boldsymbol{w}$

Но $\boldsymbol{w}$ - это какой собственный вектор C?

- C - матрица $p \times p$.
- Таким образом, он имеет p собственных векторов
- Как узнать, какой из них соответствует наибольшей дисперсии в прогнозируемом пространстве?

Но $\boldsymbol{w}$ - это какой собственный вектор C? (2)

- Наше выражение для дисперсии σ^2 имеет вид

$$\boxed{\sigma^2 = \boldsymbol{w}^T\, C\, \boldsymbol{w}}$$

- Мы также знаем, что $\boldsymbol{w}^T \boldsymbol{w} = 1$.

- Таким образом, мы можем записать уравнение σ^2 в виде

$$\boxed{\sigma^2\, \boldsymbol{w}^T \boldsymbol{w} = \boldsymbol{w}^T\, C\, \boldsymbol{w}}$$

Но $\boldsymbol{w}$ - это какой собственный вектор C? (3)

$$\boxed{{}^2\sigma\ \boldsymbol{w}^T\ \boldsymbol{w} = \boldsymbol{w}^T\ C\ \boldsymbol{w}}$$

- Убрав $\boldsymbol{w}^T$ с обеих сторон, получим

$$\boxed{{}^2\sigma\ \boldsymbol{w} = C\ \boldsymbol{w}}$$

- Заметим, что мы только что показали, что $\lambda\boldsymbol{w} = C\ \boldsymbol{w}$.

- Таким образом,

$$\boxed{{}^2\sigma\ \boldsymbol{w} = \lambda\boldsymbol{w} = C}$$

На вынос

$$\boxed{{}^{2}\sigma\ \boldsymbol{w} = \lambda \boldsymbol{w} = C\ \boldsymbol{w}}$$

- Если задана пара **(собственный вектор $\boldsymbol{w}$, собственное значение λ)** C, то λ соответствует величине дисперсии в проецируемом пространстве, определяемой $\boldsymbol{w}$

- Таким образом, если мы нашли p (собственных векторов, собственных значений) пар C, то пара с наибольшим собственным значением соответствует вектору, который в наибольшей степени максимизирует дисперсию в проектируемом пространстве

Таким образом, $\{(\boldsymbol{x}_i) |_i \boldsymbol{x} \in \mathbb{R}^p\}_{i=1}^{n}$, учитывая $X =$
PCA работает следующим образом

1. Преобразуйте данные к нулевому среднему, вычитая μ_x из каждой точки.
2. Вычислите ковариационную матрицу выборки C.
3. Найдите p (собственных векторов, собственных значений) пар C.

4. Найдите собственные векторы, соответствующие d наибольшим собственным значениям $\boldsymbol{w}_1, \boldsymbol{w}_2, \cdots, \boldsymbol{w}_d$
5. Вычислите X' как $X' = XW$, где $W = [\boldsymbol{w}_1, \boldsymbol{w}_2, \cdots, \boldsymbol{w}_d]$

Рекомендуемое чтение

1. Раздел 7.2, *"Первый курс машинного обучения"*, авторы Саймон Роджерс и Марк Джиролами

2. Раздел 6.2 из книги "*Введение в статистическое обучение"* Гарета Джеймса, Даниэлы Виттен, Тревора Хасти и Роберта Тибшриани

Резюме

- Задачи оптимизации с ограничениями
- L1 и L2 как проблемы оптимизации с ограничениями
- Высокоразмерные данные и их проблемы
- Анализ главных компонент

Отражение

1. Каким образом ограничения, накладываемые регрессией Лассо, приводят к разреженным моделям, и почему разреженность может быть желательна в некоторых задачах машинного обучения?
2. Учитывая проклятие размерности, как PCA помогает смягчить его последствия, и каковы ограничения PCA в сохранении исходной структуры данных?
3. Как выбор числа главных компонент в PCA влияет на баланс между уменьшением размерности и сохранением значимой дисперсии в данных? Какие критерии можно использовать для принятия этого решения?
4. Подумайте о математических основах PCA, в частности о роли собственных значений и собственных векторов в определении главных компонент. Как это связано с дисперсией, объясняемой каждым компонентом?
5. Обсудите практические последствия использования методов регуляризации и снижения размерности в реальных наборах данных. Как эти методы влияют на

процесс разработки модели, начиная с выбора признаков и заканчивая проверкой модели?