这一章是讲学习理论的，涉及了VC dimension以及bias,var的知识，比较难，数学推导很多，我做这些习题估计前前后后有三四十个小时，感觉还需要多体会几遍。

我的github地址：
https://github.com/Doraemonzzz

个人主页：
http://doraemonzzz.com/

Chapter 2 Training versus Testing

Part 1: Exercise

Exercise 2.1 (Page 45)

By inspection , find a break point k for each hypothesis set in Example 2.2(if there is one). Verify that $m_{\mathcal{H}}(k) < 2^k$ using the formulas derived in that Example.

Example 2.2可以参考课本43页，这里直接给出结果 对例(1)来说，$m_{\mathcal{H}}(N) = N + 1$

N	$m_{\mathcal{H}}(N)$	2^N
1	2	2
2	3	4

所以break point k可以取2

对例(2)来说，$m_{\mathcal{H}}(N) = \frac{1}{2}N^2 + \frac{1}{2}N + 1$

N	$m_{\mathcal{H}}(N)$	2^N
1	2	2
2	4	4
3	7	8

所以break point k可以取3

对例(3)来说，$m_{\mathcal{H}}(N) = 2^N$，所以没有break point

Exercise 2.2 (Page 50)

(a) Verify the bound of Theorem 2.4 in the three cases of Example 2.2:

(i) Positive rays: $\mathcal{H}$ consists of all hypotheses in one dimension of the form $h(x) = \text{sign}(x - a)$.

(ii) Positive intervals: $\mathcal{H}$ consists of all hypotheses in one dimension that are positive within some interval and negative elsewhere.

(iii) Convex sets: $\mathcal{H}$ consists of all hypotheses in two dimensions that are positive inside some convex set and negative elsewhere.

(Note: you can use the break points you found in Exercise 2.1)

(b) Does there exist a hypothesis set for which $m_{\mathcal{N}}(N) = N + 2^{\lfloor \frac{N}{2} \rfloor}$

(where $\lfloor \frac{N}{2} \rfloor$ is the largest integer $\leq \frac{N}{2}$)?

这题是让我们在具体例子下验证下之前的理论。 先回顾下之前的理论：

$$如果对于某个 \, k, m_{\mathcal{H}}(k) < 2^k, 那么对于所有 \, N$$

$$m_{\mathcal{H}}(N) \leq \sum_{i=0}^{k-1} \binom{N}{i}$$

所以对于(i)来说，之前求得$k = 2$，因此

$$m_{\mathcal{H}}(N) = N + 1 \leq \sum_{i=0}^{k-1} \binom{N}{i} = \sum_{i=0}^{1} \binom{N}{i} = 1 + N$$

对于(ii)来说，之前求得$k = 3$，因此

$$
\begin{aligned}
m_{\mathcal{H}}(N) &= \frac{1}{2}N^2 + \frac{1}{2}N + 1 \\
&\leq \sum_{i=0}^{k-1} \binom{N}{i} \\
&= \sum_{i=0}^{2} \binom{N}{i} \\
&= 1 + N + \frac{1}{2}N(N-1) \\
&= \frac{1}{2}N^2 + \frac{1}{2}N + 1
\end{aligned}
$$

对于(iii)来说，之前没有break point，但是$m_{\mathcal{H}}(N) = 2^N$，所以我们可以在定理中取$k = N + 1$

$$
\begin{aligned}
m_{\mathcal{H}}(N) &= 2^N \\
&\leq \sum_{i=0}^{k-1} \binom{N}{i} \\
&= \sum_{i=0}^{N} \binom{N}{i} = 2^N
\end{aligned}
$$

(b)这题我的理解为否，因为如果成立，那么

$$m_{\mathcal{N}}(N) = N + 2^{\lfloor \frac{N}{2} \rfloor} \leq \sum_{i=0}^{k-1} \binom{N}{i}$$

不等式左边是N的指数函数,右边是N的多项式，所以当N充分大时不可能成立。

Exercise 2.3 (Page 50)

Compute the VC dimension of $\mathcal{H}$ for the hypothesis sets in parts (i), (ii),(iii) of Exercise 2.2(a).

这题需要计算VC维，由公式$k = d_{vc} + 1$，k为最小的break point，可以轻松得到。

(i)$k = 2, d_{vc} = 1$
(ii)$k = 3, d_{vc} = 2$
(iii)$k = +\infty, d_{vc} = +\infty$

Exercise 2.4 (Page 52)

Consider the input space $\mathcal{X} = \{1\} \times \mathbb{R}^d$(including the constant coordinate $x_0 = 1$). Show that the VC dimension of the perceptron (with $d + 1$ parameters, counting w_0) is exactly $d + 1$ by showing that it is at least $d + 1$ and at most $d + 1$, as follows.

(a) To show that $d_{vc} \geq d + 1$, find $d + 1$ points in $\mathcal{X}$ that the perceptron can shatter. [Hint: Construct a nonsingular $(d + 1) \times (d + 1)$ matrix whose rows represent the $d + 1$ points, then use the nonsingularity to argue that the perceptron can shatter these points.]

(b) To show that $d_{vc} \leq d + 1$, show that no set of $d + 2$ points in X can be shattered by the perceptron. [Hint: Represent each point in as a vector of length $d + 1$, then use the fact that any $d + 2$ vectors of length $d + 1$ have to be linearl dependent. This means that some vector is a linear combination of all the other vectors. Now, if you choose the cass of these other vectors carefully then the classification of the dependent vector will be dictated. Conclude that there is some dichotomy that cannot be implemented, and therefre that for $N \geq d + 2, m_{\mathcal{H}}(N) < 2^N$

这题实际上给出了证明$d_{vc} = d$的一般思路：证明

$$d_{vc} \leq d$$

以及

$$d_{vc} \geq d$$

证明$d_{vc} \geq d$只要找出d个点可以被shattered就可以，证明$d_{vc} \leq d$则需要说明任何$d + 1$个点都不能被shattered

回到原题，先看(a)，构造一个$(d + 1) \times (d + 1)$矩阵

$$\begin{pmatrix} 1 & 1 & 1 & \ldots & 1 \\ 1 & 2 & 2^2 & \ldots & 2^d \\ 1 & 3 & 3^2 & \ldots & 3^d \\ \ldots & \ldots & \ldots & \ldots & \ldots \\ 1 & (d+1) & (d+1)^2 & \ldots & (d+1)^d \end{pmatrix}$$

它对应的行列式为范德蒙行列式，行列式的值不为0，因此对于任意的$y = (y_1, y_2, \ldots y_{d+1})^T$，以下方程组恒有解。

$$
\begin{pmatrix}
1 & 1 & 1 & \ldots & 1 \\
1 & 2 & 2^2 & \ldots & 2^d \\
1 & 3 & 3^2 & \ldots & 3^d \\
\ldots & \ldots & \ldots & \ldots & \ldots \\
1 & (d+1) & (d+1)^2 & \ldots & (d+1)^d
\end{pmatrix}
\times
\begin{pmatrix}
x_1 \\
x_2 \\
x_3 \\
\ldots \\
x_{d+1}
\end{pmatrix}
=
\begin{pmatrix}
y_1 \\
y_2 \\
y_3 \\
\ldots \\
y_{d+1}
\end{pmatrix}
$$

换句话说，这$d+1$个特殊的点可以被shattered，因此

$$d_{vc} \geq d + 1$$

(b)不妨设感知机的模型为

$$f(x) = \mathrm{sign}(w^T x)$$

我们知道在$d+1$维空间中的任何$d+2$个向量均线性相关，因此不妨设z_{d+2}可以被线性表出

$$z_{d+2} = \sum_{i=1}^{d+1} k_i z_i$$

$$w^T z_{d+2} = \sum_{i=1}^{d+1} w^T k_i z_i$$

$$\mathrm{sign}(w^T z_{d+2}) = \mathrm{sign}(\sum_{i=1}^{d+1} k_i w^T z_i)$$

也就是说z_{d+2}的分类被$z_1, z_2 \ldots z_{d+1}$确定，那么

$$(\mathrm{sign}(z_1), \mathrm{sign}(z_2) \ldots \mathrm{sign}(z_{d+1}), -\mathrm{sign}(z_{d+2}))$$

这种情形必然无法被划分出来，因此

$$d_{vc} \leq d + 1$$

综上所述$d_{vc} = d + 1$

Exercise 2.5 (Page 56)

Suppose we have a simple learning model whose growth function is $m_{\mathcal{H}}(N) = N + 1$, hence $d_{vc} = 1$. Use the VC bound (2.12) to estimate the probability that E_{out} will be within 0.1 of E_{in} given 100 training examples. [Hint: The estimate wil be ridiculous.}

这里回顾下(2.12) (Page 53)
Theorem 2.5 (VC generalization bound) . For any tolerance $\delta > 0$,

$$E_{\mathrm{out}}(g) \leq E_{\mathrm{in}}(g) + \sqrt{\frac{8}{N} \ln \frac{4 m_{\mathcal{H}}(2N)}{\delta}}$$

with probability$\geq 1 - \delta$.

令

$$\sqrt{\frac{8}{N} \ln \frac{4 m_{\mathcal{H}}(2N)}{\delta}} = \epsilon$$

那么

$$\frac{8}{N} \ln \frac{4 m_{\mathcal{H}}(2N)}{\delta} = \epsilon^2$$

$$\ln \frac{4 m_{\mathcal{H}}(2N)}{\delta} = \frac{N \epsilon^2}{8}$$

$$\frac{4 m_{\mathcal{H}}(2N)}{\delta} = e^{\frac{N \epsilon^2}{8}}$$

$$\delta = \frac{4 m_{\mathcal{H}}(2N)}{e^{\frac{N \epsilon^2}{8}}}$$

对于此题来说

$$\epsilon = 0.1, N = 100, m_{\mathcal{H}}(N) = N + 1$$

带入计算得 $\delta = 709.5275096780147$，所以概率大于等于 -708.5275096780147，这个结果太宽松了，等于没说。

```
# -*- coding: utf-8 -*-
"""
Created on Sun Feb 17 21:46:18 2019

@author: qinzhen
"""

import numpy as np
def m(n):
    return n + 1

def delta(N, epsilon):
    return 4 * m(2 * N) / (np.exp(N * (epsilon ** 2) / 8))

print(delta(100, 0.1))
```

```
709.5275096780147
```

Exercise 2.6 (Page 60)

A data set has 600 examples. To properly test the performance of the final hypothesis, you set aside a randomly selected subset of 200 examples which are never used in the training phase; these form a test set. You use a learning model with 1,000 hypotheses and select the final hypothesis g based on the 400 training examples. We wish to estimate $E_{\text{out}}(g)$. We have access to two estimates: $E_{\text{in}}(g)$, the in sample error on the 400 training examples; and, $E_{\text{test}}(g)$, the test error on the 200 test examples that were set aside.

(a) Using a 5% error tolerance ($\delta = 0.05$), which estimate has the higher 'error bar' ? (b) Is there any reason why you shouldn 't reserve even more examples for testing?

(a)$E_{\text{test}}(g)$是$E_{\text{out}}(g)$的无偏估计，但$E_{\text{in}}(g)$不是，所以$E_{\text{in}}(g)$的error bar更高。

(b)如果测试集过多的话，会使得训练集较少，这样训练出来的结果就比较差。

Exercise 2.7 (Page 62)

For binary target functions, show that $\mathbb{P}[h(x) \neq f(x)]$ can be written as an expected value of a mean squared error measure in the following cases.

(a) The convention used for the binary function is 0 or 1 (b) The convention used for the binary function is ± 1 [int: The difference between (a) and (b) is just a scale.

这题是要将之前离散情况的误差函数改写为平方误差函数。

(a)注意此时$h(x) \neq f(x)$要么是$h(x) = 1, f(x) = 0$或者$h(x) = 0, f(x) = 1$ 因此

$$\mathbb{P}[h(x) \neq f(x)] = \mathbb{P}(|h(x) - f(x)| = 1)$$

而

$$\mathbb{E}[(h(x) - f(x)^2] = 1 \times \mathbb{P}(|h(x) - f(x)| = 1) + 0 \times \mathbb{P}(|h(x) - f(x)| = 0) = \mathbb{P}(|h(x) - f(x)| = 1)$$

所以

$$\mathbb{P}[h(x) \neq f(x)] = \mathbb{E}[(h(x) - f(x))^2]$$

(a)注意此时$h(x) \neq f(x)$要么是$h(x) = 1, f(x) = -1$或者$h(x) = -1, f(x) = 1$ 因此

$$\mathbb{P}[h(x) \neq f(x)] = \mathbb{P}(|h(x) - f(x)| = 2)$$

而

$$\mathbb{E}[(h(x) - f(x))^2] = 4 \times \mathbb{P}(|h(x) - f(x)| = 2) + 0 \times \mathbb{P}(|h(x) - f(x)| = 0) = 4\mathbb{P}(|h(x) - f(x)| = 2)$$

所以

$$\mathbb{P}[h(x) \neq f(x)] = \frac{1}{4}\mathbb{E}[(h(x) - f(x))^2]$$

Exercise 2.8 (Page 63)

(a) Show that if $\mathcal{H}$ is closed under linear combination (any linear combination of hypotheses in $\mathcal{H}$ is also a hypothesis in $\mathcal{H}$), then $\overline{g} \in \mathcal{H}$.
(b) Give a model for which the average function $\overline{g}$ is not in the model's hypothesis set. [int: Use a very simple model.] (c) For binary classification, do you expect $\overline{g}$ to be a binary function?

(a)$\overline{g}(x) = \frac{1}{K} \sum_{k=1}^{K} g_k(x)$,所以如果$\mathcal{H}$在线性运算下封闭，那么$\overline{g} \in \mathcal{H}$

(b)

$$\mathcal{H} = \{g_1, g_2\}$$
$$g_1(x) = 0, g_2(x) = 1$$

但是 $\overline{g}(x) = \frac{1}{2}(g_1(x) + g_2(x))$ 显然不属于 $\mathcal{H}$

(c)对于二元分类问题，$\overline{g}$ 不一定是一个 binary function，例如上例。

Part 2: Problems

Problem 2.1 (Page 69)

In Equation (2. 1), set $\delta = 0.03$ and let

$$\epsilon(M, N, \delta) = \sqrt{\frac{1}{2N} \ln \frac{2M}{\delta}}$$

(a) For $M = 1$, how many examples do we need to make $\epsilon \leq 0.05$?
(b) For $M = 100$, how many examples do we need to make $\epsilon \leq 0.05$?
(c) For $M = 10,000$, how many examples do we need to make $\epsilon \leq 0.05$?

对等式稍作变形:

$$\epsilon^2 = \frac{1}{2N} \ln \frac{2M}{\delta}$$
$$2N = \frac{\ln \frac{2M}{\delta}}{\epsilon^2}$$
$$N = \frac{\ln \frac{2M}{\delta}}{2\epsilon^2}$$

```python
# -*- coding: utf-8 -*-
"""
Created on Sun Feb 17 21:52:33 2019

@author: qinzhen
"""

import numpy as np

delta = 0.03
def f(M, epsilon):
    return np.log(2 * M / delta)/(2 * epsilon ** 2)
```

```python
#(a)
print(f(1, 0.05))
```

```
839.9410155759853
```

```
#(b)
print(f(100, 0.05))
```

```
1760.9750527736032
```

```
#(c)
print(f(10000, 0.05))
```

```
2682.0090899712213
```

所以 (a)839.9410155759853 (b)1760.9750527736032 (c)2682.0090899712213

Problem 2.2 (Page 69)

Show that for the learning model of positive rectangles (aligned horizontally or vertically), $m_{\mathcal{H}}(4) = 2^4$ and $m_{\mathcal{H}}(5) < 2^5$. Hence, give a bound for $m_{\mathcal{H}}(N)$.

题目中的$\mathcal{H}$如上图所示，其实就是边平行于x轴y轴的矩形区域。$m_{\mathcal{H}}(4) = 2^4$这个比较简单，画图就可以了。对于$m_{\mathcal{H}}(5)$，我们可以按如下方式考虑：

假设5个点为$(x_i, y_i), i = 1, \ldots, 5$，不失一般性，我们假设

$$x_i \leq x_{i+1}, i = 1, \ldots, 4$$

我们要证明任意5个点无法被shatter，只要证明上图右边的情形必然发生即可，即证明必然存在

$$(x_i, y_i), (x_j, y_j), (x_k, y_k), l < j < k$$

满足

$$y_i \leq y_j \leq y_k 或$$
$$y_i \geq y_j \geq y_k$$

注意到上述条件可以化为一个条件：

$$(y_i - y_j)(y_j - y_k) \geq 0$$

那么为什么上述不等式成立必然可以推出$m_{\mathcal{H}}(5) < 2^5$呢？这是因为如果该不等式成立，当$(x_i, y_i), (x_k, y_k)$同时标记为$+1$或-1时，那么(x_j, y_j)必然也有相同的标记。（如上右图所示，3个点在同一个矩形内。）

下面证明这种情形必然发生，利用反证法，假设上述情形不会发生，那么对任意$i < j < k$，我们有

$$(y_i - y_j)(y_j - y_k) < 0$$

取$i = 1, k = 5$可得

$$(y_1 - y_j)(y_j - y_5) < 0, j = 2, \ldots, 4$$

取$i = 1, k = 4$可得

$$(y_1 - y_j)(y_j - y_4) < 0, j = 2, \ldots, 3$$

取$i = 1, k = 3$可得

$$(y_1 - y_j)(y_j - y_3) < 0, j = 2$$

所以

$$(y_1 - y_2)(y_2 - y_5) < 0, (y_1 - y_2)(y_2 - y_4) < 0$$
$$(y_1 - y_2)(y_2 - y_5) < 0, (y_1 - y_2)(y_2 - y_3) < 0$$
$$(y_1 - y_3)(y_3 - y_5) < 0, (y_1 - y_3)(y_3 - y_4) < 0$$

因此

$$(y_2 - y_5)(y_2 - y_4) > 0 \tag{1}$$
$$(y_2 - y_5)(y_2 - y_3) > 0 \tag{2}$$
$$(y_3 - y_5)(y_3 - y_4) > 0 \tag{3}$$

如果$y_4 > y_5$，由(1)，(3)可得

$$y_3 < y_5 或 y_3 > y_4 \tag{4}$$
$$y_2 < y_5 或 y_2 > y_4 \tag{5}$$

对(4)分情况讨论：

- 如果$y_3 > y_4$，那么

$$y_3 > y_4 > y_5$$
$$(y_3 - y_4)(y_4 - y_5) > 0$$

  这就与我们的假设矛盾。

- 如果$y_3 < y_5$，那么由(2)可得

$$y_2 < y_3 或 y_2 > y_5$$

  ○ 如果$y_2 < y_3$，那么

$$y_2 < y_3 < y_5$$
$$(y_2 - y_3)(y_3 - y_5) > 0$$

    这就与我们的假设矛盾。

  ○ 如果$y_2 > y_5$，由(5)可得必然有$y_2 > y_4$，所以

$$y_2 > y_4 > y_5$$
$$(y_2 - y_4)(y_4 - y_5) > 0$$

    这就与我们的假设矛盾。

因此，$y_4 > y_5$会产生矛盾。

如果，$y_4 < y_5$，由(1)，(3)可得

$$y_3 < y_4 或 y_3 > y_5 \tag{6}$$
$$y_2 < y_4 或 y_2 > y_5 \tag{7}$$

对(6)分情况讨论：

- 如果$y_3 < y_4$，那么

$$y_3 < y_4 < y_5$$
$$(y_3 - y_4)(y_4 - y_5) > 0$$

这就与我们的假设矛盾。

- 如果$y_3 > y_5$，那么由(2)可得

$$y_2 < y_5 或 y_2 > y_3$$

  ○ 如果$y_2 > y_3$，那么

$$y_2 > y_3 > y_5$$
$$(y_2 - y_3)(y_3 - y_5) > 0$$

    这就与我们的假设矛盾。

  ○ 如果$y_2 < y_5$，那么由(7)可得$y_2 < y_4$，所以

$$y_2 < y_4 < y_5$$
$$(y_2 - y_4)(y_4 - y_5) > 0$$

    这就与我们的假设矛盾。

因此，$y_4 < y_5$会产生矛盾。

综上，无论哪种情形都会产生矛盾，所以必然存在$i < j < k$，使得

$$(y_i - y_j)(y_j - y_k) \geq 0$$

结合之前论述可知

$$m_{\mathcal{H}}(5) < 2^5$$

结合以上两点可得$d_{vc} = 4$，所以

$$m_{\mathcal{H}}(N) \leq N^{d_{vc}} + 1 = N^4 + 1$$

Problem 2.3 (Page 69)

Compute the maximum number of dichotomies, $m_{\mathcal{H}}(N)$,for these learning models, and consequently compute d_{vc} , the VC dimension.
(a) Positive or negative ray: $\mathcal{H}$ contains the functions which are + 1 on $[a, +\infty)$(for some a) together with those that are +1 on $(-\infty, a]$ (for some a).
(b) Positive or negative interval : $\mathcal{H}$ contains the functions which are + 1 on an interval $[a, b]$ and -1 elsewhere or -1 on an interval $[a, b]$ and +1 elsewhere.
(c) Two concentric spheres in $\mathbb{R}^d$: $\mathcal{H}$ contains the functions which are +1 for $a \leq \sqrt{x_1^2 + \ldots + x_d^2} \leq b$.

计算$m_{\mathcal{H}}(N)$以及d_{vc}
(a)这题其实就相当于在N个点中画了一条直线，左边的为-1，右边的为1，或者左边的为1，右边的为-1，除去全1和全-1的情形，一共有$2(N-1)$种，那么加上全-1和全1，一共有$2N-2+2 = 2N$种情形。因此
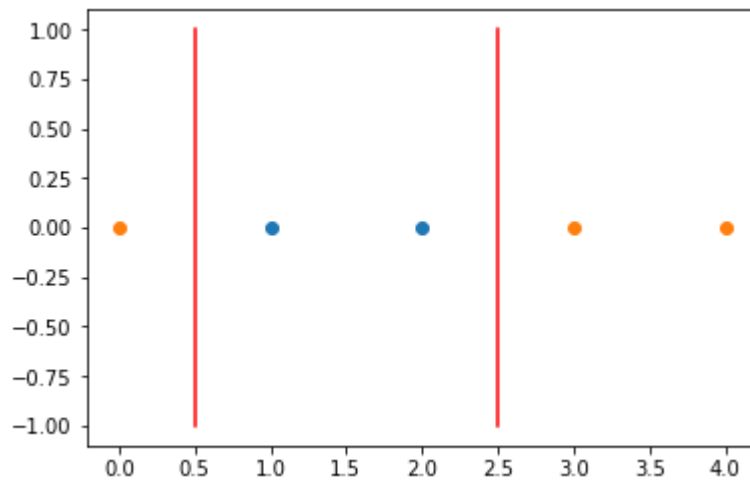
$$m_{\mathcal{H}}(N) = 2N, d_{vc} = 2$$

(b)分两种情形考虑，第一种如下

```
# -*- coding: utf-8 -*-
"""
Created on Sun Feb 17 22:54:04 2019

@author: qinzhen
"""

#(b)
import matplotlib.pyplot as plt

plt.scatter([1, 2], [0, 0])
plt.scatter([0, 3, 4], [0, 0, 0])
plt.plot([0.5, 0.5], [-1, 1], color='red')
plt.plot([2.5, 2.5], [-1, 1], color='red')
plt.show()
```
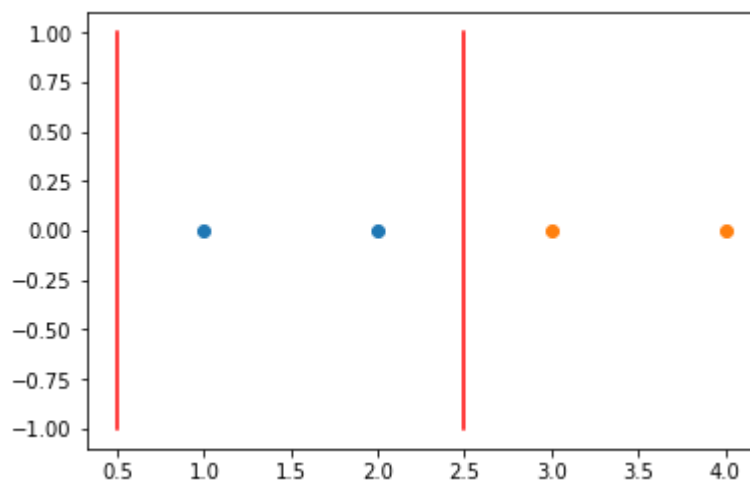

第二种如下

```
plt.scatter([1, 2], [0, 0])
plt.scatter([3, 4], [0, 0])
plt.plot([0.5, 0.5], [-1, 1], color='red')
plt.plot([2.5, 2.5], [-1, 1], color='red')
plt.show()
```

第一种情形为分界区间$[a,b]$两侧都有元素，这种情形不考虑全-1或者全1的情况，所以这种情形共有
$2C_{N-1}^2 = (N-1)(N-2)$，乘以2是因为矩形内部可以为-1，也可以为1。

第二种情形为分界区间$[a,b]$只有一侧有元素，这种情形共有$2(N+1)-2 = 2N$，减2是因为全1或者全-1的情形被
重复算了。

所以成长函数为

$$m_{\mathcal{H}}(N) = (N-1)(N-2) + 2N = N^2 - N + 2$$

因此

$$d_{vc} = 3$$

(c)做映射$y = \sqrt{x_1^2 + \ldots + x_d^2}$，所以这个问题可以化为类似(b)的问题(注意这里不同之处在于$[a,b]$区间内的元素只能
为$+1$)。注意这里两圆之间的部分为$+1$，因此问题为d个点之间以及两侧共$d+1$个间隔中挑选2个，最后要需要考虑
全-1的情形，因此

$$m_{\mathcal{H}}(N) = C_{N+1}^2 + 1 = \frac{N(N+1)}{2} + 1$$
$$d_{vc} = 2$$

Problem 2.4 (Page 69)

Show that $B(N, k) = \sum_{i=0}^{k-1} \binom{N}{i}$ by showing the otherd irection to Lemma 2.3, namely that

$$B(N, k) \geq \sum_{i=0}^{k-1} \binom{N}{i}$$

To do so, construct a specific set of $\sum_{i=0}^{k-1} \binom{N}{i}$ dichotomies that does not shatter any subset of k variables.
[Hint: Try limiting the number of -1 's in each dichotomy]

考虑N个元素的点集，点集中最多有$k-1$个-1，那么这样的点集有$\sum_{i=0}^{k-1} \binom{N}{i}$种。考虑$k$个元素的子集，然么
$(-1, -1, \ldots, -1)$(全为-1的情形)必然无法被表出，因为最多有$k-1$个点为-1。那么由$B(N, k)$的定义知

$$B(N, k) \geq \sum_{i=0}^{k-1} \binom{N}{i}$$

因此

$$B(N, k) = \sum_{i=0}^{k-1} \binom{N}{i}$$

Problem 2.5 (Page 69)

Prove by induction that $\sum_{i=0}^{D} \binom{N}{i} \leq N^D + 1$, hence

$$m_{\mathcal{H}}(N) \leq N^{d_{vc}} + 1$$

使用数学归纳法。
$D = 0$时，

$$\sum_{i=0}^{D} \binom{N}{i} = 1$$
$$N^D + 1 = 1 + 1 = 2$$

所以$D = 0$时，不等式成立。
假设$D = k$时不等式成立，$D = k + 1$时

$$\sum_{i=0}^{k+1} \binom{N}{i} = \sum_{i=0}^{k} \binom{N}{i} + \binom{N}{k+1} \leq N^k + 1 + \binom{N}{k+1}$$

对$\binom{N}{k+1}$进行放缩

$$\binom{N}{k+1} = \frac{N(N-1)\dots(N-k)}{(k+1)!} \leq N(N-1)\dots(N-k) \leq N^k(N-1)$$

因此

$$\sum_{i=0}^{k+1} \binom{N}{i} \leq N^k + 1 + \binom{N}{k+1}$$
$$\leq N^k + 1 + N^k(N-1)$$
$$= N^{k+1} + 1$$

所以对$D = k + 1$时不等式也成立。 由之前结论我们知道

$$m_{\mathcal{H}}(N) \leq \sum_{i=0}^{d_{vc}} \binom{N}{i}$$

那么

$$m_{\mathcal{H}}(N) \leq \sum_{i=0}^{d_{vc}} \binom{N}{i} \leq N^{d_{vc}} + 1$$

Problem 2.6 (Page 70)

Prove that for $N \geq d$,

$$\sum_{i=0}^{d} \binom{N}{i} \leq (\frac{eN}{d})^d$$

We suggest you first show the following intermediate steps.

(a) $\sum_{i=0}^{d} \binom{N}{i} \le \sum_{i=0}^{d} \binom{N}{i}(\frac{N}{d})^{d-i} \le (\frac{N}{d})^d \sum_{i=0}^{N} \binom{N}{i}(\frac{d}{N})^i$

(b) $\sum_{i=0}^{N} \binom{N}{i}(\frac{d}{N})^i \le e^d$ [Hints: Binomial theorem; $(1+\frac{1}{x})^x \le e$ for $x \ge 0$ Hence, argue that $m_{\mathcal{H}}(N) \le (\frac{eN}{d_{vc}})^{d_{vc}}$

(a)因为$N \ge d, d \ge i$，所以$(\frac{N}{d})^{d-i} \ge 1$那么

$$\sum_{i=0}^{d} \binom{N}{i} \le \sum_{i=0}^{d} \binom{N}{i}(\frac{N}{d})^{d-i} = (\frac{N}{d})^d \sum_{i=0}^{d} \binom{N}{i}(\frac{d}{N})^i \le (\frac{N}{d})^d \sum_{i=0}^{N} \binom{N}{i}(\frac{d}{N})^i$$

最后一步是因为$d \le N$

(b)由二项式定理

$$\sum_{i=0}^{N} \binom{N}{i}(\frac{d}{N})^i = \sum_{i=0}^{N} \binom{N}{i}(\frac{d}{N})^i 1^{N-i} = (1+\frac{d}{N})^N$$

由$(1+\frac{1}{x})^x \le e$，我们知道

$$(1+\frac{d}{N})^N = (1+\frac{d}{N})^{\frac{N}{d}d} \le e^d$$

因此

$$\sum_{i=0}^{N} \binom{N}{i}(\frac{d}{N})^i = (1+\frac{d}{N})^N \le e^d$$

$$\sum_{i=0}^{d} \binom{N}{i} \le (\frac{N}{d})^d \sum_{i=0}^{N} \binom{N}{i}(\frac{d}{N})^i \le (\frac{N}{d})^d e^d = (\frac{eN}{d})^d$$

由之前结论我们知道

$$m_{\mathcal{H}}(N) \le \sum_{i=0}^{d_{vc}} \binom{N}{i}$$

那么由这题的结论可得

$$m_{\mathcal{H}}(N) \le (\frac{eN}{d_{vc}})^{d_{vc}}$$

这题和之前一题都是对$m_{\mathcal{H}}(N)$给出一个多项式上界。

Problem 2.7 (Page 70)

Plot the bounds for $m_{\mathcal{H}}(N)$ given in Problems 2.5 and 2.6 for $d_{vc} = 2$ and $d_{vc} = 5$. When do you prefer one bound over the other?

作图题

```
# -*- coding: utf-8 -*-
```

```
"""
Created on Sun Feb 17 22:57:08 2019

@author: qinzhen
"""

import matplotlib.pyplot as plt
import numpy as np

def f1(N, d):
    return (N ** d) + 1

def f2(N, d):
    return (np.exp(1) * N / d) ** d

def draw(d):
    n = range(1,50)
    m1 = [f1(i, d) for i in n]
    m2 = [f2(i, d) for i in n]

    plt.plot(n, m1, label='f1')
    plt.plot(n, m2, label='f2')
    plt.legend()
    plt.title('dvc='+str(d))
    plt.show()
```
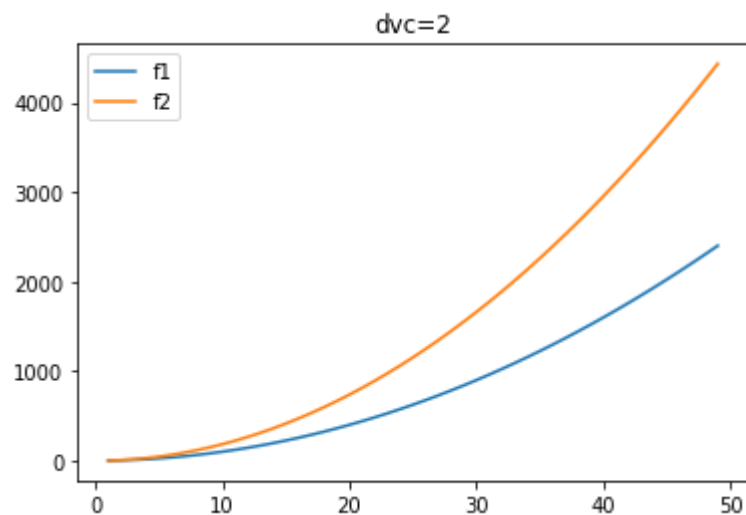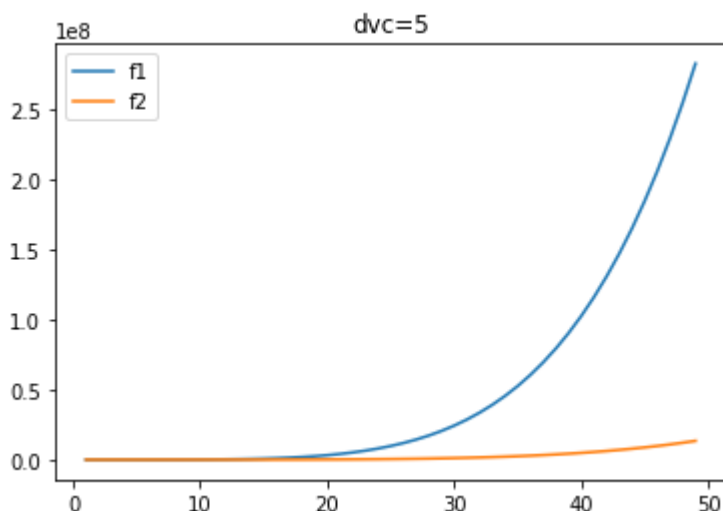
```
#d=2
draw(2)
```



```
#d=3
draw(5)
```

上图中，f1表示2.5得到的上界，f2表示2.6得到的上界，可以看到，当$d_{vc} = 2$时，2.5得到的上界要明显好于2.6得到的上界，而当$d_{vc} = 5$时，2.6得到的上界要明显好于2.5得到的上界，我们来简单分析下原因。

2.5得到的上界

$$N^{d_{vc}} + 1$$

2.6得到的上界

$$(\frac{eN}{d_{vc}})^{d_{vc}}$$

当$d_{vc} = 2$时，$\frac{e}{d_{vc}} > 1$，所以当N充分大时，$(\frac{eN}{d_{vc}})^{d_{vc}}$明显大于$N^{d_{vc}} + 1$而当$d_{vc} = 5$时，$\frac{e}{d_{vc}} < 1$，所以当$N$充分大时，$(\frac{eN}{d_{vc}})^{d_{vc}}$明显小于$N^{d_{vc}} + 1$

所以当$d_{vc} \leq 2$时，选$N^{d_{vc}} + 1$作为上界，其余情形选择$(\frac{eN}{d_{vc}})^{d_{vc}}$作为上界。

Problem 2.8 (Page 70)

Which of the following are possible growth functions $m_{\mathcal{H}}(N)$ for some hypothesis set:

$$1 + N; 1 + N + \frac{N(N-1)}{2}; 2^N; 2^{\lfloor\sqrt{N}\rfloor}; 2^{\lfloor\frac{N}{2}\rfloor}; 1 + N + \frac{N(N-1)(N-2)}{6}$$

我们知道$m_{\mathcal{H}}(N)$要么等于2^N，要么有一个多项式的上界，根据这两点，除了$2^{\lfloor\sqrt{N}\rfloor}, 2^{\lfloor\frac{N}{2}\rfloor}$以外，其余函数均有可能成为$m_{\mathcal{H}}(N)$

Problem 2.9 (Page 70)

[hard] For the perceptron in d dimensions, show that

$$m_{\mathcal{H}}(N) = 2\sum_{i=0}^{d}\binom{N-1}{i}$$

Hint: Cover(1965) in Further Reading.

Use this formula to verify that $d_{vc} = d + 1$ by evaluating $m_{\mathcal{H}}(d + 1)$ and $m_{\mathcal{H}}(d + 2)$. Plot $m_{\mathcal{H}}(N)/2N$ for $d = 10$ and $N \in [1, 40]$. If you generate a random dichotomy on N points in 10 dimensions, give an upper bound on the probability that the dichotomy will be separable for N = 10, 20, 40.

这题比较难，我也是查阅了网上资料才思考出来。参考资料

为了后续叙述方便，记N个数据，维度为d的$m_{\mathcal{H}}(N)$为$f(N, d)$，我们考虑$f(N, d)$的递推关系。

假设$B = \{(b_1, b_2 \dots b_N), b_i \in \{1, -1\}\}$是维度$d$，$N$个点所有可以被表示出来的所有组合，也就是说$\forall (b_1, b_2 \dots b_N) \in B, \exists w \in R^d, (x_1, x_2 \dots x_N), (x_i \in R^d)$使得

$$(\text{sign}(w^T x_1), \text{sign}(w^T x_2) \dots \text{sign}(w^T x_N)) = (b_1, b_2 \dots b_N)$$

那么此时增加一个点x_{N+1}

$$(\text{sign}(w^T x_1), \text{sign}(w^T x_2) \dots \text{sign}(w^T x_N), \text{sign}(w^T x_{N+1})) = (b_1, b_2 \dots b_N, \text{sign}(w^T x_{N+1}))$$

现在考虑以下两种情况：

- 情形1 $w^T x_{N+1}! = 0$,那么或$\text{sign}(w^T x_{N+1}) = 1$或$-1$,那么$(b_1, b_2 \dots b_N, \text{sign}(w^T x_{N+1}))$可以直接被表示出来，记满足这个条件的数量为$C_1$

- 情形2 $w^T x_{N+1} = 0$,那么此时需要对w进行旋转，可以进行如下处理。

  令

$$t = \max_{(i=1 \dots N)} |x_{N+1}^T x_i|$$
$$M = \min_{(i=1 \dots N)} |w^T x_i|$$
$$k = \frac{M}{2t}$$

那么对于$w_1 = w \pm ky$

$$w_1^T x_i = (w \pm k x_{N+1})^T x_i = w^T x_i \pm k x_{N+1}^T x_i$$

$$|k x_{N+1}^T x_i| = |\frac{M}{2t} x_{N+1}^T x_i| = \frac{M}{2t} |x_{N+1}^T x_i| \leq \frac{M}{2t} \times t = \frac{M}{2} \leq \frac{|w^T x_i|}{2}$$

对于任何非零的数a，$a \pm \frac{|a|}{2}$与a同号，那么$w_1^T x_i$与$w^T x_i$同号。再来观察$w_1^T x_{N+1}$的符号，注意$w^T x_{N+1} = 0$

$$w_1^T x_{N+1} = (w \pm k x_{N+1})^T x_{N+1} = w^T x_{N+1} \pm k x_{N+1}^T x_{N+1} = \pm k x_{N+1}^T x_{N+1}$$

所以当$w_1 = w + ky$时,$w_1^T y > 0$,当$w_1 = w - ky$时,$w_1^T y < 0$
综上所述，当$w^T x_{N+1} = 0$时，通过旋转w，$(b_1, b_2 \dots b_N, \pm 1)$均可以被表示出来，记满足这个条件的数量为$C_2$

接下来分析C_1以及C_2。首先看C_2，C_2是有一个约束条件$w^T x_{N+1} = 0$下的最大可表示出来的数量，所以这相当于$d - 1$维（有一个约束条件，所以减小一个维度）下N个点的最大可表示出来的数量乘以2（$\text{sign}(w^T x_{N+1})$可正可负），也就是说$C_2 = 2f(N, d - 1)$。而C_1相当于所有$f(N, d)$减去满足$w^T x_{N+1} = 0$条件下的表示数量$f(N, d - 1)$，即$C_1 = f(N, d) - f(N, d - 1)$

因此

$$f(N + 1, d) = C_1 + C_2 = 2f(N, d - 1) + f(N, d) - f(N, d - 1) = f(N, d) + f(N, d - 1)$$

其余的步骤由数学归纳法即可。

$$f(N+1, d) = f(N, d) + f(N, d-1) = 2 \sum_{i=0}^{d} \binom{N-1}{i} + 2 \sum_{i=0}^{d-1} \binom{N-1}{i}$$

我们知道

$$\binom{n}{k} = \binom{n-1}{k} + \binom{n-1}{k-1}$$

因此

$$\begin{aligned}
f(N+1, d) &= 2 \sum_{i=0}^{d} \binom{N-1}{i} + 2 \sum_{i=0}^{d-1} \binom{N-1}{i} \\
&= 2 + 2 \sum_{i=1}^{d} \binom{N-1}{i} + 2 \sum_{i=0}^{d-1} \binom{N-1}{i} \\
&= 2 + 2 \sum_{i=1}^{d} \binom{N-1}{i} + 2 \sum_{i=1}^{d} \binom{N-1}{i-1} \\
&= 2 + 2 \sum_{i=1}^{d} \binom{N}{i} \\
&= 2 \sum_{i=0}^{d} \binom{N}{i}
\end{aligned}$$

因此结论成立。

下面计算下 d_{vc}

$$m_{\mathcal{H}}(d+1) = 2 \sum_{i=0}^{d} \binom{d}{i} = 2^{d+1}$$

$$m_{\mathcal{H}}(d+2) = 2 \sum_{i=0}^{d} \binom{d+1}{i} = 2(2^{d+1} - 1) = 2^{d+2} - 2 < 2^{d+2}$$

所以 $d_{vc} = d + 1$，这也符合我们之前的论述。

剩余部分是作图，注意这里的概率其实就是 $\frac{m_{\mathcal{H}}(N)}{2^N}$，因为一共有 2^N 种，而可表示出来的一共有 $m_{\mathcal{H}}(N)$ 种

```python
# -*- coding: utf-8 -*-
"""
Created on Sun Feb 17 23:08:39 2019

@author: qinzhen
"""

from scipy.special import comb
import matplotlib.pyplot as plt

def m(N, d):
```
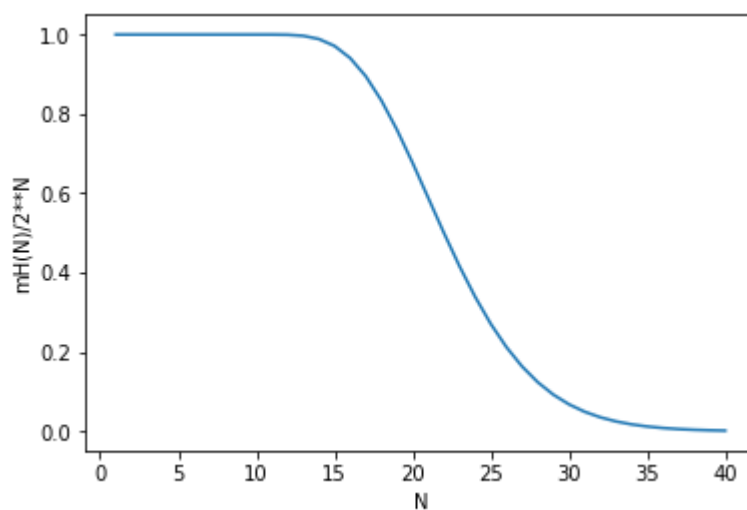
```
    result = 0
    k = min(d, N-1)
    for i in range(k + 1):
        result += comb(N-1, i)
    return 2 * result

x = range(1,41)
d = 10
y = [m(i, d) / (2 ** i) for i in x]

plt.plot(x,y)
plt.xlabel('N')
plt.ylabel('mH(N)/2**N')
plt.show()
```



```
print([m(i, d) / (2 ** i) for i in [10, 20, 40]])
```

```
[1.0, 0.67619705200195313, 0.0016889239559532143]
```

概率如上所示，可以看到对于 $d = 10$，只要有40个点就几乎不可能表出了。

Problem 2.10 (Page 70)

Show that $m_{\mathcal{H}}(2N) \leq m_{\mathcal{H}}(N)^2$, and hence obtain a generalization bound which only involves $m_{\mathcal{H}}(N)$

将2N个点分为前N个和后N个，记前N个点的区分方法为$f_1(N)$，后N个点的区分方法为$f_2(N)$,由乘法原理，$m_{\mathcal{H}}(2N) \leq f_1(N) \times f_2(N)$，显然$f_1(N) \leq m_{\mathcal{H}}(N), f_2(N) \leq m_{\mathcal{H}}(N)$。因此

$$m_{\mathcal{H}}(2N) \leq m_{\mathcal{H}}(N)^2$$

Problem 2.11 (Page 70)

Suppose $m_{\mathcal{H}}(N) = N + 1$, so $d_{vc} = 1$. You have 100 training examples. Use the generalization bound to give a bound for E_{out} with confidence 90%. Repeat for $N = 10,000$.

这题要使用上面证明的不等式，先回顾下书本第58页的公式

$$E_{\text{out}}(g) \le E_{\text{in}}(g) + \sqrt{\frac{8}{N}\ln(\frac{4m_{\mathcal{H}}(2N)}{\delta})}$$ 的 概 率 大 于 等 于 $1 - \delta$

对这题来说，$\delta = 0.1$。下面编程计算下

```
# -*- coding: utf-8 -*-
"""
Created on Sun Feb 17 23:14:03 2019

@author: qinzhen
"""

import numpy as np

delta = 0.1

def f(N, delta):
    return np.sqrt(8 / N * np.log(4 * (2 * N + 1) /delta))
```

```
print(f(100, delta))
```

```
0.8481596247015304
```

```
print(f(10000, delta))
```

```
0.10427815497178729
```

Problem 2.12 (Page 71)

For an $\mathcal{H}$ with $d_{vc} = 10$, what sample size do you need (as prescribed by the generalization bound) to have a 95% confidence that your generalization error is at most 0.05?

在此回顾下课本58页的公式

$$E_{\text{out}}(g) \le E_{\text{in}}(g) + \sqrt{\frac{8}{N}\ln(\frac{4((2N)^{d_{vc}} + 1)}{\delta})}$$ 的 概 率 大 于 等 于 $1 - \delta$

回到这题，题目中 $\delta = 0.05, \sqrt{\frac{8}{N}\ln(\frac{4((2N)^{d_{vc}}+1)}{\delta})} = 0.05$ 这个方程直接解的话不好解，可以作图看一下

```
# -*- coding: utf-8 -*-
"""
```

```
Created on Sun Feb 17 23:20:11 2019

@author: qinzhen
"""

import numpy as np
import matplotlib.pyplot as plt

delta=0.05
dvc=10

def f(N):
    return (8 / N * np.log(4 * ((2 * N) ** dvc + 1) / delta)) ** 0.5 - 0.05

n = 1
while(True):
    if(f(n) <= 0):
        break
    else:
        n += 1

print(n)
```
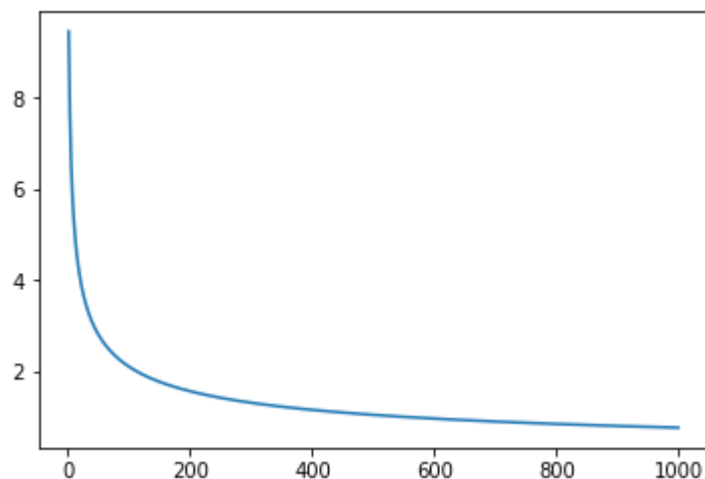
```
452957
```

可以看到这个数字非常大，我们再来作图看一下

```
#作图
x = range(1, 1000)
y = [f(i) for i in x]

plt.plot(x, y)
plt.show()
```


Problem 2.13 (Page 71)

(a) Let $\mathcal{H} = \{h_1, h_2, \ldots, h_M\}$ with some finite M. Prove that $d_{vc}(\mathcal{H}) \leq \log_2 M$.

(b) For hypothesis sets $\mathcal{H}_1, \mathcal{H}_2, \ldots \mathcal{H}_k$ with finite VC dimensions $d_{vc}(\mathcal{H}_k)$, derive and prove the tightest upper and lower bound that you can get on $d_{vc}(\bigcap_{k=1}^{K} \mathcal{H}_k)$.

(c) For hypothesis sets $\mathcal{H}_1, \mathcal{H}_2, \ldots, \mathcal{H}_k$ with finite VC dimensions $d_{vc}(\mathcal{H}_k)$, derive and prove the tightest upper and lower bounds that you can get on $d_{vc}(\bigcup_{k=1}^{K} \mathcal{H}_k)$.

(a)因为有M个假设，所以一共能区分M种情况。$d_{vc}(\mathcal{H})$表示对于$n = d_{vc}(\mathcal{H})$组数据，这M个假设可以区分出来所有$2^{d_{vc}(\mathcal{H})}$种情形，而一共最多能区分M种情形，因此

$$2^{d_{vc}(\mathcal{H})} \leq M$$
$$d_{vc}(\mathcal{H}) \leq \log_2(M)$$

(b)首先有直觉反应：

$$如果\ A \subseteq B, 那么 d_{vc}(A) \leq d_{vc}(B)$$

所以猜测

$$0 \leq d_{vc}(\bigcap_{k=1}^{K} \mathcal{H}_k) \leq \min\{(d_{vc}(\mathcal{H}_k))\}_{k=1}^{K}$$

$0 \leq d_{vc}(\bigcap_{k=1}^{K} \mathcal{H}_k)$是显然的，另一边使用反证法，记

$$d_1 = d_{vc}(\bigcap_{k=1}^{K} \mathcal{H}_k), d_2 = \min\{(d_{vc}(\mathcal{H}_k))\}_{k=1}^{K}$$

若$d_1 \geq d_2 + 1$，那么$\bigcap_{k=1}^{K} \mathcal{H}_k$可以shatter $d_2 + 1$个点，那么至少存在一个$\mathcal{H}_i(i = 1\ldots k)$，使得$\mathcal{H}_i$也可以shatter $d_2 + 1$个点，这就与$d_2 = \min\{(d_{(vc)}(H_k))\}_{k=1}^{K}$相矛盾了，所以

$$d_{vc}(\bigcap_{k=1}^{K} \mathcal{H}_k) \leq \min\{(d_{vc}(\mathcal{H}_k))\}_{k=1}^{K}$$

并且该上界是最强上界。

接着这个思路，也可以证明以下结论：

$$如果\ A \subseteq B, 那么 d_{vc}(A) \leq d_{vc}(B)$$

反证法，如果$d_{vc}(A) \geq d_{vc}(B) + 1$，那么$A$可以shatter $d_{vc}(B) + 1$个点，由$A \subseteq B$。那么B也可以shatter $d_{vc}(B) + 1$个点，矛盾。因此该结论成立。

(c)这题参考了别人的笔记，[笔记地址](#)

这题说实话要不是之前做过林老师的作业，结论我是猜不出的，所以这里就直接证明了，先上结论。

$$\max\{d_{vc}(\mathcal{H}_k)\}_{k=1}^{K} \leq d_{vc}(\bigcup_{k=1}^{K} \mathcal{H}_k) \leq K - 1 + \sum_{k=1}^{K} d_{vc}(\mathcal{H}_k)$$

左边比较简单，由之前的结论即可。

因为

$$H_k \subseteq \bigcup_{k=1}^{K} H_k$$

所以

$$d_{vc}(\mathcal{H}_k) \leq d_{vc}(\bigcup_{k=1}^{K} \mathcal{H}_k)(k = 1...K)$$

$$\max\{d_{vc}(\mathcal{H}_k)\}_{k=1}^{K} \leq d_{vc}(\bigcup_{k=1}^{K} \mathcal{H}_k)$$

观察下右边的形式，其实只要证明 $K = 2$ 的情形然后使用数学归纳法即可，$K = 2$ 时的结论为

$$d_{vc}(\mathcal{H}_1 \bigcup \mathcal{H}_2) \leq 1 + d_{vc}(\mathcal{H}_1) + d_{vc}(\mathcal{H}_2)$$

下面记 $d_1 = d_{vc}(\mathcal{H}_1), d_2 = d_{vc}(\mathcal{H}_2)$，考虑成长函数 $m_{\mathcal{H}}(N)$，首先

$$m_{\mathcal{H}_1 \bigcup \mathcal{H}_2}(N) \leq m_{\mathcal{H}_1}(N) + m_{\mathcal{H}_2}(N)$$

对于 N 个元素，$\mathcal{H}_1 \bigcup \mathcal{H}_2$ 最多可以表示出 $m_{\mathcal{H}_1 \bigcup \mathcal{H}_2}(N)$ 种分类，对于每种分类，或者由 $\mathcal{H}_1$ 表示，或者由 $\mathcal{H}_2$ 表示，所以 $m_{\mathcal{H}_1 \bigcup \mathcal{H}_2}(N)$ 应该小于 $\mathcal{H}_1$ 和 $\mathcal{H}_2$ 表示出来的分类之和。而 $\mathcal{H}_1$ 最多可以表示出 $m_{\mathcal{H}_1}(N)$ 种分类，$\mathcal{H}_2$ 最多可以表示出 $m_{\mathcal{H}_2}(N)$ 种分类，因此上述不等式成立。

回到原题，使用上述结论以及VC不等式，我们可得

$$m_{\mathcal{H}_1 \bigcup \mathcal{H}_2}(N) \leq m_{\mathcal{H}_1}(N) + m_{\mathcal{H}_2}(N) \leq \sum_{i=0}^{d_1} \binom{N}{i} + \sum_{i=0}^{d_2} \binom{N}{i}$$

接着我们使用反证法证明

$$d_{vc}(\mathcal{H}_1 \bigcup \mathcal{H}_2) \leq 1 + d_{vc}(\mathcal{H}_1) + d_{vc}(\mathcal{H}_2)$$

如果

$$d_{vc}(\mathcal{H}_1 \bigcup \mathcal{H}_2) \geq 2 + d_{vc}(\mathcal{H}_1) + d_{vc}(\mathcal{H}_2) = d_1 + d_2 + 2$$

那么

$$m_{\mathcal{H}_1 \bigcup \mathcal{H}_2}(d_1 + d_2 + 2) \geq 2^{d_1+d_2+2}$$

把 $N = d_1 + d_2 + 2$ 带入 $\sum_{i=0}^{d_1} \binom{N}{i} + \sum_{i=0}^{d_2} \binom{N}{i}$ 得

$$\sum_{i=0}^{d_1}\binom{N}{i}+\sum_{i=0}^{d_2}\binom{N}{i}=\sum_{i=0}^{d_1}\binom{d_1+d_2+2}{i}+\sum_{i=0}^{d_2}\binom{d_1+d_2+2}{i}$$

$$=\sum_{i=0}^{d_1}\binom{d_1+d_2+2}{i}+\sum_{i=0}^{d_2}\binom{d_1+d_2+2}{d_1+d_2+2-i}$$

令 $j=d_1+d_2+2-i$，那么 j 的范围从 d_1+2 到 d_1+d_2+2

$$=\sum_{i=0}^{d_1}\binom{d_1+d_2+2}{i}+\sum_{j=d_1+2}^{d_1+d_2+2}\binom{d_1+d_2+2}{j}$$

$$=2^{d_1+d_2+2}-\binom{d_1+d_2+2}{d_1+1}$$

$$<2^{d_1+d_2+2}$$

所以

$$m_{\mathcal{H}_1\bigcup\mathcal{H}_2}(d_1+d_2+2)\leq\sum_{i=0}^{d_1}\binom{d_1+d_2+2}{i}+\sum_{i=0}^{d_2}\binom{d_1+d_2+2}{i}<2^{d_1+d_2+2}$$

与之前所述矛盾。因此

$$d_{vc}(\mathcal{H}_1\bigcup\mathcal{H}_2)\leq 1+d_{vc}(\mathcal{H}_1)+d_{vc}(\mathcal{H}_2)=d_1+d_2+1$$

因此当 $K=2$ 时，结论成立。

假设 $K=n$ 时不等式成立，$K=n+1$ 时

$$d_{vc}(\bigcup_{k=1}^{n+1}\mathcal{H}_k)=d_{vc}((\bigcup_{k=1}^{n}\mathcal{H}_k)\bigcup\mathcal{H}_{n+1})$$

$$\leq 1+d_{vc}(\bigcup_{k=1}^{n}\mathcal{H}_k)+d_{vc}(\mathcal{H}_{n+1})$$

$$\leq 1+n-1+\sum_{k=1}^{n}d_{vc}(\mathcal{H}_k)+d_{vc}(\mathcal{H}_{n+1})$$

$$=n+\sum_{k=1}^{n+1}d_{vc}(\mathcal{H}_k)$$

因此 $K=n+1$ 时不等式也成立。

Problem 2.14 (Page 71)

Let $\mathcal{H}_1,\mathcal{H}_2,\ldots,\mathcal{H}_K$ be K hypothesis sets with finite VC dimension d_{vc}. Let $\mathcal{H}=\mathcal{H}_1\cup\mathcal{H}_2\cup\ldots\cup\mathcal{H}_K$ be the union of these models.

(a) Show that $d_{vc}(\mathcal{H})<K(d_{vc}+1)$.

(b) Suppose that l satisfies $2^l>2Kl^{d_{vc}}$. Show that $d_{vc}(\mathcal{H})\leq l$.

(c) Hence, show that

$$d_{vc}(\mathcal{H}) \leq \min(K(d_{vc} + 1), 7(d_{vc} + K)\log_2(d_{vc}K))$$

That is, $d_{vc}(H) = O(\max(d_{vc}, K)\log_2 \max(d_{vc}, K))$ is not too bad .

(a)使用上题证明的结论

$$d_{vc}(\bigcup_{k=1}^{K} \mathcal{H}_k) \leq K - 1 + \sum_{k=1}^{K} d_{vc}(\mathcal{H}_k)$$

将 $\mathcal{H} = \mathcal{H}_1 \cup H_2 \cup \ldots \cup H_K, d_{vc}(H_k) = d_{vc}(k = 1, 2 \ldots K)$带入得

$$d_{vc}(\mathcal{H}) \leq K - 1 + K d_{vc} < K(d_{vc} + 1)$$

(b)先回顾一个不等式

$$m_{\mathcal{H}}(N) \leq N^{d_{vc}} + 1$$

这里对这个不等式再加个更宽的上界

$$m_{\mathcal{H}}(N) \leq N^{d_{vc}} + 1 \leq 2N^{d_{vc}}$$

在Problem 2.13中我们说过一个结论

$$m_{\mathcal{H}_1 \bigcup \mathcal{H}_2}(N) \leq m_{\mathcal{H}_1}(N) + m_{\mathcal{H}_2}(N)$$

这个结论显然可以推广为K个集合的不等式：

$$m_{\bigcup_{i=1}^{K} \mathcal{H}_i}(N) \leq \sum_{i=1}^{K} m_{\mathcal{H}_i}(N)$$

因此对这题来说，我们取N为条件中的l

$$m_{\mathcal{H}}(l) = m_{\bigcup_{i=1}^{K} \mathcal{H}_i}(l) \leq \sum_{i=1}^{K} m_{\mathcal{H}_i}(l) \leq \sum_{i=1}^{K} 2l^{d_{vc}} = 2K l^{d_{vc}}$$

最后一个不等号是因为$m_{\mathcal{H}}(N) \leq N^{d_{vc}} + 1 \leq 2N^{d_{vc}}$，以及这题中的每个$\mathcal{H}_i$的 VC dimension均为$d_{vc}$。这时我们来看下题目给的条件

$$2^l > 2K l^{d_{vc}}$$

那么

$$m_{\mathcal{H}}(l) \leq 2K l^{d_{vc}} < 2^l$$

所以$\mathcal{H}$无法shatter l个点，因此我们可以得到比题目中更强的结论

$$d_{vc}(\mathcal{H}) < l$$

(c)由(a)我们知道

$$d_{vc}(\mathcal{H}) < K(d_{vc} + 1)$$

由(b)我们知道，要证明

$$d_{vc}(\mathcal{H}) \le 7(d_{vc} + K)\log_2(d_{vc}K) \triangleq s$$

我们只要验证

$$2^s > 2Ks^{d_{vc}}$$

或者取对数之后的情形

$$s > 1 + \log_2 K + d_{vc}\log_2 s$$

为了叙述方便，这里令$d_{vc} = x, K = y$，从而

$$s = 7(d_{vc} + K)\log_2(d_{vc}K) = 7(x + y)\log_2(xy)$$

所以上述式子化为

$$2^s > 2ys^x$$
$$s > 1 + \log_2 y + x\log_2 s$$

不失一般性，这里只考虑$K = y \ge 2$的情形。

先考虑$d_{vc} = x = 1$的情形，那么此时$s = 7(1 + y)\log_2(y)$，我们只要验证$2^s > 2ys$即可。由二项式定理

$$2^s = (1 + 1)^s \ge s + \frac{s(s-1)}{2} > \frac{s^2}{2} = \frac{s(7(1+y)\log_2(y))}{2}$$

由$y \ge 2$，我们知道$\log_2(y) \ge 1$。因此

$$2^s > \frac{s(7(1+y)\log_2(y))}{2} \ge \frac{s(7(1+y))}{2} > \frac{s(4y)}{2} = 2ys$$

因此$d_{vc} = x = 1$时结论成立。

接下来考虑$d_{vc} = x \ge 2$的情形，此时$s = 7(x + y)\log_2(xy)$，接下来我们要验证

$$s > 1 + \log_2 y + x\log_2 s$$

首先有俩个很显然的结论

$$\log_2\log_2(xy) \le \log_2(x + y)$$
$$\text{当}x, y \ge 2\text{时，} 7(x + y) < (xy)^6$$
$$x, y \text{均为正整数}$$

证明如下。

第一个结论：

$$\log_2\log_2(xy) \le \log_2(x + y) \qquad \Leftrightarrow$$
$$\log_2(xy) \le x + y \qquad \Leftrightarrow$$
$$xy \le 2^{x+y} \qquad \Leftrightarrow$$

注意到

$$xy \le \frac{(x+y)^2}{4}$$

所以我们只要证明

$$\frac{(x+y)^2}{4} \le 2^{x+y} \qquad \Leftrightarrow$$
$$(x+y)^2 \le 2^{x+y+2} \qquad \Leftrightarrow$$
$$2\log_2(x+y) \le x+y+2$$

即可。令

$$t = x+y, f(t) = t+2-2\log_2 t$$

那么

$$f'(t) = 1 - \frac{2}{t\ln 2}$$

注意到 x,y 都为正整数，所以 $t \ge 2$，因此

$$当 t = 2 时，\ f'(t) = 1 - \frac{1}{\ln 2} < 0$$
$$当 t \ge 3 时，\ f'(t) \ge 1 - \frac{2}{3\ln 2} > 0$$

所以

$$f(t) \ge f(3) > 0, (t \ge 3)$$
$$f(2) = 2 > 0$$
$$f(t) > 0, (t \ge 2)$$

从而

$$f(x+y) = x+y+2-2\log_2(x+y) > 0$$

第二个结论：

$$(x^3-8)(y^3-8) \ge 0$$
$$(xy)^6 \ge 8x^3 + 8y^3 - 8 > 7x + 32y - 8 = 7(x+y) + 25y - 8 > 7(x+y)$$

从而结论成立。

回到原题：

$$1 + \log_2 y + x \log_2 s = \log_2(2y) + x \log_2(7(x+y)\log_2(xy))$$
$$\leq \log_2(xy) + x \log_2(7(x+y)) + x \log_2 \log_2(xy) \text{ (这一步是因为因为 } x \geq 2)$$
$$< \log_2(xy) + x \log_2((x+y)^6) + x \log_2(x+y) \text{ (这一步是因为因为之前介绍的结论)}$$
$$= \log_2(xy) + 7x \log_2(x+y))$$
$$< 7y \log_2(xy) + 7x \log_2(x+y)$$
$$= s$$

因此结论成立，所以

$$d_{vc}(\mathcal{H}) \leq 7(d_{vc} + K) \log_2(d_{vc} K)$$

综上所述

$$d_{vc}(\mathcal{H}) \leq \min(K(d_{vc} + 1), 7(d_{vc} + K) \log_2(d_{vc} K))$$

接着验证下推论

$$d_{vc}(\mathcal{H}) = O(\max(d_{vc}, K) \log_2 \max(d_{vc}, K))$$

由之前的结论我们可得

$$d_{vc} \leq 7(d_{vc} + K) \log_2(d_{vc} K)$$
$$\leq 7 \max(d_{vc}, K) \times 2 \log_2(\max(d_{vc}, K)^2)$$
$$= 28 \max(d_{vc}, K) \log_2 \max(d_{vc}, K)$$

所以推论成立。

总结下，这题是对并集的VC dimension证明一些不等式，进行放缩，技巧性比较强，但是放缩的过程中发现 $7(d_{vc} + K) \log_2(d_{vc} K)$ 其实是一个很宽松的上界，完全可以取一个更小的下界，不过这里最终的结论应该是要得出 $d_{vc}(\mathcal{H}) = O(\max(d_{vc}, K) \log_2 \max(d_{vc}, K))$，因此更小的下界其实也不一定有必要了。

Problem 2.15 (Page 71)

The monotonically increasing hypothesis set is

$$\mathcal{H} = \{h | x_1 \geq x_2 \Rightarrow h(x_1) \geq h(x_2)\}$$

where $x_1 \geq x_2$ if and only if the inequality is satisfied for every component.

(a) Give an example of a monotonic classifier in two dimensions, clearly showing the +1 and -1 regions.

(b) Compute $m_{\mathcal{H}}(N)$ and hence the VC dimension. {Hint: Consider a set of N points generated by first choosing one point, and then generating the next point by increasing the first component and decreasing the second component until N points are obtained.}

(a)对二维空间中的点 $x = (x_1, x_2)$，定义 $h(x) = x_1 + x_2$,显然 $h(x)$ 是递增函数，对应的边界是二维平面中的一条直线

```
# -*- coding: utf-8 -*-
"""
```

```
Created on Tue Feb 19 00:12:24 2019

@author: qinzhen
"""

import matplotlib.pyplot as plt

x = range(-10, 10)
y = [-i for i in x]
up = [10] * len(x)

plt.xlabel('x')
plt.ylabel('y')
plt.plot(x, y)

plt.fill_between(x, y, up)
plt.show()
```
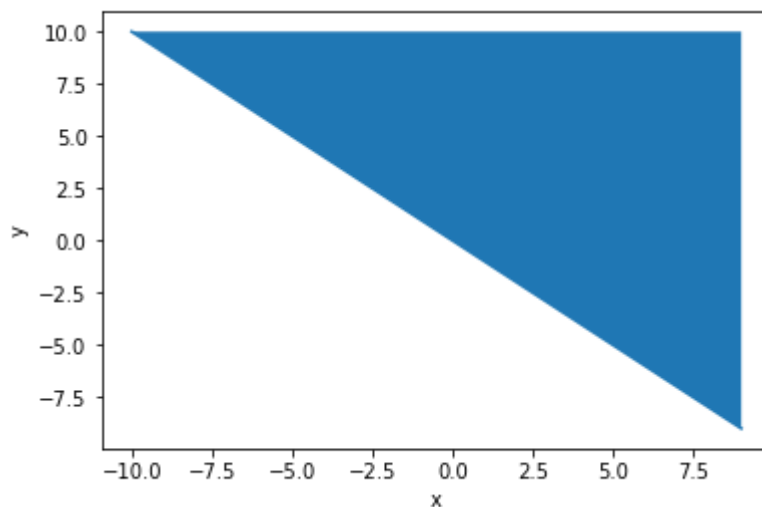

蓝色区域的点标记为$+1$,白色区域的点标记为-1

(b)这题没有说维度，我们先考虑一维的情形，假设有N个点

$$x_1 \le x_2 \le \ldots \le x_N$$

由单调函数的定义我们知道，如果$h(x_k) < 0, h(x_{k+1}) > 0$，那么

$$h(x_i) < 0(i = 1, 2...k)$$
$$h(x_i) > 0(i = k + 1, \ldots N)$$

因此对于一维的情形，相当于一刀切，左边为-1，右边为$+1$，因此一共有$m_{\mathcal{H}}(N) = N + 1$

但是对于二维及以上情形就完全不一样了，我们如提示中这样构造点，先随意取一个点x_1，接着取第二个点x_2，使得x_2的第一个分量比x_1的第一个分量大，但是第二个分量比x_1的第二个分量小，换句话说，我们不让x_2大于$x1$，也不让x_2小于x_1，用同样的方式可以构造$x_3 \ldots x_N$，使得这N的点任意两个点是无序的，对于这样N个点，我们可以shatter 他们，因为此时$h(x_i)$和$h(x_j)$的关系没有限制。所以对于二维及以上情形$m_{\mathcal{H}}(N) = 2^N$

Problem 2.16 (Page 72)

In this problem , we will consider $\mathcal{X} = \mathbb{R}$.That is, x=x is a one dimensional variable. For a hypothesis set

$$\mathcal{H} = \{h_c | h_c(x) = \text{sign}(\sum_{i=0}^{D} c_i x^i)\}$$

prove that the VC dimension of $\mathcal{H}$ is exactly $(D+1)$ by showing that

(a) There are $(D+1)$ points which are shattered by $\mathcal{H}$.

(b) There are no $(D+2)$ points which are shattered by $\mathcal{H}$.

这题和Exercise 2.4基本一样，所以做法也差不多。

(a)记$y_j = \sum_{i=0}^{D} c_i x_j^i$，现在取$D+1$个点，$x_1, x_2 \ldots x_{D+1}, x_j = j$

那么

$$\begin{pmatrix} y_1 \\ y_2 \\ \ldots \\ y_{D+1} \end{pmatrix} = \begin{pmatrix} \sum_{i=0}^{D} c_i x_1^i \\ \sum_{i=0}^{D} c_i x_2^i \\ \ldots \\ \sum_{i=0}^{D} c_i x_{D+1}^i \end{pmatrix} = \begin{pmatrix} 1 & 1 & 1 & \ldots & 1 \\ 1 & 2 & 2^2 & \ldots & 2^D \\ 1 & 3 & 3^2 & \ldots & 3^D \\ \ldots & \ldots & \ldots & \ldots & \ldots \\ 1 & (D+1) & (D+1)^2 & \ldots & (D+1)^D \end{pmatrix} \times \begin{pmatrix} c_0 \\ c_1 \\ \ldots \\ c_D \end{pmatrix}$$

我们知道

$$\begin{pmatrix} 1 & 1 & 1 & \ldots & 1 \\ 1 & 2 & 2^2 & \ldots & 2^D \\ 1 & 3 & 3^2 & \ldots & 3^D \\ \ldots & \ldots & \ldots & \ldots & \ldots \\ 1 & (D+1) & (D+1)^2 & \ldots & (D+1)^D \end{pmatrix}$$ 对应的行列式为范德蒙行列式，不为0

那么对任意的$y = (y_1, y_2, \ldots, y_{D+1})^T$

$$\begin{pmatrix} 1 & 1 & 1 & \ldots & 1 \\ 1 & 2 & 2^2 & \ldots & 2^D \\ 1 & 3 & 3^2 & \ldots & 3^D \\ \ldots & \ldots & \ldots & \ldots & \ldots \\ 1 & (D+1) & (D+1)^2 & \ldots & (D+1)^D \end{pmatrix} \times \begin{pmatrix} c_0 \\ c_1 \\ \ldots \\ c_D \end{pmatrix} = \begin{pmatrix} y_1 \\ y_2 \\ \ldots \\ y_{D+1} \end{pmatrix}$$

关于$c = (c_0, c_1, \ldots c_D)^T$有解，因此存在$D+1$个点可以被shatter

(b)记$z_j = (1, x_j, x_j^2 \ldots x_j^D) \in R^{D+1}$，那么现在任取$D+2$个点$x_1, x_2 \ldots x_{D+2}$，对应可以得到$z_1, z_2 \ldots z_{D+2}$。
R^{D+1}空间任意$D+2$个点必然线性相关，所以不妨设z_{D+2}可以被线性表出，注意$c = (c_0, c_1, \ldots c_D)^T$

$$z_{D+2} = \sum_{i=1}^{D+1} k_i z_i$$

$$c^T z_{D+2} = \sum_{i=1}^{D+1} c^T k_i z_i$$

$$\operatorname{sign}(c^T z_{D+2}) = \operatorname{sign}(\sum_{i=1}^{D+1} k_i c^T z_i)$$

也就是说z_{D+2}的分类被$z_1, z_2 \ldots z_{D+1}$确定，那么

$$(\operatorname{sign}(z_1), \operatorname{sign}(z_2) \ldots \operatorname{sign}(z_{D+1}), -\operatorname{sign}(z_{D+2}))$$

这种情形必然无法被划分出来，因此$d_{vc} \le D+1$

结合(a),(b)我们可得

$$d_{vc} = D+1$$

回顾书上的说明，d_{vc}可以理解为自由分量的个数，对于D次多项式，显然有$D+1$个自由分量。

Problem 2.17 (Page 72)

The VC dimension depends on the input space as well as $\mathcal{H}$. For a fixed $\mathcal{H}$, consider two input spaces $\mathcal{X}_1 \subseteq \mathcal{X}_2$. Show that the VC dimension of $\mathcal{H}$ with respect to input space $\mathcal{X}_1$ is at most the VC dimension of $\mathcal{H}$ with respect to input space $\mathcal{X}_2$.

How can the result of this problem be used to answer part (b) in Problem 2.16? [Hint: How is Problem 2.16 related to a perceptron in D dimensions?}

反正法，假设$d_1 = d_{vc}(\mathcal{X}_1, \mathcal{H}) > d_{vc}(\mathcal{X}_2, \mathcal{H}) = d_2$。那么在$\mathcal{X}_1$中必然有$d_1$个点可以被shatter，由$\mathcal{X}_1 \subseteq \mathcal{X}_2$我们知道这$d_1$个点也在$\mathcal{X}_2$中，因为假设空间均为$\mathcal{H}$，我们知道$\mathcal{X}_2$中这$d_1$个点也可以被shatter，这就与$d_1 = d_{vc}(\mathcal{X}_1, \mathcal{H}) > d_{vc}(\mathcal{X}_2, \mathcal{H}) = d_2$矛盾。因此$d_{vc}(\mathcal{X}_1, \mathcal{H}) \le d_{vc}(\mathcal{X}_2, \mathcal{H})$

回到上题的(b)部分，我们知道$z_j = (1, x_j, x_j^2 \ldots x_j^D) \in \mathbb{R}^{D+1}$，所以整个输入空间的集合构成了整个$\mathbb{R}^{D+1}$的子集，而我们知道维度为$D$（不包括偏置项）的感知机的VC dimension 为$D+1$，因此$d_{vc} \le D+1$。

Problem 2.18 (Page 72)

The VC dimension of the perceptron hypothesis set corresponds to the number of parameters $(w_0, w_1, \ldots, w_d)$ of the set, and this observation is 'usually' true for other hypothesis sets. However, we will present a counter example here. Prove that the following hypothesis set for $x \in \mathbb{R}$ has an infinite VC dimension :

$$\mathcal{H} = \{h_\alpha | h_\alpha = (-1)^{\lfloor \alpha x \rfloor}, \text{where } \alpha \in \mathbb{R}\}$$

where $\lfloor A \rfloor$ is the biggest integer $\le A$ (the floor function). This hypothesis has only one parameter α but 'enjoys' an infinite VC dimension. [Hint: Consider $x_1, \ldots, x_N$, where $x_n = 10^n$, and show how to implement an arbitrary dichotomy $y_1, \ldots, y_N$.]

由感知机模型的例子我们似乎可以得出VC dimension和变量的自由分量有关，这题就给出一个反例，只有一个参数，但是VC dimension是无穷大。

任取$(y_1, \ldots, y_N) \in \{-1, +1\}^N$，我们构造这样一个$z = (z_1, \ldots, z_N)$，使得

$$\text{当} y_i = +1 \text{时，} z_i = 0$$
$$\text{当} y_i = -1 \text{时，} z_i = 1$$
$$\text{简单来说，就是} (-1)^{z_i} = y_i$$

现在取

$$\alpha = 0.z_1 z_2 \ldots z_N$$
$$x = (x_1, \ldots, x_N), x_n = 10^n$$

那么

$$\begin{aligned} h_\alpha(x_n) &= (-1)^{\lfloor \alpha x_n \rfloor} \\ &= (-1)^{\lfloor 0.z_1 z_2 \ldots z_N \times 10^n \rfloor} \\ &= (-1)^{z_1 z_2 \ldots z_n} \\ &= (-1)^{z_n} \\ &= y_n (\text{回顾之前的结论}) \end{aligned}$$

所以存在$x = (x_1, \ldots, x_N)$可以被shatter，由N的任意性可得$\mathcal{H}$的VC dimension为无穷大。

Problem 2.19 (Page 72)

This problem derives a bound for the VC dimension of a complex hypothesis set that is built from simpler hypothesis sets via composition. Let $\mathcal{H}_1, \ldots, \mathcal{H}_k$ be hypothesis sets with VC dimension $d_1, \ldots, d_K$. Fix $h_1, \ldots, h_k$, where $h_i \in \mathcal{H}_i$. Define a vector z obtained from x to have components $h_i(x)$. Note that $x \in \mathbb{R}^d$, but z $z \in \{-1, +1\}^K$. Let $\tilde{\mathcal{H}}$ be a hypothesis set of functions that take inputs in $\mathbb{R}^K$. So

$$\tilde{h} \in \tilde{\mathcal{H}} : z \in \mathbb{R}^K \mapsto \{+1, -1\}$$

and suppose that $\tilde{\mathcal{H}}$ has VC dimension $\tilde{d}$.

We can apply a hypothesis in $\tilde{\mathcal{H}}$ to the z constructed from $(h_1, \ldots, h_K)$. This is the composition of the hypothesis set $\tilde{\mathcal{H}}$ with $(\mathcal{H}_1, \ldots, \mathcal{H}_k)$. More formally, the composed hypothesis set $H = \tilde{\mathcal{H}} \circ (\mathcal{H}_1, \ldots, \mathcal{H}_k)$ is defined by $h \in \mathcal{H}$ if

$$h(x) = \tilde{h}(h_1(x), \ldots, h_K(x)), \tilde{h} \in \tilde{\mathcal{H}}; h_i \in \mathcal{H}_i$$

(a) Show that

$$m_{\mathcal{H}}(N) \leq m_{\tilde{\mathcal{H}}}(N) \prod_{i=1}^{K} m_{\mathcal{H}_i}(N) \tag{2.18}$$

[Hint: Fix N points $x_1, \ldots, x_N$ and fix $h_1, \ldots, h_K$. This generates N transformed points $z_1, \ldots, z_N$. These $z_1, \ldots, z_N$ can be dichotomized in at most $m_{\tilde{h}}(N)$ ways, hence for fixed $(h_1, \ldots, h_k)$, $(x_1, \ldots, x_N)$ can be dichotomized in at most $m_{\tilde{\mathcal{H}}}(N)$ ways. Through the eyes of $x_1, \ldots, x_N$, at most how many hypotheses are there (effectively) in $\mathcal{H}_i$? Use this bound to bound the effective number of K-tuples $(h_1, \ldots, h_K)$ that need to

be considered. Finally, argue that you can bound the number of dichotomies that can be implemented by the product of the number of possible K-tuples $(h_1, \ldots, h_K)$ and the number of dichotomies per K-tuple.]

(b) Use the bound $m(N) \leq (\frac{eN}{d_{vc}})^{d_{vc}}$ to get a bound for $m_{\mathcal{H}}(N)$ in terms of $\tilde{d}, d_1, \ldots, d_K$.

(c) Let $D = \tilde{d} + \sum_{i=1}^{K} d_i$, and assume that $D > 2e \log_2 D$. Show that

$$d_{vc}(\mathcal{H}) \leq 2D \log_2 D.$$

(d) If $\mathcal{H}_i$ and $\tilde{H}$ are all perceptron hypothesis sets, show that

$$d_{vc}(\mathcal{H}) = O(dK \log(dK)).$$

In the next chapter, we will further develop the simple linear model. This linear model is the building block of many other models, such as neural networks. The results of this problem show how to bound the VC dimension of the more complex models built in this manner.

这题希望解释的是复合函数的VC dimension以及$m_{\mathcal{H}}(N)$，下面分别看一下

(a)我们根据提示里一步步来。

首先对于固定的N个点 $x_1, \ldots, x_N$和固定的 $h_1, \ldots, h_K$，我们可以得到N个转换后的点$z_1, \ldots, z_N$，那么对于这N个点和假设$\tilde{\mathcal{H}}$，最多可以可以表示$m_{\tilde{\mathcal{H}}}(N)$种组合。因此对于固定的$(x_1, \ldots, x_N)$和$(h_1, \ldots, h_K)$最多可以表示$m_{\tilde{\mathcal{H}}}(N)$种组合，所以接下来只要看$(h_1, \ldots, h_K)$可以表示出多少种有效的组合，这部分可以用数学公式写下来：

$$m_{\mathcal{H}}(N) \leq m_{\tilde{h}}(N)((h_1, \ldots, h_K)_{最多可以表示的组合数量})$$

我们还是用之前类似的思路考虑，对于固定的$h_2, h_3, \ldots, h_K$，$(h_1(x_1), h_1(x_2), \ldots, h_1(x_N))$最多可以表示出$m_{\mathcal{H}_1}(N)$种组合，同理对于任意$h_i$，固定其他的$h_j(i \neq j)$，最多可以表示出$m_{\mathcal{H}_i}$种组合，因此由乘法原理可得：

$$m_{\mathcal{H}}(N) \leq m_{\tilde{\mathcal{H}}}(N) \prod_{i=1}^{K} m_{\mathcal{H}_i}(N)$$

(b)这个就比较简单了，题目中的不等式可以查看Problem 2.6。由$m(N) \leq (\frac{eN}{d_{vc}})^{d_{vc}}$，我们可得：

$$m_{\mathcal{H}}(N) \leq m_{\tilde{H}}(N) \prod_{i=1}^{K} m_{\mathcal{H}_i}(N)$$

$$\leq (\frac{eN}{\tilde{d}})^{\tilde{d}} \prod_{i=1}^{K} (\frac{eN}{d_i})^{d_i}$$

$$= \frac{(eN)^{\tilde{d} + \sum_{i=1}^{K} d_i}}{\tilde{d}^{\tilde{d}} \prod_{i=1}^{K} d_i^{d_i}}$$

(c)只要证明

$$当 N = 2D \log_2 D 时, \; m(N) < 2^N 即可$$

由上题我们知道

$$m_{\mathcal{H}}(N) \leq \frac{(eN)^{\tilde{d}+\sum_{i=1}^{K} d_i}}{\tilde{d}^{\tilde{d}} \prod_{i=1}^{K} d_i^{d_i}} = \frac{(eN)^D}{\tilde{d}^{\tilde{d}} \prod_{i=1}^{K} d_i^{d_i}} = \frac{(eN)^D}{e^{\tilde{d} \ln(\tilde{d})+\sum_{i=1}^{K} d_i \ln(d_i)}}$$

下面对分母 $e^{\tilde{d} \ln(\tilde{d})+\sum_{i=1}^{K} d_l n(d_i)}$ 进行估计，首先介绍 **琴生不等式** **(Jensen's inequality)**

若 $f(x)$ 是下凸函数 $(f''(x) \geq 0)$，那么对于任意 $\lambda_1 + \lambda_2 + \ldots + \lambda_n = 1 (\lambda_i \geq 0)$ 有

$$\sum_{i=1}^{n} \lambda_i f(x_i) \geq f(\sum_{i=1}^{n} \lambda_i x_i)$$

特别的，取 $\lambda_i = \frac{1}{n}$ 时可得

$$\frac{\sum_{i=1}^{n} f(x_i)}{n} \geq f(\frac{\sum_{i=1}^{n} x_i}{n})$$

回到原题，我们令 $f(x) = x \ln x$，那么

$$f'(x) = \ln x + 1$$
$$f''(x) = \frac{1}{x} > 0$$

因此 $f(x)$ 是下凸函数，现在令 $\tilde{d} = d_{K+1}$，通过变形以及琴生不等式可得

$$\begin{aligned}
e^{\tilde{d} \ln(\tilde{d})+\sum_{i=1}^{K} d_i \ln(d_i)} &= e^{\sum_{i=1}^{K+1} d_i \ln(d_i)} \\
&= e^{\sum_{i=1}^{K+1} f(d_i)} \\
&\geq e^{(K+1)f(\frac{\sum_{i=1}^{K+1} d_i}{K+1})} \left(\text{这一步是由琴生不等式}\right) \\
&= e^{(K+1)f(\frac{D}{K+1})} \left(\text{注意} D = \tilde{d} + \sum_{i=1}^{K} d_i = \sum_{i=1}^{K+1} d_i\right) \\
&= e^{(K+1) \times \frac{D}{K+1} \ln(\frac{D}{K+1})} \\
&= e^{D \ln(\frac{D}{K+1})} \\
&= (\frac{D}{K+1})^D
\end{aligned}$$

因此

$$m_{\mathcal{H}}(N) \leq \frac{(eN)^D}{e^{\tilde{d} \ln(\tilde{d})+\sum_{i=1}^{K} d_i \ln(d_i)}} \leq \frac{(eN)^D}{(\frac{D}{K+1})^D}$$

现在取 $N = 2D \log_2 D$，那么上述不等式右边可以化为

$$\frac{(eN)^D}{(\frac{D}{K+1})^D} = \frac{(e2D \log_2 D)^D}{(\frac{D}{K+1})^D} = (2e \log_2 D)^D (K+1)^D \left(\text{这一步是约去分子分母的} D^D\right)$$

回顾题目中的假设 $D > 2e \log_2 D$，以及一个比较显然但是容易忽略的结论

$$D = \tilde{d} + \sum_{i=1}^{K} d_i \geq K + 1$$

注意$N = 2D \log_2 D$，那么带入上式可得

$$m_{\mathcal{H}}(2D \log_2 D) = m_{\mathcal{H}}(N) \leq \frac{(eN)^D}{\left(\frac{D}{K+1}\right)^D} = (2e \log_2 D)^D (K+1)^D < D^D D^D = D^{2D} = 2^{2D \log_2 D}$$

因此当$N = 2D \log_2 D$时，$m_{\mathcal{H}}(N) < 2^N$，所以

$$d_{vc}(\mathcal{H}) \leq 2D \log_2 D.$$

这题总体来说还是有一定难度的，比较难的一点是琴生不等式。

(d)结合(c)来考虑这题，对于$\mathbb{R}^n$中的感知机模型，我们知道它的VC dimension为$n + 1$，这题中$(\mathcal{H}_1, \ldots, \mathcal{H}_k)$为$\mathbb{R}^d$中的感知机，$\tilde{\mathcal{H}}$为$\mathbb{R}^K$中的感知机，因此

$$\begin{aligned}
D &= \tilde{d} + \sum_{i=1}^{K} di \\
&= K + 1 + \sum_{i=1}^{K} (d+1) \\
&= Kd + K + K + 1 \\
&= Kd + 2K + 1
\end{aligned}$$

带入(c)中可得

$$d_{vc}(\mathcal{H}) \leq 2D \log_2 D = 2(Kd + 2K + 1) \log_2(Kd + 2K + 1)$$

因此$d_{vc} = O(dK \log(dK))$

可能有人会说(c)需要$D > 2e \log_2 D$成立，这个不等式左边是一次式，右边是对数式，所以只要D取较大的数，例如32，这个不等式就必然成立。由$D = \tilde{d} + \sum_{i=1}^{K} d_i$可得$D \geq K + 1$，所以只要$K$稍大一些就能满足不等式。

总结一下，我认为这题的非常好，这个对于复合函数的VC dimension估计实际上非常好的解释了有些特征转换后的模型为什么可以shatter维度非常高的数据，从数学角度证明了看起来显然的事情。

Problem 2.20 (Page 73)

There are a number of bounds on the generalization error ϵ, all holding with probability at least $1 - \delta$.

(a) Original VC-bound:

$$\epsilon \leq \sqrt{\frac{8}{N} \ln \frac{4m_{\mathcal{H}}(2N)}{\delta}}$$

(b) Rademacher Penalty Bound:

$$\epsilon \leq \sqrt{\frac{2 \ln(2N m_{\mathcal{H}}(N))}{N}} + \sqrt{\frac{2}{N} \ln \frac{1}{\delta}} + \frac{1}{N}$$

(c) Parrondo and Van den Broek:

$$\epsilon \leq \sqrt{\frac{1}{N}\left(2\epsilon + \ln\frac{6m_{\mathcal{H}}(2N)}{\delta}\right)}$$

(d) Devroye:

$$\epsilon \leq \sqrt{\frac{1}{2N}\left(4\epsilon(1+\epsilon) + \ln\frac{4m_{\mathcal{H}}(N^2)}{\delta}\right)}$$

Note that (c) and (d) are implicit bounds in ϵ. Fix $d_{vc} = 50$ and $\delta = 0.05$ and plot these bounds as a function of N. Which is best?

这题就是按照公式估计上界并作图，注意这里我们对$m_{\mathcal{H}}(N)$的估计为

$$m_{\mathcal{H}}(N) \leq \sum_{i=0}^{d_{vc}}\binom{N}{i}$$

但(c),(d)左右两边都有ϵ，应该要处理一下。

先看(c)，令$a = \frac{1}{N}, b = \ln\frac{6m_{\mathcal{H}}(2N)}{\delta}$，那么不等式可以化为

$$\epsilon \leq \sqrt{2a\epsilon + ab}$$

两边平方可得

$$\epsilon^2 \leq 2a\epsilon + ab$$
$$(\epsilon - a)^2 \leq ab + a^2$$
$$-\sqrt{ab + a^2} \leq \epsilon - a \leq \sqrt{ab + a^2}$$
$$a - \sqrt{ab + a^2} \leq \epsilon \leq a + \sqrt{ab + a^2}$$

注意这里只要考虑上界，所以

$$\epsilon \leq a + \sqrt{ab + a^2}$$

将$a = \frac{1}{N}, b = \ln\frac{6m_{\mathcal{H}}(2N)}{\delta}$带入可得

$$\epsilon \leq \frac{1}{N} + \sqrt{\frac{1}{N^2} + \frac{1}{N}\ln\frac{6m_{\mathcal{H}}(2N)}{\delta}}$$

对于(d)，令$a = \frac{1}{2N}, b = \ln\frac{4m_{\mathcal{H}}(N^2)}{\delta}$，那么不等式可以化为

$$\epsilon \leq \sqrt{a(4\epsilon(1+\epsilon) + b)}$$

两边平方可得

$$\epsilon^2 \leq 4a\epsilon(1+\epsilon) + ab$$
$$(1 - 4a)\epsilon^2 - 4a\epsilon \leq ab$$

注意这里我们只要考虑N很大的情形，所以

$$1 - 4a = 1 - \frac{2}{N} > 0$$

从而

$$(1 - 4a)(\epsilon - \frac{2a}{1 - 4a})^2 \leq ab + \frac{4a^2}{1 - 4a}$$

$$(\epsilon - \frac{2a}{1 - 4a})^2 \leq \frac{ab + \frac{4a^2}{1-4a}}{1 - 4a}$$

$$-\sqrt{\frac{ab + \frac{4a^2}{1-4a}}{1 - 4a}} \leq \epsilon - \frac{2a}{1 - 4a} \leq \sqrt{\frac{ab + \frac{4a^2}{1-4a}}{1 - 4a}}$$

这里只要考虑上界，因此

$$\epsilon \leq \sqrt{\frac{ab + \frac{4a^2}{1-4a}}{1 - 4a}} + \frac{2a}{1 - 4a} = \sqrt{\frac{ab}{1 - 4a} + \frac{4a^2}{(1 - 4a)^2}} + \frac{2a}{1 - 4a}$$

将$a = \frac{1}{2N}, b = \ln \frac{4m_{\mathcal{H}}(N^2)}{\delta}$带入可得

$$\epsilon \leq \sqrt{\frac{\ln \frac{4m_{\mathcal{H}}(N^2)}{\delta}}{2(N - 2)} + \frac{1}{(N - 2)^2}} + \frac{1}{N - 2}$$

```python
# -*- coding: utf-8 -*-
"""
Created on Tue Feb 19 15:19:48 2019

@author: qinzhen
"""

from scipy.special import comb
import numpy as np
import matplotlib.pyplot as plt
plt.rcParams['font.sans-serif']=['SimHei'] #用来正常显示中文标签
plt.rcParams['axes.unicode_minus']=False #用来正常显示负号

dvc = 50
delta = 0.05

#计算m(N)
def m(n):
    k = min(dvc, n)
    s = 0
    for i in range(k+1):
        s += comb(n, k)
    return s
```

```python
#Original VC-bound
def f1(n):
    result = (8 / n) * np.log(4 * m(2 * n) / delta)
    result = result ** 0.5
    return result

#Rademacher Penalty Bound
def f2(n):
    k1 = (2 * np.log(2 * n * m(n))) / n
    k2 = (2 / n) * np.log(1 / delta)
    k3 = 1 / n
    result = k1 ** 0.5 + k2 ** 0.5 + k3
    return result

#Parrondo and Van den Broek
def f3(n):
    k1 = 1 / n
    k2 = 1 / (n ** 2) + (1 / n) * np.log(6 * m(2 * n) / delta)
    k2 = k2 ** 0.5
    result = k1 + k2
    return result

#Devroye
def f4(n):
    k1 = 1 / (n - 2)
    k2 = np.log(4 * m(n ** 2) / delta) / (2 * (n - 2)) + 1 / ((n - 2) ** 2)
    k2 = k2 ** 0.5
    result = k1 + k2
    return result

#产生点集
x = np.arange(100, 2000)

y1 = [f1(i) for i in x]
y2 = [f2(i) for i in x]
y3 = [f3(i) for i in x]
y4 = [f4(i) for i in x]

plt.plot(x, y1, label="Original VC-bound")
plt.plot(x, y2, label="Rademacher Penalty Bound")
plt.plot(x, y3, label="Parrondo and Van den Broek")
plt.plot(x, y4, label="Devroye")
plt.legend()
plt.show()
```
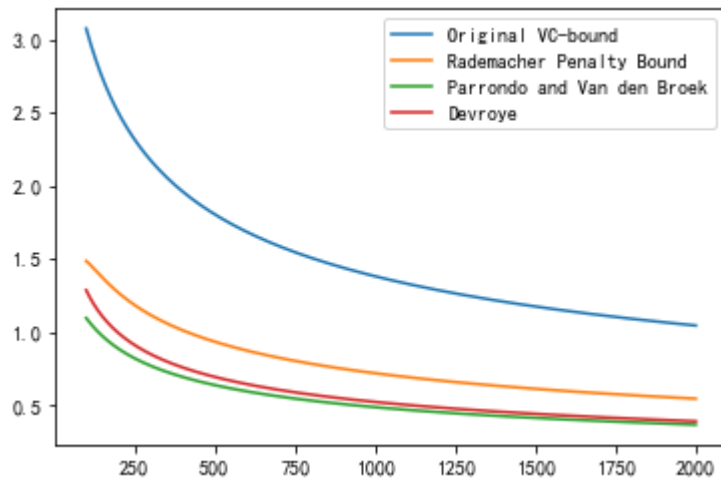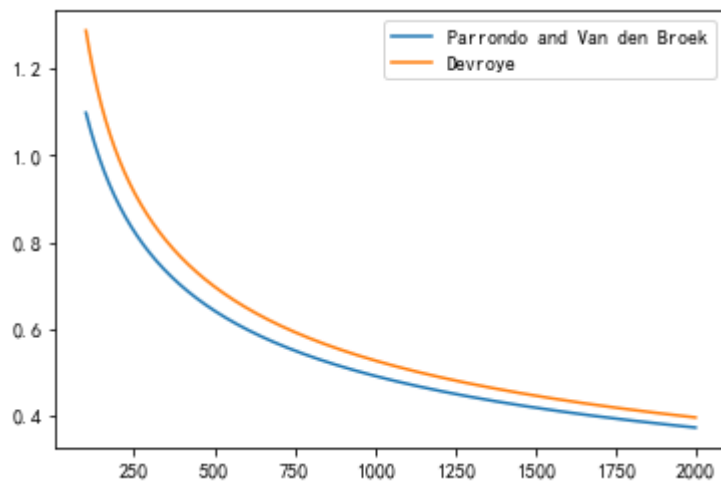
比较Parrondo and Van den Broek和Devroye

```
#比较y3, y4
plt.plot(x, y3, label="Parrondo and Van den Broek")
plt.plot(x, y4, label="Devroye")
plt.legend()
plt.show()
```


所以最好的上界为Parrondo and Van den Broek。

Problem 2.21 (Page 74)

Assume the following theorem to hold

Theorem

$$\mathbb{P}\left[\frac{E_{\text{out}}(g) - E_{\text{in}}(g)}{\sqrt{E_{\text{out}}(g)}} > \epsilon\right] \le c.\, m_{\mathcal{H}}(2N) \exp\left(-\frac{\epsilon^2 N}{4}\right)$$

where c is a constant that is a little bigger than 6.

This bound is useful because sometimes what we care about is not the absolute generalization error but instead a relative generalization error (one can imagine that a generalization error of 0.01 is more significant when $E_{\text{out}} = 0.01$ than when $E_{\text{out}} = 0.5$). Convert this to a generalization bound by showing that with probability at least $1 - \delta$,

$$E_{\text{out}}(g) \le E_{\text{in}}(g) + \frac{\xi}{2}\left[1 + \sqrt{1 + \frac{4E_{\text{in}}(g)}{\xi}}\right]$$

where $\xi = \frac{4}{N}\log\frac{c.m_{\mathcal{H}}(2N)}{\delta}$.

由题目中给出的不等式我们知道

$$\frac{E_{\text{out}}(g) - E_{\text{in}}(g)}{\sqrt{E_{\text{out}}(g)}} \le \epsilon 的概率大于等于 1 - c.m_{\mathcal{H}}(2N)\exp\left(-\frac{\epsilon^2 N}{4}\right)$$

现在要证明的结论中要求的概率为 $1 - \delta$，因此我们令 $\delta = c.m_{\mathcal{H}}(2N)\exp\left(-\frac{\epsilon^2 N}{4}\right)$，对此式化简可得

$$\ln\frac{\delta}{c.m_{\mathcal{H}}(2N)} = -\frac{\epsilon^2 N}{4}$$

$$-\frac{4}{N}\ln\frac{\delta}{c.m_{\mathcal{H}}(2N)} = \epsilon^2$$

我们令

$$\xi = \epsilon^2 = \frac{4}{N}\ln\frac{c.m_{\mathcal{H}}(2N)}{\delta}$$

因此我们现在有

$$\frac{E_{\text{out}}(g) - E_{\text{in}}(g)}{\sqrt{E_{\text{out}}(g)}} \le \epsilon 的概率大于等于 1 - \delta$$

$$\xi = \epsilon^2 = \frac{4}{N}\ln\frac{c.m_{\mathcal{H}}(2N)}{\delta}$$

$$\delta = c.m_{\mathcal{H}}(2N)\exp\left(-\frac{\epsilon^2 N}{4}\right)$$

为了更接近我们的结论，现在要对 $\frac{E_{\text{out}}(g) - E_{\text{in}}(g)}{\sqrt{E_{\text{out}}(g)}} \le \epsilon$ 处理

$$\frac{E_{\text{out}}(g) - E_{\text{in}}(g)}{\sqrt{E_{\text{out}}(g)}} \le \epsilon$$

$$E_{\text{out}}(g) - E_{\text{in}}(g) \le \epsilon\sqrt{E_{\text{out}}(g)}$$

配方可得

$$E_{\text{out}}(g) - \epsilon\sqrt{E_{\text{out}}(g)} \le E_{\text{in}}(g)$$

$$E_{\text{out}}(g) - \epsilon\sqrt{E_{\text{out}}(g)} + \frac{\epsilon^2}{4} \le E_{\text{in}}(g) + \frac{\epsilon^2}{4}$$

$$(\sqrt{E_{\text{out}}(g)} - \frac{\epsilon}{2})^2 \le E_{\text{in}}(g) + \frac{\epsilon^2}{4}$$

$$-\sqrt{E_{\text{in}}(g) + \frac{\epsilon^2}{4}} \le \sqrt{E_{\text{out}}(g)} - \frac{\epsilon}{2} \le \sqrt{E_{\text{in}}(g) + \frac{\epsilon^2}{4}}$$

$$\frac{\epsilon}{2} - \sqrt{E_{\text{in}}(g) + \frac{\epsilon^2}{4}} \le \sqrt{E_{\text{out}}(g)} \le \frac{\epsilon}{2} + \sqrt{E_{\text{in}}(g) + \frac{\epsilon^2}{4}}$$

注意到

$$\frac{\epsilon}{2} - \sqrt{E_{\text{in}}(g) + \frac{\epsilon^2}{4}} \le 0,$$

所以这部分没什么意义，只需考虑右边的不等式

$$\sqrt{E_{\text{out}}(g)} \le \frac{\epsilon}{2} + \sqrt{E_{\text{in}}(g) + \frac{\epsilon^2}{4}}$$

两边平方可得

$$E_{\text{out}}(g) \le \left(\frac{\epsilon}{2} + \sqrt{E_{\text{in}}(g) + \frac{\epsilon^2}{4}}\right)^2 = \frac{\epsilon^2}{4} + E_{\text{in}}(g) + \frac{\epsilon^2}{4} + \epsilon\left(\sqrt{E_{\text{in}}(g) + \frac{\epsilon^2}{4}}\right)$$

$$E_{\text{out}}(g) \le \frac{\epsilon^2}{2} + E_{\text{in}}(g) + \frac{\epsilon^2}{2}\sqrt{1 + \frac{4E_{\text{in}}(g)}{\epsilon^2}}$$

现在将 $\xi = \epsilon^2$ 带入可得

$$E_{\text{out}}(g) \le \frac{\xi}{2} + E_{\text{in}}(g) + \frac{\xi}{2}\sqrt{1 + \frac{4E_{\text{in}}(g)}{\xi}} = E_{\text{in}}(g) + \frac{\xi}{2}\left[1 + \sqrt{1 + \frac{4E_{\text{in}}(g)}{\xi}}\right]$$

因此结论成立。其实这题就是纯粹计算，没什么技巧性。

Problem 2.22 (Page 74)

When there is noise in the data , $E_{\text{out}}(g^{(\mathcal{D})}) = \mathbb{E}_{x,y}[(g^{(\mathcal{D})}(x) - y(x))^2]$, where $y(x) = f(x) + \epsilon$. If ϵ is a zero mean noise random variable with variance δ^2 , show that the bias variance decomposition becomes

$$\mathbb{E}_{\mathcal{D}}[E_{\text{out}}(g^{(\mathcal{D})})] = \delta^2 + \text{bias} + \text{var}$$

方差和偏差的公式可以查看课本63页，这里我们直接计算。

$$\mathbb{E}_{\mathcal{D}}\big[E_{\text{out}}\big(g^{(\mathcal{D})}\big)\big] = \mathbb{E}_{\mathcal{D}}\left[\mathbb{E}_{x,y}[(g^{(\mathcal{D})}(x) - y(x))^2]\right]$$

$$= \mathbb{E}_{\mathcal{D}}\left[\mathbb{E}_x[(g^{(\mathcal{D})}(x) - f(x) - \epsilon)^2]\right]$$

$$= \mathbb{E}_{\mathcal{D}}\left[\mathbb{E}_x[(g^{(\mathcal{D})}(x) - f(x))^2] + \mathbb{E}_x(\epsilon^2) - 2\mathbb{E}_x[\epsilon(g^{(D)}(x) - f(x))]\right]$$

我们分别看下上述三项，首先看第一项

$$\mathbb{E}_{\mathcal{D}}\left[\mathbb{E}_x[(g^{(D)}(x) - f(x))^2]\right] = \mathbb{E}_x\left[\mathbb{E}_{\mathcal{D}}[(g^{(D)}(x) - f(x))^2]\right]$$
$$= \text{bias} + \text{var}(这一步可以参考课本63页)$$

再来看下第二项，注意ϵ数学期望为0，方差为δ^2

$$\mathbb{E}_{\mathcal{D}}\left[\mathbb{E}_x(\epsilon^2)\right] = \mathbb{E}_{\mathcal{D}}\left[\mathbb{E}_x(\epsilon^2)\right] = \mathbb{E}_{\mathcal{D}}(\delta^2) = \delta^2$$

最后来看下第三项，注意ϵ数学期望为0

$$\mathbb{E}_{\mathcal{D}}\left[-2\mathbb{E}_x[\epsilon(g^{(\mathcal{D})}(x) - f(x))]\right] = \mathbb{E}_{\mathcal{D}}\left[-2\mathbb{E}_x(\epsilon)\mathbb{E}_x[g^{(\mathcal{D})}(x) - f(x)]\right] = 0$$

集合上述三点可得

$$\mathbb{E}_{\mathcal{D}}\big[E_{\text{out}}\big(g^{(\mathcal{D})}\big)\big] = \delta^2 + \text{bias} + \text{var}$$

Problem 2.23 (Page 74)

Consider the learning problem in Example 2.8, where the input space is $\mathcal{X} = [-1, +1]$, the target function is $f(x) = \sin(\pi x)$, and the input probability distribution is uniform on $\mathcal{X}$. Assume that the training set $\mathcal{D}$ has only two data points (picked independently), and that the learning algorithm picks the hypothesis that minimizes the in sample mean squared error. In this problem, we will dig deeper into this case.

For each of the following learning models, find (analytically or numerically) (i) the best hypothesis that approximates f in the mean squared error sense (assume that f is known for this part) , (ii) the expected value (with respect to $\mathcal{D}$) of the hypothesis that the learning algorithm produces, and (iii) the expected out of sample error and its bias and var components.

(a) The learning model consists of all hypotheses of the form $h(x) = ax + b$(if you need to deal with the infinitesimal probability case of two identical data points, choose the hypothesis tangential to f).

(b) The learning model consists of all hypotheses of the form $h(x) = ax$. This case was not covered in Example 2.8 .

(c) The learning model consists of all hypotheses of the form $h(x) = b$.

这题我的理解是这样的:

(i)先根据两个样本求一下三种模型的在平方误差下的最优解。

(ii)学习算法产生的假设的数学期望，也就是课本63页中的$E_{\text{out}}(g^{(D)}) = \mathbb{E}_x\left[(g^{(D)}(x) - f(x))^2\right]$。这部分要用编程模拟，直接求的话感觉不大可行。

(iii)(ii)已经求出$E_{\text{out}}(g^{(D)})$,只要求var,bias

下面分别做一下，这里统一假设两个样本点为$(x_1, y_1), (x_2, y_2), y_i = \sin(\pi x_i)$

(a)

(i)平方误差$S_1 = (ax_1 + b - y_1)^2 + (ax_2 + b - y_2)^2$，这个实际上就是一元线性回归的特殊情形，这里直接套公式了

$$使得 \sum_{i=1}^{n}(ax_i + b - y_i)^2 最小的 a, b 满足$$

$$\tilde{a} = \frac{l_{xy}}{l_{xx}}$$

$$\tilde{b} = \overline{y} - \tilde{a}\overline{x}$$

$$其中 \overline{y} = \frac{\sum_{i=1}^{n} y_i}{n}, \overline{x} = \frac{\sum_{i=1}^{n} x_i}{n}$$

$$l_{xx} = \sum_{i=1}^{n}(x_i - \overline{x})^2, l_{xy} = \sum_{i=1}^{n}(x_i - \overline{x})(y_i - \overline{y})$$

不过由于这题是两个点，所以最好的直线实际上就是过这两个点的直线，因此

$$y_1 = ax_1 + b$$
$$y_2 = ax_2 + b$$
$$a = \frac{y_2 - y_1}{x_2 - x_1}$$
$$b = y_1 - ax_1 = y_1 - \frac{y_2 - y_1}{x_2 - x_1}x_1 = \frac{y_1 x_2 - y_2 x_1}{x_2 - x_1}$$

后续部分也会采用这个公式

(ii)这部分采用编程模拟,我们先画个图看一下,第一步先把点构造出来

```python
# -*- coding: utf-8 -*-
"""
Created on Tue Feb 19 16:15:56 2019

@author: qinzhen
"""

import numpy as np
import matplotlib.pyplot as plt
import math
from scipy import integrate
plt.rcParams['font.sans-serif']=['SimHei'] #用来正常显示中文标签
plt.rcParams['axes.unicode_minus']=False #用来正常显示负号

def f(x):
    return math.sin(math.pi * x)


#### Part 1: y=kx+b
```

```
#已知(x1,y1),(x2,y2)过直线y=kx+b，此函数返回k,b
def treat(x1,x2):
    y1 = f(x1)
    y2 = f(x2)
    k = (y2 - y1) / (x2 - x1)
    b = (y1 * x2 - y2 * x1) / (x2 - x1)
    return k, b

#产生-1到1的20个点，对每个点，求每个点与其后面一个点连线的直线
x = np.arange(-1, 1, 0.05)

#求出对应的k,b
u = []
for i in range(x.size-1):
    u.append(treat(x[i], x[i+1]))

#求出直线
X1 = np.array([-1.0, 1.0])
Y1 = []
for i in u:
    temp = X1 * i[0] + i[1]
    Y1.append(temp)

#y=sin(pix)
X2 = np.arange(-1,1,0.01)
Y2 = np.sin(2 * np.pi * X2)
```
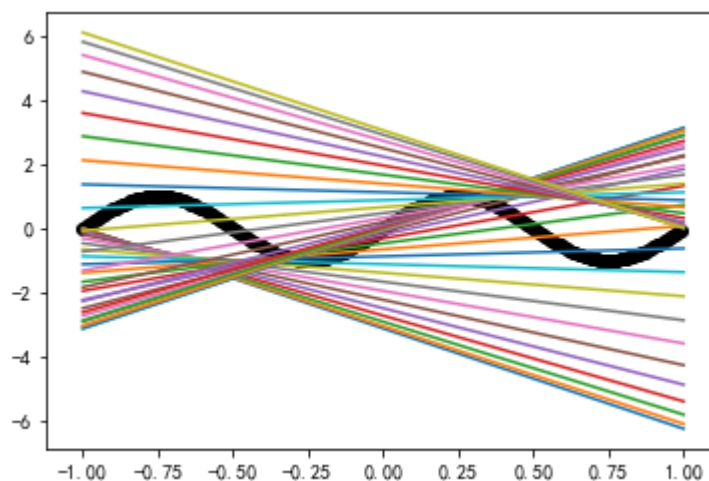
作图

```
#作图
for i in Y1:
    plt.plot(X1, i)
plt.scatter(X2, Y2, c='BLACK')
plt.show()
```


计算E_{out}的公式如下

$$E_{out} = \mathbb{E}_{x,x_1,x_2} \left[\sin(\pi x) - \frac{y_2 - y_1}{x_2 - x_1} x - \frac{y_1 x_2 - y_2 x_1}{x_2 - x_1} \right]^2$$

$$= \frac{1}{8} \int_{-1}^{1} \int_{-1}^{1} \int_{-1}^{1} \left[\sin(\pi x) - \frac{\sin(x_2) - \sin(x_1)}{x_2 - x_1} x - \frac{\sin(x_1) x_2 - \sin(x_2) x_1}{x_2 - x_1} \right]^2 dx_1 dx_2 dx$$

这个式子计算量太大，直接编程进行计算。

```
#定义误差函数
def s1(x, x1, x2):
    #此处用numpy计算会产生错误，暂时没找到原因
    y1 = f(x1)
    y2 = f(x2)
    #为了防止分母为0
    try:
        k = (y2 - y1) / (x2 - x1)
        b = (y1 * x2 - y2 * x1) / (x2 - x1)
    except:
        k = (y2 - y1) / (x2 - x1 + 10 ** (-10))
        b = (y1 * x2 - y2 * x1) / (x2 - x1 + 10 ** (-10))
    y = f(x)
    y0 = k * x + b
    return 1 / 8 * (y - y0) ** 2

print(integrate.tplquad(s1, -1, 1, lambda x: -1, lambda x: 1,lambda x, y: -1, lambda x, y:
1))
```

```
(1.8829992352626945, 1.409809326862331e-10)
```

这里采用了scipy库来计算积分，我们看下书上65页的结果,$\text{bias} = 0.21, \text{var} = 1.69$，加起来为1.90，还是非常接近的，这样这部分就完成了。

(iii)查看课本63页，要计算bias与var，我们需要计算出$\overline{g}(x)$，这里我们用模拟的方式求出数值解。

```
#计算bias
#模拟求出系数
def simulation(n):
    k = 0
    b = 0
    #模拟n次
    for i in range(n):
        #产生-1,1之间均匀分布随机数
        x1 = np.random.uniform(-1, 1)
        x2 = np.random.uniform(-1, 1)
        k1, b1 = treat(x1, x2)
        k += k1
        b += b1
    return k / n, b / n

#比较下10000次以上的结果
n = range(10 ** 4, 10 ** 5+1, 10 ** 4)
result =[]
```

```
for i in n:
    temp = simulation(i)
    print(temp)
    result.append(temp)
```

```
(0.7553055579826999, 0.003415753884397186)
(0.7723534275990537, -0.00044689199911268575)
(0.7759823827175366, -0.0011020087898786115)
(0.7681754907999855, 0.001278068061675665)
(0.7803466811748968, 0.0021760164599298427)
(0.7850619662155226, -0.009981846491328482)
(0.7687570588654568, 0.0023050518096932714)
(0.7817829859625752, -0.0021574306952086794)
(0.7830268807195201, -0.0037297425276534724)
(0.7676139340544786, -0.002281884820206641)
```

可以看到，模拟10000次以上得出的结果都比较接近，这里我们取第一组数据，然后计算bias，计算公式如下

$$\text{bias} = \frac{1}{2} \int_{-1}^{1} (\overline{g}(x) - f(x))^2 dx$$

```
#取第一组计算bias
a1, b1 = result[0]

def bias(x):
    y1 = a1 * x + b1
    y2 = f(x)
    return (y1 - y2) ** 2 / 2

print(integrate.quad(bias,-1,1))
```

```
(0.20933137695686926, 5.534401525802478e-10)
```

四舍五入之后就是书上65页的0.21。实际上这时候var已经不需要计算了，只要拿(ii)的结果减去这里计算的bias即可，但是为了严谨起见，这里依旧把剩余部分模拟完。

计算公式如下

$$\text{var} = \frac{1}{8} \int_{-1}^{1} \int_{-1}^{1} \int_{-1}^{1} \left[\overline{g}(x) - \frac{\sin(x_2) - \sin(x_1)}{x_2 - x_1} x - \frac{\sin(x_1)x_2 - \sin(x_2)x_1}{x_2 - x_1} \right]^2 dx_1 dx_2 dx$$

```
#计算var
def var(x, x1, x2):
    y1 = f(x1)
    y2 = f(x2)
    try:
        k = (y2 - y1) / (x2 - x1)
        b = (y1 * x2 - y2 * x1) / (x2 - x1)
    except:
        k = (y2 - y1) / (x2 - x1 + 10 ** (-10))
```

```
        b = (y1 * x2 - y2 * x1) / (x2 - x1 + 10 ** (-10))
    #之前计算的平均洗漱
    yavg= a1 * x + b1
    #真实值
    yrea = k * x + b
    return 1 / 8 * (yavg - yrea) ** 2

print(integrate.tplquad(var, -1, 1, lambda x: -1, lambda x: 1,lambda x, y: -1, lambda x,
y: 1))
```

```
(1.676435840266893, 2.313869161255777e-12)
```

这里的值和书上比差了一些，推测是$\overline{g}(x)$选择的问题。

(b)

(i)现在误差函数为$S_1 = (ax_1 - y_1)^2 + (ax_2 - y_2)^2$，我们来对这个式子做简单的处理

$$S_1 = (ax_1 - y_1)^2 + (ax_2 - y_2)^2$$
$$= (x_1^2 + x_2^2)a^2 - 2(x_1 y_1 + x_2 y_2)a + (y_1^2 + y_2^2)$$

由二次函数的性质，我们知道当$a = \frac{x_1 y_1 + x_2 y_2}{x_1^2 + x_2^2}$时，$S_1$最小

(ii)和上一题一样的处理方法。

```
#### Part 2: y=kx
#定义误差函数
def s1_b(x, x1, x2):
    y1 = f(x1)
    y2 = f(x2)
    #为了防止分母为0
    try:
        a = (x1 * y1 + x2 * y2) / (x1 ** 2 + x2 ** 2)
    except:
        a = (x1 * y1 + x2 * y2) / (x1 ** 2 + x2 ** 2 + 10 ** (-6))
    y = f(x)
    y0 = a*x
    return 1 / 8 * (y - y0) ** 2

print(integrate.tplquad(s1_b, -1, 1, lambda x: -1, lambda x: 1,lambda x, y: -1, lambda x,
y: 1))
```

```
(0.507219461635839, 1.3595152276862434e-08)
```

(iii)依旧和上一题同样的处理方法

```
#bias
#模拟求出系数
def simulation_b(n):
    a = 0
    #模拟n次
```

```python
    for i in range(n):
        #产生-1,1之间均匀分布随机数
        x1 = np.random.uniform(-1, 1)
        x2 = np.random.uniform(-1, 1)
        y1 = f(x1)
        y2 = f(x2)
        a1 = (x1 * y1 + x2 * y2) / (x1 ** 2 + x2 ** 2)
        a += a1
    return a / n

#比较下10000次以上的结果
n = range(10 ** 4, 10 ** 5+1, 10 ** 4)
result =[]
for i in n:
    temp = simulation_b(i)
    print(temp)
    result.append(temp)
```

```
1.4242798331604158
1.425690405949237
1.424071907712383
1.43219359628694
1.4247275146654388
1.4260006827489529
1.4299107366837769
1.4275041630558862
1.429846774549229
1.4254709994266355
```

可以看到，模拟10000次以上得出的结果都比较接近，这里我们取第一组数据来计算bias：

```python
#取第一组计算bias
a1 = result[0]
#bias
def bias_b(x):
    y1 = a1 * x
    y2 = f(x)
    return (y1 - y2) ** 2 / 2

print(integrate.quad(bias_b, -1, 1))
```

```
(0.26946631120816666, 3.818684931252901e-10)
```

计算var

```
#var
def var_b(x, x1, x2):
    y1 = f(x1)
    y2 = f(x2)
    try:
        a = (x1 * y1 + x2 * y2) / (x1 ** 2 + x2 ** 2)
    except:
        a = (x1 * y1 + x2 * y2) / (x1 ** 2 + x2 ** 2 + 10 ** (-6))
    yavg = a1 * x
    yrea = a * x
    return 1 / 8 * (yavg - yrea) ** 2

print(integrate.tplquad(var_b, -1, 1, lambda x: -1, lambda x: 1,lambda x, y: -1, lambda
x, y: 1))
```

```
(0.2365806067772056, 1.4880304077628016e-08)
```

可看看到 $var + bias$ 和(ii)中求出的数值几乎相等，再一次验证了课本上的结论。

(c)

(i)现在误差函数为 $S_1 = (b - y_1)^2 + (b - y_2)^2$，我们来对这个式子做简单的处理

$$S_1 = 2b^2 - 2(y_1 + y_2)b + y_1^2 + y_2^2$$

由二次函数的性质，我们知道当 $b = \frac{y_1 + y_2}{2}$ 时，S_1 最小

(ii)和上一题一样的处理方法。

```
#### Part 3: y=b
#定义误差函数
def s1_c(x, x1, x2):
    y1 = f(x1)
    y2 = f(x2)
    b = (y1 + y2) / 2
    y = f(x)
    y1 = b
    return 1 / 8 * (y - y1) ** 2
print(integrate.tplquad(s1_c, -1, 1, lambda x: -1, lambda x: 1,lambda x, y: -1, lambda
x, y: 1))
```

```
(0.75, 9.174802559227526e-11)
```

(iii)依旧和上一题同样的处理方法

```
#bias
#模拟求出系数
def simulation_c(n):
    b = 0
    #模拟n次
    for i in range(n):
```

```
        #产生-1,1之间均匀分布随机数
        x1 = np.random.uniform(-1, 1)
        x2 = np.random.uniform(-1, 1)
        y1 = f(x1)
        y2 = f(x2)
        b1 = (y1 + y2) / 2
        b += b1
    return(b / n)

#比较下10000次以上的结果
n = range(10 ** 4, 10 ** 5+1, 10 ** 4)
result =[]
for i in n:
    temp = simulation_c(i)
    print(temp)
    result.append(temp)
```

```
-0.00013963559105002233
-0.00035921811407854556
0.002640937676081866
-0.002458642684280539
0.0011231675022783125
-0.0010286024743622996
0.00409087821365123
0.0008138348591112101
0.0011649215739660676
-0.002208601419948738
```

依旧选择第一组数据来计算bias。

```
#取第一组计算bias
b1 = result[0]
def bias_c(x):
    y1 = b1
    y2 = f(x)
    return (y1 - y2) ** 2 / 2

print(integrate.quad(bias_c, -1, 1))
```

```
(0.500000194980984, 3.669920676267717e-10)
```

计算var

```
#var
def var_c(x, x1, x2):
    y1 = f(x1)
    y2 = f(x2)
    b = (y1 + y2)/2
    yavg = b1
    yrea = b
    return 1 / 8 * (yavg - yrea) ** 2

print(integrate.tplquad(var_b, -1, 1, lambda x: -1, lambda x: 1,lambda x, y: -1, lambda
x, y: 1))
```

```
(0.2365806067772056, 1.4880304077628016e-08)
```

和书上基本一致。

最后总结一下，这题设计的目的是让我们从实验中验证bias,var的等式，我一开始这题完全没有头绪，不过后来做完之后对这部分理解更加深刻了。

Problem 2.24 (Page 75)

Consider a simplified learning scenario. Assume that the input dimension is one. Assume that the input variable x is uniformly distributed in the interval $[-1, 1]$. The data set consists of 2 points $\{x_1, x_2\}$ and assume that the target function is $f(x) = x^2$. Thus, the full data set is $\mathcal{D} = \{(x_1, x_1^2), (x_2, x_2^2)\}$. The learning algorithm returns the line fitting these two points as g ($\mathcal{H}$ consists of functions of the frm $h(x) = ax + b$). We are interested in the test performance (E_{out}) of our learning system with respect to the squared error measure, the bias and the var.

(a) Give the analytic expression fr the average function $\overline{g}(x)$.

(b) Describe an experiment that you could run to determine (numerically) $\overline{g}(x)$, E_{out}, bias, and var.

(c) Run your experiment and report the results. Compare Bout with bias+var. Provide a plot of your $\overline{g}(x)$ and $f(x)$ (on the same plot) .

(d) Compute analytically what E_{out}. bias and var should be.

这题其实和上一题的(a)基本一样，只不过这里的$f(x) = x^2$

(a)对于固定的两个点$D = \{(x_1, x_1^2), (x_2, x_2^2)\}$，训练误差最小的直线为过这两个点的直线，假设直线为
$g(x) = ax + b$，因此

$$x_2^2 = ax_2 + b$$
$$x_1^2 = ax_1 + b$$
$$a = (x_1 + x_2)$$
$$b = -x_1 x_2$$

那么$g(x) = (x_1 + x_2)x - x_1 x_2$

假设有$\mathcal{D}_1, \ldots, \mathcal{D}_K$这$K$组数据，每个$\mathcal{D}_i$有两个点$x_{i1}, x_{i2}$，那么$\mathcal{D}_i$训练出来的$g_i(x) = (x_{i1} + x_{i2})x - x_{i1}x_{i2}$，回到课本63页$\overline{g}(x)$的定义可知

$$\overline{g}(x) = \frac{1}{K} \sum_{i=1}^{K} g_i(x)$$

(b)这部分就是采用数值解，和上一题采用的方法同理。

$\overline{g}(x)$我采取的方法是在$[-1, 1]$上随机取两个点(均匀分布)，然后计算对应的$g_i(x)$，重复这实验多次，利用
$\overline{g}(x) = \frac{1}{K} \sum_{i=1}^{K} g_i(x)$计算。

再来看下$E_{\text{out}}(g^{(D)})$,注意书上的公式

$$E_{\text{out}}(g^{(D)}) = \mathbb{E}_x \left[(g^{(D)}(x) - f(x))^2 \right]$$

我们已经知道对于$\mathcal{D} = \{(x_1, x_1^2), (x_2, x_2^2)\}$，$g(x) = (x_1 + x_2)x - x_1 x_2$，带入上式计算关于$x_1, x_2, x$的三重积分即可。

bias可以参考课本63页的公式

$$\text{bias}(x) = (\overline{g}(x) - f(x))^2, \text{bias} = \mathbb{E}_x[\text{bias}(x)]$$

之前已经计算出了$\overline{g}(x)$，带入上式求积分即可。

var可以参考课本64页的公式

$$\text{var}(x) = \mathbb{E}_D[(g^{(D)}(x) - \overline{g}(x))^2], \text{var} = \mathbb{E}_x[\text{var}(x)]$$

我们已经知道对于$D = \{(x_1, x_1^2), (x_2, x_2^2)\}$，$g(x) = (x_1 + x_2)x - x_1 x_2$，带入上式计算关于$x_1, x_2, x$的三重积分即可。

(c)下面通过模拟计算一下，先计算$\overline{g}(x)$。

```python
# -*- coding: utf-8 -*-
"""
Created on Tue Feb 19 17:23:17 2019

@author: qinzhen
"""

import numpy as np
import matplotlib.pyplot as plt
from scipy import integrate
plt.rcParams['font.sans-serif']=['SimHei'] #用来正常显示中文标签
plt.rcParams['axes.unicode_minus']=False #用来正常显示负号

def f(x):
    return x ** 2

#模拟求出系数
def simulation(n):
    a = 0
    b = 0
    #模拟n次
    for i in range(n):
        #产生-1,1之间均匀分布随机数
```

```
            x1 = np.random.uniform(-1, 1)
            x2 = np.random.uniform(-1, 1)
            a1 = (x1 + x2)
            b1 = - x1 * x2
            a += a1
            b += b1
        return a / n, b / n

#比较下10000次以上的结果
n = range(10 ** 4, 10 ** 5+1, 10 ** 4)
result =[]
for i in n:
    temp = simulation(i)
    print(temp)
    result.append(temp)
```

```
(-0.007259431812949206, -0.006505637449944645)
(0.01054551337977565, 0.0007205658774392578)
(0.002200261748429362, 0.00039501424061546197)
(-0.0036746772681782273, 3.6611066267928454e-05)
(0.0005707483200652478, 0.0012540525909730687)
(-0.0035353924654228748, -0.0027067893024173384)
(0.002037735570941735, 0.0008453013796700488)
(0.0003824037342788735, -0.000651045180896328 2)
(-0.002044260427503321, 0.0010197292343565425)
(-0.002699490891714745, 0.0017938900684602123)
```
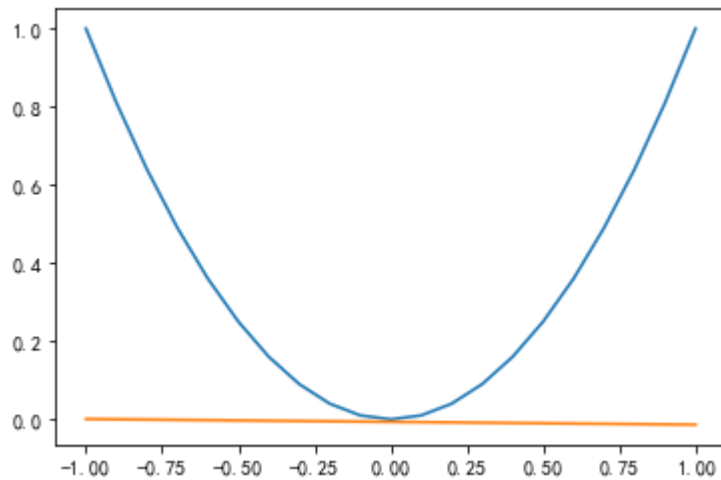
可以看到后面的结果已经比较接近了，我们这里取第一组数据，然后作图。

```
#选择第一组数据并作图
a, b = result[0]
x1 = np.arange(-1, 1.1, 0.1)
y1 = [f(i) for i in x1]
x2 = np.array([-1, 1])
y2 = a * x2 + b

plt.plot(x1,y1)
plt.plot(x2,y2)
plt.show()
```

计算 $E_{\text{out}}(g^{(D)})$

```
#计算积分
#定义误差函数
def s1(x, x1, x2):
    a = (x1 + x2)
    b = - x1 * x2
    y = f(x)
    y1 = a * x + b
    return 1 / 8 * (y - y1) ** 2


print(integrate.tplquad(s1, -1, 1, lambda x: -1, lambda x: 1,lambda x, y: -1, lambda x, y:
1))
```

```
(0.5333333333333333, 8.785771717918444e-15)
```

bias

```
#bias
def bias(x):
    y1 = a * x + b
    y2 = f(x)
    return (y1 - y2) ** 2 / 2


print(integrate.quad(bias, -1, 1))
```

```
(0.2043969814020088, 2.2692623491639006e-15)
```

var

```
#var
def var(x, x1, x2):
    yavg = a * x + b
    a1 = (x1 + x2)
    b1 = -x1 * x2
    yrea = a1 * x + b1
    return 1 / 8 * (yavg - yrea) ** 2

print(integrate.tplquad(var, -1, 1, lambda x: -1, lambda x: 1,lambda x, y: -1, lambda x,
y: 1))
```

```
(0.3333932231020458, 6.3874396220908005e-15)
```

可以看到$E_{\text{out}}(g^{(D)}) = \text{bias} + \text{var}$

(d)计算$E_{\text{out}}, \text{bias}, \text{var}$的解析解，只要分别计算积分即可。

先看E_{out}

$$
\begin{aligned}
E_{\text{out}} &= \mathbb{E}_x\left[((x_1+x_2)x - x_1x_2 - x^2)^2\right] \\
&= \frac{1}{8}\int_{-1}^1\int_{-1}^1\int_{-1}^1\left[(x-x_2)x_1 + x_2x - x^2\right]^2 dx_1 dx_2 dx \\
&= \frac{1}{8}\int_{-1}^1\int_{-1}^1\int_{-1}^1\left[(x-x_2)^2x_1^2 + 2(x-x_2)x_1(x_2x-x^2) + (x_2x-x^2)^2\right]dx_1 dx_2 dx \\
&= \frac{1}{8}\int_{-1}^1\int_{-1}^1\left[\frac{2(x-x_2)^2}{3} + 2(x_2x-x^2)^2\right]dx_2 dx \\
&= \frac{1}{8}\int_{-1}^1\int_{-1}^1\left[\frac{2}{3}(x^2-2xx_2+x_2^2) + 2x^2(x_2^2-2xx_2+x^2)\right]dx_2 dx \\
&= \frac{1}{8}\int_{-1}^1(\frac{2}{3}+2x^2)(\frac{2}{3}+2x^2)dx \\
&= \frac{1}{8}\int_{-1}^1(\frac{4}{9}+4x^4+\frac{8}{3}x^2)dx \\
&= \frac{1}{8}(\frac{8}{9}+\frac{8}{5}+\frac{1}{6}9) \\
&= \frac{1}{3}+\frac{1}{5} \\
&\approx 0.53
\end{aligned}
$$

可以看到和我们之前模拟的数值解基本一致。

求bias, var之前，先得求$\overline{g}(x)$，设$\overline{g}(x) = ax + b$。我们知道对于$D = \{(x_1, x_1^2), (x_2, x_2^2)\}$，$g(x) = (x_1+x_2)x - x_1x_2$，那么由$\overline{g}(x)$的定义我们知道

$$a = \mathbb{E}(x_1+x_2), b = \mathbb{E}(-x_1x_2)$$

那么

$$a = \mathbb{E}(x_1 + x_2)$$
$$= \frac{1}{4} \int_{-1}^{1} \int_{-1}^{1} (x_1 + x_2) dx_1 dx_2$$
$$= \frac{1}{4} \int_{-1}^{1} 2x_2 dx_2$$
$$= 0$$
$$b = \mathbb{E}(-x_1 x_2)$$
$$= \int_{-1}^{1} \int_{-1}^{1} (-x_1 x_2) dx_1 dx_2$$
$$= - \int_{-1}^{1} x_1 dx_1 \int_{-1}^{1} x_2 dx_2$$
$$= 0$$

因此$\overline{g}(x) = 0$，接下来求bias

$$\text{bias} = \mathbb{E}_x [(\overline{g}(x) - f(x))^2]$$
$$= \int_{-1}^{1} x^4 dx$$
$$= 0.2$$

和我们之前模拟的结果一致。

最后求var

$$\text{var} = \mathbb{E}_x [\mathbb{E}_{\mathcal{D}} [(g^{(D)}(x) - \overline{g}(x))^2]]$$
$$= \frac{1}{8} \int_{-1}^{1} \int_{-1}^{1} \int_{-1}^{1} \Big[(x_1 + x_2)x - x_1 x_2 \Big]^2 dx dx_1 dx_2$$
$$= \frac{1}{8} \int_{-1}^{1} \int_{-1}^{1} \int_{-1}^{1} \Big[(x_1 + x_2)^2 x^2 + x_1^2 x_2^2 - 2x_1 x_2 (x_1 + x_2)x \Big] dx dx_1 dx_2$$
$$= \frac{1}{8} \int_{-1}^{1} \int_{-1}^{1} \Big[\frac{2(x_1^2 + x_2^2 + 2x_1 x_2)}{3} + 2x_1^2 x_2^2 \Big] dx_1 dx_2$$
$$= \frac{1}{8} \int_{-1}^{1} \Big[\frac{4}{9} + \frac{4}{3} x_2^2 + \frac{4}{3} x_2^2 \Big] dx_2$$
$$= \frac{1}{8} (\frac{8}{9} + \frac{16}{9})$$
$$= \frac{1}{3}$$
$$\approx 0.33$$

和我们之前模拟的结果也一致。