从数据拟合到积分变换

纪浩正
地球与空间科学系，南方科技大学
jihz2023@mail.sustech.edu.cn

June 14, 2025

Abstract

本文系统探讨了数据拟合与积分变换的理论框架及方法应用。研究以函数空间正交投影为核心，阐释了Gram-Schmidt正交化在构建基函数中的作用，通过多项式、傅里叶基、指数函数等典型基的对比实验，揭示了基函数选择对拟合效果的影响机制。进一步推导了傅里叶变换的正交投影本质，拓展至拉普拉斯变换的积分表达式，到各类积分变换，建立了微分方程求解的新视角,并且在最后简要介绍了，向量内积在统计学中的应用。

Keywords: 数据拟合；Gram-Schmidt正交化；傅里叶变换；拉普拉斯变换；积分变换；函数空间投影

Contents

1 拟合

基本定义：
拟合指的是寻找一个函数 $f(x)$ 来近似描述给定的数据集 $\{x_i, y_i\}_{i\in\mathbf{N}}$，使得每个 $f(x_i)$ 都尽可能接近对应的 y_i。衡量拟合效果的一个常用指标是残差平方和

$$\sum_{i=1}^{N} |f(x_i) - y_i|^2.$$

类似地，若要比较两个函数 $f(x)$ 与 $g(x)$ 之间的差异，也可以采用类似的标准，其连续形式可写为

$$\int_0^\infty |f(x) - g(x)|^2 \, dx.$$

在函数向量空间中，我们定义内积为

$$\langle f, g\rangle = \int_0^\infty f(x)g(x)\,dx,$$

进而定义范数

$$\|f(x)\| = \sqrt{\int_0^\infty f^2(x)\,dx},$$

从而 $\|f-g\|$ 自然地代表了 f 与 g 之间的距离。

如何进行拟合

在高中阶段，我们学习了利用最小二乘法拟合数据点，但该方法仅适用于线性关系，而自然界中的线性现象往往较少。

最小二乘法利用形式如

$$y = \hat{a}\cdot 1 + \hat{b}\cdot x,$$

的方程，将 $\{1, x\}$ 作为基来拟合数据。随后，通过泰勒展开，我们开始使用 $\{1, x, x^2, x^3, \dots\}$ 作为基函数来逼近函数。进一步，我们又引入了傅里叶级数

$$\{1, \cos x, \cos 2x, \cos 3x, \dots, \sin x, \sin 2x, \sin 3x, \dots\}$$

作为拟合工具。

以 $f(x) = 3x^2 + 2x - 1$ 为例，若采用最小二乘法，只能在有限区间内捕捉其大致变化趋势；而若采用泰勒级数，仅用前三项便可完全表达 $f(x)$ 所蕴含的信息，实现反演与预测时都能获得精确结果。反之，若采用傅里叶级数，在给定闭区间内

$$\sum_{i=-n}^{n} C_i e^{j\omega_i x}$$

可以绝对收敛于 $f(x) = 3x^2 + 2x - 1$；但当超出该区间时，由于傅里叶级数由正弦与余弦函数构成，表现出明显周期性，而 $f(x)$ 则无此周期性，因此拟合将失效。这正是所选拟合基不同所带来的差异。虽然我们无法总是选出完美的基函数进行预测，但在给定范围内使用适当的基函数可以有效反映数据或函数的变化。

2 Gram–Schmidt 正交化

函数拟合方法

我们利用基函数集 $\{v_1(x), v_2(x), v_3(x), \dots, v_n(x)\}$ 来拟合目标函数。该集合张成的函数空间 $\mathbf{V}$ 中的任一函数都可表示为

$$u(x) = \sum_{i=1}^{n} C_i v_i(x).$$

我们的目标是确定一个 $u(x)$，使得残差平方和

$$\sum_{i=1}^{N} |f(x_i) - u(x_i)|^2$$

最小。也可引入权函数 $r(x)$ 考虑不同数据点的重要性，此时残差平方和变为

$$\sum_{i=1}^{N} |f(x_i) - u(x_i)|^2 r(x)$$

或其连续形式

$$\int_a^b |f(x) - u(x)|^2 r(x)\, dx.$$

为简便起见，我们通常取 $r(x) = 1$。因此，表达式

$$\int_a^b |f(x) - u(x)|^2\, dx$$

反映了 f 与 u 在整体空间 $\mathbf{W}$ 中的距离。要使

$$\int_0^\infty |f(x) - u(x)|^2\, dx$$

最小，则 u 必须为 f 在空间 $\mathbf{V}$ 上的正交投影$P_v f$。

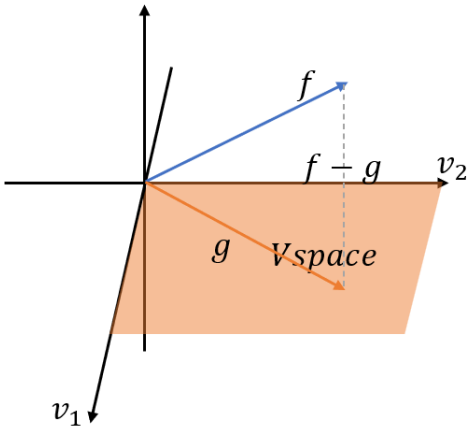

Figure 1: 正交投影示意图

向量化函数

在数值计算中，函数 f 的离散化处理通常转化为向量形式: $\begin{bmatrix} f(x_1) \\ f(x_2) \\ f(x_3) \\ \vdots \\ f(x_n) \end{bmatrix}$ 这种向量化处理方式使得函数 f 可以视作一个向量。通过将其投影到由向量化函数 v_i 构成的子空间 $W = \text{span}\{v_i\}$ 中，获得的$P_v f$就是我们想要寻找的拟合函数。

构造正交归一基

在具有内积定义的向量空间中，我们可采用 Gram–Schmidt 正交化过程将基 $\{v_1(x), v_2(x), v_3(x), \ldots, v_n(x)\}$

转化为正交归一基 $\{u_1(x), u_2(x), u_3(x), \ldots, u_n(x)\}$：

$$\textbf{Step 1:}\quad u_1(x) = \frac{v_1(x)}{\sqrt{\langle v_1, v_1\rangle}} = \frac{v_1(x)}{\sqrt{\int_a^b v_1(x)^2\,dx}},$$

$$\textbf{Step 2:}\quad u_2'(x) = v_2(x) - \langle v_2, u_1\rangle\, u_1(x), \quad u_2(x) = \frac{u_2'(x)}{\sqrt{\langle u_2', u_2'\rangle}},$$

$$\vdots$$

$$\textbf{Step n:}\quad u_n'(x) = v_n(x) - \sum_{i=1}^{n-1}\langle v_n, u_i\rangle\, u_i(x), \quad u_n(x) = \frac{u_n'(x)}{\sqrt{\langle u_n', u_n'\rangle}}.$$

不难验证，$\langle u_i, u_j\rangle = \delta_{ij}$。例如：

$$\textbf{例子:}\quad \langle u, v\rangle = \int_{-1}^{1} u(x)v(x)\,dx,$$

$$\{1, x, x^2\} \Rightarrow \left\{\frac{1}{\sqrt{2}}, \frac{\sqrt{3}\,x}{\sqrt{2}}, \frac{3\sqrt{10}}{4}x^2 - \frac{\sqrt{10}}{4}\right\},$$

$$\{1, \cos\pi x, \cos 2\pi x\} \Rightarrow \left\{\frac{1}{\sqrt{2}}, \cos\pi x, \cos 2\pi x\right\}.$$

获得投影

有了正交归一基后，我们便可将 f 投影到 $\mathbf{V}$ 上，其投影记为 $P_V f$：

$$P_V f = \sum_{i=1}^{n}\langle f, u_i\rangle\, u_i.$$

这样，f 与空间 $\mathbf{V}$ 之间的距离（即残差平方和 $\int_0^\infty |f(x) - P_V f(x)|^2\,dx$）便取得最小值。

例如，我们使用基 $\{\sin\pi x, \sin 2\pi x, \sin 3\pi x, \ldots\}$ 来拟合函数

$$u(x) = \begin{cases} 0, & -1 < x < 0, \\ 1, & 0 \le x < 1, \end{cases}$$

可以清楚地看到，这正是傅里叶级数展开的过程；由于各项正交，每次新增的项均展现出正弦函数的形状。

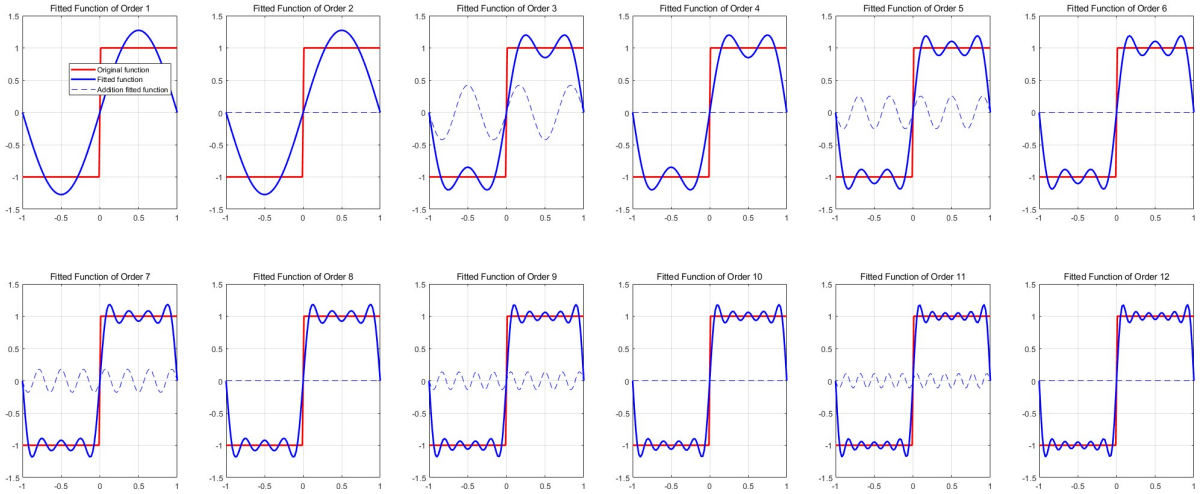

Figure 2: 正弦级数拟合示意图

下图展示了使用基

$$\left\{\sin\left(\left(\frac{1-(-1)^n+2n}{4}\right)\pi x+\frac{\pi\left(1+(-1)^n\right)}{4}\right)\right\}_{0\le n\le N}$$

（即余弦与正弦交替构成的基）进行展开的效果。可以看出，在余弦项上，附加项均为零，这是因为余弦函数为偶函数，而待拟合的方波为奇函数，在对称区间上它们天然正交，因此投影为零。由此可见，余弦函数不能反映出方波函数的特性，我们需要补充一组奇函数基来拟合目标函数。

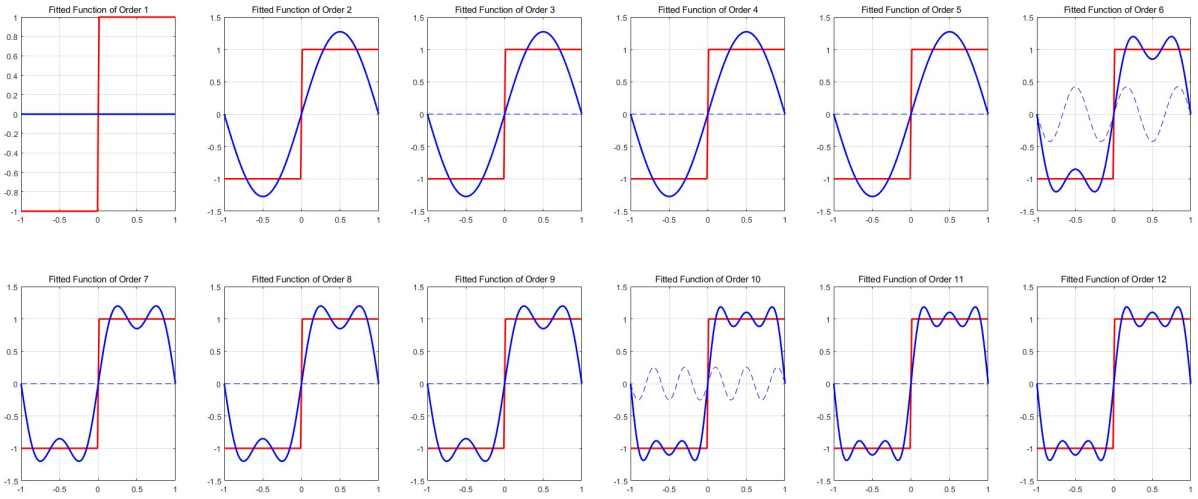

Figure 3: 交替余弦与正弦基的傅里叶展开

同理，使用基 $\{1, x, x^2, x^3, \dots\}$（需先进行正交归一化）拟合时，前几项表现出明显的多项式函数的特性，而后续项则因是多个多项式函数的线性组合而表现出模糊的多项式函数特性。

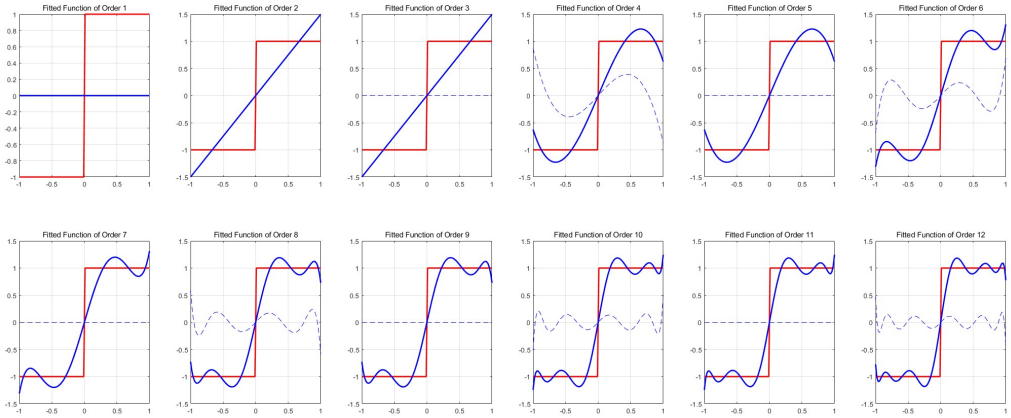

Figure 4: 多项式基拟合示意图

此外，我们还可以采用其他函数进行拟合，例如基 $\{1, e^x, e^{2x}, e^{3x}, \dots\}$（需先进行正交归一化）。

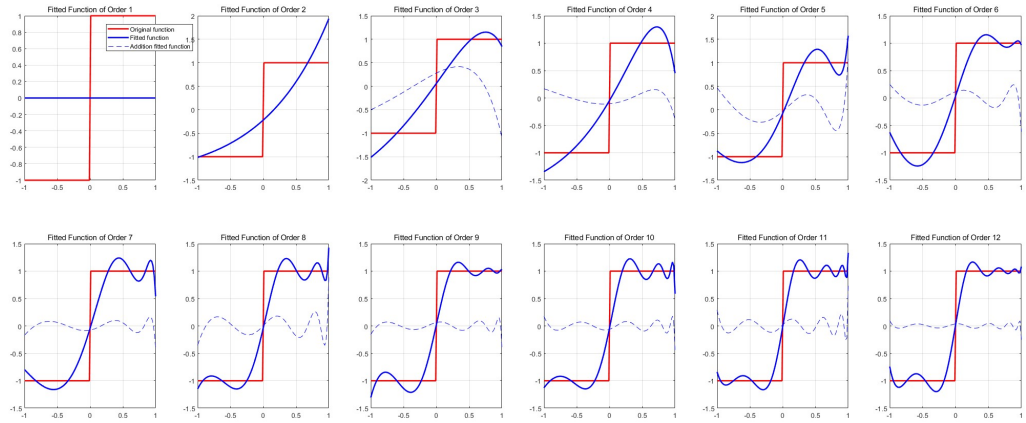

Figure 5: 指数函数拟合示意图

下图展示了利用$\{ln(x), ln(x+1), ln(x+2), ln(x+3), \dots\}$对数函数基在区间$[2\pi, 6\pi]$拟合$sin(x)$的示意图，

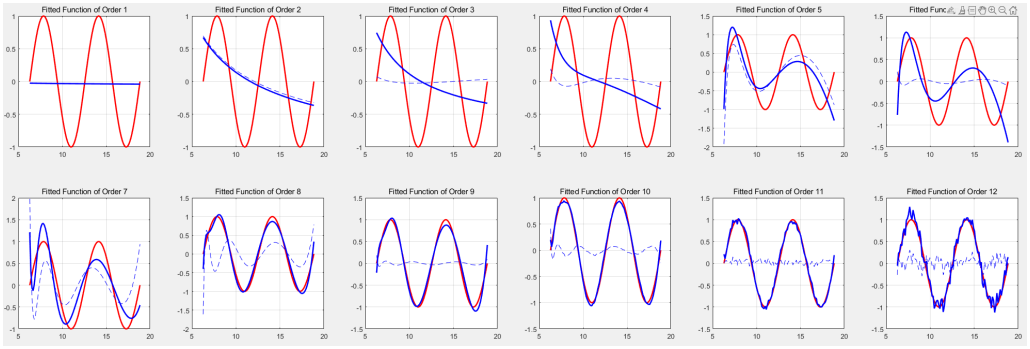

Figure 6: 对数基拟合示意图

此外，我们可以组合使用不同的基，例如利用

$$\{1, x, x^2, x^3, x^4, x^5\} + \{e^x, e^{2x}, e^{3x}, e^{4x}, e^{5x}, e^{6x}\}$$

在区间 $[-\pi, \pi]$ 内拟合 $\sin(x)$。左图显示先使用多项式基进行拟合，在第七项引入指数基；右图则相反。可以看出，若先采用多项式基，其拟合效果更为高效。

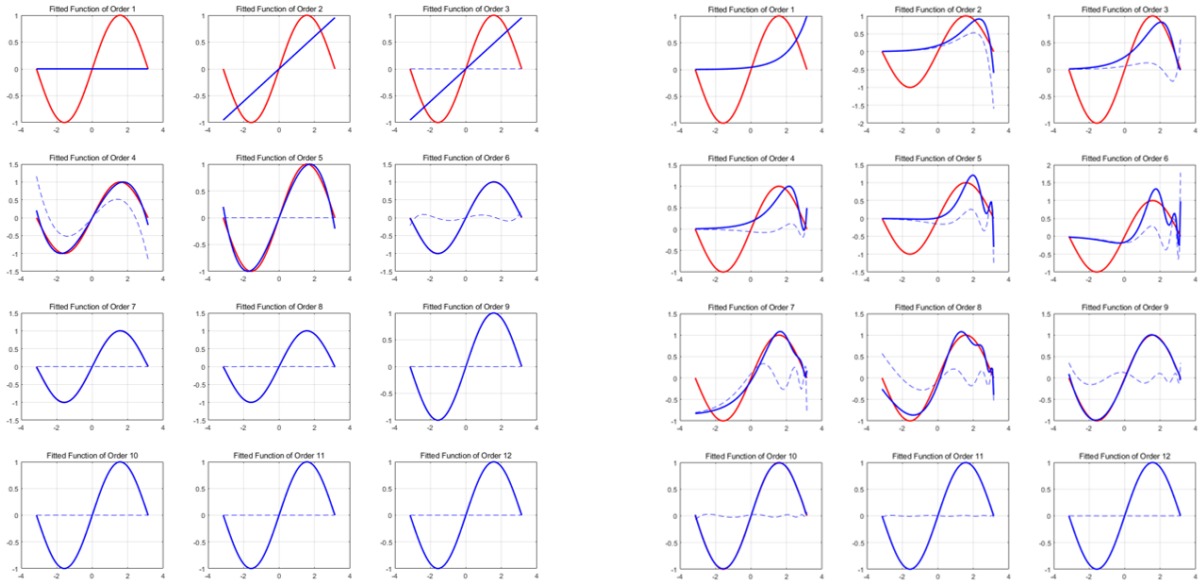

Figure 7: 不同基组合顺序对 $\sin(x)$ 拟合效果的比较

在实际编程过程中，我们可以任意选择基底 $\{\varphi_n\}_{n\in N}$ 来张成函数空间 $\mathbf{V}$，再通过投影的方法获得 f 在该基下的近似表示。为方便起见，常用的基包括 $\{x^n\}_{n\in\mathbf{N}}$、$\{\sin nx\}_{n\in\mathbf{N}}$、$\{\cos nx\}_{n\in\mathbf{N}}$ 以及 $\{\sin nx, \cos nx\}_{n\in\mathbf{N}}$，这些基函数具有明显的优越性。

3 积分变换

傅里叶变换

函数 f 在空间 V 下的投影为

$$P_V f = \sum_{n=1}^{N} \langle \varphi_n, f \rangle \varphi_n.$$

若认为 $P_V f$ 能很好地逼近 f（在紧致区间内一致收敛，在开区间内逐点收敛），则可写成

$$f = \sum_{n=1}^{N} \langle \varphi_n, f \rangle \varphi_n.$$

延用上述内积定义，采用正弦基

$$\left\{ \sin \frac{n \cdot 2\pi}{T} x \right\}_{n\in\mathbf{N}},$$

对定义在 $\left[-\frac{T}{2}, \frac{T}{2}\right]$ 内的函数 f 进行拟合。首先对该基进行正交归一化，得到

$$\left\{ \sqrt{\frac{2}{T}} \sin \frac{n \cdot 2\pi}{T} x \right\}_{n\in\mathbf{N}}.$$

此时 f 在该空间下的投影为

$$\begin{aligned} P_V f &= \sum_{n=1}^{N} \langle \varphi_n, f \rangle \varphi_n \\ &= \frac{2}{T} \sum_{n=1}^{N} \left(\int_{-\frac{T}{2}}^{\frac{T}{2}} f(\tau) \sin \frac{n \cdot 2\pi}{T} \tau \, d\tau \right) \sin \frac{n \cdot 2\pi}{T} x. \end{aligned}$$

利用 $T = \frac{2\pi}{\omega_0}$ 可重写为

$$P_V f = \frac{\omega_0}{\pi} \sum_{n=1}^{N} \left(\int_{-\frac{T}{2}}^{\frac{T}{2}} f(\tau) \sin(n\omega_0 \tau)\, d\tau \right) \sin(n\omega_0 t).$$

这便得到了 f 的正弦傅里叶展开。当 $T \to \infty$（即 $\omega_0 \to 0$）时，令 $\omega_0 = d\omega$ 且 $n\omega_0 = \omega$，则

$$P_V f = \frac{1}{\pi} \int_0^{N\cdot\omega_0} \left(\int_{-\infty}^{\infty} f(\tau) \sin(\omega\tau)\, d\tau \right) \sin(\omega t)\, d\omega.$$

考虑到外层积分原本从 $\omega_0 = 0$ 到 $N\omega_0$，为使拟合足够精确，可用无限项进行逼近，上式上限改为 ∞。由于内外层函数均为关于 ω 的奇函数，其乘积为偶函数，因此积分形式亦可调整为

$$P_V f = \frac{1}{2\pi} \int_{-\infty}^{\infty} \left(\int_{-\infty}^{\infty} f(\tau) \sin(\omega\tau)\, d\tau \right) \sin(\omega t)\, d\omega.$$

这便是傅里叶变换正弦表达式的由来。回顾最初版本

$$P_V f = \sum_{n=1}^{N} \langle \sqrt{\frac{2}{T}} \sin \frac{n\cdot 2\pi}{T} x, f \rangle \sqrt{\frac{2}{T}} \sin \frac{n\cdot 2\pi}{T} x,$$

可将其视为一组奇函数的叠加，因此 f 在 V 上的投影依然为奇函数。

同理，我们可求得 f 在空间

$$W = \mathbf{span} \left\{ \cos \frac{n\cdot 2\pi}{T} x \right\}_{0 \le n \le N}$$

下的投影：

$$P_W f = \frac{1}{2\pi} \int_{-\infty}^{\infty} \left(\int_{-\infty}^{\infty} f(\tau) \cos(\omega\tau)\, d\tau \right) \cos(\omega t)\, d\omega.$$

由于正弦基反映 f 的奇函数特性，而余弦基反映其偶函数特性，因此 $P_V f$ 与 $P_W f$ 互相正交。将两者相加，可更全面地表达 f：

$$\begin{aligned} P_V f + P_W f &= \frac{1}{2\pi} \int_{-\infty}^{\infty} \left(\int_{-\infty}^{\infty} f(\tau) \Big[\sin(\omega\tau)\sin(\omega t) + \cos(\omega\tau)\cos(\omega t) \Big]\, d\tau \right) d\omega \\ &= \frac{1}{2\pi} \int_{-\infty}^{\infty} \left(\int_{-\infty}^{\infty} f(\tau) \cos(\omega t - \omega\tau)\, d\tau \right) d\omega. \end{aligned}$$

$j^2 = -1$，且内层函数为关于ω的奇函数，则下面的主值积分为0

$$\frac{j}{2\pi} \int_{-\infty}^{\infty} \left(\int_{-\infty}^{\infty} f(\tau)\, sin\,(\omega t - \omega\tau)\, d\tau \right) d\omega = 0$$

$$\begin{aligned} P_V f + P_W f &= \frac{1}{2\pi} \int_{-\infty}^{\infty} \left(\int_{-\infty}^{\infty} f(\tau) \cos(\omega t - \omega\tau)\, d\tau \right) d\omega + \frac{j}{2\pi} \int_{-\infty}^{\infty} \left(\int_{-\infty}^{\infty} f(\tau)\, sin\,(\omega t - \omega\tau)\, d\tau \right) d\omega \\ &= \frac{1}{2\pi} \int_{-\infty}^{\infty} \left(\int_{-\infty}^{\infty} f(\tau)\, (cos\,(\omega t - \omega\tau) + j sin\,(\omega t - \omega\tau))\, d\tau \right) d\omega \\ &= \frac{1}{2\pi} \int_{-\infty}^{\infty} \left(\int_{-\infty}^{\infty} f(\tau)\, e^{j\omega(t-\tau)} d\tau \right) d\omega \\ &= \frac{1}{2\pi} \int_{-\infty}^{\infty} \left(\int_{-\infty}^{\infty} f(\tau)\, e^{-j\omega\tau} d\tau \right) e^{j\omega t} d\omega \end{aligned}$$

从而得到标准傅里叶积分表达式：

$$f(t) = \frac{1}{2\pi} \int_{-\infty}^{\infty} \left(\int_{-\infty}^{\infty} f(\tau) e^{-j\omega\tau}\, d\tau \right) e^{j\omega t}\, d\omega.$$

讨论收敛性

在上述例子中，我们所采用的基均为连续函数，因此 $P_V f + P_W f$ 也为连续函数，可反映 f 的连续性质。在适当条件下，可以证明该投影逐点或一致收敛于 f。

需要注意的是，傅里叶变换也存在局限性。比如将函数投影到正弦函数空间仅能得到其奇函数部分，无法完整反映函数的全部信息。例如，将 $g(x) = x + x^2$ 投影到正弦空间时，通过与各正交归一基求内积得到的系数仅包含 x 项，而 x^2 部分被舍去，从而丢失了相应的信息。

比较系数

在有限维基下的投影过程中，可能出现不同函数具有相同投影的情况，难以区分原始函数。为此，可以采用无限维基，通过比较投影系数来区分不同函数。如果一个函数在无限维空间下的所有投影系数均已确定，则该投影能够唯一标识原始函数。拉普拉斯变换正是基于这一思想来求解微分方程。

拉普拉斯变换

在有限维空间中，通过对基 $\{v_i\}_{0\le i\le n}$ 进行 Gram–Schmidt 正交化得到的投影，其结果仅为这些基的线性组合，且由于基的线性无关性与唯一性，无论正交化顺序如何，各元素前的系数均不变。

采用基 $\{e^{-st}\}_{s\in N}$ 张成函数空间时，我们可以选取其中的 e^{-st} 作为最后一步正交化的对象。除最后得到的标准基外，其他标准基均不含 e^{-st} 的线性项，故 e^{-st} 仅在最终投影中出现。

设最终得到的标准基为

$$u_s = c\,e^{-st} + \sum a_{si}u_i,$$

其中 c ， a_{si}与u_i 均为系统固有性质，与 f 无关。函数 f 在 u_s 上获得的投影向量为

$$\langle f, c\,e^{-st} + \sum a_{si}u_i\rangle\Big(c\,e^{-st} + \sum a_{si}u_i\Big),$$

而获得的仅与 e^{-st} 相关的系数仅为 $\langle f, c\,e^{-st}\rangle$。对所有基均可作类似处理，故 $\langle f, c\,e^{-st}\rangle$ 可看作仅与 s 相关的积分系数，与其他 s' 的积分系数共同构成投影向量，从而唯一确定 f。利用拉普拉斯变换求解微分方程实际上就是在比较这些系数的对应关系。

推广

上述过程并未利用 $\{e^{-st}\}$ 的特殊函数性质，因此无论是傅里叶变换还是拉普拉斯变换，都适用类似的方法。我们还可以采用如

$$\int_0^\infty f(x)x^s\,dx$$

（涉及幂函数）或

$$\int_0^\infty f(x)s^x\,dx$$

（涉及指数函数）或

$$\int_a^b f(x)ln(x+s)\,dx$$

（涉及对数函数）等变换，建立与幂、指数、对数等函数相关的关系，从而可以进一步得到 f 与其 n 阶导数 $\frac{d^n f}{dx^n}$ 之间的联系，以求解各类微分方程。

4 统计学应用

向量空间中的内积与夹角

在向量空间 $\mathbf{V}$ 中，我们定义向量 f 与 g 的内积为

$$\langle f, g\rangle = \|f\|\,\|g\|\cos\theta,$$

其中 θ 是两向量的夹角。于是，我们有

$$\cos\theta = \frac{\langle f, g\rangle}{\|f\|\,\|g\|}.$$

当 $\cos\theta = 1$ 时，$\theta = 0$，此时 f 与 g 同向，即 f 可表示为 g 的标量倍数 $f = \lambda g$, $\lambda \in \mathcal{R}$，两向量完全线性相关；当 $\theta = \frac{\pi}{2}$ 时，$\langle f, g\rangle = 0$，f 与 g 正交，表示两向量在空间中相互独立无关。

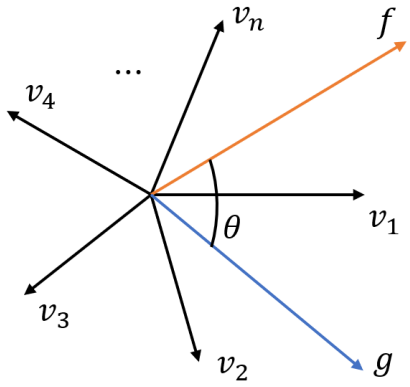

Figure 8: 向量夹角示意图

相关系数的定义

由此，我们可以定义向量 v_1 与 v_2 的相关系数为

$$r = \frac{\langle v_1, v_2\rangle}{\|v_1|\,\|v_2|}.$$

沿用函数空间的内积定义，令

$$r = \frac{\int_a^b f(x)\,g(x)\,\mathrm{d}x}{\sqrt{\int_a^b f^2(x)\,\mathrm{d}x}\sqrt{\int_a^b g^2(x)\,\mathrm{d}x}} = \frac{\sum_{i=0}^{n} f(x_i)\,g(x_i)}{\sqrt{\sum_{i=0}^{n} f^2(x_i)}\sqrt{\sum_{i=0}^{n} g^2(x_i)}}.$$

皮尔逊相关系数

若我们关注两个数据集的中心化变化相关性，则需先将各自减去均值 $\bar{x}, \bar{y}$，得到经典的皮尔逊相关系数：

$$r = \frac{\sum_{i=0}^{n}\left(x_i - \bar{x}\right)\left(y_i - \bar{y}\right)}{\sqrt{\sum_{i=0}^{n}\left(x_i - \bar{x}\right)^2}\sqrt{\sum_{i=0}^{n}\left(y_i - \bar{y}\right)^2}}.$$

该系数衡量了两个数据集在排除整体水平差异后的线性相关程度，取值范围为 $[-1, 1]$，其中 ± 1 分别对应完全正相关与完全负相关，0 表示无线性相关性。

5 Appendix

```
n = 12;              % Fitting order
a = 2*pi; b = 6*pi;  % Interval
orifun = @(x) sin(x);% Function to be fitted
createfun = @(ii,x) log(x+ii-1);
% Options for basis:
% Exponential: exp((ii-1)*x)
% Polynomial: x.^(ii-1)
% Sine: sin(ii*2*pi/(b-a)*(x-(a+b)/2))
% Cosine: cos((ii-1)*2*pi/(b-a)*(x-(a+b)/2))
% Logarithm: log(ii+x)

x = linspace(a,b,100);

fun = cell(1,n);        % Basis functions (not orthonormalized)
normalfun = cell(1,n); % Orthonormalized basis functions

% Generate the required bases
for ii = 1:n
    fun{ii} = @(x) createfun(ii,x);
    normalfun{ii} = fun{ii};
end

% Normalize the first vector
coeff = integral(@(x) normalfun{1}(x).^2, a, b);
normalfun{1} = @(x) normalfun{1}(x) / sqrt(coeff);
% Orthonormalize the n vectors
for ii = 2:n
    for jj = 1:ii-1
        coeff = integral(@(x) normalfun{ii}(x) .* normalfun{jj}(x), a,
           b);
        normalfun{ii} = @(x) normalfun{ii}(x) - coeff .* normalfun{jj}(
           x);
    end
    coeff = integral(@(x) normalfun{ii}(x).^2, a, b);
    normalfun{ii} = @(x) normalfun{ii}(x) / sqrt(coeff);
end

% Fitting process
coeff = zeros(1, n);
fitfun = @(x) 0;

% Create figure layout
figure;
rows = ceil(n/6);
cols = 6;
% Iterative fitting
for ii = 1:n
    coeff(ii) = integral(@(x) orifun(x) .* normalfun{ii}(x), a, b);
    fitfun = @(x) fitfun(x) + coeff(ii) .* normalfun{ii}(x);
    % Plot each fitting curve
    subplot(rows, cols, ii);
    plot(x, orifun(x), 'r', 'LineWidth', 2, 'DisplayName', 'Original␣
       function');
    hold on;
    plot(x, fitfun(x), 'b', 'LineWidth', 2, ...
         'DisplayName', ['Fitted␣function␣(Order␣' num2str(ii) ')']);
```

```
    contribution = @(x) coeff(ii) .* normalfun{ii}(x);
    plot(x, contribution(x), '--b', 'DisplayName', ...
         ['Additional fitted function (Order ' num2str(ii) ')']);
    % Set title and legend for each subplot
    title(['Fitted Function of Order ' num2str(ii)]);
    grid on;
end
```