Computer Science and Application 计算机科学与应用, 2018, 8(3), 339-353
Published Online March 2018 in Hans. http://www.hanspub.org/journal/csa
https://doi.org/10.12677/csa.2018.83039

Hans汉斯

Sinc Function-Based Regression Algorithm

Dong Chen, Yongbin Yu, Yuxin Guo, Chengtao Wang, Nyima Tashi

School of Information and Software Engineering, University of Electronic Science and Technology of China, Chengdu Sichuan
Email: 1040155788@qq.com

Received: Mar. 12th, 2018; accepted: Mar. 21st, 2018; published: Mar. 28th, 2018

Abstract

In this paper, the reconstruction of discrete signals in sampling theorem is introduced, and a linear and logistic regression model based on Sinc function is proposed. In order to minimize the mean square error of the regression function, Sinc function is designed in the independent variable domain of data, and then the regression curve is reconstructed directly. On the basis of the linear and logistic regression model based on Sinc function and the algorithm analysis, a sufficient simulation experiment is carried out. And compared with the traditional linear regression, regression algorithm based on Sinc function has obvious advantages when the regression function is not obvious. Finally, the linear regression algorithm based on Sinc function is succeeding to predict the monthly minimum temperature.

Keywords

Sinc, Linear Regression, Logistic Regression

基于Sinc函数的回归算法

陈　董，于永斌，郭雨欣，王承韬，Nyima Tashi

电子科技大学信息与软件工程学院，四川 成都
Email: 1040155788@qq.com

收稿日期：2018年3月12日；录用日期：2018年3月21日；发布日期：2018年3月28日

摘　要

本文引入采样定理中离散信号的重构思想，提出基于Sinc函数的线性与逻辑回归模型及其算法。为使回归函数的均方误差最小化，在数据的自变量域中设计出Sinc函数，然后直接重构出回归曲线。在基于Sinc函数的线性与逻辑回归模型推导与算法分析基础上，进行了充分的仿真实验验证。与传统线性回归相比，

文章引用: 陈董, 于永斌, 郭雨欣, 王承韬, Nyima Tashi. 基于 Sinc 函数的回归算法[J]. 计算机科学与应用, 2018, 8(3): 339-353. DOI: 10.12677/csa.2018.83039

基于Sinc的回归算法在回归函数关系不明显时具有明显优势。最后，将基于Sinc的线性回归算法成功用于月最低气温预测。

关键词

Sinc，线性回归，逻辑回归

1. 引言

回归分析(regression analysis)是一种确定两种或两种以上变量间相互依赖关系的统计分析方法。回归分析按照涉及变量的多少，分为一元回归和多元回归分析；按照自变量和因变量之间的关系类型，可分为线性回归分析和非线性回归分析。如果回归模型的因变量和自变量是非线性函数形式，回归规律在图形上表现为形态各异的各种曲线，称为非线性回归[1]。回归分析已广泛应用于社会，经济，工程，医学，卫生，农业，气象和水文[2]。

非线性函数的求解一般有两种情况：非线性可变换成线性和不可变换成线性。处理可线性化的非线性回归的基本方法是，通过变量变换，将非线性回归化为线性回归，然后用线性回归方法处理。这种处理方式需要根据理论或经验、已获得输出变量与输入变量之间的非线性表达式、输入输出的 n 次观察结果来确定系数的值[3]。

本文针对一元非线性可变换成线性的回归问题中存在的需要根据经验进行变量带换、求表达式系数需要训练、较复杂回归方程中因变量和自变量函数关系不易得出的问题，通过引入采样定理中离散信号的重构思想，提出基于 Sinc 函数的回归算法。在信号处理领域，Sinc 滤波器是一种理想的电子滤波器，可消除给定带宽的信号分量，仅保留低频信号[4]。由于具有无限延迟的理想 Sinc 滤波器(称为矩形滤波器)，现实世界中的滤波器只是其近似值之一。但它在概念验证和论证验证中仍然被广泛应用，如采样定理和 Whittaker-Shannon 插值公式[5]。该算法引入数字信号处理中时域重构原理，首先根据数据集找到自变量间的最小间隔，使得数据集对应所要拟合的函数呈现过采样，再基于局部线性回归模型的思想(详细介绍见[6])，通过在每个数据点求解一个单独的最小二乘问题来逼近目标函数[7]。最后根据数据的最小间隔对应的 Sinc 函数进行时域重构，得出使代价函数最小的回归曲线。基于相同思想，本文进一步提出该算法在逻辑回归算法中的应用。最后，考虑到算法只能在数据范围内预测的局限性，本文将基于 Sinc 的线性回归算法应用于月最低温度预测。

2. 回归模型

我们做出如下假设：

假设 I：所要求得的回归曲线可以通过低通滤波获得其频域的主频范围。由数字信号处理的相关知识可知，时限信号的频域不可能为带限信号，即由目标函数采样而来的数据集在频域无绝对带宽。又考虑到一般要求得到的回归函数的泛化能力，回归函数应为较简单曲线，所以目标函数频域的主频范围可通过低通滤波获得。

假设 II：数据集足够大。这里数据集足够大是指对于自变量的最小值 x_0，每个间隔点上 x_0+nT (其中 T 为自变量最小间隔， $n=1,2,3,\cdots$)都存在数据点。

2.1. 基于 Sinc 的线性回归

2.1.1. 代价函数最小化

定义 $x^{(k)}\left(k=0,1\cdots,m\right)$ 为第 k 个自变量的值，且满足 $x^{(k)}=x^{(1)}+\left(k-1\right)T$。假定 $x^{(k)}$ 上有 p 个数据点 $y_k^{(i)}\left(i=1,2,3,\cdots,p\right)$.

则 $x^{(k)}$ 上的代价函数为：

$$J\left(\theta\right)^k=\frac{1}{p}\sum_{i=1}^{p}\frac{1}{2}\left(h_\theta\left(x^{(k)}\right)-y_k^{(i)}\right)^2 \tag{1}$$

1) 中 $h_\theta\left(x^{(k)}\right)$ 表示最终的回归函数在 $x^{(k)}$ 上对应因变量的值。

$$\begin{aligned}&\frac{\mathrm{d}J\left(\theta\right)^k}{\mathrm{d}h_\theta\left(x^{(k)}\right)}=\frac{1}{p}\sum_{i=1}^{p}\left(h_\theta\left(x^{(k)}\right)-y_k^{(i)}\right)=h_\theta\left(x^{(k)}\right)-\frac{\sum_{i=1}^{p}y_k^{(i)}}{p}=0\\&h_\theta\left(x^{(k)}\right)=\frac{\sum_{i=1}^{p}y_k^{(i)}}{p}=y_k\end{aligned} \tag{2}$$

由(2)可知，当回归函数经过每一个自变量 $x^{(k)}$ 上数据的平均值点时，该点处代价函数将最小。又因为总的代价函数为每个自变量点上代价函数的加和，所以当回归函数经过每个自变量上因变量的均值点时，总的代价函数达到最小。

这里，我们称使代价函数最小的每个自变量上的点成为最优点，用 $x\left[n\right]$ 表示：

$$x\left[n\right]=y_k\ \left(n=k=1,2,3,\cdots\right) \tag{3}$$

我们称经过最优点的回归曲线(函数)为最优曲线(最优函数)，用 $f\left(x\right)$ 表示。

2.1.2. 基于低通滤波的时域重构

为了方便用数字信号处理知识进行相关推导，我们在此用 t 代替自变量 x，用 Ω 表示其对应频域变量。

F 的傅里叶变换为：

$$F\left(f\right)=\int \mathrm{e}^{-2\pi j\left(x\Omega\right)}f\left(x\right)\mathrm{d}x \tag{4}$$

令 T_s 表示数据集中自变量 t 间的最小间隔， $x_a\left(t\right)$ 为最优曲线 $f\left(x\right)$ 的原函数。

为了方便表述，以下用带限信号举例(非带限信号整个过程不变)。

下图为最优曲线采样时，时域与频域的变换情况，其中 A、B、C 表示时域，a、b、c 表示频域，且 C 中采样后的时域数据即为由数据集得来的最优点。

图 1 中 B 到 C 的采样过程：

$$\delta_T\left(t\right)=\sum_{n=-\infty}^{\infty}\delta\left(t-nT\right) \tag{5}$$

$$x_d\left(t\right)=x_a\left(t\right)\delta_T\left(t\right)=\sum_{n=-\infty}^{\infty}x_a\left(nT\right)\delta\left(t-nT\right) \tag{6}$$

由于时域离散化，对应频域周期化，所以频域对应 b 到 c 变化：

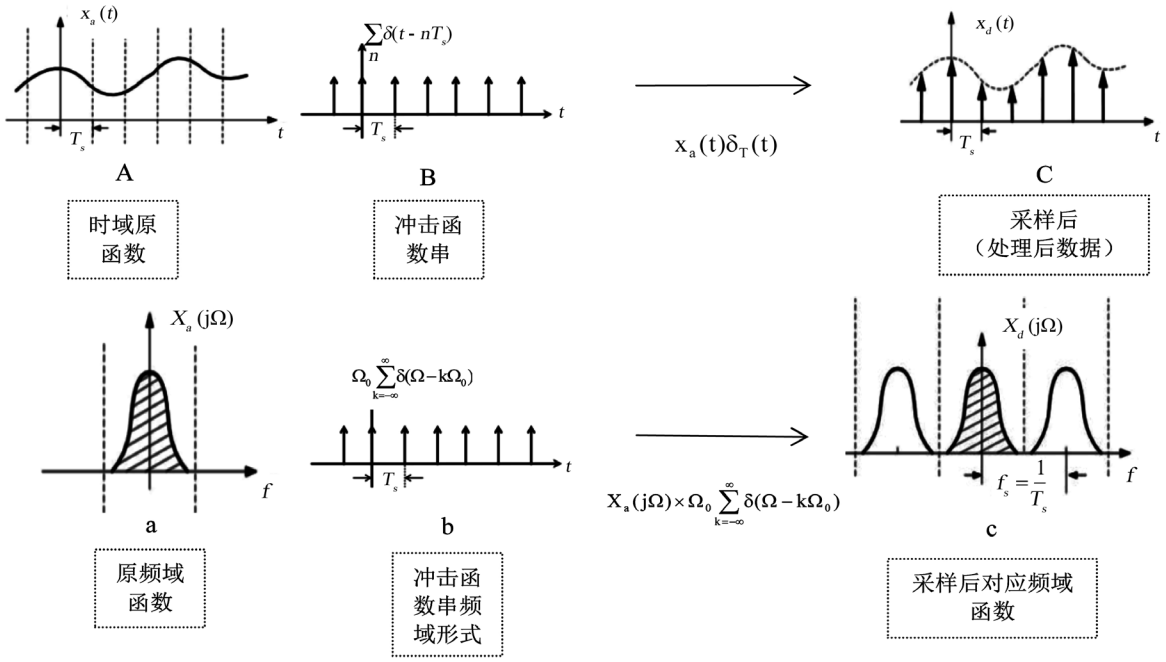

Figure 1. Sampling process
图 1. 最优曲线的采样过程中时域、频域变化情况

$$X_d\left(j\Omega\right)=\frac{1}{T_s}\sum_{k=-\infty}^{\infty}X_a\left(j\left(\Omega-k\Omega_0\right)\right) \tag{7}$$

在 c 的基础上，经过低通滤波，可以得到最优函数频域的主频频范围。在假设 I 和假设 II 的前提下，$\frac{\Omega_s}{2}$ 可以包含最优函数频域的主频范围。低通滤波器的频域表达式如下：

$$H_r\left(j\Omega\right)=\begin{cases}T, & \left|\Omega\right|\le\Omega_{s/2}\\ 0, & \left|\Omega\right|>\Omega_{s/2}\end{cases}$$

$$\Omega_{\frac{s}{2}}=\frac{\pi}{T_s}$$

上述低通滤波的过程，在频域表现为：$X_d\left(j\Omega\right)\times H_r\left(j\Omega\right)$. 对应如图 2。
。
则对应时域为：

$$\begin{aligned}x_a\left(t\right)&=x_d\left(t\right)*h_r\left(t\right)=\sum_{n=-\infty}^{\infty}x_a\left(nT_s\right)h_r\left(t-nT_s\right)\\&=\sum_{n=-\infty}^{\infty}x\left[n\right]h_r\left(t-nT_s\right)\end{aligned} \tag{8}$$

其中：

$$h_r\left(t\right)=\frac{1}{2\pi}\int_{-\infty}^{\infty}H_r\left(j\Omega\right)e^{j\Omega t}d\Omega=\frac{T}{2\pi}\int_{-\Omega_s/2}^{\Omega_s/2}e^{j\Omega t}d\Omega=\frac{\sin\left(\Omega_s t/2\right)}{\Omega_s t/2},\quad -\infty\le t\le\infty \tag{9}$$

由图 1 流程，在我们假定 A 中最优曲线存在的前提下，可以由数据集与公式(2)得到 C 中各个最优点。此时要求得时域原函数只需根据数据自变量间最小间隔 T_s 对应的 Sinc 函数(Sinc 函数图形如图 4)进行时域重构。重构后的时域图像如图 3。

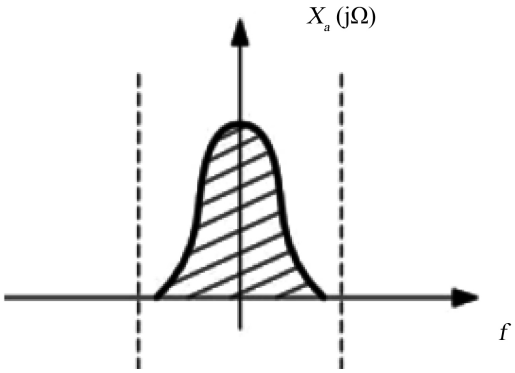

Figure 2. Obtain the frequency domain function by the low pass filter
图 2. 由低通滤波获得主频范围

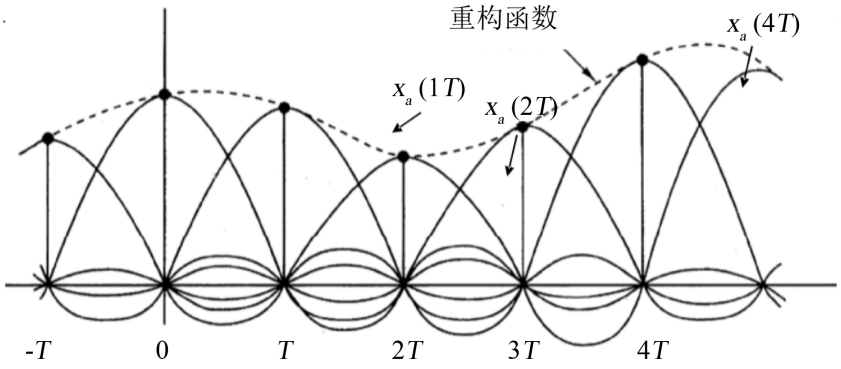

Figure 3. Time domain reconstruction
图 3. 时域重构

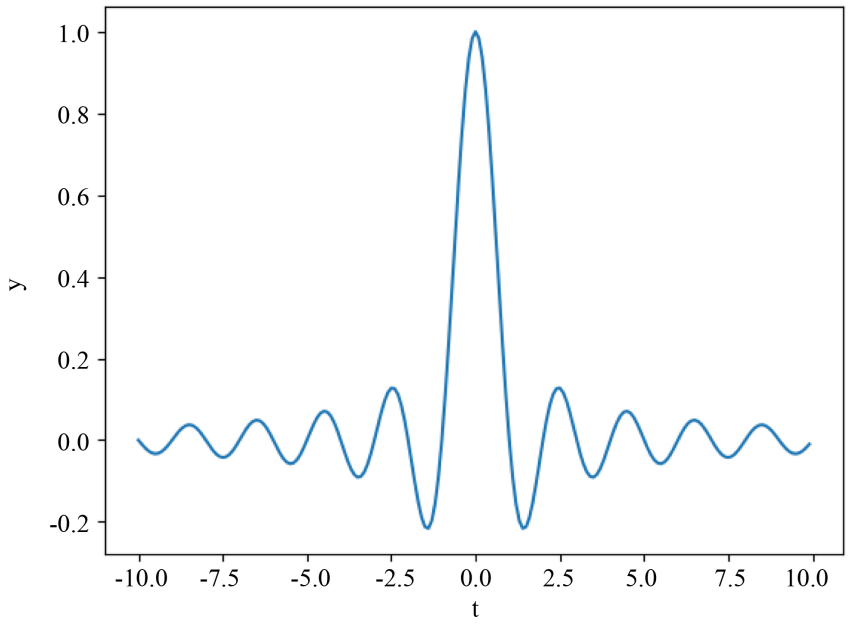

Figure 4. Sinc
图 4. Sinc

2.1.3. 基于 DTFT 基本公式的时域重构

更纯粹的公式推导如下：

$x_a(t)$ 的正反傅里叶变换：

$$\begin{aligned} X_a(j\Omega) &= \int_{t=-\infty}^{\infty} x_a(t)\mathrm{e}^{(-j\Omega t)}\mathrm{d}t \\ x_a(t) &= \frac{1}{2\pi}\int_{\Omega=-\infty}^{\infty} X(\Omega)\mathrm{e}^{(j\Omega t)}\mathrm{d}\Omega = \frac{1}{2\pi}\int_{\Omega=-\Omega_0}^{\Omega_0} X(\Omega)\mathrm{e}^{(j\Omega t)}\mathrm{d}\Omega \end{aligned} \tag{10}$$

$x[n]$的正反 DTFT 变换：

$$\begin{aligned} X(\omega) &= \sum_{n=-\infty}^{\infty} x[n]\mathrm{e}^{(-j\omega n)} \\ x[n] &= \frac{1}{2\pi}\int_{\omega=-\pi}^{\pi} X(\omega)\mathrm{e}^{(j\omega n)}\mathrm{d}\omega \end{aligned} \tag{11}$$

其中 $\omega = \Omega T_s$。

假定 $X_a(j\Omega) = X(\omega)$.

$$x[n] = \frac{1}{2\pi}\int_{\omega=-\pi}^{\pi} X(\omega)\mathrm{e}^{(j\omega n)}\mathrm{d}\omega = \frac{1}{2\pi}\int_{\omega=-\pi}^{\pi} X(\Omega)\mathrm{e}^{(j\Omega T_s)}\mathrm{d}\Omega T_s = T_s x(nT_s) \tag{12}$$

得 $X(\omega) = \dfrac{X_a(j\Omega)}{T_s}$.

$$\begin{aligned} x_a(t) &= \frac{1}{2\pi}\int_{\Omega=-\infty}^{\infty} X(\Omega)\mathrm{e}^{(j\Omega t)}\mathrm{d}\Omega \\ &= \frac{T_s}{2\pi}\int_{\omega=-\pi}^{\pi} X(\omega)\mathrm{e}^{\left(\frac{j\omega t}{T_s}\right)}\mathrm{d}\frac{\omega}{T_s} \\ &= \frac{T_s}{2\pi}\int_{\omega=-\pi}^{\pi}\left(\sum_{n=-\infty}^{\infty} x[n]\mathrm{e}^{(-j\omega n)}\right)\mathrm{e}^{\left(\frac{j\omega t}{T_s}\right)}\mathrm{d}\frac{\omega}{T_s} \\ &= \frac{T_s}{2\pi}\sum_{n=-\infty}^{\infty} x[n]\left(\int_{\omega=-\pi}^{\pi}\mathrm{e}^{(-j\omega n)}\mathrm{e}^{\left(\frac{j\omega t}{T_s}\right)}\mathrm{d}\frac{\omega}{T_s}\right) \\ &= \sum_{n=-\infty}^{\infty} x[n]\frac{\sin\left(\left(t-nT_s\right)\frac{\pi}{T_s}\right)}{\left(t-nT_s\right)\frac{\pi}{T_s}} = \sum_{n=-\infty}^{\infty} x[n]h_r(t) \end{aligned} \tag{13}$$

可以看出由 DTFT 推导的时域重构结果与低通滤波原理的时域重构结果相同，且在由数据集经量化等处理过程后，在仅知道最小数据间隔 T_s 的情况下，就可以求出使代价函数最小的回归函数。

2.1.4. 基于 Sinc 函数的线性回归算法步骤

基于Sinc函数的线性回归算法沿用传统的代价函数公式(1)，但不再进行自变量与因变量形式的代换，也不再训练自变量前的系数。其算法的具体步骤如下：

输入：数据矩阵 $x_{1\times a}$，其中 a 代表 a 个数据；$y_{1\times a}$。

输出：因变量预测值 y.

算法步骤：

对 x 操作，适当量化，求出尽可能小的自变量间隔，放入 T，同时将不同自变量按间隔以从小到大顺序放入 $x_{1\times N}$，这里假定在间隔 T 时，有 N 个不同自变量。

根据 $x_{1\times N}$，将 y 变为 $y_{m\times N}$，其中 m 代表每个不同自变量上 m 个因变量的值。

1) 求出每个自变量上的最优点，对 y 做如下操作：

for i = 1:N

对 $y_{m\times N}$ 第 i 列加和除以 m，放入 $x[i]$中

end for。

2) 将 T 带入公式(9)求出相应 Sinc 函数。

3) 最优点分别与 4)中所得 Sinc 函数相乘并累加，得 y。

4) 算法结束。

时间复杂度：

$$T(n)=O(a+N) \tag{14}$$

2.2. 基于 Sinc 的逻辑回归

2.2.1. 旋转

为了方便陈述基于 Sinc 函数的逻辑回归算法思想，以二分类问题为例，且令每个自变量上，两类因变量个数相当。当不满足该条件时，对数据集进行适当旋转，以使数据集尽量满足该条件。以图 5 和图 6 为例。

2.2.2. 代价函数最小化

$$Cost\left(h_\theta(x),y\right)=\begin{cases}-\log\left(1-h_\theta(x)\right), & \text{if } y=0\\ -\log\left(h_\theta(x)\right), & \text{if } y=1\end{cases} \tag{15}$$

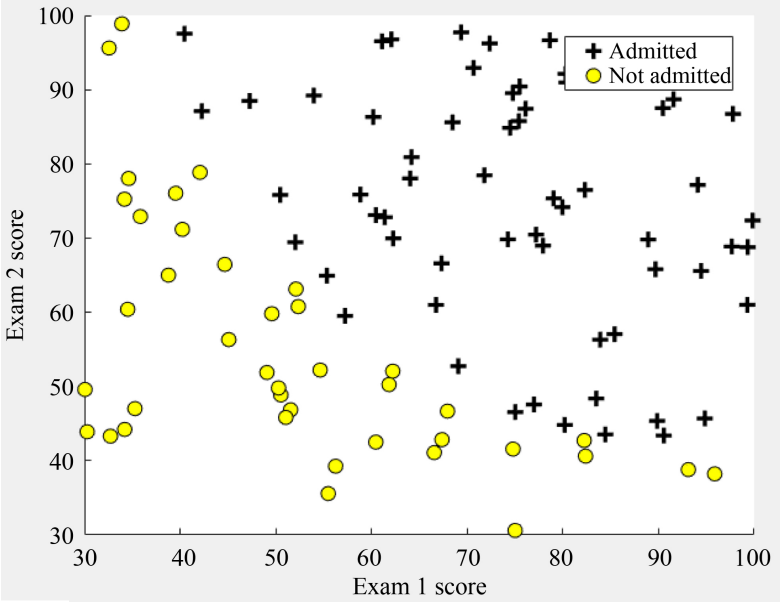

Figure 5. Admission data [8]
图 5. 入学成绩数据[8]

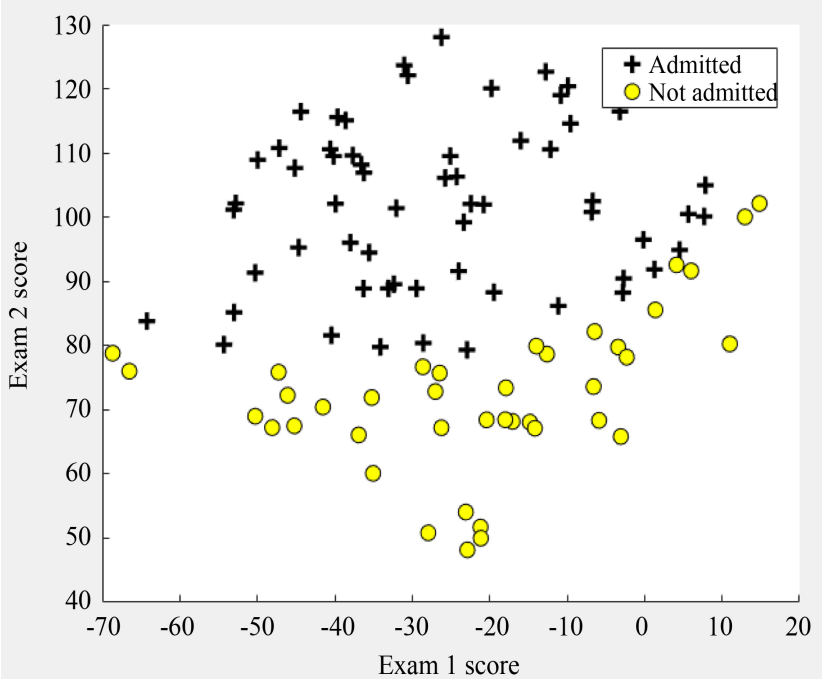

Figure 6. Data rotated 60˚
图 6. 逆时针旋转 60˚

$$h_\theta(x)=\text{sigmoid}(z)=\frac{1}{1+e^{-z}}$$
$$z=W^{T}X \tag{16}$$

从公式(15)可以看出，当 $y=1$ 时 $h_\theta(x)$ 越接近 1，代价函数越小；当 $y=0$ 时，$h_\theta(x)$ 越接近 0 代价函数越小。

由图 7 可以看出，z 越大 h 越大，为了使 z 尽可能的大，应满足当 z = 0 时，数据点尽可能的远离 $z=f(x_1,x_2)$ 在 x_1Ox_2 上的投影。这里我们称该投影为边界函数，其中 x_1,x_2 分别表示两种自变量。

2.2.3. 确定边界函数

为了使用和线性回归相同的原理，我们需要找到使代价函数最小的边界函数上的点 x_0^k。当 x_1^i 被确定，x_0^k 表示 x_2^i 的数值。

假设当 $x_1^{(i)}=x_1^{(k)}\ (k=0,1,\cdots,m)$ 时，有 p 个 x_2^i 在 x_1^k 上，其中 m 表示满足假设 I 的 x_1 个数。

对每个 x_1^k，我们应当使得所有 x_2^i 离边界函数尽可能的远。所以定义如下距离函数:

$$\text{distince}^k=\sum_{i=1}^{p}\left(x_2^i-x_0^k\right)^2 \tag{17}$$

我们需要使每一类数据公式(17)的值尽可能的大，又由于每个 x_1^i 上有两类数据,可得图 8。

图中蓝色的线代表 $y=0$ 的距离函数值随 x_0^k 的变化情况，橘色线代表 $y=1$ 的距离函数值随 x_0^k 变化的情况；实线之外代表不存在数据点的部分，红色点和绿色点仅表示曲线的趋势；红色“x”表示 $y=0$ 的数据的 x_2 在 x_1^k 上的均值；绿色“x”表示 $y=1$ 的数据的 x_2 在 x_1^k 上的均值。

当 $y=0$ 的一部分 x_2^i 大于 x_0^k 时，它们将对距离函数值做负贡献；同样，当 $y=1$ 的一部分 x_2^i 小于 x_0^k 时，它们也将对距离函数值做负贡献。假设一共有 q 个做出负贡献的点，我们可以得到如下新的距离函数：

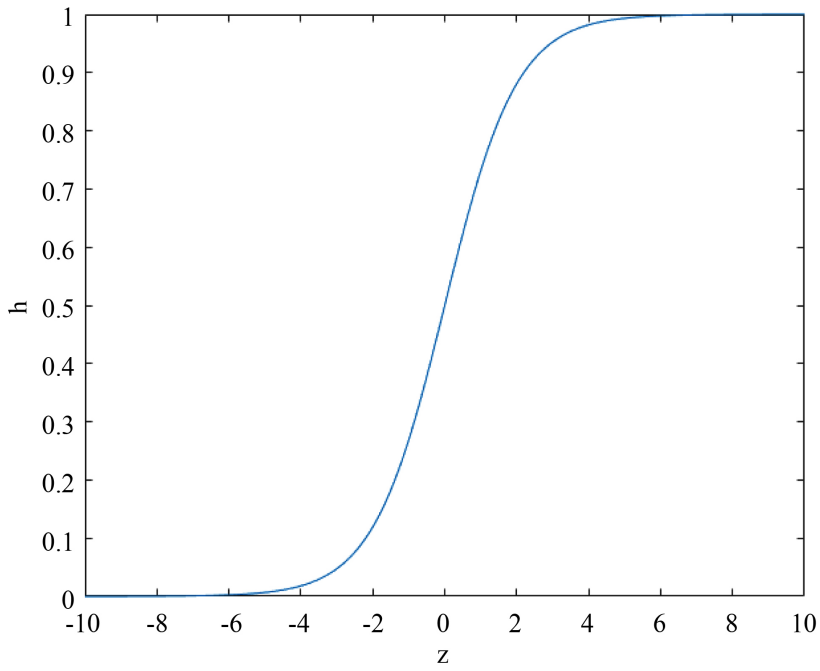

Figure 7. Sigmoid function
图 7. Sigmoid 函数

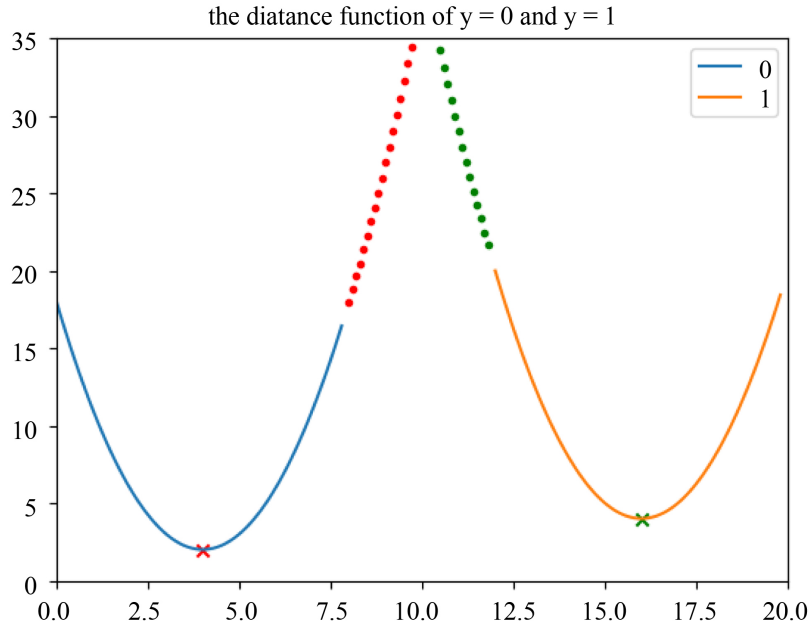

Figure 8. Distance function
图 8. 距离函数

$$\text{distince}^k = \sum_{i=1}^{p-q}\left(x_2^i - x_0^k\right)^2 - \sum_{j=1}^{q}\left(x_2^i - x_0^k\right)^2 \tag{18}$$

对于此处讨论的逻辑回归问题，由于之前假定 $y = 0$ 与 $y = 1$ 在每个 x_1 上个数相当。在求 x_0^k 时，我们令两类距离函数值相等。

$$\sum_{i=1}^{p_2}\left(x_2^{1i}-x_0^k\right)^2=\sum_{j=1}^{p_1}\left(x_2^{0j}-x_0^k\right)^2$$

$$x_0^k=\frac{\left(2\overline{x^0}p_1-2\overline{x^1}p_2\right)\pm\sqrt{\left(2\overline{x^0}p_1-2\overline{x^1}p_2\right)^2-4\left(p_1-p_2\right)N}}{2\left(p_1-p_2\right)}$$

$$\overline{x^0}=\frac{\sum_{j=1}^{p_1}x_2^{0j}}{p_1} \tag{19}$$

$$\overline{x^1}=\frac{\sum_{i=1}^{p_2}x_2^{1i}}{p_2}$$

$$N=\sum_{j=1}^{p_1}\left(x_2^{0j}\right)^2-\sum_{i=1}^{p_2}\left(x_2^{1i}\right)^2$$

其中 x_2^{0i} 代表 $y=0$ 的数据， x_2^{1i} 代表 $y=1$ 的数据。

2.2.4. 基于 Sinc 函数的逻辑回归算法步骤

基于 Sinc 的逻辑回归算法关键在于找到使代价函数最小的边界函数上的点，再对这些点进行“时域重构”。其算法的具体步骤如下：

输入：数据矩阵 $x_{2\times a}$，其中 a 代表 a 个数据， $x_{2\times a}$ 为一个 2 × a 大小的矩阵，每一行为一个自变量，两个自变量分别表示为 x_1、x_2； $y_{1\times a}$。

输出：因变量预测值边界函数。

算法步骤：

1) 对 $x_{2\times a}$ 进行旋转操作，使得每个 x_1 上两类 x_2 数量尽可能相当。

2) 对 x_1 操作，适当量化，求出尽可能小的自变量间隔，放入 T，同时将不同自变量按间隔以从小到大顺序放入 $x1_{1\times N}$，这里假定在间隔 T 时，有 N 个不同自变量 $x1$。

3) 根据 $x1_{1\times N}$，将 $x2$ 变为 $x2_{m\times N}$，其中 m 代表每个不同自变量上 m 个 x_2 的值。

4) 求出每个 x_1 上的边界函数上的点，对 x_2 做如下操作：

for i = 1:N

对 $x2_{m\times N}$ 第 i 列加和除以 m，放入 $x[i]$中

end for。

5) 将 T 带入公式(9)求出相应 Sinc 函数。

6) 最优点分别与 4)中所得 Sinc 函数相乘并累加，得边界函数。

7) 算法结束。

时间复杂度与基于 Sinc 的线性回归基本相当在不考虑旋转等步骤的情况下，可以近似为 $T(n)=O(a+k)$。

3. 仿真实验

3.1. 线性回归仿真与对比

定义 $y=\sqrt{x}$ 为回归的目标曲线，按正态分布在每个自变量上随机生成 30 个点，按间距 0.5 在[0:50]范围内生成 3000 个数据点。数据图 9 如下：

基于 Sinc 的线性回归生成如图 10 曲线。

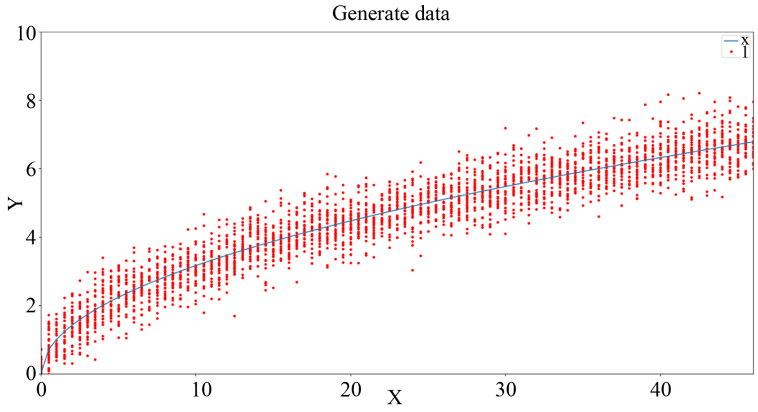

Figure 9. Generate data
图 9. 生成数据

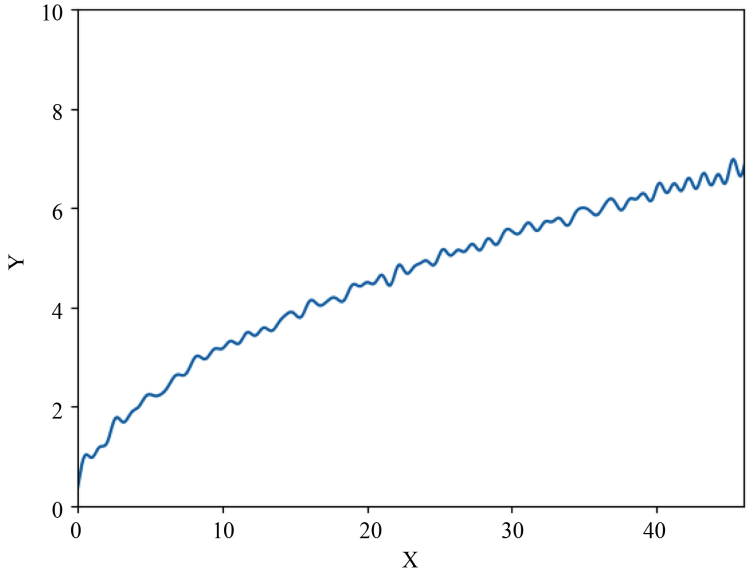

Figure 10. Sinc linear regression
图 10. 基于 Sinc 的回归

使用相同分布，按间隔 0.7 生成测试集数据对回归函数进行检验。

与传统线性回归进行对比。首先进行变量代换，令 $x' = \sqrt{x}$ 。按线性做法，使用梯度下降法进行训练，令学习率为 0.01，迭代次数为 10，得回归曲线，后用同样测试集进行测试。

两种算法测试集代价函数值与运行时间，见下表 1：

对比图 11 与图 12 可知，传统线性回归所求的函数泛化能力更强，同时由表 1 可知，对于 $y = \sqrt{x}$ 两种算法性能基本相同。但本实验可以由数据明显看出自变量与因变量的关系，进而进行变量代换，当这种关系不明显时基于 Sinc 的回归将具有明显优势。但同时应注意到基于 Sinc 的回归只能在数据范围内进行预测。

3.2. 逻辑回归仿真

围绕 $y = \sqrt{x}$ 和 $y = \sqrt{x} + 6$ 在[0:50]上以间隔 0.5，按正太分布随机生成 3000 个点，每个 x 上总共生成 30 个点。

Table 1. The comparison of two algorithms
表 1. 两种算法对比

Algorithm	Running time	Cost
Sinc regression	0.08746	0.25843
Linear regression	0.09534	0.25062

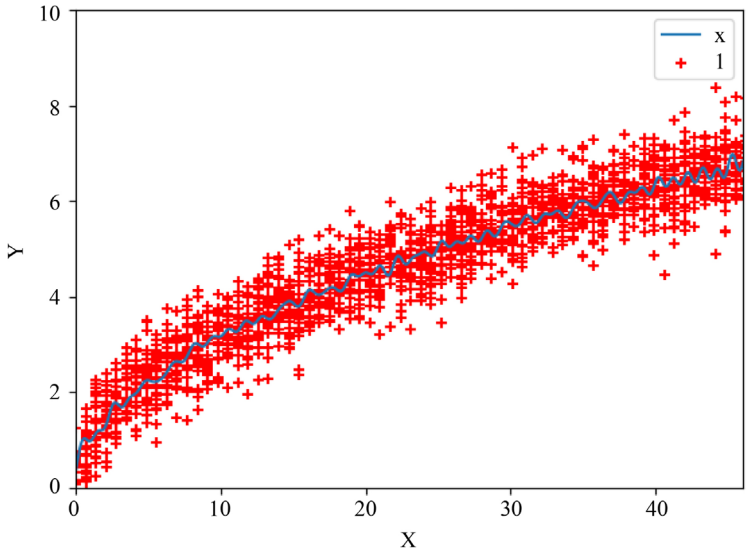

Figure 11. Sinc linear regression test
图 11. 基于 Sinc 的回归测试

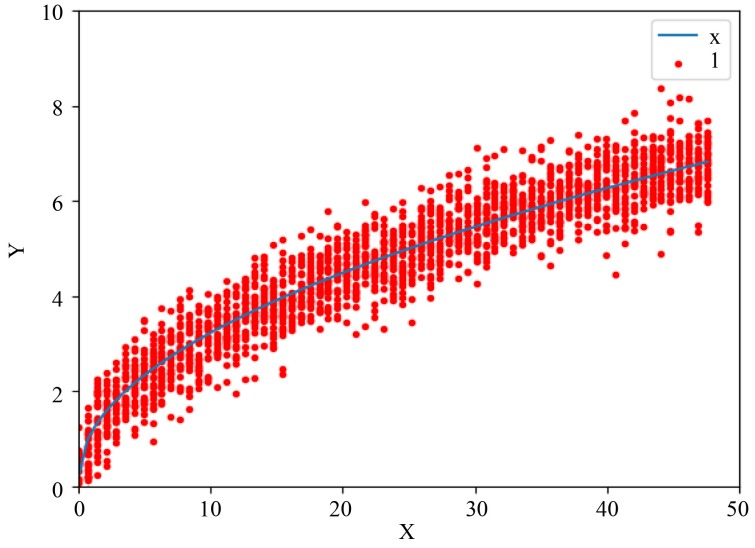

Figure 12. Traditional linear regression
图 12. 传统线性回归

按照基于 Sinc 的逻辑回归算法找到最优的边界曲线上的点，如下图 13 黄色的“*x*”。

边界点乘以对应的边界函数得下图 14：

可以看出基于 Sinc 的逻辑回归算法可以较好的解决数据数量分布相当的线性可分的二分类问题。

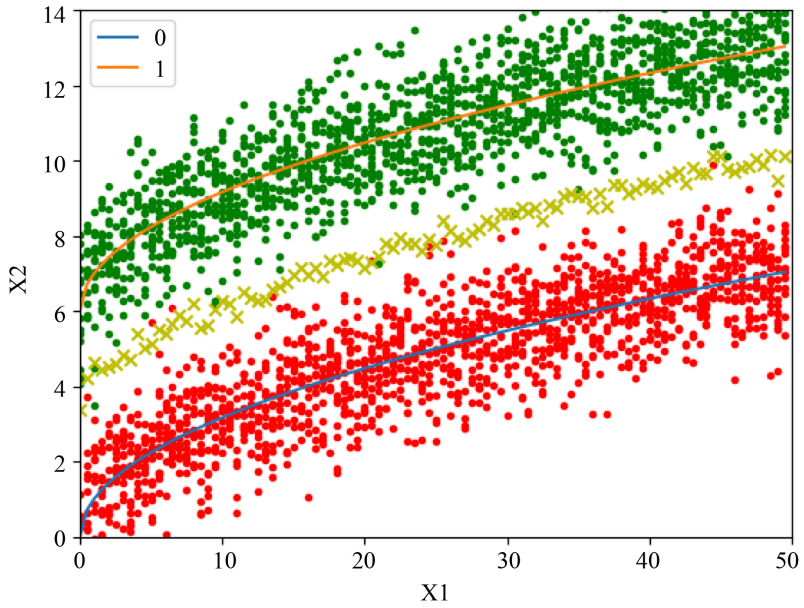

Figure 13. Generate data
图 13. 数据生成

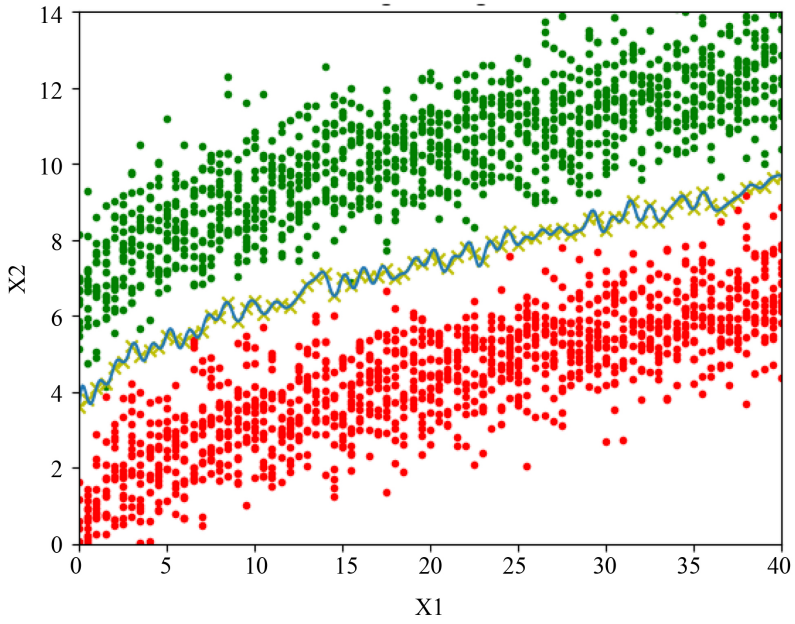

Figure 14. Sinc logistic regression
图 14. 逻辑回归边界函数

4. 基于 Sinc 的回归算法的气候预测

由以上过程可知基于 Sinc 函数的线性回归适合对周期性数据进行预测。而天文气候类数据正好符合(如年太阳黑子活动时间，月降水量等)。我们从 NOAA 拿到 BIJIE 从 1952 年到 2016 年的月最低气温(TMIN)数据，将数据按时间分布排列，如图 15。

图 16 中，使用历年 2 月、4 月、6 月、8 月、10 月 12 月数据为样本数据，对 3 月、5 月、7 月、9 月、11 月的最低温度进行预测。代价函数值为 7.26322704418。

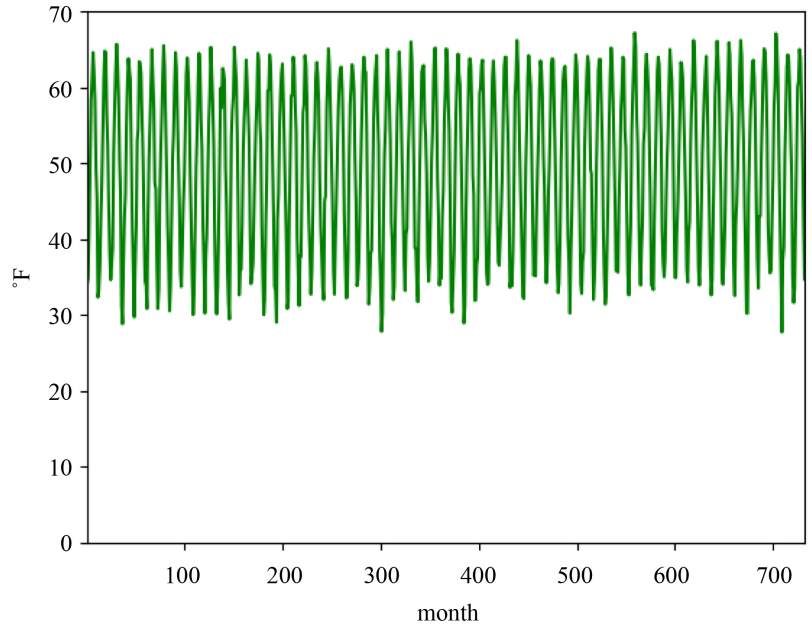

Figure 15. Months minimum temperature of 732 data
图 15. BIJIE 月最低温度的 732 条数据

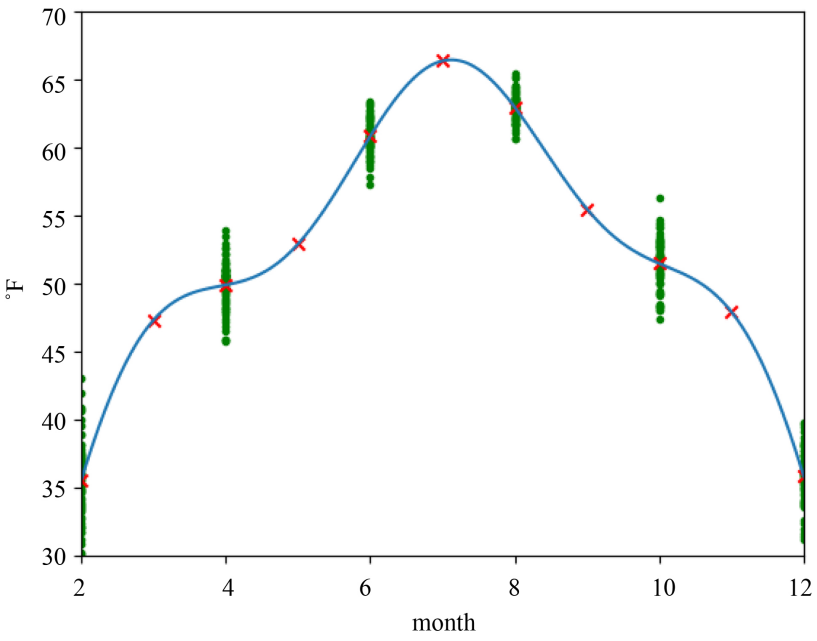

Figure 16. TMIN of BIJIE based on Sinc linear regression
图 16. 基于 Sinc 的线性回归的 BIJIE TMIN 预测

5. 结论

本文研究基于 Sinc 函数的线性与逻辑回归算法，主要工作归纳如下。

1) 在采样定理中离散信号的重构思想启示下，最小化回归函数的均方误差，在数据的自变量域中设计出 Sinc 函数，然后直接重构出回归曲线；

2) 通过理论推导、算法分析与仿真实验，提出基于 Sinc 函数的线性与逻辑回归模型及其算法；

3) 将基于 Sinc 的线性回归算法成功用于月最低气温预测。

致 谢

这项工作由中国国家自然科学基金国际青年科学家基金(NSFC Grant No. 61550110248)和西藏自治区重点科研项目(批准号：Z2014A18G2-13)资助。同时，作者要感谢编辑和审稿人的认真负责。

参考文献

[1] Yahia, M., Hamrouni, T.-A. and Abdelfattah, R. (2017) Infinite Number of Looks Prediction in SAR Filtering by Linear Regression. *IEEE Geoscience & Remote Sensing Letters*, **14**, 2205-2209. https://doi.org/10.1109/LGRS.2017.2749322

[2] Watts, B. 非线性回归分析及其应用[M]. 北京: 中国统计出版社, 1998.

[3] 徐全智, 吕恕. 概率论与数理统计[M]. 第二版. 北京: 高等教育出版社, 2010: 201-202.

[4] Dooley, S.R. and Nandi, A.K. (2000) Notes on the Interpolation of Discrete Periodic Signals Using Sinc Function Related Approaches. *IEEE Transactions on Signal Processing*, **48**, 1201-1203. https://doi.org/10.1109/78.827555

[5] Schanze, T. (1995) Sinc Interpolation of Discrete Periodic Signals. *IEEE Transactions on Signal Processing*, **43**, 1502-1503. https://doi.org/10.1109/78.388863

[6] Hastie, T., Tibshirani, R. and Friedman, J. (2009) The Elements of Statistical Learning. 2nd Edition, Springer, New York.

[7] Cervellera, C. and Macciò, D. (2014) Local Linear Regression for Function Learning: An Analysis Based on Sample Discrepancy. *IEEE Transactions on Neural Networks & Learning Systems*, **25**, 2086-2098. https://doi.org/10.1109/TNNLS.2014.2305193

[8] Ng, A. (2017) Machine Learning. https://www.coursera.org/learn/machine-learning