



基于高斯混合-变分自编码器的轨迹预测算法

张显炆, 朱晓宇, 林浩申, 刘 刚, 安喜彬

(火箭军工程大学 核工程学院, 西安 710025)

摘 要: 海面舰船的轨迹预测对预测精度和实时性具有较高要求, 而舰船轨迹数据特征的高复杂度特性, 导致传统预测算法精度低、耗时长, 难以达到良好的预测效果。为此, 提出一种基于变分自编码器的海面舰船轨迹预测算法。将轨迹坐标数据集转化为轨迹移动矢量集, 使用变分自编码器完成轨迹运动特征的提取与生成预测。同时为提高轨迹预测精度, 将变分自编码网络的隐空间分布设定为混合高斯分布, 使其更符合真实的数据分布特征, 并在隐空间完成轨迹特征的分类, 实现端到端的轨迹预测。仿真结果表明, 相较于传统预测算法 GMMTP 和 VAETP, 该算法的预测误差分别降低了 85.48% 和 35.59%。

关键词: 轨迹预测; 变分自编码器; 混合高斯模型; 无监督学习; 端到端学习

开放科学(资源服务)标志码(OSID):



中文引用格式: 张显炆, 朱晓宇, 林浩申, 等. 基于高斯混合-变分自编码器的轨迹预测算法[J]. 计算机工程, 2020, 46(7): 50-57.

英文引用格式: ZHANG Xianyang, ZHU Xiaoyu, LIN Haoshen, et al. Trajectory prediction algorithm based on Gaussian mixture-variational autoencoder[J]. Computer Engineering, 2020, 46(7): 50-57.

Trajectory Prediction Algorithm Based on Gaussian Mixture-Variational Autoencoder

ZHANG Xianyang, ZHU Xiaoyu, LIN Haoshen, LIU Gang, AN Xibin

(School of Nuclear Engineering, Rocket Force University of Engineering, Xi'an 710025, China)

[Abstract] The trajectory prediction of warships requires high accuracy and real-time performance, but the high complexity of trajectory data features of warships causes the traditional prediction algorithms to be inaccurate and time-consuming, reducing prediction performance. To address the problem, this paper proposes a warship trajectory prediction algorithm based on Variational Autoencoder(VAE). The trajectory coordinate data set is transformed into a trajectory motion vector set, and the trajectory motion features are extracted and generated by using variational autoencoder. Also, in order to improve the prediction accuracy, the hidden space distribution of the variational autoencoding network is set to be mixture Gaussian distribution, which is closer to the features of real data distribution. Then the classification of trajectory features is accomplished in hidden space to implement end-to-end trajectory prediction. Simulation results show that compared with the traditional trajectory prediction algorithms, GMMTP and VAETP, the proposed algorithm can reduce the prediction error rate by 85.48% and 35.59% respectively.

[Key words] trajectory prediction; Variational Autoencoder(VAE); Gaussian Mixture Model(GMM); unsupervised learning; end-to-end learning

DOI: 10.19678/j.issn.1000-3428.0055074

0 概述

轨迹预测的目的是通过分析移动目标的历史轨迹挖掘数据特征, 从而得到轨迹数据模型, 进一步实现轨迹的时空预测^[1]。通过对舰船的历史轨迹数据

建模并研究相应的求解算法进行舰船轨迹预测, 可以为海洋作业如海上搜救、海洋运输等提供重要保障。海面舰船与地面道路车辆的不同主要有两点: 一是轨迹约束不同, 海面舰船轨道约束和速度约束弱, 其运动的随机性较大, 运动模式多样, 运动规律

基金项目: 国防科技“引领”基金(18-163-00-75-004-078-01)。

作者简介: 张显炆(1995—), 男, 硕士研究生, 主研方向为机器学习、行为预测; 朱晓宇, 硕士; 林浩申, 博士; 刘 刚, 教授、博士; 安喜彬, 博士。

收稿日期: 2019-05-30

修回日期: 2019-07-10

E-mail: zhangxyang177@163.com

多变;二是轨迹数据来源方式不同,车辆轨迹数据通常可以通过GPS定位得到,而海面舰船的轨迹一般是通过侦察卫星观测而来,其轨迹是离散的数据点,且受到卫星观测精度的影响,轨迹数据特征进一步复杂化。因此,将传统的地面车辆轨迹预测方法应用于海面舰船上不能得到良好的预测结果,但是海洋作业又对舰船轨迹预测精度和实时性提出较高的要求。

当前轨迹预测常用的方法有卡尔曼滤波器(Kalman Filter, KF)^[2-3]、隐马尔科夫模型(Hidden Markov Model, HMM)^[4]和高斯混合模型(Gaussian Mixture Model, GMM)^[5]。卡尔曼滤波器具有线性、无偏、方差小等特点,被广泛用于预测问题,但是该方法需要做大量的矩阵运算从而缺乏实时性,常用于道路车辆等运动模型简单的目标的轨迹预测。隐马尔科夫模型本质上是对时间序列的预测,但是对噪声没有较好的鲁棒性,并且随着数据集的增大,预测误差会不断累积。高斯混合模型是一种被广泛应用的基于概率统计的预测模型,模型中的轨迹数据由多个高斯过程线性组合产生,通过高斯混合建模并使用最大似然估计(Expectation-Maximization, EM)算法求取模型参数,再通过高斯混合回归实现轨迹预测。文献[6]表明,对于轨迹数据较多的情况, GMM方法在预测精度和实时性上性能均优于KF和HMM方法。目前, GMM方法在车辆轨迹预测^[7-8]、智能交通^[9]、数据聚类^[10-11]等领域都得到了较好的应用。由于车辆的行进轨迹受到道路、速度等诸多约束,其移动规律和数据特征较为简单,因此GMM的预测误差小。但是当数据的复杂度提高时, GMM方法不再具有良好的适用性,并且随着数据集的增大, GMM方法的计算代价也会指数级增加。因此,针对舰船轨迹的特征,亟需研究计算代价低、预测精度高的轨迹预测算法。

轨迹预测的本质是通过训练模型获得模型参数,再利用模型进行预测生成,这与变分自编码器(Variational Autoencoder, VAE)^[12]的设计思想相近。变分自编码器是当前被广泛应用的一种无监督生成模型^[13-14],能够解决无标签的机器学习问题^[15-16]。它通过神经网络将原始的高维数据编码到低维的隐空间,再根据解码网络生成原始数据的逼近值。在经典的变分自编码器中,隐变量的后验分布通常被假设为简单的分布,例如标准正态分布,这导致低维隐空间的表示和对隐空间的特征描述过于简单,一旦原始训练数据集复杂度提高,模型生成的精度将达不到要求。为解决该问题,文献[17-18]通过假设更复杂的先验分布形式来提高模型生成和预测的准确性。如何将VAE方法应用到轨迹预测问题上,并通过合理假设隐空间的先验分布,使其更符合真实的数据分布,是当前需要解决的问题。

本文研究海面舰船的轨迹预测问题,提出一种基于高斯混合先验-变分自编码器(GMVAE)的轨迹预测算法。将轨迹坐标转化为轨迹矢量,并以位移矢量表示预测目标的位置坐标,利用VAE完成轨迹特征的提取与生成预测。针对海面舰船的轨迹数据特点,将隐空间的先验分布假设为混合高斯分布,使其更符合真实的数据特征,同时在隐空间引入类别标签,完成对数据特征的分类。

1 海面舰船轨迹预测模型

1.1 轨迹预测问题描述

海面舰船的轨迹数据通常由卫星观测获得^[19],因此,舰船的轨迹由离散的经纬坐标点构成。本文假设舰船在二维平面上移动,将上述经纬度坐标看作平面坐标,以 T 表示 n 条轨迹组成的数据集:

$$T = \{T_1, T_2, \dots, T_n\} \quad (1)$$

$$T_i = \{(x_{i1}, y_{i1}), (x_{i2}, y_{i2}), \dots, (x_{il}, y_{il})\} \quad (2)$$

其中, T_i 表示第 i 条轨迹, (x_{ij}, y_{ij}) 表示第 i 条轨迹在 j 时刻的坐标点, l 是第 i 条轨迹数据的长度,即该条轨迹包含坐标点的数量。

假设舰船轨迹在 x 和 y 方向上相互独立,在本文中,均以 x 方向上的轨迹预测问题为例, y 方向上的同理可得。设 x 轴方向上的真实轨迹数据集为 X :

$$X = (x_1, x_2, \dots, x_n)^T \quad (3)$$

其中, $x_i = (x_{i1}, x_{i2}, \dots, x_{il})$ 表示第 i 条轨迹, l 是第 i 条轨迹的长度。设在 x 轴方向上预测得到的轨迹数据集为 \hat{X} :

$$\hat{X} = (\hat{x}_1, \hat{x}_2, \dots, \hat{x}_i, \dots, \hat{x}_n)^T \quad (4)$$

其中, $\hat{x}_i = (\hat{x}_{i1}, \hat{x}_{i2}, \dots, \hat{x}_{ik})$ 表示第 i 条轨迹的预测结果, k 表示预测轨迹的长度,即轨迹中包含坐标点的数量。轨迹预测问题可以描述为:

$$\hat{X} = \psi_\lambda(X) \quad (5)$$

其中, ψ_λ 表示轨迹预测模型, λ 表示模型参数。

在通常情况下,轨迹数据集 X 较大,如本文的仿真数据集就包含约10 000条轨迹。KF方法是对每条轨迹进行逐条预测,加上大量的矩阵运算必然导致计算代价高。HMM方法虽然不必逐条预测,但是其预测误差会随着预测长度的递增不断累积,本身的迭代计算复杂度也非常高。因此,上述两种算法均不适用于舰船的轨迹预测。GMM方法虽然可以避免大量的矩阵运算,在一些简单的轨迹预测问题上,其实时性和精度均优于上述两种算法^[6],但是该方法需要计算完整轨迹数据的对数似然关于某个分布的期望,对于复杂度很高的舰船轨迹数据,其耗时会随着数据集的增大呈指数级增长,这在实际的预测中是不可取的。为解决样本数据集增大带来的计算代价增大问题,同时又保证轨迹预测精度,本文提出一种基于变分自编码的轨迹预测算法。

1.2 基于变分自编码器的轨迹预测算法

最简单的变分自编码器包含 3 层神经网络,即输入层、隐层和输出层,如图 1 所示。输入层到隐层是一个编码过程,目的是将原始的数据编码到隐层中。隐层到输出层是一个解码的过程,目的是将隐层中的变量解码得到对应于输入数据 X 的逼近值 \hat{X} 。编码器与解码器通过最小化重构误差^[20]来更新其中的神经网络参数。

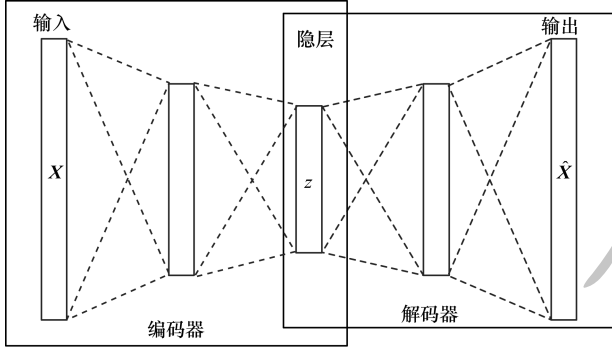


图 1 变分自编码器结构

Fig.1 Structure of variational autoencoder

VAE 首先在经典编码器基础上融入变分贝叶斯思想,通过对原始轨迹数据集的训练,使得高维复杂轨迹输入数据 X 经过编码网络将目标运动特征映射到一个低维隐层概率分布中,然后通过从隐层概率分布中采样,并经过解码网络生成预测轨迹数据 \hat{X} 。可以看出,原始轨迹数据集 X 的复杂度虽然很高,但是经过编码后的隐层分布往往是一个简单的易处理的分布。编码过程实现了数据的降维,并提取到了轨迹数据的关键特征,由此可根据关键特征完成预测生成。相比于 GMM 方法,VAE 避免了对庞大的原始轨迹数据集直接进行运算带来的计算代价高昂的问题,适用于海面舰船的轨迹预测问题。

2 基于改进变分自编码器的轨迹预测算法

本节首先对舰船轨迹数据集进行矢量化和归一化处理,得到轨迹运动矢量集,从而与变分自编码器结合完成轨迹的特征提取与生成预测;然后改进了经典 VAE 的隐空间网络结构,从而进一步提高轨迹预测精度。

2.1 海面舰船轨迹数据预处理

2.1.1 轨迹矢量集转化

VAE 进行轨迹预测的本质是使输出 \hat{X} 尽可能地逼近输入 X 。假设输入轨迹为位置序列 X ,其输出便是与训练集轨迹位置序列相似的预测轨迹,这种预测方法没有用到目标当前的位置信息,即预测生成的轨迹与当前目标的位置没有任何关联,并且在算法中难以使用测试轨迹数据评价算法性能。因

此,本文将舰船的轨迹数据集转化为运动矢量集作为编码器的输入,将预测舰船下一时刻的坐标转化为预测舰船在下一时刻的位移。对轨迹数据集 X 进行如下转化:

$$V = \{V_1, V_2, \dots, V_n\} \quad (6)$$

其中, $V_i = (x_1^i, x_2^i, \dots, x_{i-1}^i)$ 表示第 i 条轨迹在 x 轴方向上的矢量集,且 $x_j^i = (x_{i(j+1)} - x_{ij})$ 。利用 VAE 完成若干个时刻间舰船位移的预测,然后将预测结果按时间序列进行累加,还原出预测的轨迹位置序列,即可完成轨迹预测。

2.1.2 数据归一化

为使 VAE 的训练过程能够收敛到最优解,需要对上述矢量集数据进行归一化处理。归一化公式如下:

$$x_{\text{normal}} = \frac{1}{x_{\text{max}} - x_{\text{min}}} (x - x_{\text{min}}) \quad (7)$$

其中, x_{max} 、 x_{min} 为输入轨迹数据的最大值和最小值。按照上述归一化公式,可以得到模型的输入 V_{normal} 。

2.2 VAETP 算法

在基于经典 VAE 的轨迹预测算法 (VAETP) 中,编码网络将轨迹数据映射到隐空间 Z 以提取数据特征,得到隐变量 z 的后验分布。引入标准高斯分布作为 z 的先验分布,通过最小化后验与先验分布之间的 KL 距离,达到训练编码网络参数的目的。预测的轨迹数据通过从 z 中采样和解码获得。在采样过程中,为了能够进行参数更新,使用了参数重构^[11]方法,对于小样本轨迹数据可以获得良好的预测结果。VAETP 网络架构如图 2 所示。

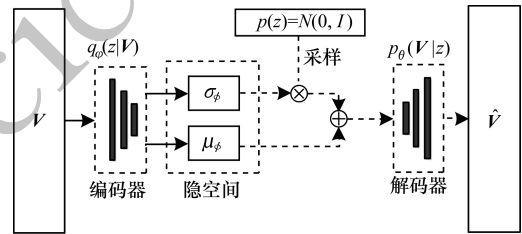


图 2 VAETP 网络架构

Fig.2 Network architecture of VAETP

VAE 的基本思想是假定轨迹数据 V 是对某个随机过程采样得到的,每条轨迹样本由隐变量 z 决定。轨迹数据的生成过程为:首先从先验概率分布 $p_\theta(z)$ 中采样得到样本 z ,然后根据条件概率分布函数 $p_\theta(V|z)$ 解码得到轨迹样本,但是生成过程中解码参数 θ 和隐变量 z 的信息未知,并且无法计算隐变量 z 的后验概率 $q_\phi(z|V)$ 。所以,VAE 统一用神经网络进行拟合。根据边缘概率公式,似然函数^[12]可以写为:

$$\log_a p(V) = D_{\text{KL}}[q_\phi(z|V) || p_\theta(z|V)] + L(\phi, \theta; V) \quad (8)$$

$$L(\phi, \theta; V) = E_{q_\phi}[-\log_a q_\phi(z|V) + \log_a p_\theta(z, V)] = -D_{\text{KL}}(q_\phi(z|V) || p_\theta(z)) + E_{q_\phi}[\log_a p_\theta(V|z)] \quad (9)$$

其中, $L(\varphi, \theta; \mathbf{V})$ 为变分下界,也是 VAE 网络的损失函数; φ, θ 为编码和解码网络的参数。在式(9)中:第1项称为 KL 散度项^[21],用于衡量后验分布与先验分布的相似程度,通过优化该项可以使得后验 $q_\varphi(z|\mathbf{V})$ 逼近先验 $p_\theta(z)$,从而防止模型过拟合;第2项是一个对数似然估计,相当于自编码器中的重构误差损失函数,用于重构原始样本数据。由于 KL 散度的非负性,最大化似然函数问题转化为了最大化变分下界的问题。

VAETP 运用 Adam 算法^[22]进行编码和解码网络参数的更新。编码网络的参数 φ 更新过程如式(10)~式(13)所示,解码网络的参数 θ 更新过程同理可得。

$$v_i = \beta_1 \times v_{i-1} - (1 - \beta_1) \times g_i \quad (10)$$

$$s_i = \beta_2 \times s_{i-1} - (1 - \beta_2) \times g_i^2 \quad (11)$$

$$\Delta \varphi_i = -\eta \frac{v_i}{\sqrt{s_i + \varepsilon}} \times g_i \quad (12)$$

$$\varphi_{i+1} = \varphi_i + \Delta \varphi_i \quad (13)$$

其中, η 为初始的学习率, $\beta_1, \beta_2, \varepsilon$ 为超参数, g_i 为 φ 在第 i 次训练时的梯度, v_i 为 g_i 的指数加权平均, s_i 为 g_i^2 的指数加权平均。通过式(10)和式(11)分别得到参数 φ 的梯度指数平均和梯度平方的指数平均,通过式(12)得到第 i 次训练时梯度的变化量,通过式(13)得到参数 φ 更新以后的值。当变分下界收敛时,编码网络的输出是提取的目标轨迹特征,解码网络的输出是生成的预测轨迹,并且轨迹预测长度可以通过设置解码网络输出层的节点数进行调整。

当数据集的复杂度提高,如轨迹数据包含若干种运动模式,或者 \mathbf{V} 的分布是复杂的高维分布时,映射得到的隐空间复杂度随之提高。此时式(9)中 z 的先验分布对隐空间的描述不能完全刻画原始数据隐空间的特征,从而导致 VAETP 算法进行轨迹生成和预测精度不够准确。对此,通常的解决方法是先对数据进行分类,再针对每个类别的轨迹数据分别使用 VAE 训练,最终得到若干个网络。轨迹生成和预测的过程,首先判断轨迹数据所属的类别,再从该类别隐空间中采样得到预测的轨迹。但在实际中,原始轨迹数据往往很难实现有效分类,而且该方法模型的整体契合度不高^[23]。

本文提出的轨迹预测算法,通过调整隐空间的先验分布形式使其更符合真实的分布,从而有效地解决 VAETP 算法采用固定高斯先验缺少变异性的问题,提高预测的精度。该方法不需要事先对轨迹数据进行分类处理,而是直接将轨迹矢量数据编码到隐空间,在隐空间完成轨迹数据的分类,再经过解码得到预测的轨迹矢量,实现端到端的轨迹预测。

2.3 基于 GMVAE 的轨迹预测算法

本节设计了基于 GMVAE 的轨迹预测算法。通过 2.2 节的分析可以看出,在 VAETP 方法中,隐空

间的先验假设过于简单,难以逼近真实的轨迹特征,需要用更复杂的分布来描述。而混合高斯分布几乎可以近似任意复杂的概率分布^[24],所以,将隐空间的先验分布假设为混合高斯分布,可以更准确地描述真实的轨迹特征。

GMVAE 的网络结构如图3所示,其中,实线箭头表示编码过程,虚线箭头表示解码过程。在编码过程中,轨迹数据 \mathbf{V} 首先经过编码网络 $q(y|\mathbf{V}; \varphi_y)$ 得到分类标签 y ,用来描述轨迹的类别,然后将标签 y 和原始轨迹数据 \mathbf{V} 共同编码到隐空间 z ,实现轨迹的分类和轨迹特征的提取;在解码过程中,根据轨迹类别 y 从对应的隐空间 z 中采样,再解码生成预测的轨迹 $\hat{\mathbf{V}}$ 。

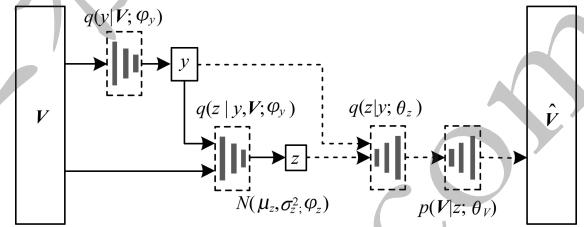


图3 GMVAE 网络架构

Fig.3 Network architecture of GMVAE

编码和解码过程中变量之间的关系如图4所示,其中, \mathbf{V} 表示轨迹数据, y 代表类别标签, z 代表隐变量, φ_y, φ_z 为编码网络参数, θ_z, θ_v 为解码网络参数。

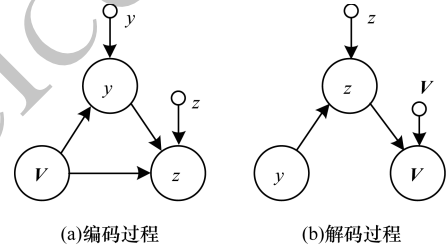


图4 GMVAE 编码和解码过程中的变量关系

Fig.4 Variable relationship of GMVAE encoding process and decoding process

隐变量 y, z 编码过程表示为:

$$q_\varphi(y, z|\mathbf{V}) = q(y|\mathbf{V}; \varphi_y) q(z|\mathbf{V}, y; \varphi_z) \quad (14)$$

其中, $q(y|\mathbf{V}; \varphi_y)$ 表示 y 的编码过程, $q(z|\mathbf{V}, y; \varphi_z)$ 表示 z 的编码过程。由此,原始轨迹数据经过编码网络可以得到隐变量 z 的后验分布:

$$z|y, \mathbf{V} \sim N(\mu_z, \sigma_z^2; \varphi_z) \quad (15)$$

在解码过程中, \mathbf{V}, y 和 z 的联合概率分布可以描述为:

$$\begin{cases} p(\mathbf{V}, y, z) = p(y)p(z|y; \theta_z)p(\mathbf{V}|z; \theta_v) \\ y \sim \text{Cat}(\boldsymbol{\pi}) \\ z|y \sim N(\mu(y; \theta_z), \sigma^2(y; \theta_z)) \\ \mathbf{V}|z \sim B(\mu(z; \theta_v)) \end{cases} \quad (16)$$

其中, y 服从分类分布, 其参数 π 是一个向量 $\pi = (\pi_1, \pi_2, \dots, \pi_K)$, π_i 表示某条轨迹属于第 i 个类别的概率, K 为总的类别个数。每个类别对应的隐空间先验 $p(z|y; \theta_z)$ 是均值为 $\mu(y; \theta_z)$ 、方差为 $\sigma^2(y; \theta_z)$ 的高斯分布。因此, 隐变量 z 的分布为:

$$z \sim \sum_{i=1}^K \pi_i N(\mu_{zi}, \sigma_{zi}^2) \quad (17)$$

可以看出, 隐变量 z 的先验实际上是一个混合高斯分布。不同的轨迹类别映射到隐空间后对应着不同的高斯分量, 每个分量的系数即为 y 分布的参数 π 。生成模型 $p(V|z; \theta_v)$ 通常假设为伯努利分布, 损失函数为:

$$\begin{aligned} L(\theta, \varphi; V) &= E_{q(y, z|V)} [\ln p(V, y, z) - \ln q(y, z|V)] = \\ &E_{q(y, z|V)} \left[\ln \frac{p(y)}{q(y|V)} + \ln \frac{p(z|y)}{q(z|V, y)} + \ln p(V|y, z) \right] \end{aligned} \quad (18)$$

其中, $\ln \frac{p(y)}{q(y|V)}$ 可看作标签损失项, 用于衡量生成的轨迹类别与先验的差异, $\ln \frac{p(z|y)}{q(z|V, y)}$ 和 $\ln p(V|y, z)$ 分别是 KL 散度项和重构项。得到变分下界即损失函数后, 利用式 (10) ~ 式 (13) 描述的 Adam 算法更新参数, 当变分下界收敛时, 从 y 中即可得到轨迹数据的标签类别, 从隐变量 z 中可以得到轨迹特征, 而编码输出即为各个类别的预测轨迹。

3 算法设计与仿真

本节设计 GMVTP 算法流程, 并通过与传统算法 GMMTP、VAETP 进行比较, 检验其算法性能。

3.1 算法流程

轨迹数据经过预处理后, 得到的轨迹矢量集 V_{normal} 作为 VAE 网络的输入。训练过程中选取 $n = 400$ 条轨迹, 每条轨迹包含 200 个轨迹点, 设定轨迹类别总数为 $K = 3$ 。本文使用小批量 (mini_batch) [25] 方法进行训练, 将整个训练集划分成若干个小的子训练集, 用 $V_minibatch$ 表示, 每个子集的大小 $batchsize = m$ 。参数更新过程可以表示为:

$$\theta_j = \theta_j - \frac{\partial L}{\partial \theta_j} = \theta_j - \frac{1}{m} \sum_{i=1}^m \frac{\partial L(V_i)}{\partial \theta_j} \quad (19)$$

算法每次只在一个 $V_minibatch$ 上更新梯度, 而不用遍历完整的数据集, 大幅减少了算法收敛所需的迭代次数, 而且可以实现并行化计算。

算法 1 GMVTP 算法

输入 轨迹矢量集 $V_{\text{normal}} = \{V_1, V_2, \dots, V_n\}$

输出 网络参数 φ, θ

初始化 随机初始化网络参数 φ, θ

For $i = 1 : epochs$

For $j = 1 : n/m$

```

V_minibatch ← {V_j, V_{j+1}, ..., V_{j+m-1}} // 取 m 条轨迹数据
For k = 1 : m
    y ← q(y|V_k; φ_y) // 通过编码网络 φ_y 生成标签 y
    μ_z, σ_z, z ← q(z|y, V_k; φ_z) // 通过编码网络 φ_z 生成 z 的后验
    z ← N(μ(y; θ_z), σ^2(y; θ_z)) // 根据 y 从相应的先验高斯
    // 分布中采样得到 z
    V_k | z ← B(μ(z; θ_v)) // 根据 z 解码出轨迹 V_k
    L(θ, φ; V_k) // 根据式 (17) 计算损失函数值
    g_k ← g_k - ∇_{φ, θ} L(φ, θ; V_k) // 计算梯度
End for
g_j ← g_k / m // 单个 batch 数据训练后的梯度更新
φ_j, θ_j ← φ_j, θ_j // 根据式 (10) ~ 式 (13) 进行参数更新
End for
End for

```

轨迹生成模型 $p(V|z; \theta_v)$ 的神经网络输入层和输出层节点数 200, 其他网络输出层节点设置为 64, 迭代次数为 epochs。轨迹预测过程为将已知的轨迹前 $k' (k' < 200)$ 个点作为网络输入, 经过编码可以得到其类别标签 y , 再根据标签 y 从对应的隐空间 z 中采样解码生成完整的轨迹, 即可得到预测出的第 $k' + 1$ 个 ~ 第 200 个轨迹点。

3.2 仿真条件设置

本文使用的轨迹数据集是 MIT Trajectory Data。该数据集来源于卫星采集数据, 移动目标是在平面上中速移动的机动车辆。因为该轨迹数据在采集类型以及目标运动速度、自由度、路径约束等方面与真实的舰船轨迹类似, 所以通过与真实舰船轨迹数据的仿真结果比较, 该数据集的仿真结果能够较为真实地反映算法的性能。本文的仿真环境配置为 CPU Intel® Core™ i5-8300H @ 2.3 GHz, 内存 8 GB, 显卡 NVIDIA GeForce GTX 1050Ti, 操作系统为 Ubuntu 16.04。训练过程中的超参数设置如表 1 所示。

表 1 训练过程超参数设置
Table 1 Setting of hyperparameters in training process

超参数	参数值
β_1	0.9
β_2	0.999
ε	10^{-8}
η	0.001
$\pi_i = 1/K$	1/3
m	20
epochs	300

3.3 仿真结果分析

3.3.1 轨迹训练结果

本节从轨迹数据集中选取 600 条轨迹作为训练集。图 5 为 600 条轨迹经过矢量化和归一化后的结果, 从中可以看出, 目标运动轨迹主要包含了左高右低、左低右高、中间高两边低 3 种情况。这 3 种轨迹对应着隐空间的 3 类特征, 在仿真中给定 $K = 3$ 。图 6 为 VAETP 经过训练后从隐空间采样得到的

120 条轨迹,从中可以看出,VAE 虽然能够生成和预测轨迹,但是相对于原始轨迹,3 种运动特征之间的区别不明显,这是因为在 VAE 中,轨迹数据被视为从一个标准正态分布中采样得到。图 7 为 GMVTP 的生成预测结果,从中可以看出,3 类轨迹之间区分更加明显,正是由于隐空间分布是混合高斯分布,因此能够更准确地描述轨迹特征。图 8 为随机选取的 120 轨迹使用 GMVTP 进行分类的结果,不同深浅线条分别代表不同特征的轨迹,从中可以看出,GMVTP 可基本实现轨迹的无监督分类。

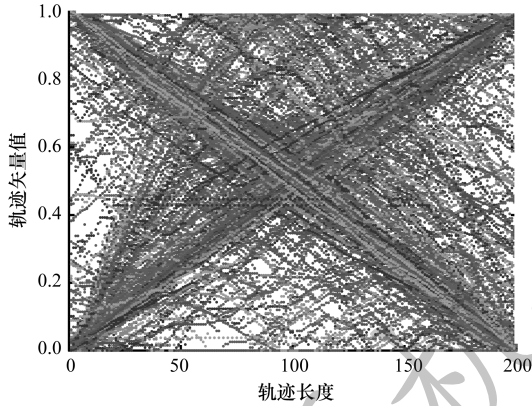


图 5 训练轨迹矢量集

Fig. 5 Vector set of training trajectories

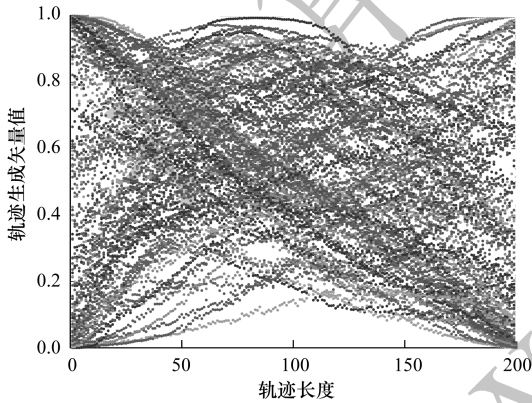


图 6 VAETP 算法的轨迹生成结果

Fig. 6 Trajectory generation result of VAETP algorithm

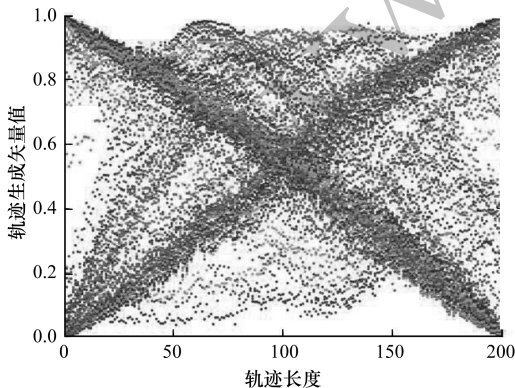


图 7 GMVTP 算法的轨迹生成结果

Fig. 7 Trajectory generation result of GMVTP algorithm

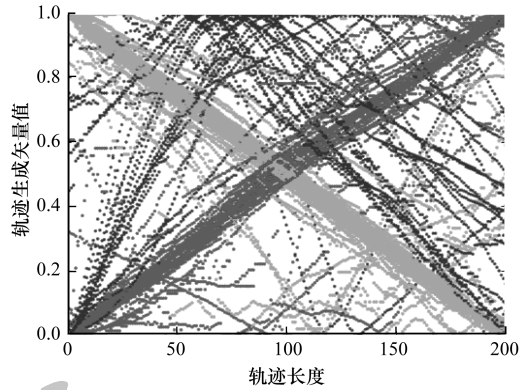


图 8 GMVTP 算法的轨迹分类结果

Fig. 8 Trajectory classification result of GMVTP algorithm

3.3.2 算法性能比较

本节比较 GMVTP 和 VAETP 2 种算法在相同训练集大小(800 条轨迹)下的收敛性,迭代步数设置为 300。图 9 为损失函数变化曲线。从中可以看出,VAETP 在迭代 150 步左右收敛,而 GMVTP 在 50 步左右即收敛。损失函数值间接反映出生成轨迹与原始轨迹之间的差异性,由于 GMVTP 的隐空间结构被假设为混合高斯分布,在训练开始阶段就能够精确近似真实的分布,因此 GMVTP 的损失函数值始终小于 VAETP 并很快收敛。

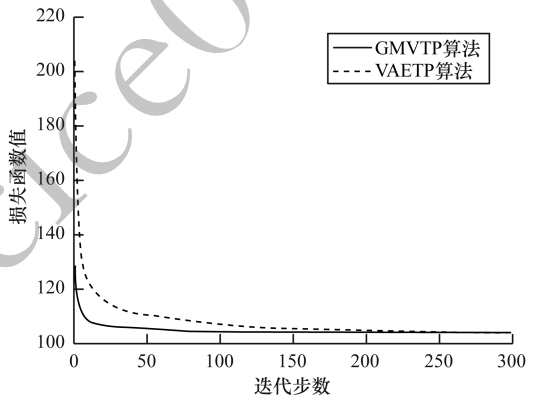


图 9 损失函数变化曲线

Fig. 9 Change curves of loss function

3.3.3 预测精度和时间比较

采用均方根误差(RMSE)计算轨迹预测误差:

$$\text{RMSE} = \sqrt{\frac{1}{k} \sum_{i=1}^k (x_i - \hat{x}_i)^2}$$

其中, x_i 为轨迹的真实值, \hat{x}_i 为预测值, k 为预测轨迹点的数量。在仿真中给定轨迹前 100 个点的值,预测后 100 个点的值即 $k = 100$ 。图 10 给出了 3 种预测算法在不同测试集大小情况下的预测误差。可以看出,相对于 GMMTP, VAETP 的 RMSE 平均降低了 41.19%,原因在于 VAETP 中的编码网络提取到了原始轨迹的特征,从而能够更精确地生成预测轨

迹,而 GMVTP 的轨迹预测误差在三者中最小,相对于 GMMTP 平均降低了85.48%,说明 GMVTP 显著提高了轨迹预测的精度。随着训练集轨迹数量增多,GMMTP 算法的预测误差会减小,这是由于历史轨迹数据的增多引入了的更多先验,使得预测误差不断降低。VAETP 有略微上升是因为随着训练集中轨迹数量的增多,原始数据的复杂度逐渐提高,其隐空间真实分布的复杂度也随即提高,而经典 VAE 由于隐空间采用了标准正态分布的先验假设,对真实分布的拟合度必然下降,进一步导致生成预测误差的增加。相比之下,GMVTP 由于隐空间的先验是混合高斯分布,能够更准确近似真实分布,预测误差几乎不会受到训练集大小影响,始终处于较低水平。

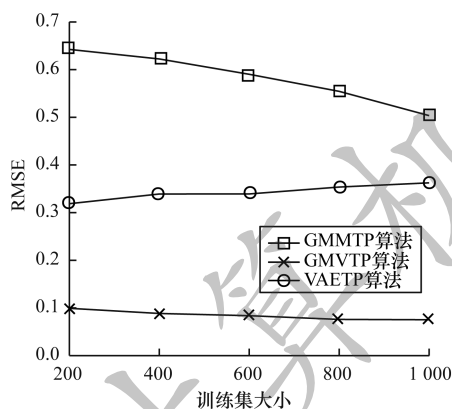


图 10 3 种算法的预测误差比较

Fig. 10 Prediction error comparison of three algorithms

为比较 3 种算法的实时性,计算在相同收敛条件下不同训练集大小时算法的运行时间,实验结果取每种条件下训练 30 次的平均耗时,如图 11 所示。可以看出,GMMTP 的平均训练耗时约为 21 s,最长达到 27 s,而 VAETP 和 GMVTP 两种算法的平均耗时在 4 s 左右。相对于 GMMTP 算法,后两种算法的训练耗时平均缩短了 80.3% 和 74.9%,并且随着训练集的增大,GMMTP 的训练时间显著增大,而 VAETP 和 GMVTP 两种算法的训练时间无明显增长。这是由于变分自编码器采用了随机梯度下降法更新网络参数,避免了 GMMTP 算法中有关概率分布期望的计算,同时 TensorFlow 框架下的 GPU 与程序之间的联系加速了网络参数的训练。因此,当轨迹数据集庞大时,基于 VAE 的轨迹预测算法优势更加明显。从图中还可以看出,GMVTP 的训练时间略多于 VAETP,这是由于 GMVTP 的网络结构更加复杂,导致训练时间略微增长,但从上文的仿真可知,GMVTP 的预测误差相比于 VAETP 平均降低了 35.59%,总体而言其性能优于 VAETP。

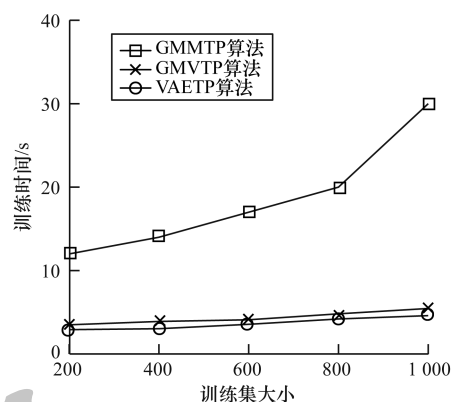


图 11 3 种算法训练时间比较

Fig. 11 Training time comparison of three algorithms

综上所述,本文算法在预测精度和实时性上均优于传统的 GMM 轨迹预测算法,有利于实现舰船轨迹的在线预测,并且由于隐空间先验分布复杂度提高,该算法能够更精确地描述轨迹数据的特征,进一步减小预测误差。

4 结束语

海面舰船由于路径约束小,移动规律多变,且受到观测手段的影响,其轨迹数据特征较为复杂。为此,本文提出一种基于高斯混合-变分自编码器的轨迹预测算法 GMVTP。将轨迹坐标转化为轨迹矢量,利用 VAE 提取轨迹特征并生成预测结果。针对海面舰船的轨迹数据特点,改进经典的 VAE 算法,将隐空间的先验分布假设为混合高斯分布,从而使其更符合真实的数据特征分布,并在隐空间中完成轨迹特征分类,实现端到端的轨迹预测。仿真结果表明,该算法在预测精度和实时性上优于传统的 GMM 模型,有利于舰船轨迹数据的在线训练与预测,并且提高了轨迹预测精度。下一步将考虑解决类别总数 K 的自适应问题以及各个类别概率向量 π 的训练问题。

参考文献

- [1] SUN Hong, CHEN Suo. Spatio-temporal trajectory prediction algorithm based on clustering based hidden Markov model[J]. Journal of Chinese Computer Systems, 2019, 40(3): 472-476. (in Chinese)
孙红,陈锁.一种聚类隐马尔可夫模型的时空轨迹预测算法[J].小型微型计算机系统,2019,40(3): 472-476.
- [2] CHENG Quan, WANG Chaoli. A method of trajectory prediction based on Kalman filtering algorithm and support vector machine algorithm[C]//Proceedings of 2017 Chinese Intelligent Systems Conference. Berlin, Germany: Springer, 2017: 495-504.
- [3] JIANG Jiaxuan, HUANG Chunlin. Tracking fast steering motion of human target based on adaptive Kalman filter

- algorithm[J]. Modern Radar, 2019, 40(7): 54-60. (in Chinese)
- 姜嘉璇, 黄春琳. 人体快转向运动自适应卡尔曼跟踪算法[J]. 现代雷达, 2019, 40(7): 54-60.
- [4] LASSOUED Y, MONTEIL J, GU Y, et al. A hidden Markov model for route and destination prediction[C]//Proceedings of the 20th IEEE International Conference on Intelligent Transportation Systems. Washington D. C., USA: IEEE Press, 2018: 126-135.
- [5] LIN Yi, ZHANG Jianwei, WU Xiping, et al. Study on algorithm for flight trajectory prediction based on GMM[J]. Journal of Sichuan University (Engineering Science Edition), 2018, 50(4): 108-113. (in Chinese)
- 林毅, 张建伟, 武喜萍, 等. 基于 GMM 的航班轨迹预测算法研究[J]. 工程科学与技术, 2018, 50(4): 108-113.
- [6] QIAO Shaojie, JIN Kun, HAN Nan, et al. Trajectory prediction algorithm based on Gaussian mixture model[J]. Journal of Software, 2015, 26(5): 1048-1063. (in Chinese)
- 乔少杰, 金琨, 韩楠, 等. 一种基于高斯混合模型的轨迹预测算法[J]. 软件学报, 2015, 26(5): 1048-1063.
- [7] HU Zhenzhen. Research on trajectory prediction of moving vehicles in uncertain environment[D]. Changsha: Changsha University of Science and Technology, 2017. (in Chinese)
- 胡珍珍. 不确定环境下移动车辆轨迹预测研究[D]. 长沙: 长沙理工大学, 2017.
- [8] XIA Zhuoqun, HU Zhenzhen, LUO Junpeng. EAVTP: an environment adaptive vehicle trajectory prediction method[J]. Journal of Chinese Computer Systems, 2016, 37(10): 2375-2379. (in Chinese)
- 夏卓群, 胡珍珍, 罗君鹏. EAVTP: 一种环境自适应车辆轨迹预测方法[J]. 小型微型计算机系统, 2016, 37(10): 2375-2379.
- [9] KUANG Xianyan, WANG Chengkun, XU Lunhui. Video-based two-wheel vehicle detection for mixed traffic flow based on combination foreground extraction[J]. Journal of Transportation Systems Engineering and Information Technology, 2014(5): 49-54, 73. (in Chinese)
- 邝先验, 王成坤, 许伦辉. 基于组合前景提取的混合交通两轮车辆视频检测[J]. 交通运输系统工程与信息, 2014(5): 49-54, 73.
- [10] GE Lige, SUN Wei. Research on floating cluster trajectory clustering of moving object in adaptive Gaussian mixture model[J]. Modern Computer, 2018(23): 3-8. (in Chinese)
- 葛丽阁, 孙伟. 自适应高斯混合模型海上移动对象浮标轨迹聚类研究[J]. 现代计算机, 2018(23): 3-8.
- [11] JING Xiaogang, GE Lige, SUN Wei. A buoy trajectory clustering algorithm based on Gaussian mixture model[J]. Modern Computer, 2017(36): 3-5, 8. (in Chinese)
- 荆晓刚, 葛丽阁, 孙伟. 一种基于高斯混合模型的海上浮标轨迹聚类算法[J]. 现代计算机, 2017(36): 3-5, 8.
- [12] KINGMA D P, WELING M. Auto-encoding variational Bayes[EB/OL]. [2019-04-30]. <https://arxiv.org/abs/1312.6114>.
- [13] ZHANG Pengsheng. The generative adversarial network based on variational auto-encoder for face frontalization[J]. Software Guide, 2018, 17(12): 48-51. (in Chinese)
- 张鹏升. 基于变分自编码器的人脸正面化产生式模型[J]. 软件导刊, 2018, 17(12): 48-51.
- [14] HUANG Guojie, JIN Hui, YU Yibiao. Voice conversion using non-parallel corpora based on enhanced variation auto-encoder[J]. Journal of Signal Processing, 2018, 34(10): 108-113. (in Chinese)
- 黄国捷, 金慧, 俞一彪. 增强变分自编码器做非平行语料语音转换[J]. 信号处理, 2018, 34(10): 108-113.
- [15] WALKER J, DOERSCH C, GUPTA A, et al. An uncertain future: forecasting from static images using variational autoencoders[C]//Proceedings of European Conference on Computer Vision. Berlin, Germany: Springer, 2016: 835-851.
- [16] PU Y, GAN Z, HENAO R, et al. Variational autoencoder for deep learning of images, labels and captions[C]//Proceedings of NIPS'16. Washington D. C., USA: IEEE Press, 2016: 2352-2360.
- [17] LI Peng. Variational automatic encoder based on Gaussian mixture model[D]. Harbin: Harbin Institute of Technology, 2017. (in Chinese)
- 李鹏. 基于高斯混合模型的变分自动编码器[D]. 哈尔滨: 哈尔滨工业大学, 2017.
- [18] WANG L, SCHWING A, LAZEBNIK S. Diverse and accurate image description using a variational auto-encoder with an additive Gaussian encoding space[C]//Proceedings of NIPS'17. Washington D. C., USA: IEEE Press, 2017: 5756-5766.
- [19] GAO Yilun. Index and prediction of ship simulation trajectory data[D]. Harbin: Harbin Engineering University, 2014. (in Chinese)
- 高翊伦. 舰船仿真轨迹数据索引及预测研究[D]. 哈尔滨: 哈尔滨工程大学, 2014.
- [20] TITOV I, KHODDAM E. Unsupervised induction of semantic roles within a reconstruction-error minimization framework[C]//Proceedings of 2015 Conference of the North American Chapter of the Association for Computational Linguistics. Denver, USA: Association for Computational Linguistics, 2015: 1-10.
- [21] LECUN Y, BENGIO Y, HINTON G. Deep learning[J]. Nature, 2015, 521(7553): 436-444.
- [22] GOODFELLOW L, BENGIO Y, COURVILLE A. Deep learning[M]. Translated by ZHAO Shenjian, LI Yujun, FU Tianfan, et al. Beijing: Posts and Telecom Press, 2017. (in Chinese)
- GOODFELLOW L, BENGIO Y, COURVILLE A. 深度学习[M]. 赵申剑, 黎彧君, 符天凡, 等译. 北京: 人民邮电出版社, 2017.
- [23] ZHU Zisheng. Research and implementation of vehicle trajectory prediction algorithm based on neural network[D]. Xi'an: Xidian University, 2018. (in Chinese)
- 朱自升. 基于神经网络的车辆轨迹预测算法的研究与实现[D]. 西安: 西安电子科技大学, 2018.
- [24] BISHOP C M. Pattern recognition and machine learning (information science and statistics) [M]. New York, USA: Springer-Verlag New York, Inc., 2006.
- [25] METEL M R. Mini-batch stochastic gradient descent with dynamic sample sizes[EB/OL]. [2019-04-30]. <https://www.researchgate.net/publication/>.