# 第一章 题目说明

## 1.1原题展示

如图所示的单自由度框架结构， 可以通过动力加载试验分析该结构的动力特征，其中该结构质量 ，总刚度 ，阻尼系数 。试验中通过助动器给该结构施加(kN)的正弦荷载作用力，并通过传感器收集结构的位移响应，假设采样频率为，时程。

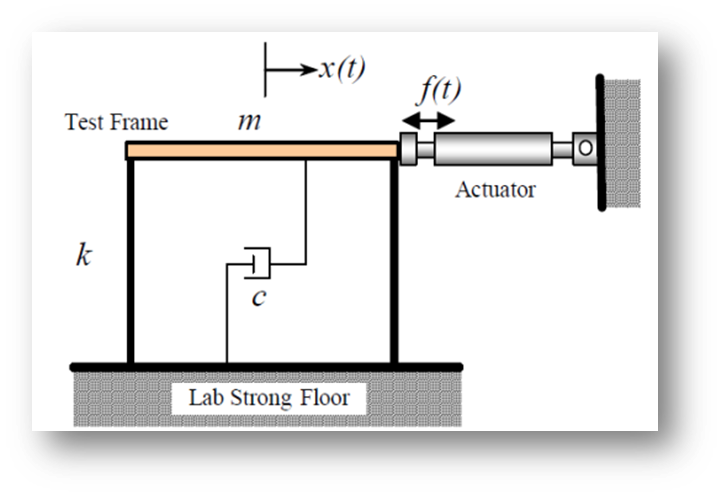


图1-1 单自由度框架结构

**问题一：**根据收集的试验数据（data\_exp），建立合适的神经网络学习该结构的动力特征，从而实现对应输入荷载下结构位移的预测。已知试验中助动器施加的正弦荷载对应的参数𝐹0和𝑓分别为 𝐹0=[110, 120, 130, 140, 150, 160, 170,180, 190, 200] kN 和𝑓 =[0.5000, 0.7000,0.9000, 1.1000,1.3000, 1.5000, 3.0000, 3.2000, 3.4000, 3.6000,3.8000, 4.0000] Hz。

**问题二：**基于（1）中训练的深度学习模型，预测𝐹=[50, 300] kN 和𝑓=[0.4000, 2.0000,2.5000, 5.0000] Hz 对应的正弦荷载激励下的结构位移（data\_pred1），并对预测结果好坏进行分析。（提示：计算结构频率）

**问题三：**基于（1）中训练的深度学习模型，分别预测结构在𝐹(𝑡) = 200 sin(6𝜋𝑡)和白噪声荷载激励下的结构位移（ data\_pred2）。

## 1.2解题思路：

题目要求根据给定的试验数据，建立适当的神经网络模型，学习单自由度框架结构的动力特征，并用于预测在不同输入荷载下的结构位移。给定的试验数据包括助动器施加的正弦荷载的参数，即施加荷载的振幅𝐹0和频率𝑓。可以按照以下步骤进行：

（1）准备数据：根据题目中给定的试验数据，收集结构的位移响应𝑥(m)和对应的输入荷载𝐹(𝑡)。

（2）数据预处理：将试验数据进行预处理，包括归一化等步骤，以便于神经网络的训练和预测。

（3）神经网络模型选择：根据题目中的要求，选择适当的神经网络模型来学习结构的动力特征。可以考虑使用深度学习中的回归模型，如多层感知机（MLP）或长短期记忆网络（LSTM）等。

（4）数据划分：将预处理后的数据划分为训练集和验证集，用于训练和评估模型的性能。

（5）模型训练：使用训练集进行神经网络模型的训练，通过迭代优化模型参数，使其能够准确地学习输入荷载与结构位移之间的映射关系。

（6）模型评估：使用验证集评估模型的性能，计算预测位移与实际位移之间的误差指标，如均方根误差（RMSE）等。

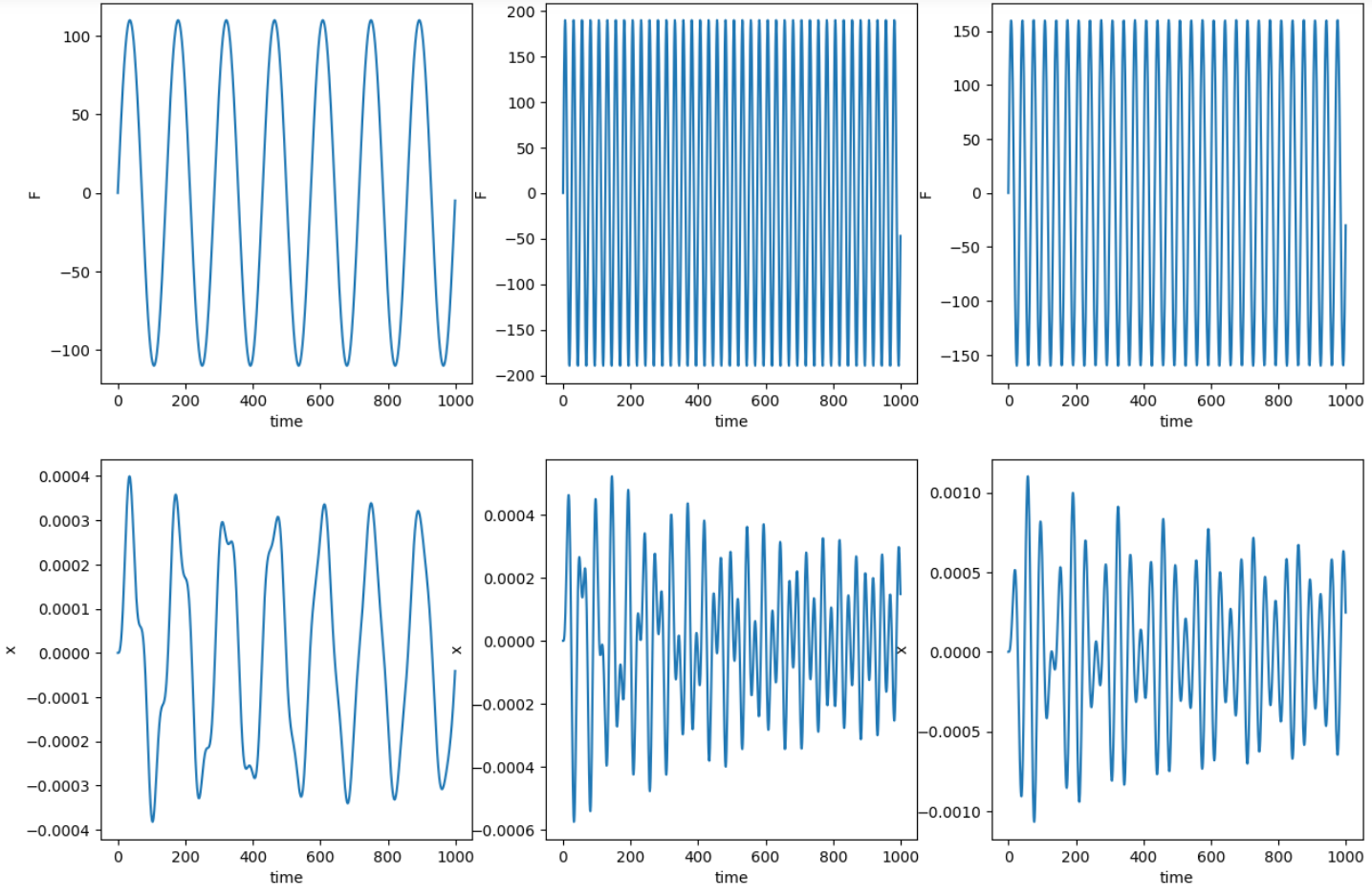
（7）预测位移：使用训练好的模型对给定的输入荷载进行预测，得到相应的结构位移。

# 第二章 数据展示与物理方程

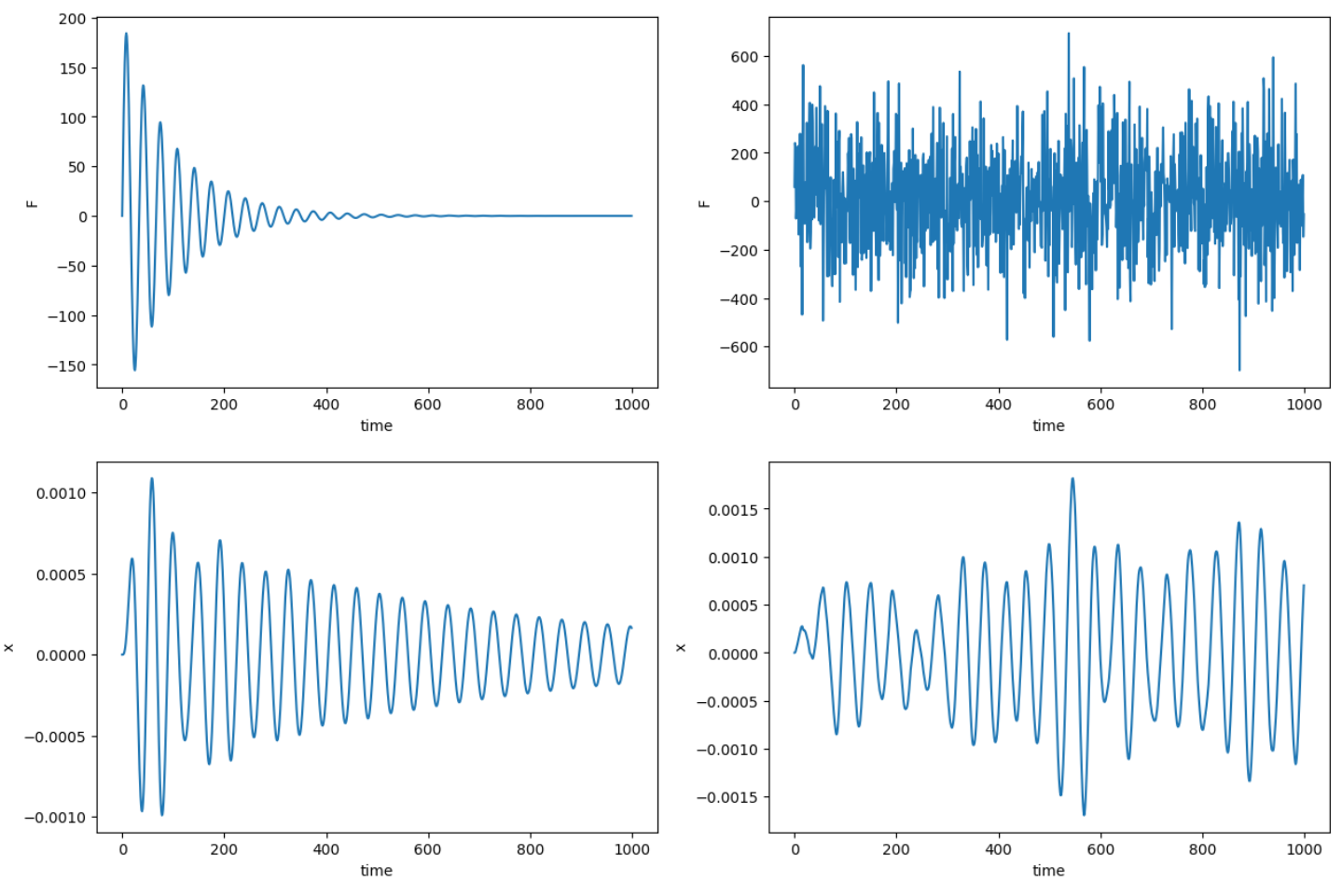
## 2.1原始数据

在给定的数据集中，“data\_exp.mat”中共有120行，每一行代表特定工况下的F值的大小变化。如第一行代表等于110KN，等于0.5Hz时F随时间变化的趋势，同时对应的x值的第一行也有随时间波动的趋势。同理，对应的“data\_pred1.mat”和“data\_pred2.mat”中的数据也为相同的方式。

问题一和问题二中的F的数据均为标准的正弦曲线图像，部分数据展示如下图所示。



问题三中的数据分别是加入了衰减系数和白噪声的驱动力的值，部分数据展示如下图所示。当加入逐渐衰减的驱动力时，在4s左右时衰减接近于0，但从位移x的图上可以看出，结构整体上并没有完全减少为0。当加入了白噪声之后的驱动力后，从图中明显看出其变得较为杂乱而且毫无章法，而其位移图上可以看出还是较为平缓的，原因可能是整体结构上存在一个较强的阻尼器帮助其减少外荷载的作用。



## 2.2求解结构自振频率

该结构为单自由度的自由振动，单自由度有阻尼振动的公式可以表示为：

其中， 是质量， 是阻尼系数， 是弹性系数 是位移。为简化该方程，引入两个系数：质量阻尼系数为和固有频率 。由此，上述方程可以进一步表示为： 。

单自由度有阻尼振动的解析解可以表示为：

其中， 和 是常数，可以通过初始条件确定。 是阻尼比， 是固有频率。如果给定如下质点的初始条件：初始位置 ，初始速度  ，则可以得到待定系数：

对于本问题，可以得到结构振动的微分方程为

其中，表示位移，表示速度，表示加速度。

由题目中的已知条件，当，总刚度时，求的,则.

# 第三章 问题解决

本章从普通的模型多层感知机（MLP）开始进行该问题的解决，接着使用了长短期记忆（LSTM）网络对该问题的进行解决。

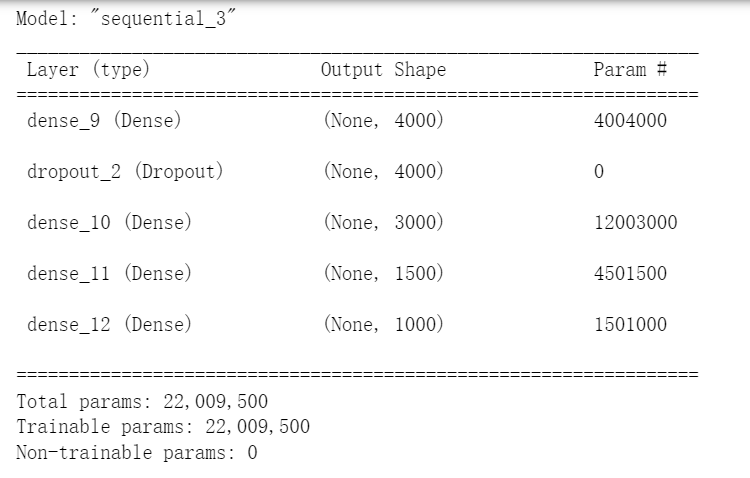
## 3.1基础模型训练效果展示

### 3.1.1 MLP

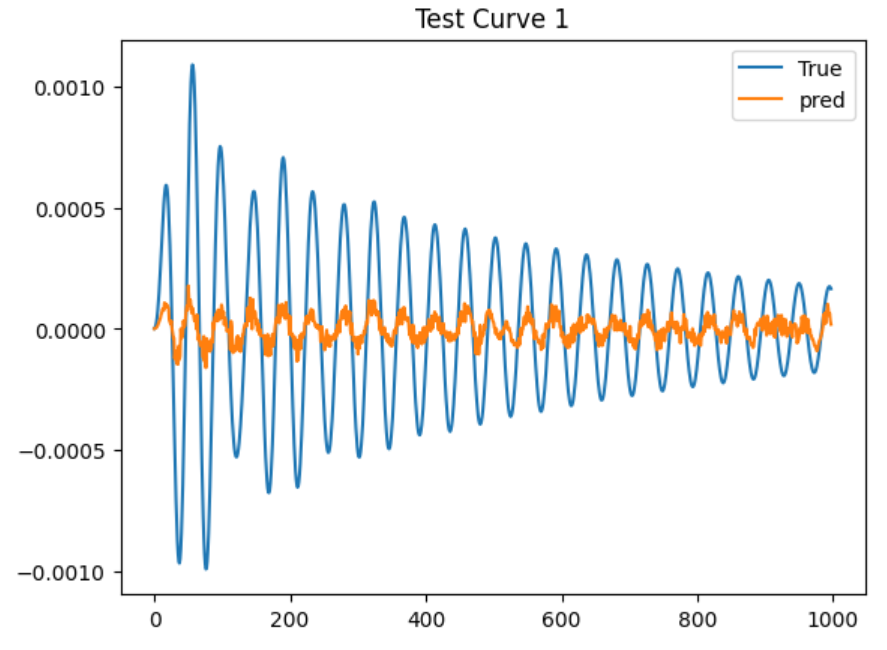
可以预见，传统解法求解位移的过程中，使用了杜哈曼（Duhamel）积分，得出最终结果为。

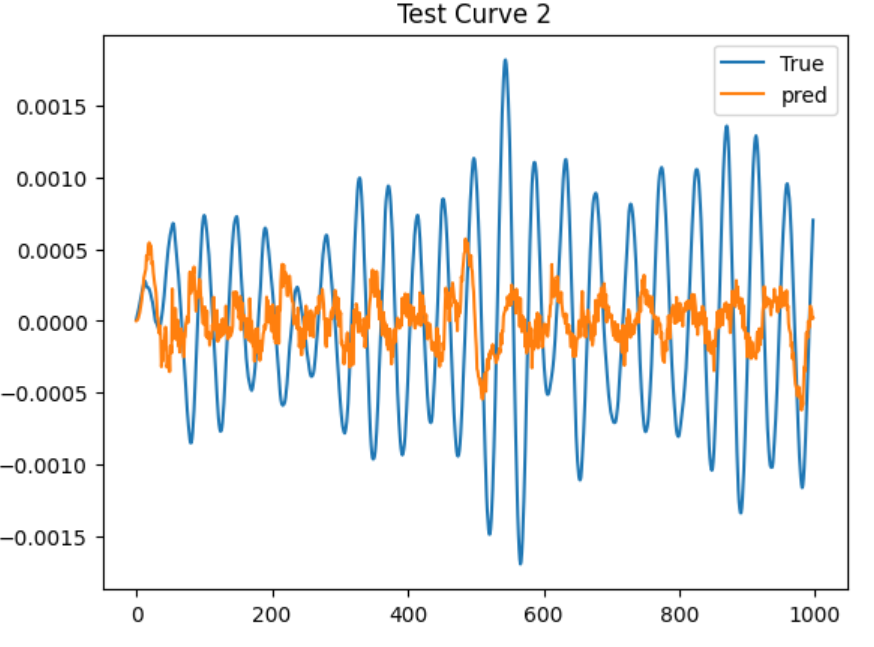
对于输入1000，输出1000的神经网络，想要得出这样一个积分的解十分苦难。因此传统时域上的数据并不能产生很好的效果是可以预见的。

建立如下的网络：



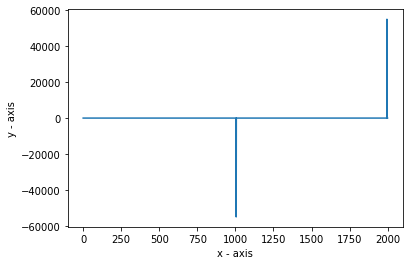
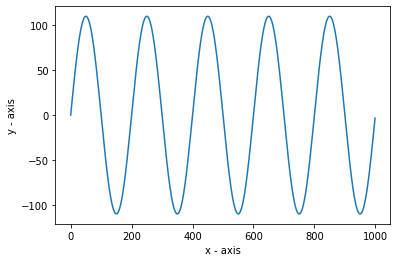
直接看一下第三问的结果：

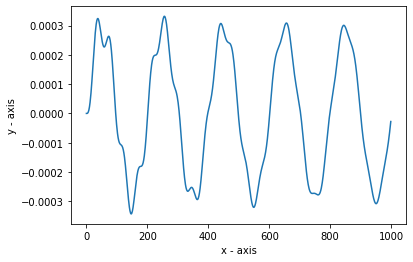
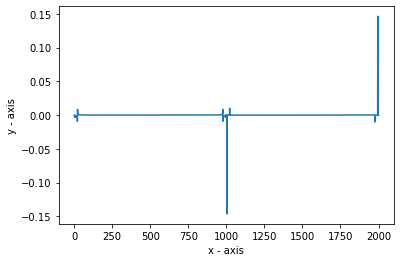




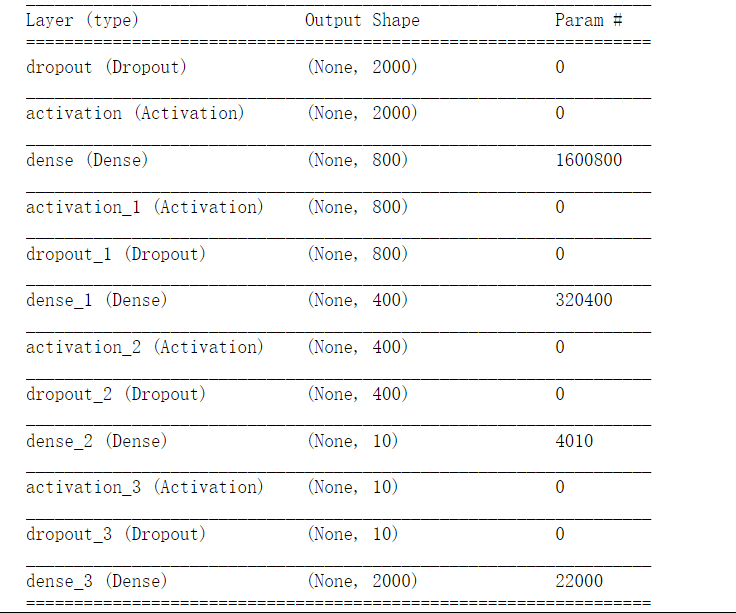
效果并不是很好。

考虑结构的响应问题，可以将外力和位移都作傅里叶变换转化到频域上，从而得到更为简洁的对应关系。



设置神经网络模型的结构。由于本研究的特征数量多而且数据集较小，所以采用了较深的神经网络结构。第一层神经网络具有800个神经元，为了防止过拟合，设置了丢弃率为0.7的Dropout网络，接着分别设置了神经元数量为400丢弃率为0.7的Dropout网络10丢弃率为0.4的隐藏层神经元。最后的输出层神经元的数量为2000，与特征的数量相同。



第三步：神经网络的训练：本模型的训练采用的是MSE损失函数，使用的优化器为Adam优化器。迭代的Epoch总数为800，在仅有训练集的情况下得到了如下图所示的训练集和验证集损失。

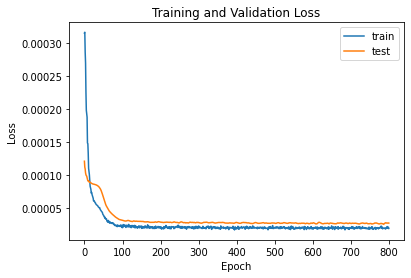


图3-3 MLP1000训练集损失

另外，在此处介绍使用到的Adam优化器：Adam优化器是一种在深度学习中用来替代随机梯度下降的优化算法，它结合了AdaGrad和RMSProp两种优化算法的优点，能够自适应地调节学习率，适用于不稳定的目标函数和梯度稀疏或噪声的情况。

**Adam优化器的基本思想是对梯度的一阶矩估计（即梯度的均值）和二阶矩估计（即梯度的未中心化的方差）进行综合考虑，计算出更新步长。具体来说，它首先计算梯度的指数移动平均数，类似于Momentum算法，综合考虑之前时间步的梯度动量；然后计算梯度平方的指数移动平均数，类似于RMSProp算法，对梯度平方进行加权均值；最后对两者进行偏差纠正，降低训练初期的影响，并用它们的比值来更新参数。**

**Adam优化器具有以下几个显著的优点：**

**- 实现简单，计算高效，对内存需求少**

**- 参数的更新不受梯度的伸缩变换影响**

**- 超参数具有很好的解释性，且通常无需调整或仅需很少的微调**

**- 更新的步长能够被限制在大致的范围内（初始学习率）**

**- 能自然地实现步长退火过程（自动调整学习率）**

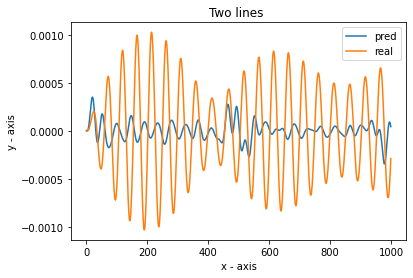
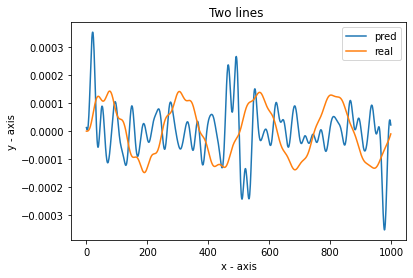
**- 很适合应用于大规模的数据及参数的场景**

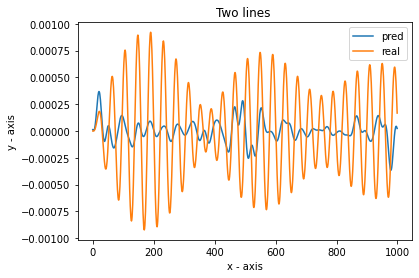
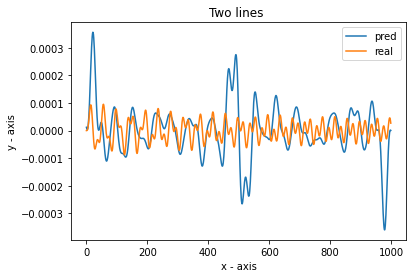
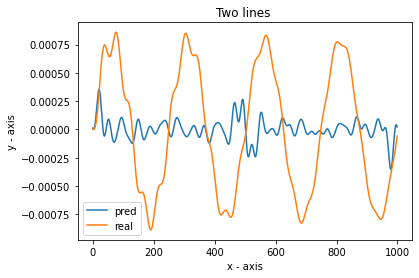
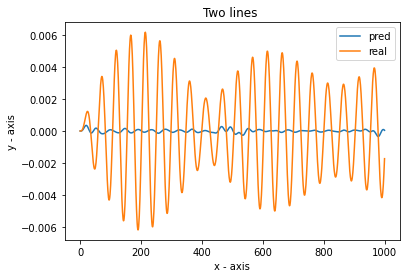
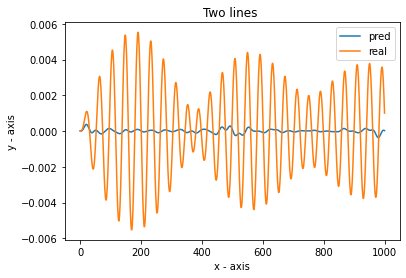
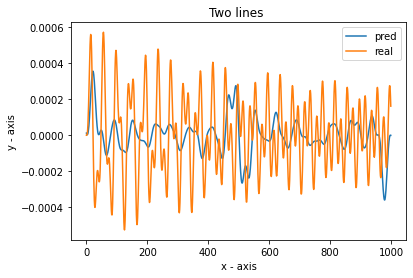
**- 适用于不稳定目标函数**

**- 适用于梯度稀疏或梯度存在很大噪声的问题**

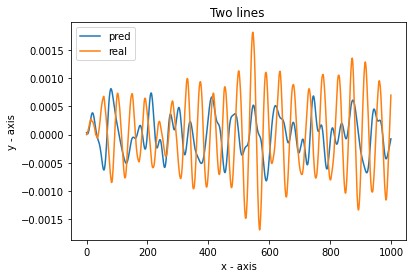
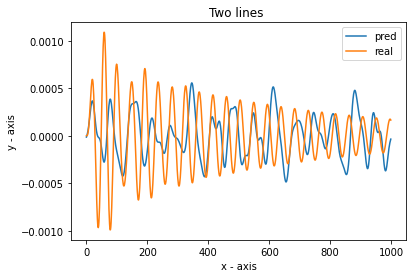
**综合来说，Adam优化器在很多情况下都表现出优秀的学习率自适应能力，是一种比较通用和高效的优化算法。**

第四步：训练结果展示。第二问测试集，在这八个图像上均没有表现出较为优秀的效果。尤其是在低频段，效果极差，并且甚至没有拟合能力。在高频段，能反映出位移的曲线的频率特性，但其幅值特性没能很好的反映出来。在噪声区段，衰减时并没有随F变化为0，而是表现出相对稳定的振动特性。同样的是，在白噪声的干扰下，其也能表现出较为优异的频率识别特征，但是对于幅值问题，依然无法得到很好的解决。



测试集上预测的结果



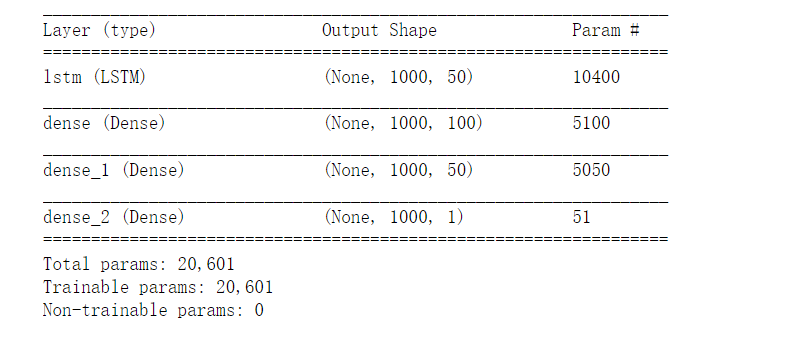
噪声条件下的预测结果

对于此类输入特征较多的数据问题，不同的神经网络结构可能适用于不同的任务和数据。并且由于数据量较少，无论是在时域上还是频域上，都无法得到一个较好的结果。但相对来说，频域上建立的神经网络显示出了一些对于响应的结果，可以推测，或许增大数据量可以使得频域上的网络更好地预测。

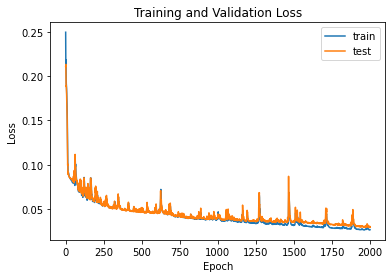
### 3.1.2 LSTM模型

基于上述的数据集，建立时间步长为1000的数据集。如图所示，对于一个样本，其特征数为1，即F的数值大小。

下图展示了该数据集下采用的神经网络模型，共设置三个隐藏层，其中第一个LSTM层有50个隐藏层，第二个隐藏层有100个全连接神经元，第三个隐藏层有50个连接神经元，最后输出层为1，对应输出的位移序列。本模型的所有参数共有20601个。

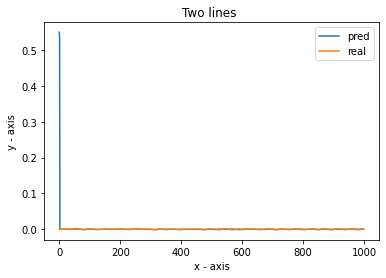
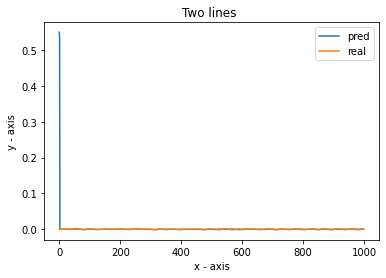


下图展示了模型在训练集和测试集上的损失，训练集和测试集的损失均随时间逐渐下降，而且可以看出，在2000步以内都在一直下降，推测效果应该比较理想。



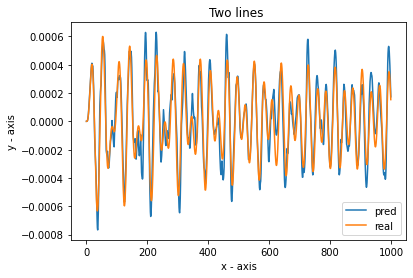
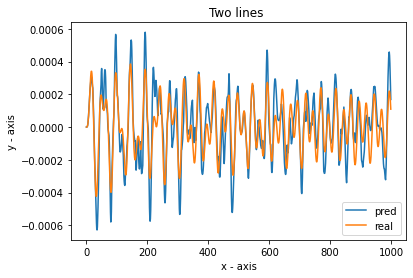
训练集和测试集损失

接下来进行网络预测结果的展示，

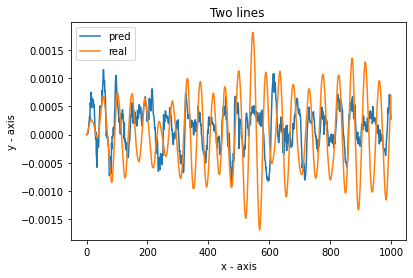
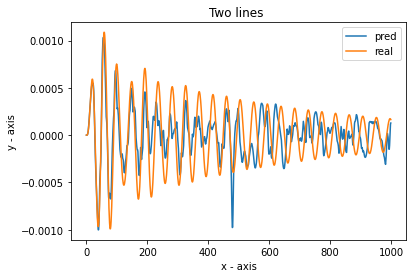
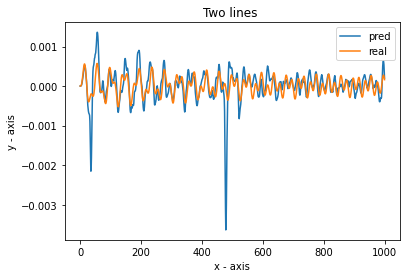
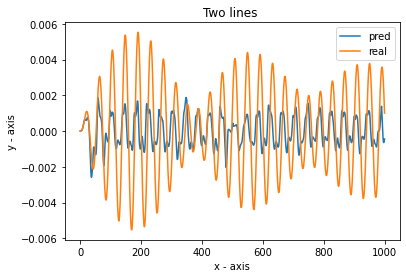
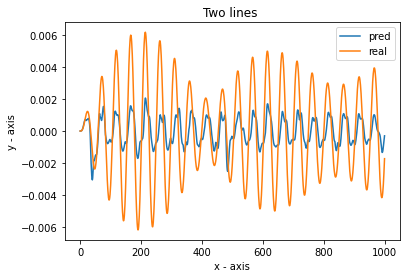
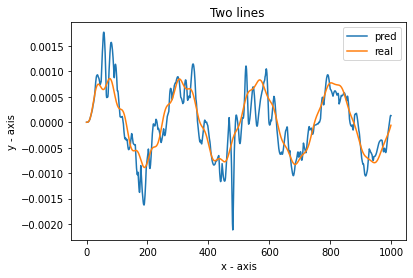
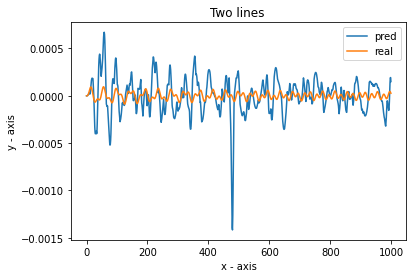
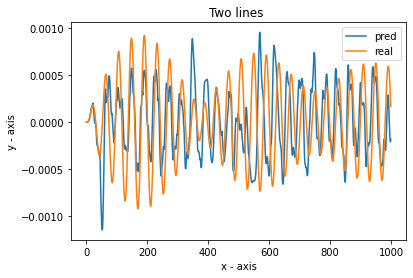
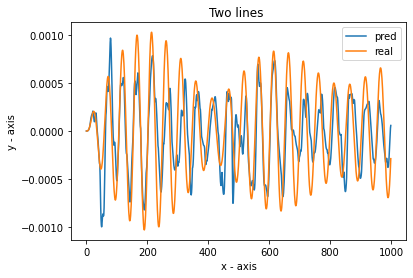
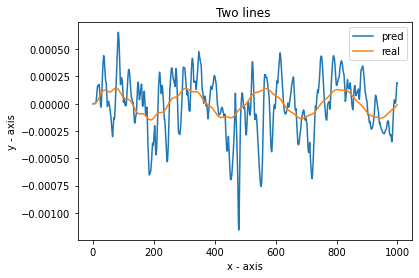
 

预测效果

可以看到，输出的第一个值居然特别大，以至于影响了整体的结果，应该是反归一化时候出现的问题，通过手动将0处的值设置为0，可以得到以下效果：

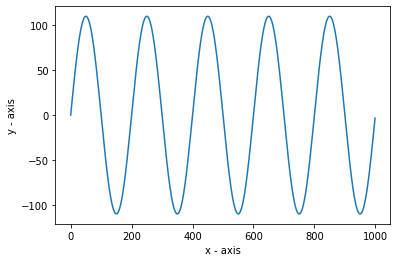
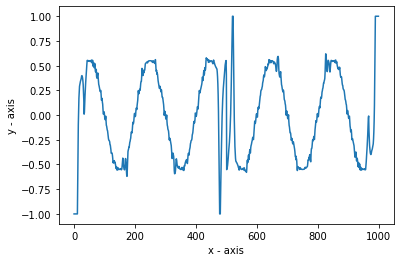
 

部分验证集的预测效果



二三问预测结果

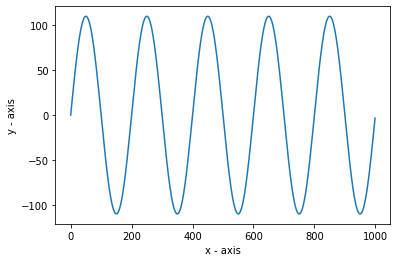
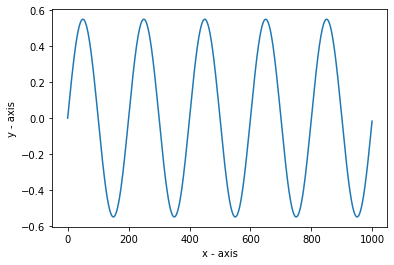
不难看出，在测试集上，LSTM已经获得了较好的结果，预测的效果也有较好的改善，但是由于有前人的经验，LSTM可以获得很好的结果。但是现在已经建立的模型已经是较为标准的LSTM模型，并且输入的数据维度和输出的数据维度也没有问题。考虑到数据反归一化时候的问题，思考数据归一化时的操作。

初始的归一化结果

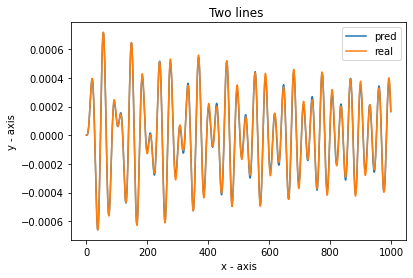
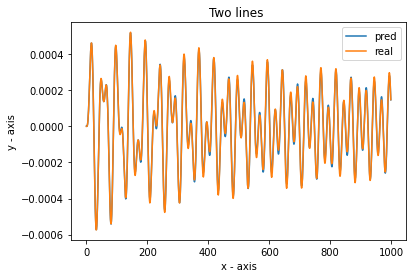
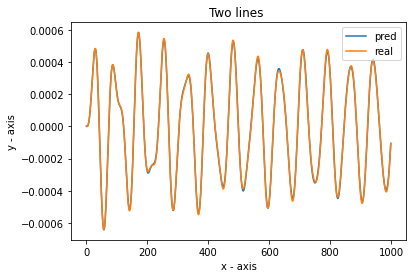
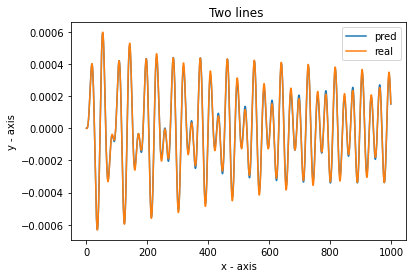
考虑到MinMaxScaler的归一化方法，只是对于列的归一化，对于MLP问题中，由于输入的1000维是特征，因此这样归一化是没有太大的问题的。但是对于建立的LSTM模型，特征为输入的F，这样归一化会导致归一化后数据和原数据产生较大的差别。

因此，现将第一问中的120个数据转化为1维列向量，进行归一化后，再转换回原样，这样就可以得到一个较为好的归一化数据如下图：

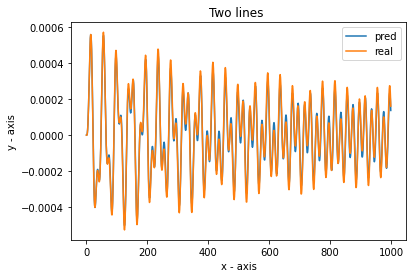
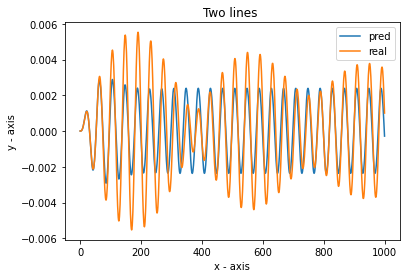
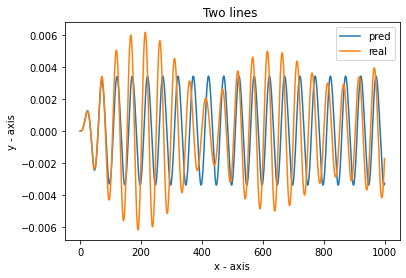
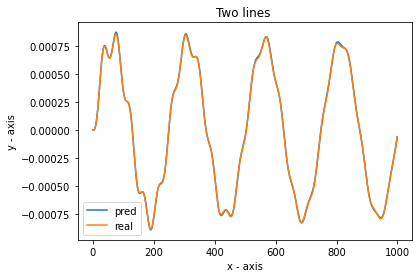
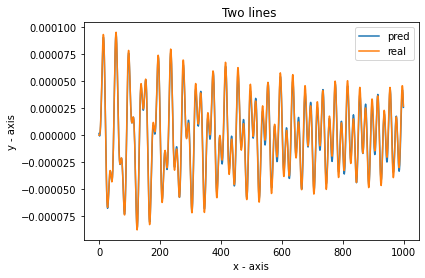
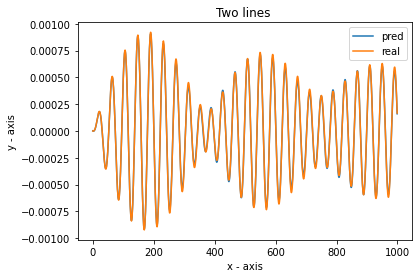
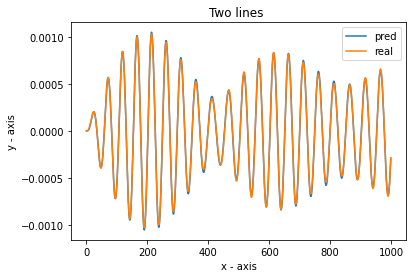
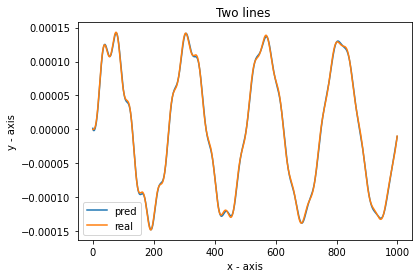
改进后的归一化结果

使用与之前同样的LSTM网络可以得到下列结果：



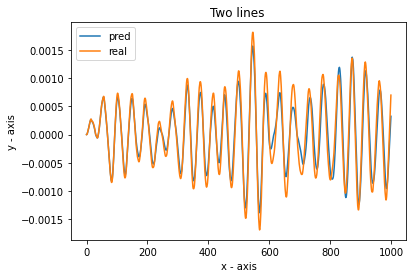
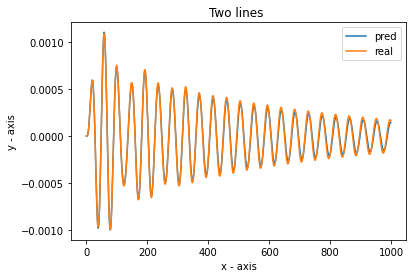
部分验证集的预测效果

可以看出，在验证集上已经可以达到几乎完全拟合的效果。



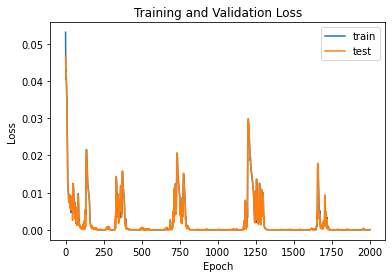
第二问的预测结果

如图所示，在第二问的数据集上，蓝色的为预测值，黄色的为真实值。可以发现除了在与结构频率相近，且外力F0较大时，预测的效果并不能完全拟合，可能原因是训练集中没有相应的接近结构频率的数据，且没有相应300的这么大的数据来训练。



第三问的预测结果

第三问中，第一个衰减噪声已经可以达到几乎完全拟合的状态，就算是白噪声也可以较为完美地拟合，没有完全拟合的原因可能是因为网络参数并没有达到最佳状态，可以预见，通过调整参数，使得第三问第二小问也达到很好的拟合效果。



通过损失图像可以看出，其实并不是步长越多，拟合的效果越好。对于Adam优化器来说，很容易产生图中这种较大的波动，为了获取训练中获得的最佳模型，可以使用earlystop等方式，保存训练过程中对于验证集损失最小的模型。

# 第四章 思考与讨论

## 4.1数据处理

在本问题中面临的第一个难题就是如何更好地划分数据集。首先在训练集上，如果按照120个样本进行划分数据集，则会出现在训练集上的训练结果好而出现过拟合现象。假如把一部分数据集按一段进行划分，则会有数据特征不明显，网络无法较好地学习单自由度系统的特性。所以合理的划分数据集是解决问题的关键。数据集的划分是非常重要的一步，可以影响模型的性能和泛化能力。经过查阅资料，得到了以下可以用于结合数据集划分的方法。训练集、验证集和测试集：通常将数据集划分为三个部分，训练集用于模型的参数更新和训练，验证集用于调整模型的超参数和监控模型的性能，测试集用于最终评估模型的泛化能力。

第二要确保数据集中的样本是随机排列的，这样可以减少数据的分布顺序对模型的影响，并且可以更好地代表整体数据的特征。同时在划分数据集时，要确保各个类别的样本在每个数据集中都是均衡的。这样可以避免模型偏向于某些类别，从而更好地学习整体数据的特征。

对于本问题时间序列划分：对于时间序列数据，划分数据集时需要考虑时间上的连续性。但是在实际操作过程中，由于数据归一化过程中犯了一些低级错误，导致效果不佳，因此，我们应该在前期画一定的时间去分析处理数据后，再进行网络的训练，这样既可以节约计算资源，也可以使得网络获得更好的效果。

## 4.2网络的选择

对于每一个深度学习问题，神经网络的选择是多样的。对于一个问题可能有很多符合的模型来解决，也可能不存在任何一个模型可以解决特定的问题，所以模型的选择也至关重要。在本研究，分别选择了MLP和LSTM神经网络模型来对问题进行解决。然而每个问题数据集的划分都有不同的解释，所以可以通过不一样的数据集进行网络的选择。在这个问题中，MLP并不能很好地拟合模型，无论是时域还是频域，但推测频域上的数据量如果增大，或许可以获得更好的结果。LSTM可以较好地解决这个问题，但是想要获得较为良好的模型，还是得进一步地调整网络超参数。但是对比两个网络，可以看出选择合理的网络结构比起一味地调参是更加合理的选择。

代码：<https://github.com/Christopher-Wan/Homework/tree/main/AI>