light_data_3.10/result/3.13：

（1） /rand_bias0.3：采样率 10M，接收速率 60M，均匀分布，偏置电流 0.3A。

1. /mix_amp: 混合幅度数据作为训练数据，且数据归一化。发送信号是均匀分布的随机信号，采样率为 10M，接收速率 150M，偏置电流 0.3A。

1.1 /Twononlinear1：

相关符号数为 3，即 h_order=3*rate_times。L=2，U=25

```
Twononlinear ,
ini learningRate = 1.000000e-02 ,
min batch size = 400 ,
DropPeriod = 5 ,
DropFactor = 0.100000 ,
amp begin = 2 , amp end = 26 , amp step = 1
data_num = 100
validationFrequency is floor(numel(xTrain)/miniBatchSize/4)
H order = 18
Hidden Units = 25
```

1.2 /Twononlinear2：

相关符号数为 5，即 h_order=5*rate_times。L=2，U=25

```
Twononlinear ,
ini learningRate = 1.000000e-02 ,
min batch size = 400 ,
DropPeriod = 5 ,
DropFactor = 0.100000 ,
amp begin = 2 , amp end = 26 , amp step = 1
data_num = 100
validationFrequency is floor(numel(xTrain)/miniBatchSize/4)
H order = 30
Hidden Units = 25
```

1.3 /Twononlinear3：

相关符号数为 8，即 h_order=8*rate_times。L=2，U=25

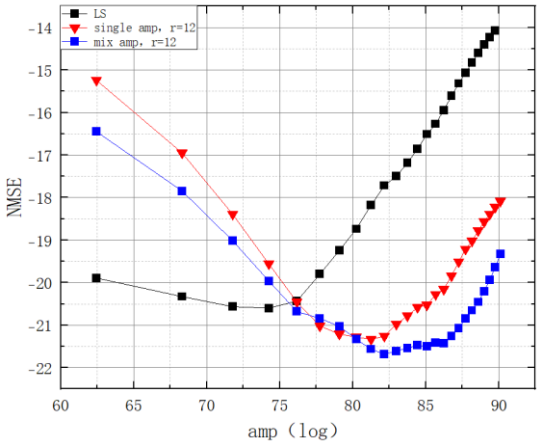
```
Twononlinear ,
ini learningRate = 1.000000e-02 ,
min batch size = 400 ,
DropPeriod = 5 ,
DropFactor = 0.100000 ,
amp begin = 2 , amp end = 26 , amp step = 1
data_num = 100
validationFrequency is floor(numel(xTrain)/miniBatchSize/4)
H order = 48
Hidden Units = 25
```

1.4 /Twononlinear4:

相关符号数为 12，即 h_order=12*rate_times。L=2，U=25

```
Twononlinear ,
ini learningRate = 1.000000e-02 ,
min batch size = 400 ,
DropPeriod = 5 ,
DropFactor = 0.100000 ,
amp begin = 2 , amp end = 26 , amp step = 1
data_num = 100
validationFrequency is floor(numel(xTrain)/miniBatchSize/4)
H order = 72
Hidden Units = 25
```

1.5 /Twononlinear5:



基于此图，决定先训练线性的部分，看能不能把线性部分的性能训练到和 LS 持平。即 amp_begin=2，amp_end=6。相关符号数为 12，即 h_order=12*rate_times。L=2，U=25

```
Twononlinear ,
ini learningRate = 1.000000e-02 ,
min batch size = 400 ,
DropPeriod = 5 ,
DropFactor = 0.100000 ,
amp begin = 2 , amp end = 6 , amp step = 1
data_num = 100
validationFrequency is floor(numel(xTrain)/miniBatchSize/4)
H order = 72
Hidden Units = 25
```

1.6 /Onenonlinear1:

基于上图，决定先训练线性的部分，看能不能把线性部分的性能训练到和 LS 持平。即 amp_begin=2，amp_end=6。相关符号数为 12，即 h_order=12*rate_times。L=1，U=25

```
Onenonlinear ,
ini learningRate = 1.000000e-02 ,
min batch size = 400 ,
DropPeriod = 5 ,
DropFactor = 0.100000 ,
amp begin = 2 , amp end = 6 , amp step = 1
data_num = 100
validationFrequency is floor(numel(xTrain)/miniBatchSize/4)
H order = 72
Hidden Units = 25
```

1.7 /Linear1:

基于上图，决定先训练线性的部分，看能不能把线性部分的性能训练到和 LS 持平。即 amp_begin=2，amp_end=6。相关符号数为 12，即 h_order=12*rate_times。一层线性隐藏层，隐藏层点数=25

```
linear ,
ini learningRate = 1.000000e-02 ,
min batch size = 400 ,
DropPeriod = 5 ,
DropFactor = 0.100000 ,
amp begin = 2 , amp end = 6 , amp step = 1
data_num = 100
validationFrequency is floor(numel(xTrain)/miniBatchSize/4)
H order = 72
Hidden Units = 25
```

1.8 /Linear2:

基于上图，决定先训练线性的部分，看能不能把线性部分的性能训练到和 LS 持平。即 amp_begin=2，amp_end=6。相关符号数为 12，即 h_order=12*rate_times。一层线性隐藏层，隐藏层点数=40

```
linear ,
ini learningRate = 1.000000e-02 ,
min batch size = 400 ,
DropPeriod = 5 ,
DropFactor = 0.100000 ,
amp begin = 2 , amp end = 6 , amp step = 1
data_num = 100
validationFrequency is floor(numel(xTrain)/miniBatchSize/4)
H order = 72
Hidden Units = 40
```

1.9 /Linear3:

基于上图，决定先训练线性的部分，看能不能把线性部分的性能训练到和 LS 持平。即 amp_begin=2，amp_end=6。相关符号数为 12，即 h_order=12*rate_times。一层线性隐藏层，隐藏层点数=60

```
linear ,
ini learningRate = 1.000000e-02 ,
min batch size = 400 ,
DropPeriod = 5 ,
DropFactor = 0.100000 ,
amp begin = 2 , amp end = 6 , amp step = 1
data_num = 100
validationFrequency is floor(numel(xTrain)/miniBatchSize/4)
H order = 72
Hidden Units = 60
```

1.10 /Linear4:

基于上图，决定先训练线性的部分，看能不能把线性部分的性能训练到和 LS 持平。即 amp_begin=2，amp_end=6。相关符号数为 12，即 h_order=12*rate_times。两层线性隐藏层，隐藏层点数=40

```
linear ,
ini learningRate = 1.000000e-02 ,
min batch size = 400 ,
DropPeriod = 5 ,
DropFactor = 0.100000 ,
amp begin = 2 , amp end = 6 , amp step = 1
data_num = 100
validationFrequency is floor(numel(xTrain)/miniBatchSize/4)
H order = 72
Hidden Units = 40
```

1.11 /Linear5:

基于上图，决定先训练线性的部分，看能不能把线性部分的性能训练到和 LS 持平。即 amp_begin=2，amp_end=6。相关符号数为 12，即 h_order=12*rate_times。两层线性隐藏层，隐藏层点数=60

```
linear ,
ini learningRate = 1.000000e-02 ,
min batch size = 400 ,
DropPeriod = 5 ,
DropFactor = 0.100000 ,
amp begin = 2 , amp end = 6 , amp step = 1
data_num = 100
validationFrequency is floor(numel(xTrain)/miniBatchSize/4)
H order = 72
Hidden Units = 60
```

light_data_3.11/result/3.13：

（1） /rand_bias0.3：采样率 10M，接收速率 150M，均匀分布，偏置电流 0.3A。

1. /norm_LS: 用 LS 算法，求出各个幅度信号的 NMSE。

2. /single_amp: 单一幅度数据作为训练数据，且数据归一化。发送信号是均匀分布的随机信号，采样率为 10M，接收速率 150M，偏置电流 0.3A。

2.1 /Threenonlinear1：
相关符号数为 5，即 h_order=5*rate_times。L=3，U=60

```
Threenonlinear ,
ini learningRate = 1.000000e-02 ,
min batch size = 200 ,
DropPeriod = 12 ,
DropFactor = 0.100000 ,
amp begin = 1 , amp end = 101 , amp step = 1
data_num = 100
validationFrequency is floor(size(xTrain{1},2)/miniBatchSize
origin rate = 1.000000e+07 , receive rate = 1.500000e+08
H order = 75
Hidden Units = 60
```

3. /mix_amp: 混合幅度数据作为训练数据，且数据归一化。发送信号是均匀分布的随机信号，采样率为 10M，接收速率 150M，偏置电流 0.3A。

3.1 /Threenonlinear1：
相关符号数为 5，即 h_order=5*rate_times。L=3，U=60

```
Threenonlinear ,
ini learningRate = 1.000000e-02 ,
min batch size = 400 ,
DropPeriod = 5 ,
DropFactor = 0.100000 ,
amp begin = 1 , amp end = 101 , amp step = 2
data_num = 25
validationFrequency is floor(numel(xTrain)/miniBatchSize/4)
H order = 75
Hidden Units = 60
```