### deep\_ensembles

- 1. 西班牙数据集
- 2. 美国数据集

# deep\_ensembles

#### deep ensemble 网络结构:

输入层节点: 7隐藏层 1 节点: 50隐藏层 2 节点: 50

● 输出层节点: 2 (mean、var)

• cost function: NLL

#### deep ensemble 参数设置:

```
1
        parser = argparse.ArgumentParser()
 2
        # Ensemble size
 3
        parser.add_argument('--ensemble_size', type=int, default=5,
                             help='Size of the ensemble')
 4
 5
        # Maximum number of iterations
        parser.add_argument('--max_iter', type=int, default=5000,
 6
 7
                             help='Maximum number of iterations')
 8
        # Batch size
 9
        parser.add_argument('--batch_size', type=int, default=10,
10
                             help='Size of batch')
11
        # Epsilon for adversarial input perturbation
12
        parser.add_argument('--epsilon', type=float, default=1e-2,
13
                             help='Epsilon for adversarial input perturbation')
        # Alpha for trade-off between likelihood score and adversarial score
14
15
        parser.add_argument('--alpha', type=float, default=0.5,
                             help='Trade off parameter for likelihood score and
16
    adversarial score')
17
        # Learning rate
        parser.add_argument('--learning_rate', type=float, default=0.005,
18
19
                             help='Learning rate for the optimization')
20
        # Gradient clipping value
        parser.add_argument('--grad_clip', type=float, default=100.,
21
                             help='clip gradients at this value')
22
        # Learning rate decay
23
24
        parser.add_argument('--decay_rate', type=float, default=0.99,
                             help='Decay rate for learning rate')
25
26
        # Dropout rate (keep prob)
27
        parser.add_argument('--keep_prob', type=float, default=0.8,
28
                             help='Keep probability for dropout')
```

## esn 参数设置:

## box-cox 变换:

```
设 wp \sim N(\mu, \sigma^2) 则 wp_{ln} = ln(wp+0.01) wp_{pred} = exp(f(X, wp_{ln})) - 0.01 预测方差: Var(wp_{pred}) = e^{2\mu+\sigma^2}(e^{\sigma^2}-1)
```

## 1. 西班牙数据集

train index: [6426, 10427] train\_len: 4000

test index: [14389, 15390] test\_len: 1000

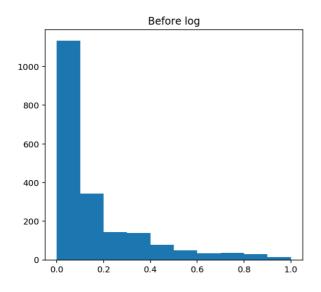
● 输入特征:

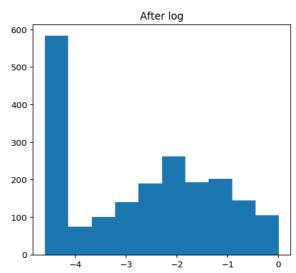
```
1 'wind_speed', 'sin(wd)', 'cos(wd)', 【t期】
2 'wind_speed-1', 'sin(wd)-1','cos(wd)-1', 'wind_power-1'【t-1期】
```

• 输出: wind\_power

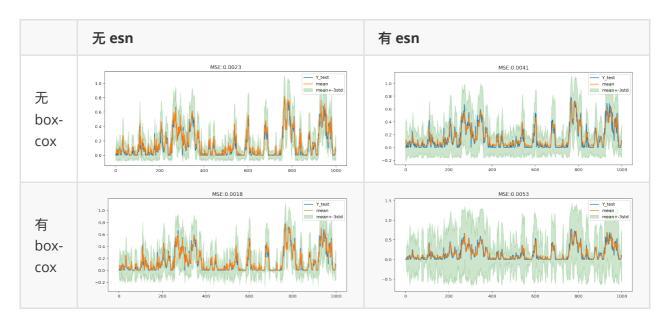
box-cox 转换前后数据分布:

: Text(0.5, 1.0, 'After log')





## 结果:



## 2. 美国数据集

train index: [3001, 7002] train\_len: 4000

test index: [2000, 3001] test\_len: 1000

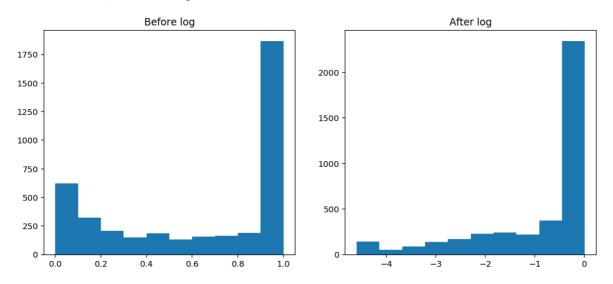
● 输入特征:

```
1 'wind_speed', 'sin(wd)', 'cos(wd)', 【t期】
2 'wind_speed-1', 'sin(wd)-1','cos(wd)-1', 'wind_power-1'【t-1期】
```

● 输出: wind\_power

```
box-cox 转换前后数据分布:
```

: Text(0.5, 1.0, 'After log')



## 结果:

