注意：

1，结果不稳定…？？？

2，标准化后，效果更好

#### 训练集构建

?lm

[ts.intersect](http://127.0.0.1:19369/help/library/stats/help/ts.intersect)(..., dframe = TRUE) %>% na.trim()

示例：

library(mlbench)

data(BostonHousing)

require(nnet)

require(caret)

mygrid <- expand.grid(.decay=c(0.1,0.5), .size=c(1:8))

**set.seed(1)**

**# range—极大极小标准化**

nnetfit <- train(medv ~ ., data=BostonHousing, method="nnet", **preProcess = "range"**, maxit=1000, tuneGrid=mygrid, trace=F, **linout=TRUE**)

print(nnetfit)

preProc <- preProcess(BostonHousing, method = "range")

training <- predict(preProc, BostonHousing)

set.seed(1)

nnet.fit <- nnet(medv ~ ., data=training, size=5, rang = 0.1, decay = 0.5, maxit = 1000, trace=F, **linout=TRUE**)

nnet.predict <- predict(nnet.fit)

# mean squared error: 16.40581

mean((nnet.predict - training$medv)^2)

plot(training$medv, nnet.predict,

main="Neural network predictions vs actual",

xlab="Actual")

更多参考：

<https://heuristically.wordpress.com/2011/11/17/using-neural-network-for-regression/>