机器学习在因果分析中的应用

郑军威

目录

1 引言

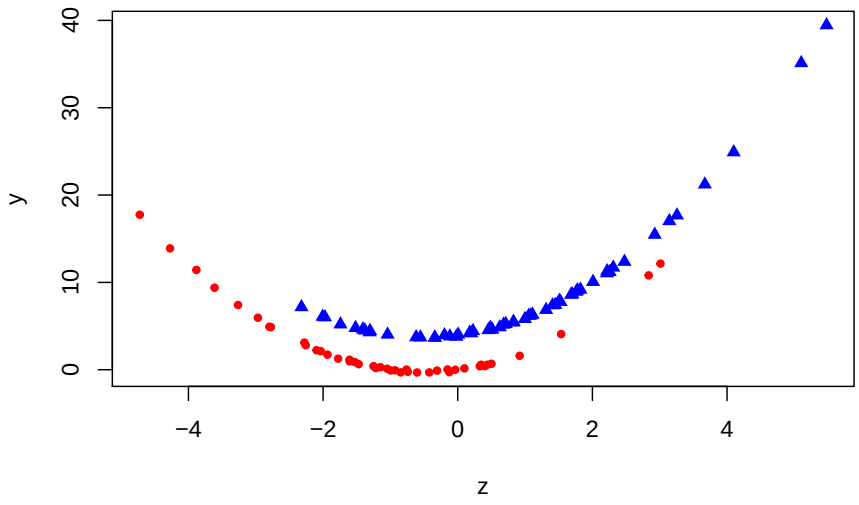
本文将使用模拟生成数据的方法，在了解因果关系的情况下，表明使用合适的统计方法，可以更好地拟合出因果关系。

2 模拟数据

z ~ N (0, 4)
x = Bernoulli ((1 + exp (-z))^-1)
y = z + z^2 + 4x + e, e ~ N (0, 0.01)

3 生成数据

```
n = 100
z = 2*rnorm(n)
x = rbinom(n,1,sigmoid(z))
y = z + z^2 + 4*x + 0.1*rnorm(n)
plot(z,y,type = "n")
points(z[x==0],y[x==0],col="red",pch=20)
points(z[x==1],y[x==1],col="blue",pch=17)
```



4 线性回归

```
data = data.frame(y=y,x=x,z=z)
fit = lm(y~.,data)
coeftest(fit)
```

```
##
## t test of coefficients:
```

```
##
##             Estimate Std. Error t value  Pr(>|t|)
## (Intercept)  4.45585    0.90639  4.9160 3.594e-06 ***
## x            2.75313    1.28607  2.1407    0.0348 *
## z            1.54723    0.32846  4.7105 8.242e-06 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

5 半参数回归

```
library(mgcv)
```

```
## Loading required package: nlme
```

```
## This is mgcv 1.8-27. For overview type 'help("mgcv-package")'.
```

```
fit = gam(y ~ x + s(z),data,family=gaussian)
summary(fit)
```

```
##
## Family: gaussian
## Link function: identity
##
## Formula:
## y ~ x + s(z)
##
## Parametric coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  3.80627    0.01827   208.4   <2e-16 ***
## x            3.97440    0.02618   151.8   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Approximate significance of smooth terms:
```

```
##        edf Ref.df     F p-value
## s(z) 8.809  8.989 34388  <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## R-sq.(adj) =      1   Deviance explained =  100%
## GCV = 0.013309  Scale est. = 0.011871  n = 100
```

6 两种方法对比

从线性回归可以看出，x 的系数偏离真实数值 4。但从半参数回归可以看出，在不用猜测 z 的具体方程情况下，可以直接使用该方法，获得 x 系数，并且极其接近真实值 4.

7 小结

在了解因果关系的情况下，使用合适的统计方法，从而不局限于线性回归，可以更好地分析因果关系。