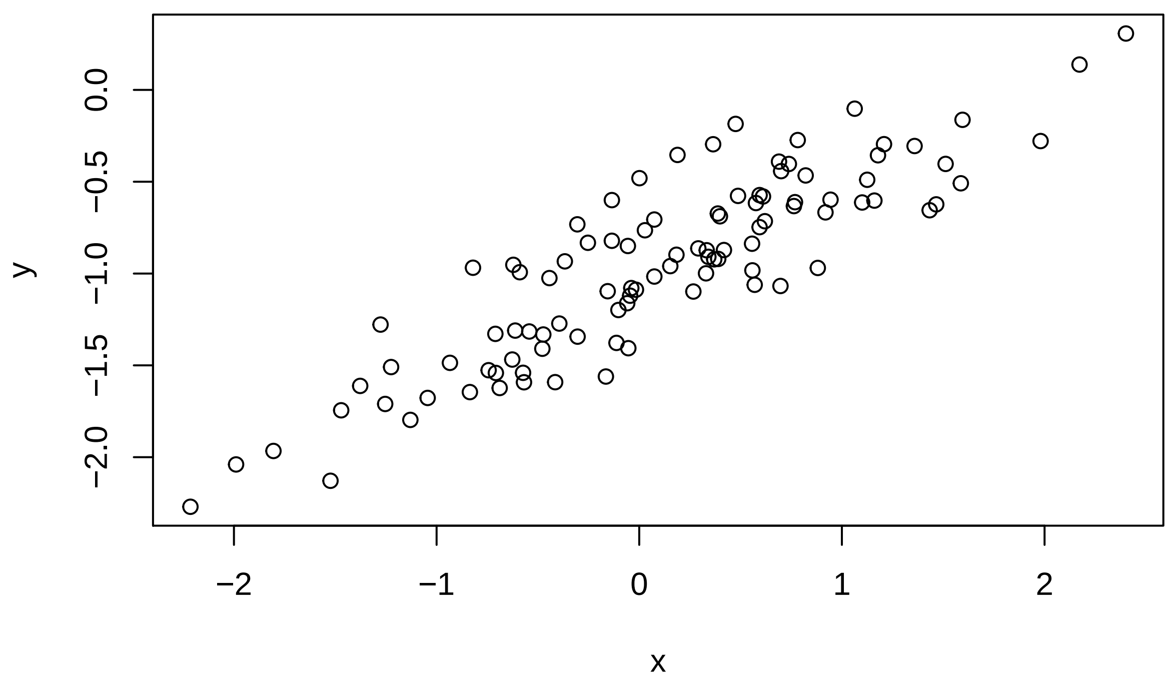
Linear Regression Simulation

Simulate data

```r
set.seed(1)
windows(width=10, height=8)
x = rnorm(100)
eps = rnorm(100, 0, 0.25)
y = -1 + 0.5*x + eps
plot(x,y)
```

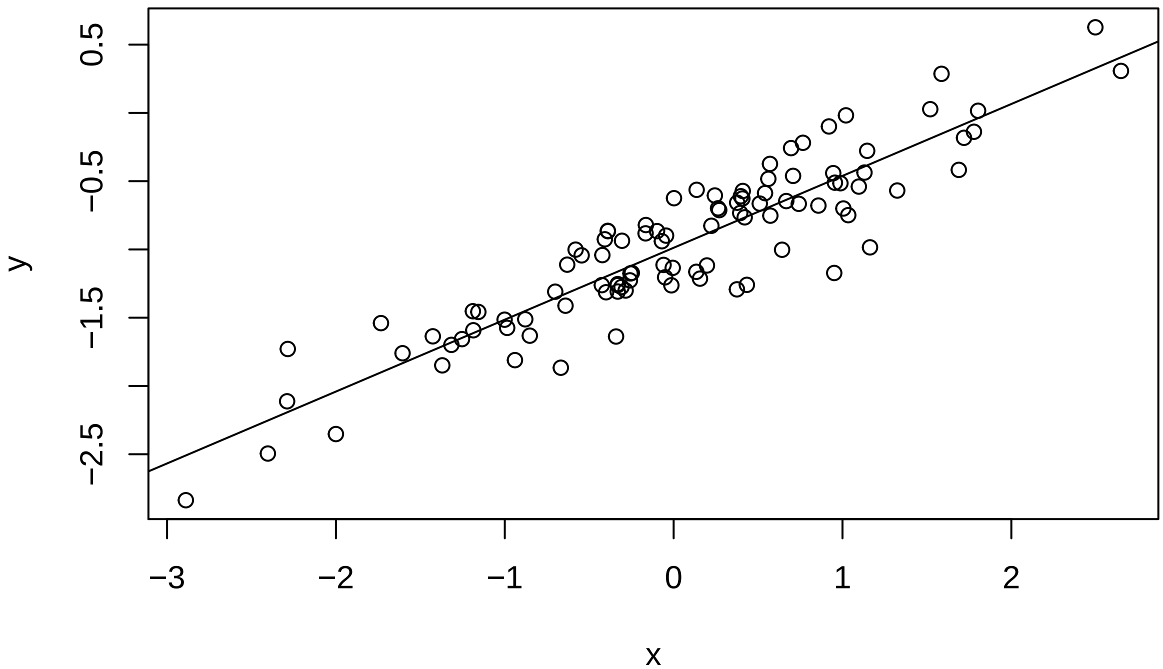


There is a linear relationship between x and y. They are positively correlated.

Predict y using x.

```r
windows(width=10, height=8)
x = rnorm(100)
eps = rnorm(100, 0, 0.25)
y = -1 + 0.5*x + eps
plot(x,y)
```

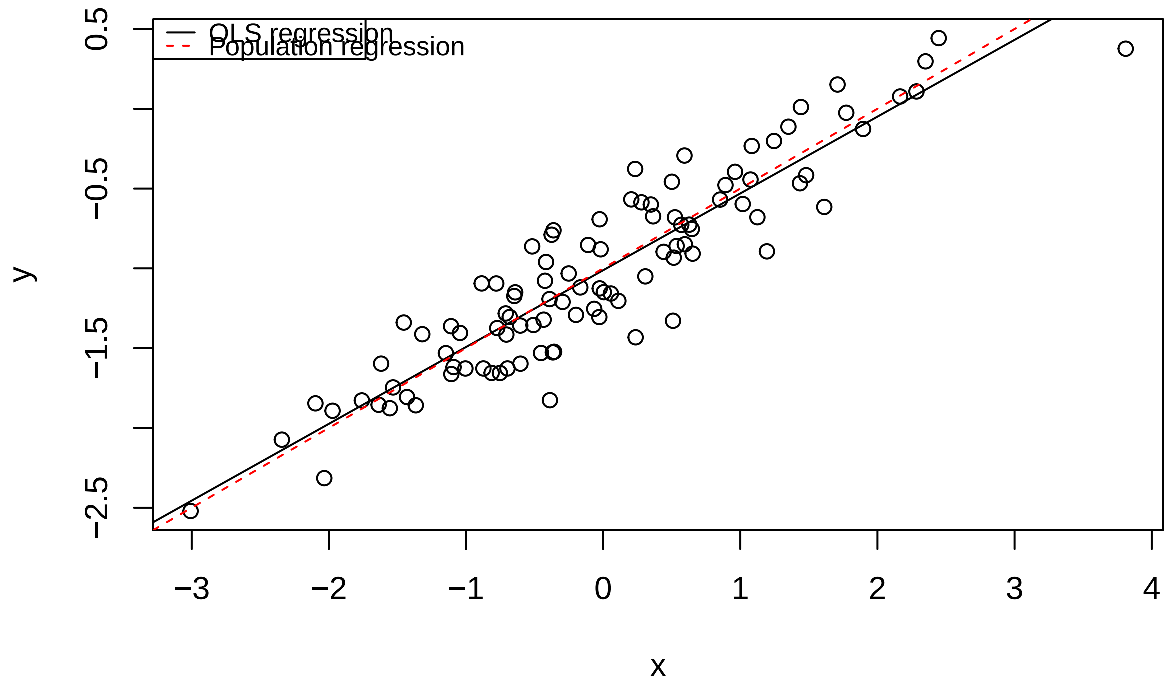
```
fit1= lm(y~x)
abline(coef(fit1))
```



The beta estimates are very close to the actual parameter. However, beta1 is statistically significantly different from 0.5.

Scatterplot:

```
windows(width=10, height=8)
x<- rnorm(100)
eps<- rnorm(100, 0, 0.25)
y<- -1 + 0.5*x + eps
plot(x,y)
fit1<- lm(y~x)
abline(coef(fit1))
abline(-1,0.5,col = 'red',lty = 2)
legend('topleft',
       legend = c("OLS regression","Population regression"),
       col = c('black','red'),
       lty = 1:2, cex = 0.8,
       )
```



Polynomial regression

```
fit2= lm(y~x+I(x^2))
summary(fit2)
```

```
##
## Call:
## lm(formula = y ~ x + I(x^2))
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.63403 -0.15501  0.00207  0.18784  0.51415
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) -1.003703   0.029171 -34.407   <2e-16 ***
## x            0.483237   0.021218  22.775   <2e-16 ***
## I(x^2)      -0.005961   0.011809  -0.505    0.615
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.2427 on 97 degrees of freedom
## Multiple R-squared:  0.846,  Adjusted R-squared:  0.8428
## F-statistic: 266.4 on 2 and 97 DF,  p-value: < 2.2e-16
```

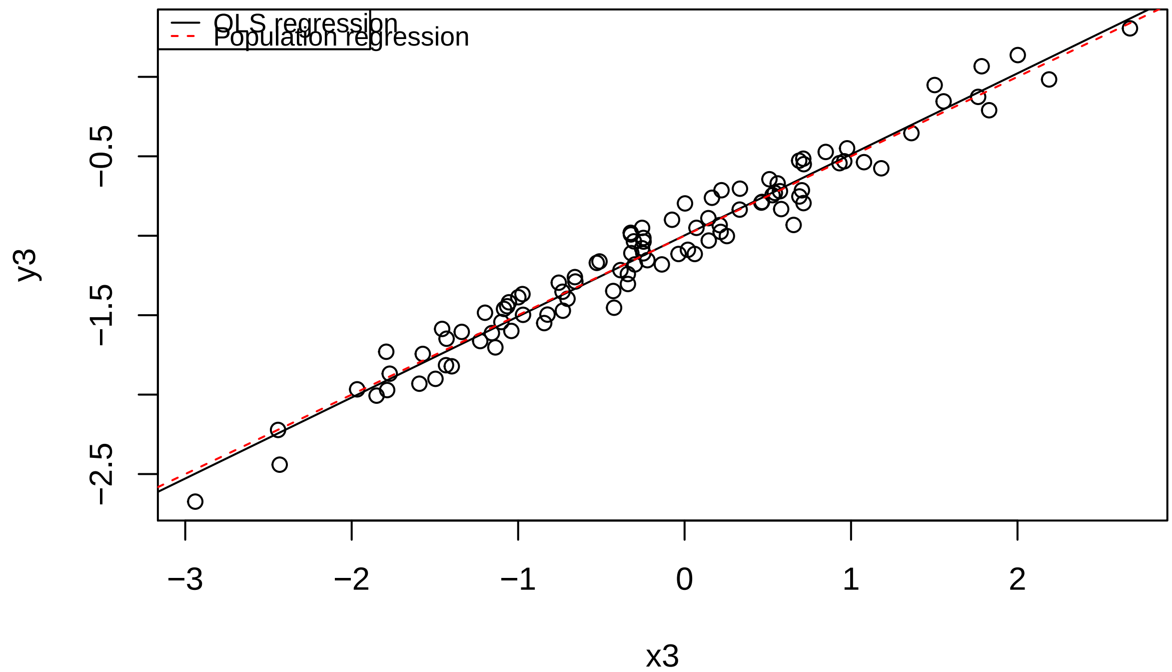
It does not help because the R-sqaured decreases.

Less noise approach:

```r
windows(width=10, height=8)
x3 = rnorm(100)
eps= rnorm(100, 0, 0.1)
y3 = -1 + 0.5*x3 + eps
plot(x3,y3)
fit3= lm(y3~x3)
summary(fit3)
```

```
##
## Call:
## lm(formula = y3 ~ x3)
##
## Residuals:
##       Min        1Q    Median        3Q       Max
## -0.268223 -0.088552 -0.007682  0.092975  0.200080
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) -0.99797    0.01116  -89.41   <2e-16 ***
## x3           0.51005    0.01021   49.98   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.1097 on 98 degrees of freedom
## Multiple R-squared:  0.9622, Adjusted R-squared:  0.9619
## F-statistic:  2498 on 1 and 98 DF,  p-value: < 2.2e-16
```

```r
abline(coef(fit3))
abline(-1,0.5,col = 'red',lty = 2)
legend('topleft',
       legend = c("OLS regression","Population regression"),
       col = c('black','red'),
       lty = 1:2, cex = 0.8,
)
```



The estimated and actual regression line is almost identical.

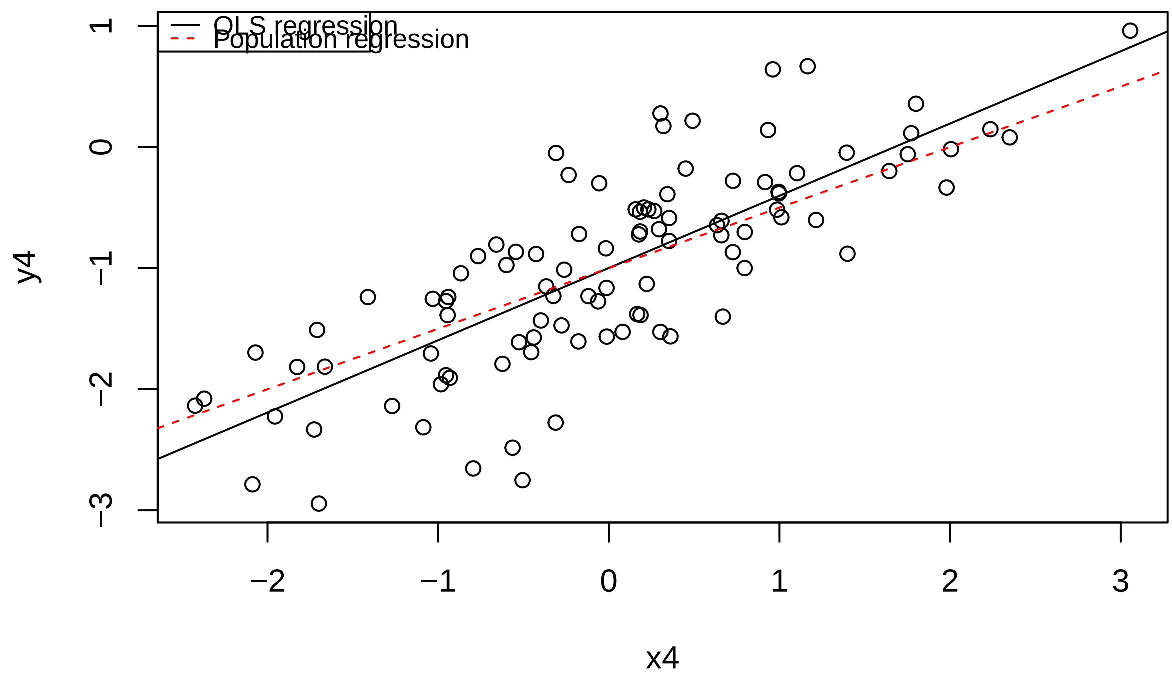
More noise approach:

```r
windows(width=10, height=8)
x4 = rnorm(100)
eps = rnorm(100, 0, 0.5)
y4 = -1 + 0.5*x4 + eps
plot(x4,y4)
fit4= lm(y4~x4)
summary(fit4)
```

```
##
## Call:
## lm(formula = y4 ~ x4)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1.45071 -0.31817  0.01701  0.34848  1.13426
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) -0.99894    0.05195  -19.23   <2e-16 ***
## x4           0.59663    0.04797   12.44   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
```

```
## Residual standard error: 0.5194 on 98 degrees of freedom
## Multiple R-squared:  0.6122, Adjusted R-squared:  0.6082
## F-statistic: 154.7 on 1 and 98 DF,  p-value: < 2.2e-16
```

```r
abline(coef(fit4))
abline(-1,0.5,col = 'red',lty = 2)
legend('topleft',
       legend = c("OLS regression","Population regression"),
       col = c('black','red'),
       lty = 1:2, cex = 0.8,
)
```



The estimates are less accurate because they have lower t value. The R-squared is lower.

```r
confint(fit1)
```

```
##                  2.5 %     97.5 %
## (Intercept) -1.0598693 -0.9638569
## x            0.4400348  0.5224908
```

```r
confint(fit3)
```

```
##                  2.5 %     97.5 %
## (Intercept) -1.0201202 -0.9758185
## x3           0.4897954  0.5302986
```

```r
confint(fit4)
```

```
##                  2.5 %     97.5 %
## (Intercept) -1.1020276 -0.8958549
## x4           0.5014353  0.6918155
```

The width of confidence interval with noisier data is higher.