

KNN Regression

```
library('caret')
```

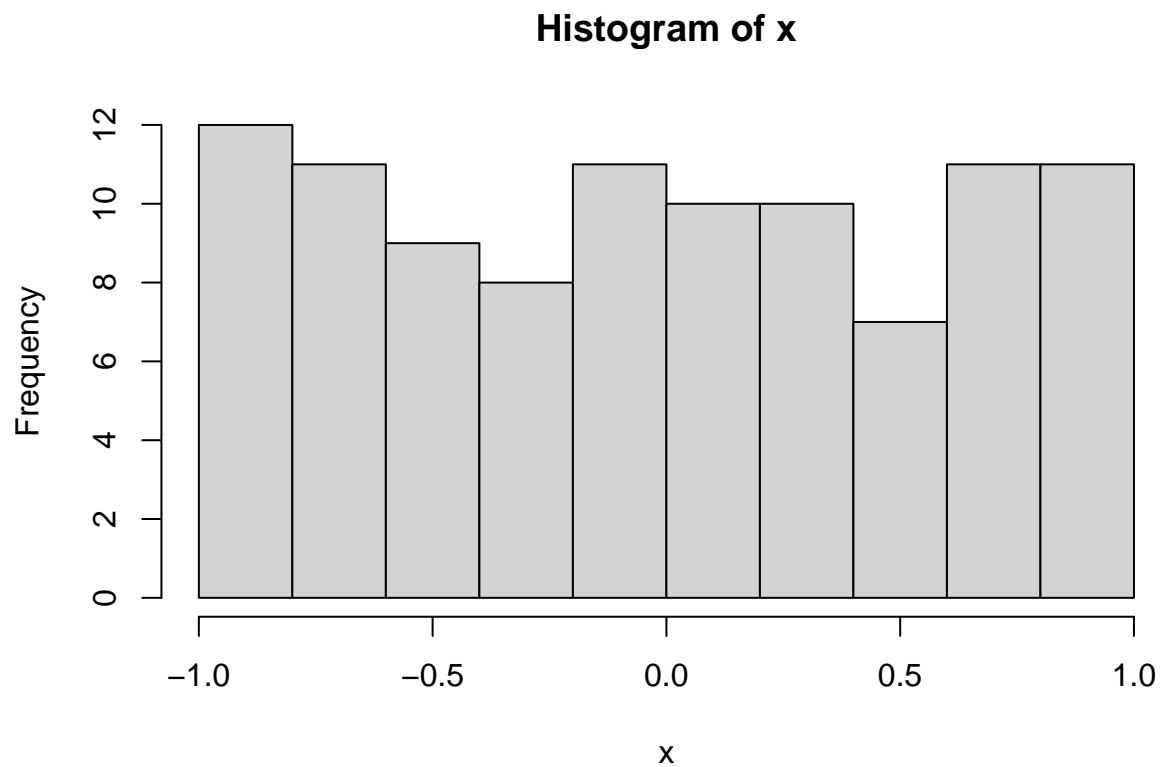
```
## Loading required package: ggplot2
```

```
## Loading required package: lattice
```

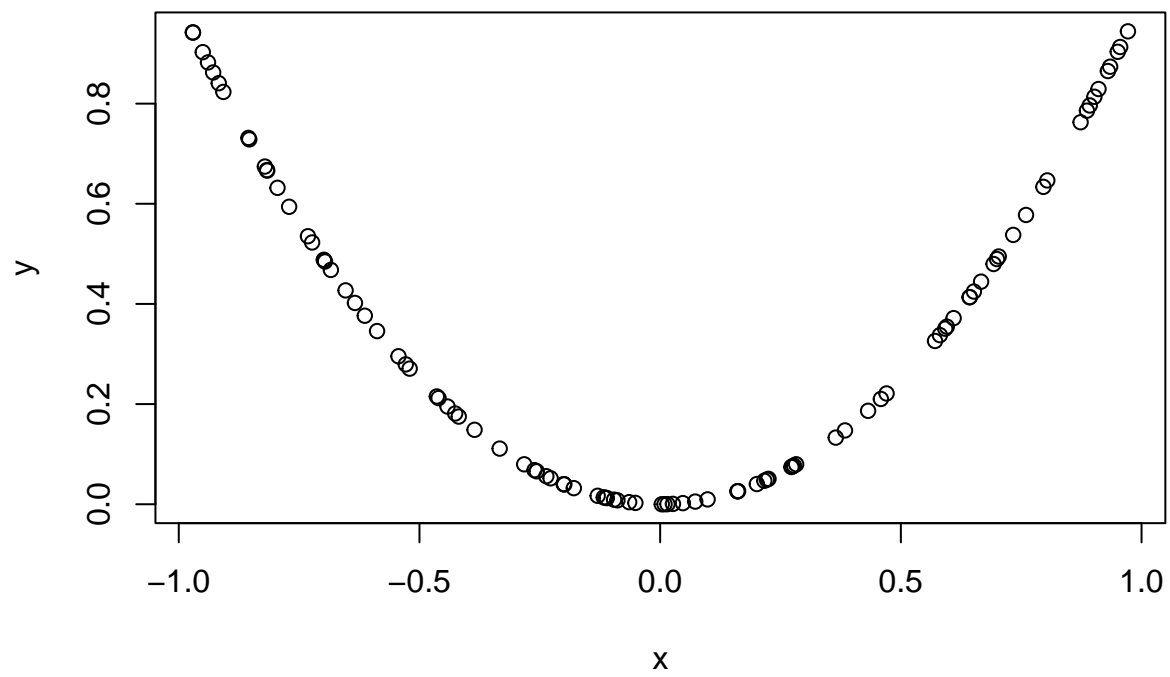
simulation

```
?runif
```

```
x = runif(100,-1,1)
x_ord = order(x)
hist(x)
```

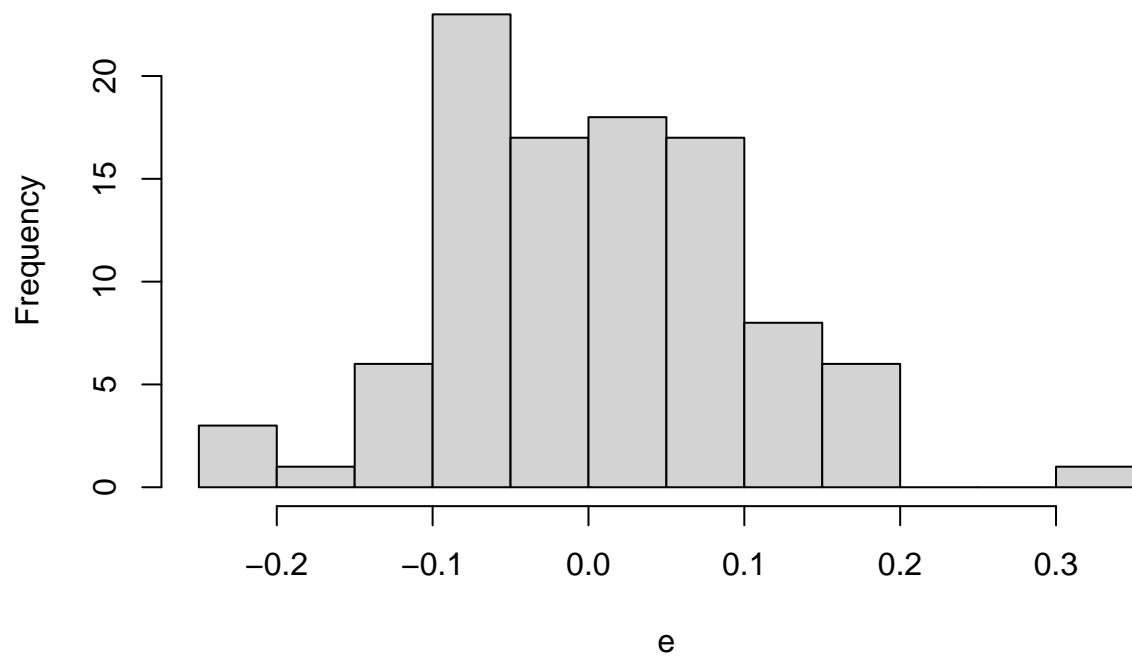


```
y = x^2 #  $f(x) = x^2$ 
plot(x,y)
```

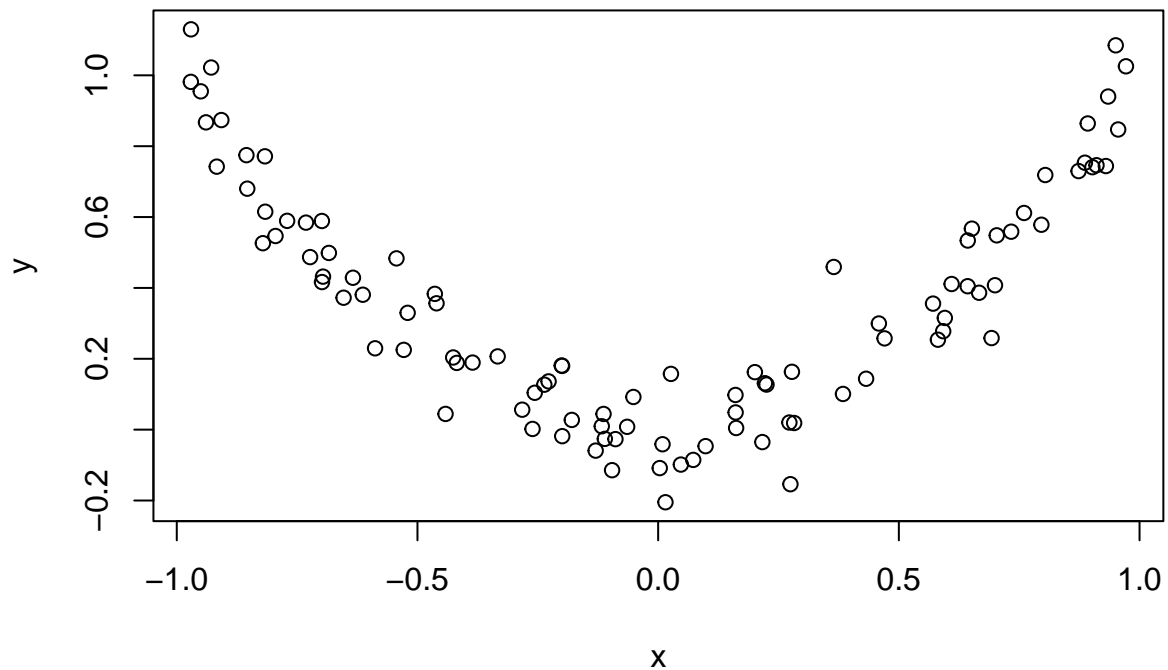


```
e = rnorm(100,0,1/10)
hist(e)
```

Histogram of e



```
y = x^2 + e  
plot(x,y)
```

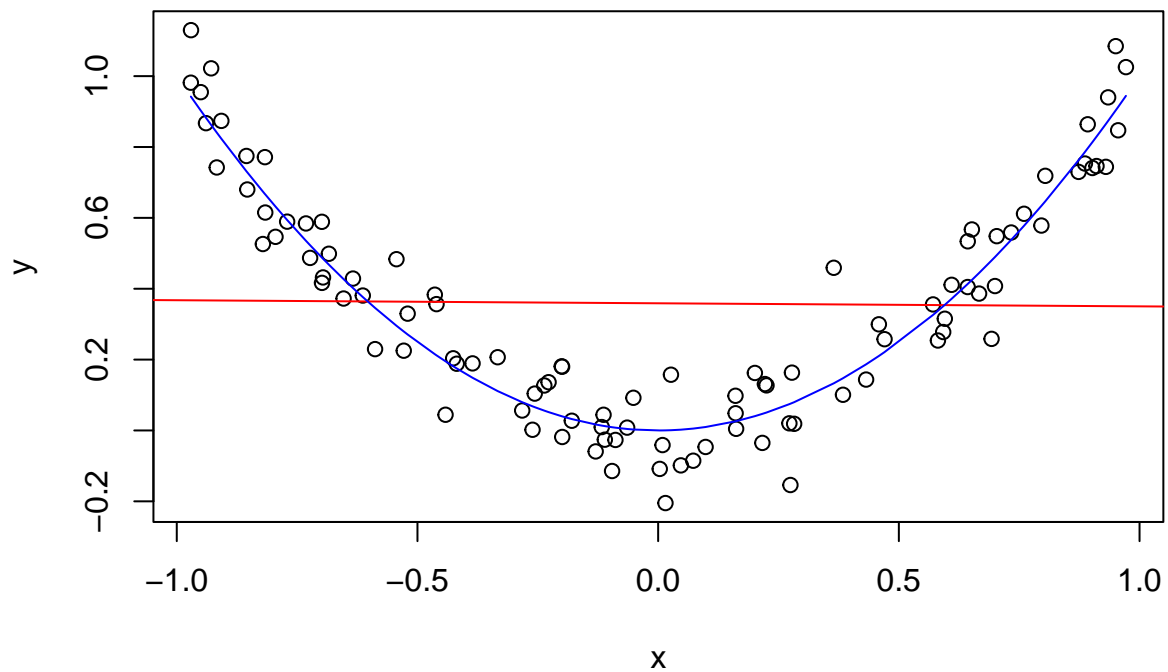


using linear regression

```
mod = lm(y~x) #  $y = b_0 + b_1x$ 
summary(mod)
```

```
##
## Call:
## lm(formula = y ~ x)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.56369 -0.26154 -0.03615  0.22379  0.76243
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  0.358937   0.033256  10.793  <2e-16 ***
## x           -0.008645   0.055945  -0.155    0.878
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.3324 on 98 degrees of freedom
## Multiple R-squared:  0.0002436, Adjusted R-squared:  -0.009958
## F-statistic: 0.02388 on 1 and 98 DF,  p-value: 0.8775
```

```
plot(x,y)
abline(coef(mod),col='red')
lines(x[x_ord],x[x_ord]^2,col='blue')
```



```
dim(model.matrix(mod))
```

```
## [1] 100 2
```

KNN

```
?knnreg
```

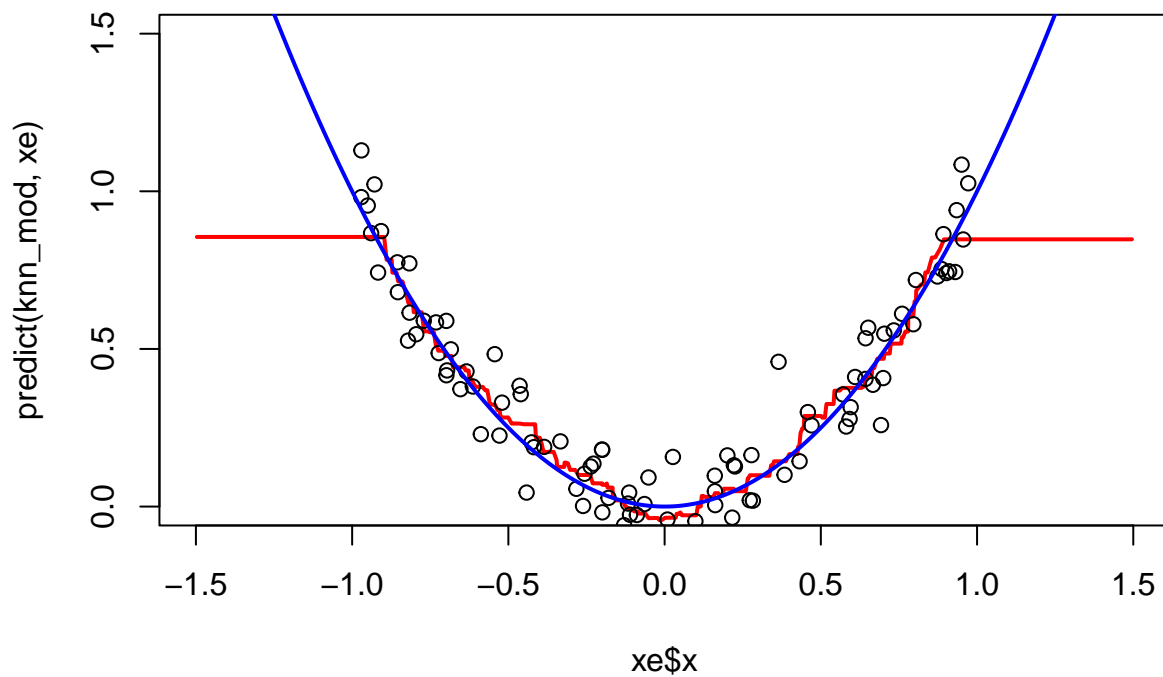
```
df = data.frame(x=x,y=y)
knn_mod = knnreg(y~.,data=df,k=10)
```

```
xe = data.frame(x=sort(runif(1000,-1.5,1.5)))
```

```
head(sample(predict(knn_mod,xe)))
```

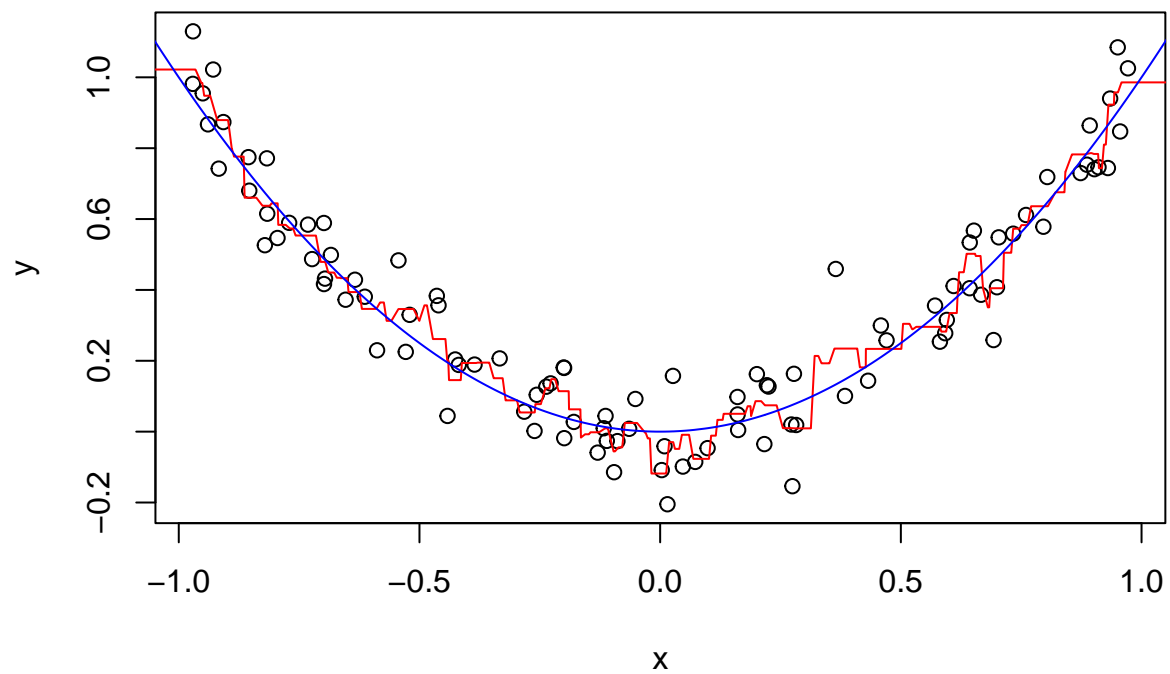
```
## [1] 0.28743444 0.84757476 0.01973762 0.26103180 0.84757476 0.84757476
```

```
plot(xe$x,predict(knn_mod,xe),col='red',lwd=2,type='l',ylim=c(0,1.5))
points(x,y)
lines(xe$x,xe$x^2,col='blue',lwd=2)
```



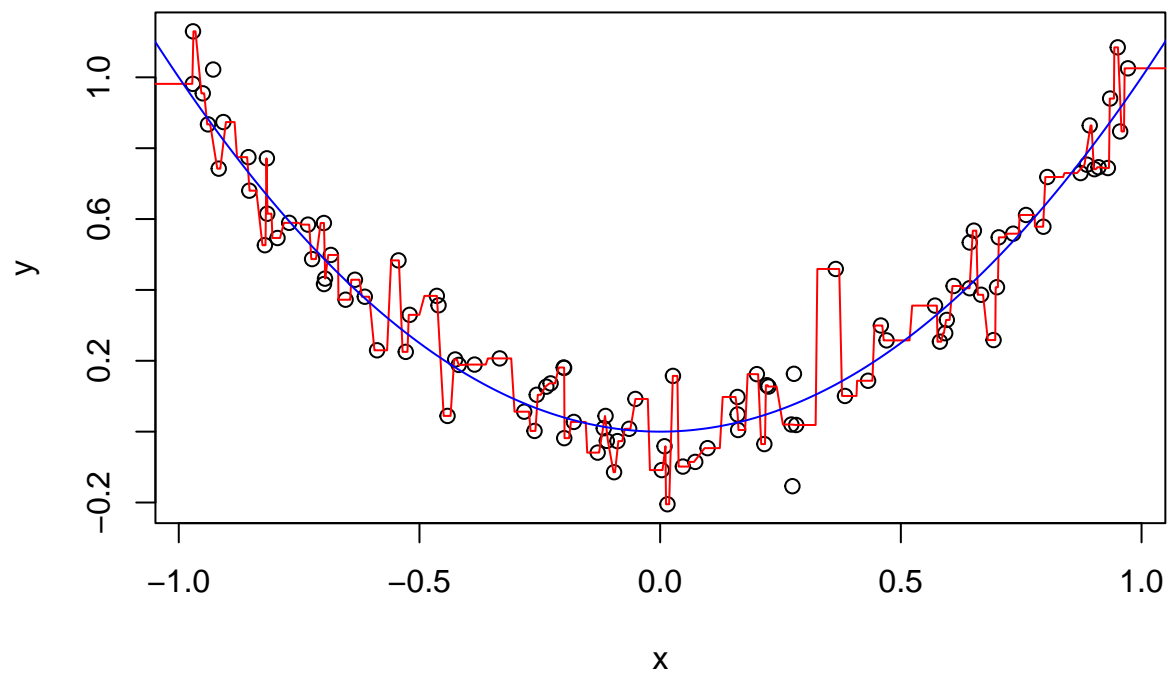
$k = 3$

```
knn_mod = knnreg(y~.,data=df,k=3)
plot(x,y)
lines(xe$x,predict(knn_mod,xe),col='red')
lines(xe$x,xe$x^2,col='blue')
```



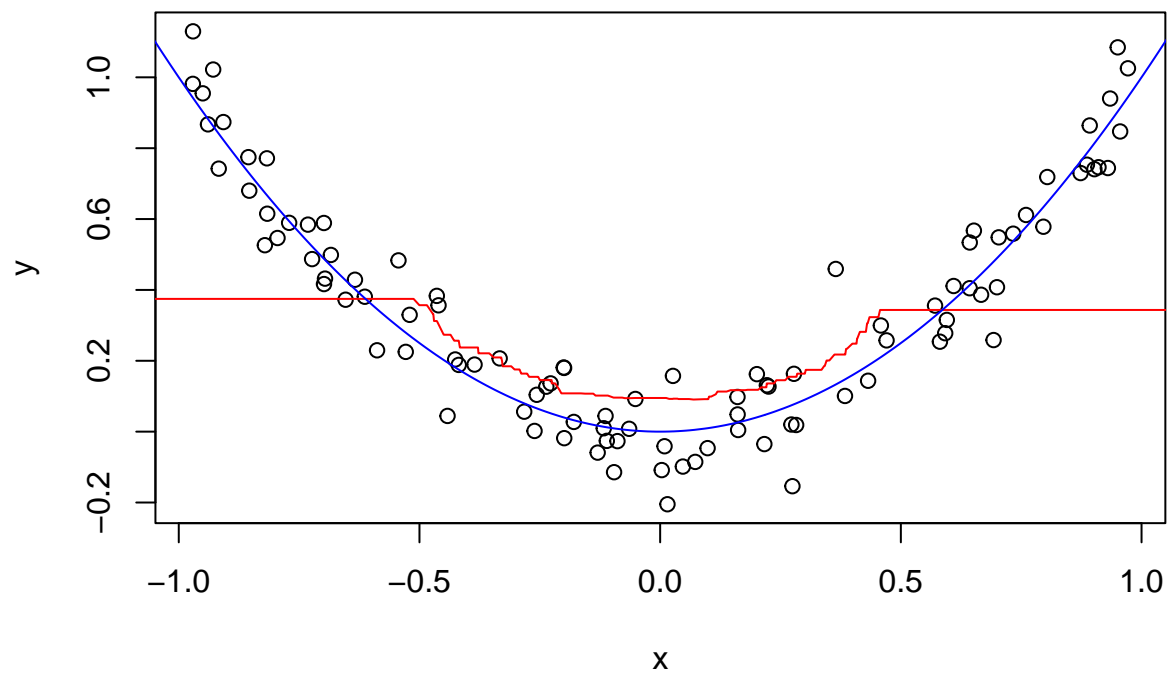
k = 1

```
knn_mod = knnreg(y~.,data=df,k=1)
plot(x,y)
lines(xe$x,predict(knn_mod,xe),col='red')
lines(xe$x,xe$x^2,col='blue')
```



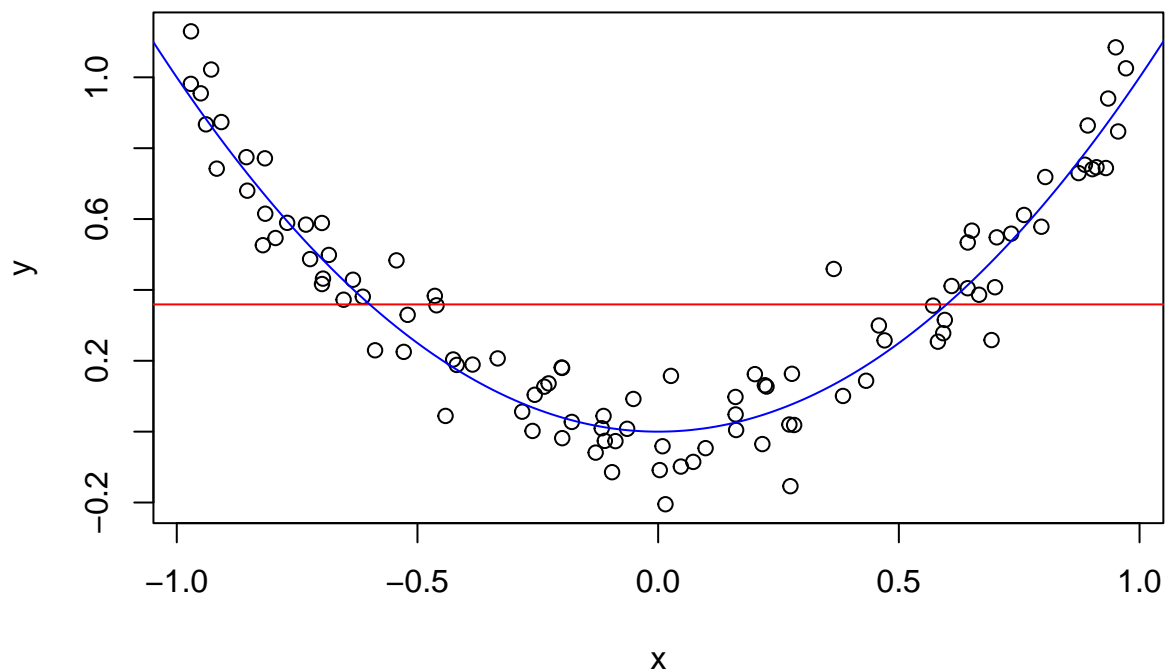
k = 50

```
knn_mod = knnreg(y~.,data=df,k=50)
plot(x,y)
lines(xe$x,predict(knn_mod,xe),col='red')
lines(xe$x,xe$x^2,col='blue')
```

k = 100

```
knn_mod = knnreg(y~.,data=df,k=100)
plot(x,y)
lines(xe$x,predict(knn_mod,xe),col='red')
lines(xe$x,xe$x^2,col='blue')
```



```
#abline(h=mean(y),lty=2,col='orange')
```

```
mean(y)
```

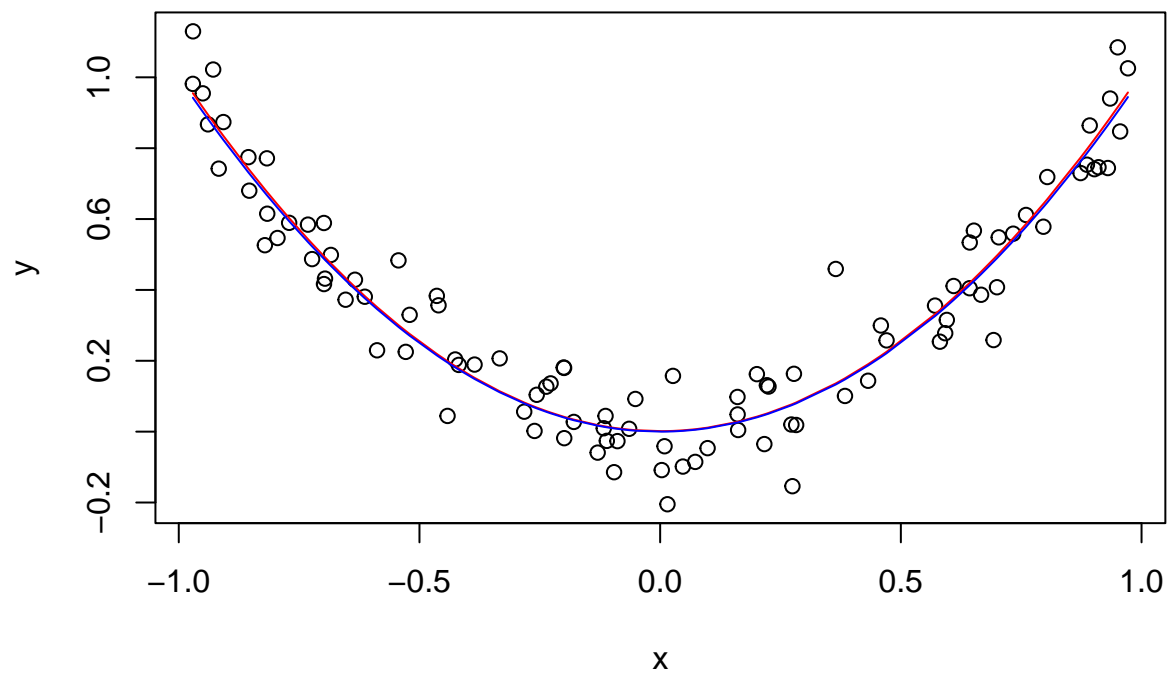
```
## [1] 0.3590736
```

```
head(predict(knn_mod))
```

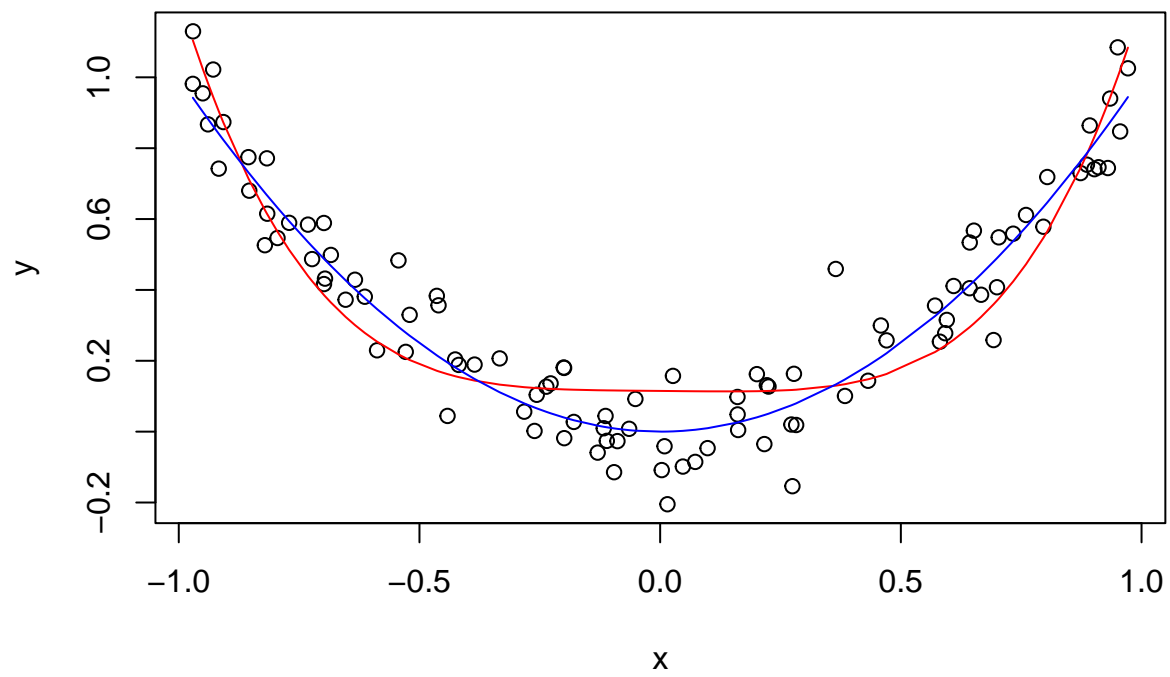
```
## [1] 0.3590736 0.3590736 0.3590736 0.3590736 0.3590736 0.3590736
```

Can we do this with a linear model?

```
lin_mod = lm(y~I(x^2)) # y = b0+b1x^2
plot(x,y)
lines(x[x_ord],predict(lin_mod)[x_ord],col='red')
lines(x[x_ord],x[x_ord]^2,col='blue')
```



```
lin_mod = lm(y~x+I(x^4)) #  $y = b_0 + b_1x^2$ 
plot(x,y)
lines(x[x_ord],predict(lin_mod)[x_ord],col='red')
lines(x[x_ord],x[x_ord]^2,col='blue')
```



```
dim(model.matrix(lin_mod))
```

```
## [1] 100  3
```