

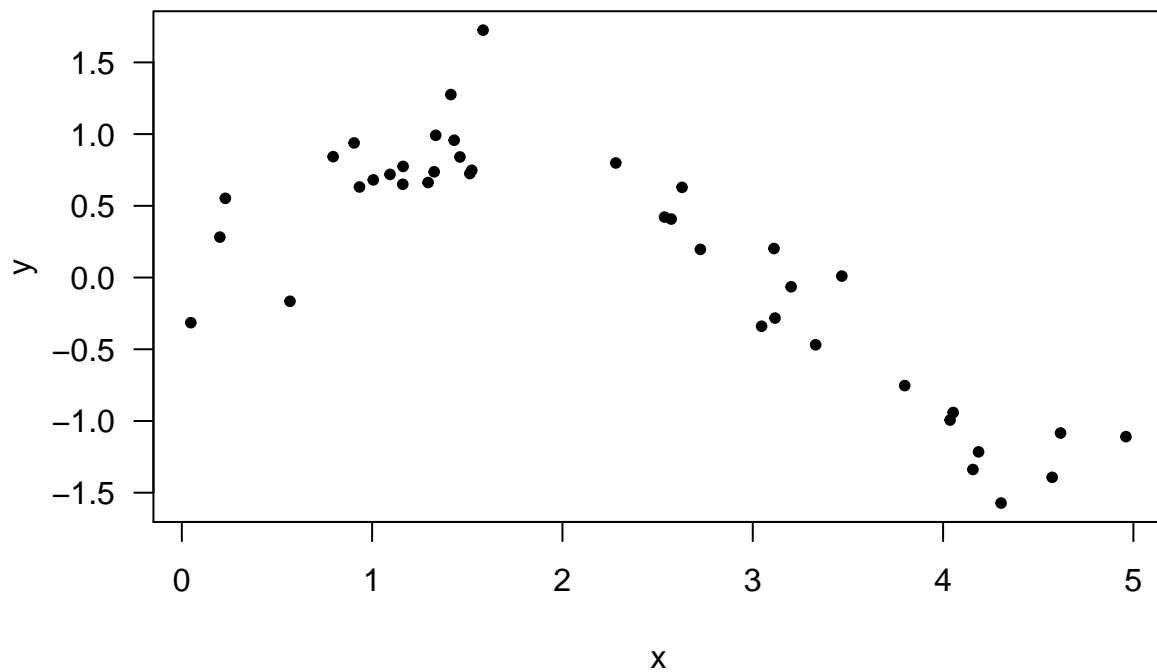
Ejemplo-01-regresion-con-kernlab.R

fhern

2020-09-23

```
# En este ejemplo se usan datos artificiales (simulados) para mostrar  
# el uso de svm en regresion
```

```
# Creando los datos -----  
set.seed(1234)  
x <- sort(runif(n=40, min=0, max=5)) # sort for convenience  
set.seed(1234)  
y <- sin(x) + rnorm(40, sd=0.3)  
plot(x, y, pch=20, las=1)
```



```
# Funcion para calcular MSE -----  
mse <- function(y, y_hat) mean((y - y_hat)^2)
```

```

# sum lineal -----
library(kernlab)

# Para ajustar el modelo
mod_lin <- ksvm(y ~ x, type="eps-svr", kernel="vanilladot",
               C=1, epsilon=0.1)

## Setting default kernel parameters

# To obtain y_hat
y_hat_lin <- predict(mod_lin)

# To obtain the correlation coefficient and MSE.
cor(y, y_hat_lin)

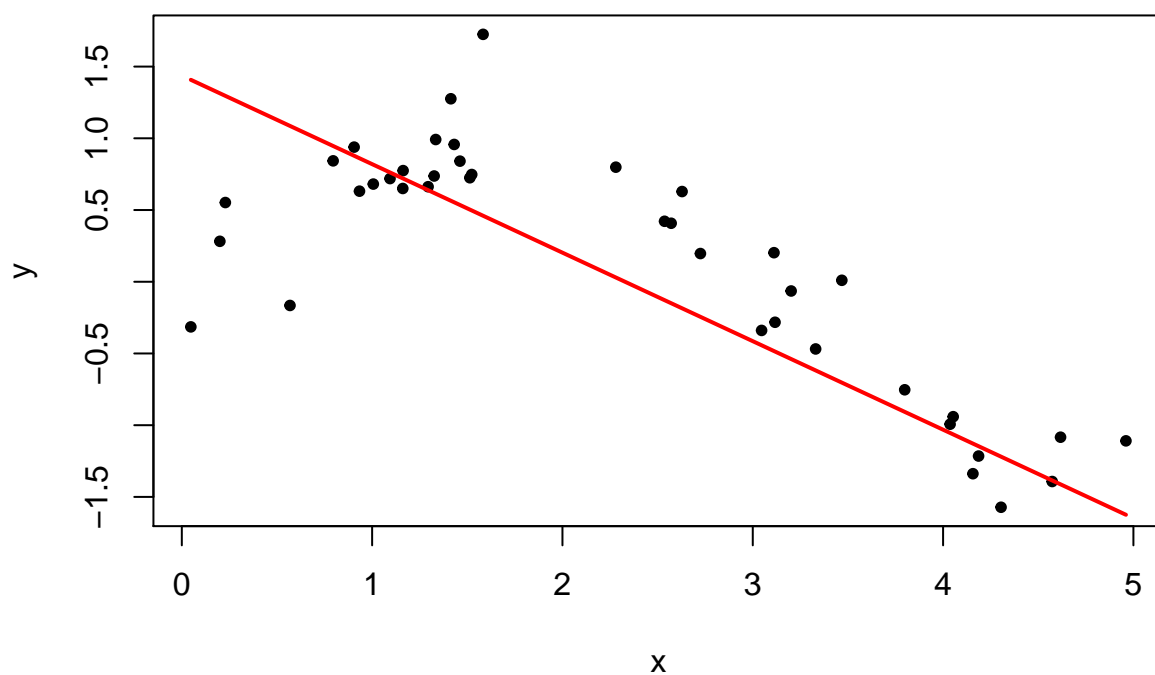
##           [,1]
## [1,] 0.7852445

mse(y, y_hat_lin)

## [1] 0.3201497

# To illustrate the results
plot(x, y, pch=20)
points(x=x, y=y_hat_lin, type="l", lwd=2, col="red")

```



```
# sum polinomial -----
```

```
# Para ajustar el modelo con los hiper-parametros por defecto
```

```
mod_pol <- ksvm(y ~ x, type="eps-svr", kernel="polydot",  
               C=1, epsilon=0.1,  
               kpar=list(degree=1, scale=1, offset=1))
```

```
# To obtain y_hat
```

```
y_hat_pol <- predict(mod_pol)
```

```
# To obtain the correlation coefficient and MSE.
```

```
cor(y, y_hat_pol)
```

```
##           [,1]
```

```
## [1,] 0.7852445
```

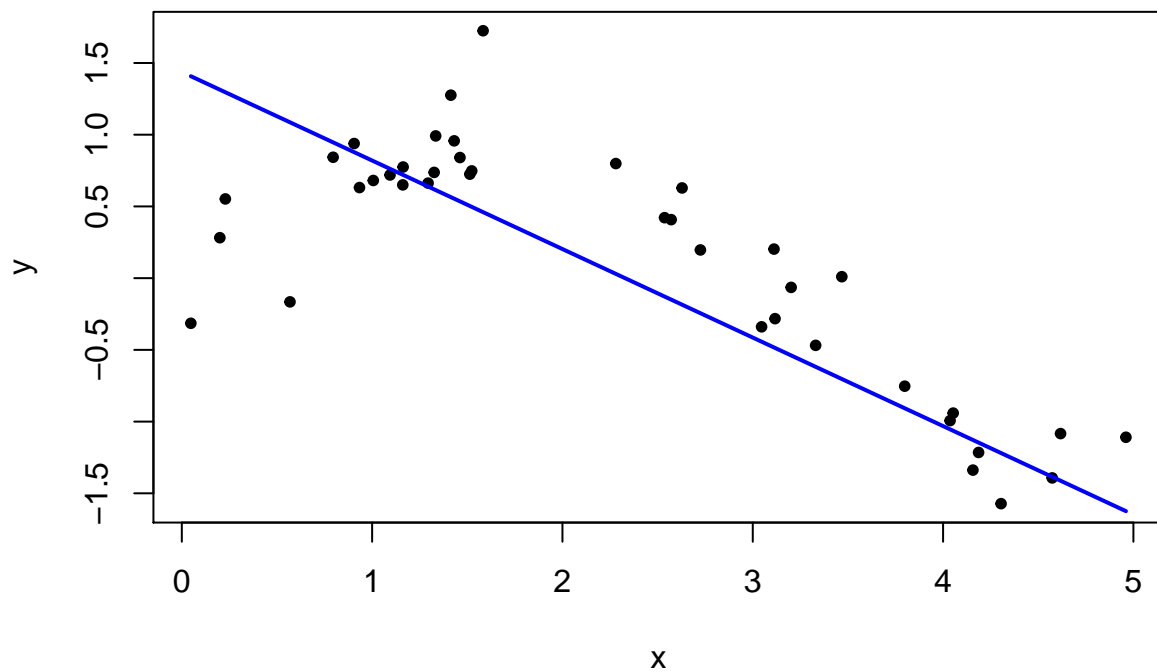
```
mse(y, y_hat_pol)
```

```
## [1] 0.3201497
```

```
# To illustrate the results
```

```
plot(x, y, pch=20)
```

```
points(x=x, y=y_hat_pol, type="l", lwd=2, col="blue")
```



```
# svm radial -----

# Para ajustar el modelo con los hiper-parametros por defecto
mod_rad <- ksvm(y ~ x, type="eps-svr", kernel="rbfdot",
               C=1, epsilon=0.1,
               kpar=list(sigma=1))
```

```
# To obtain y_hat
y_hat_rad <- predict(mod_rad)
```

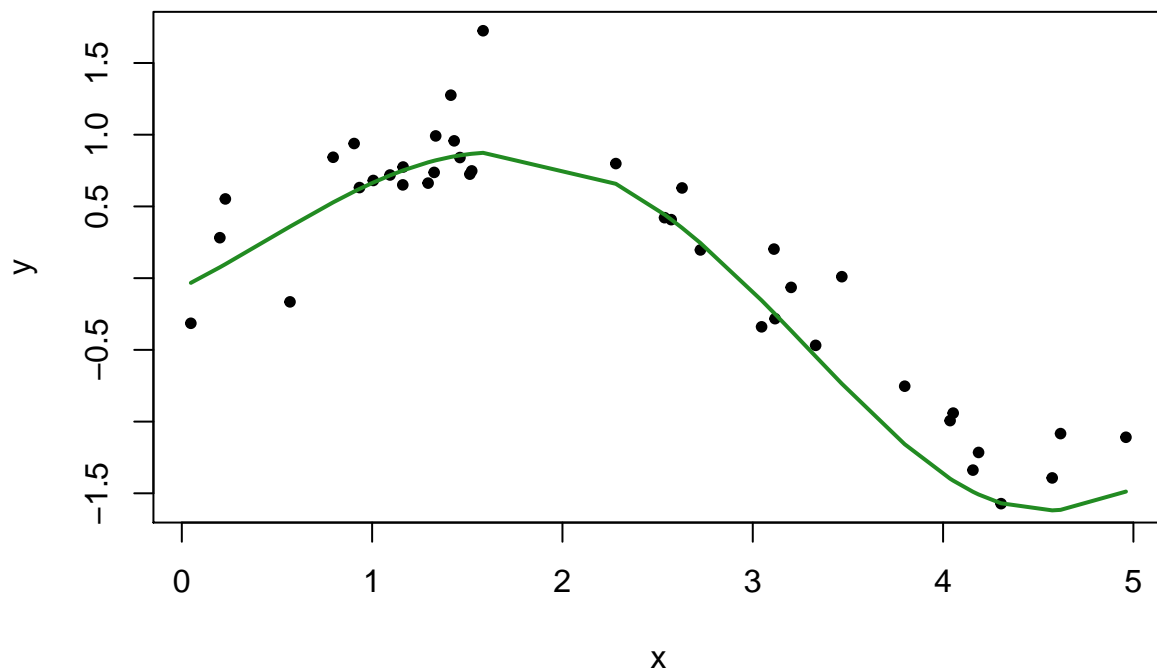
```
# To obtain the correlation coefficient and MSE.
cor(y, y_hat_rad)
```

```
##           [,1]
## [1,] 0.9506258
```

```
mse(y, y_hat_rad)
```

```
## [1] 0.09900929
```

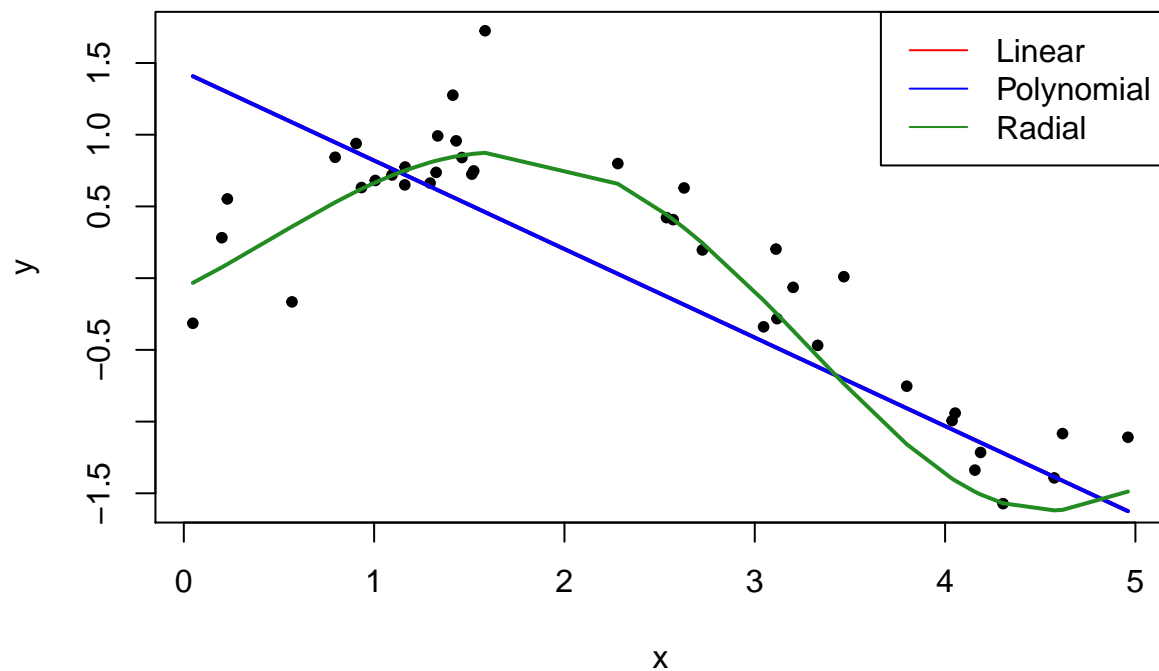
```
# To illustrate the results
plot(x, y, pch=20)
points(x=x, y=y_hat_rad, type="l", lwd=2, col="forestgreen")
```



```

# Comparing -----
plot(x, y, pch=20)
points(x=x, y=y_hat_lin, type="l", lwd=2, col="red")
points(x=x, y=y_hat_pol, type="l", lwd=2, col="blue")
points(x=x, y=y_hat_rad, type="l", lwd=2, col="forestgreen")
legend("topright", lty=1,
      col=c("red", "blue", "forestgreen"),
      legend=c("Linear", "Polynomial", "Radial"))

```



```

# Tuning parameters -----

# En stackoverflow un usuario hizo una pregunta interesante al respecto
# visitar
# https://stackoverflow.com/questions/26459650/tuning-ksum-from-kernlab

```