SVM_R

Odo Luo

March 30, 2021

Homework 4

Using R and Python, train SVM regression models using the BostonHousing data (in R: data("BostonHousing", package = "mlbench"); in Python: from sklearn import datasets; datasets.load_boston()). Try the linear, one polynomial and the RBF kernel. Tune the parameters. Compare the models on a test set using the RMSE. Create a scatterplot with the fitted values of the best model against the true values.

```
library(mlbench)
library(dplyr)
```

```
##
## Attaching package: 'dplyr'
```

```
## The following objects are masked from 'package:stats':
##
##     filter, lag
```

```
## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```
data(BostonHousing)
data<- BostonHousing[,-4]
summary(data)
```

```
##       crim                zn             indus            nox
##  Min.   : 0.00632   Min.   :  0.00   Min.   : 0.46   Min.   :0.3850
##  1st Qu.: 0.08205   1st Qu.:  0.00   1st Qu.: 5.19   1st Qu.:0.4490
##  Median : 0.25651   Median :  0.00   Median : 9.69   Median :0.5380
##  Mean   : 3.61352   Mean   : 11.36   Mean   :11.14   Mean   :0.5547
##  3rd Qu.: 3.67708   3rd Qu.: 12.50   3rd Qu.:18.10   3rd Qu.:0.6240
##  Max.   :88.97620   Max.   :100.00   Max.   :27.74   Max.   :0.8710
##        rm             age              dis              rad
##  Min.   :3.561   Min.   :  2.90   Min.   : 1.130   Min.   : 1.000
##  1st Qu.:5.886   1st Qu.: 45.02   1st Qu.: 2.100   1st Qu.: 4.000
##  Median :6.208   Median : 77.50   Median : 3.207   Median : 5.000
##  Mean   :6.285   Mean   : 68.57   Mean   : 3.795   Mean   : 9.549
##  3rd Qu.:6.623   3rd Qu.: 94.08   3rd Qu.: 5.188   3rd Qu.:24.000
##  Max.   :8.780   Max.   :100.00   Max.   :12.127   Max.   :24.000
##       tax           ptratio            b               lstat
##  Min.   :187.0   Min.   :12.60   Min.   :  0.32   Min.   : 1.73
##  1st Qu.:279.0   1st Qu.:17.40   1st Qu.:375.38   1st Qu.: 6.95
##  Median :330.0   Median :19.05   Median :391.44   Median :11.36
##  Mean   :408.2   Mean   :18.46   Mean   :356.67   Mean   :12.65
##  3rd Qu.:666.0   3rd Qu.:20.20   3rd Qu.:396.23   3rd Qu.:16.95
```

```
##  Max.   :711.0   Max.   :22.00   Max.   :396.90   Max.   :37.97
##       medv
##  Min.   : 5.00
##  1st Qu.:17.02
##  Median :21.20
##  Mean   :22.53
##  3rd Qu.:25.00
##  Max.   :50.00
```

Train Test Split

```
data <- data %>%  mutate(id=row_number())
train<- sample_frac(data,0.7)
test<- anti_join(data, train,"id")
test <- test %>%  select(-id)
train <- train %>%  select(-id)
```

Tuning

```
library(e1071)
library(ggplot2)
library(Metrics)

range = 10 ^ (-4:4)
obj = tune.svm(medv ~ crim+ zn +indus  + nox   + rm   +age   + dis +rad +tax+ ptratio    +  b +lstat, da
print(obj$best.model)
```

```
##
## Call:
## best.svm(x = medv ~ crim + zn + indus + nox + rm + age + dis + rad +
##     tax + ptratio + b + lstat, data = train, gamma = range, cost = range)
##
##
## Parameters:
##    SVM-Type:  eps-regression
##  SVM-Kernel:  radial
##        cost:  10
##       gamma:  0.1
##     epsilon:  0.1
##
##
## Number of Support Vectors:  243
```

```
obj$best.parameters
```

```
##    gamma cost
## 49   0.1   10
```

Train & Prediction

```
gamma<- 0.1
cost<- 10

linear <- svm(medv ~ crim+ zn +indus  + nox + rm  +age + dis +rad +tax+ ptratio + b +lstat, data=train,
```

```
poly <- svm(medv ~ crim+ zn +indus  + nox + rm  +age + dis +rad +tax+ ptratio + b +lstat, data=train,ke
radial <- svm(medv ~ crim+ zn +indus  + nox + rm  +age + dis +rad +tax+ ptratio + b +lstat, data=train,

test_feature <- test[,-13]
test_target <- test[,13]

linear_prediction <- predict(linear,test_feature )
poly_prediction <- predict(poly,test_feature )
radial_prediction <- predict(radial,test_feature )
```

RSME

```
library(Metrics)
rmse(linear_prediction,test_target)
```

```
## [1] 4.442527
```

```
rmse(poly_prediction,test_target)
```

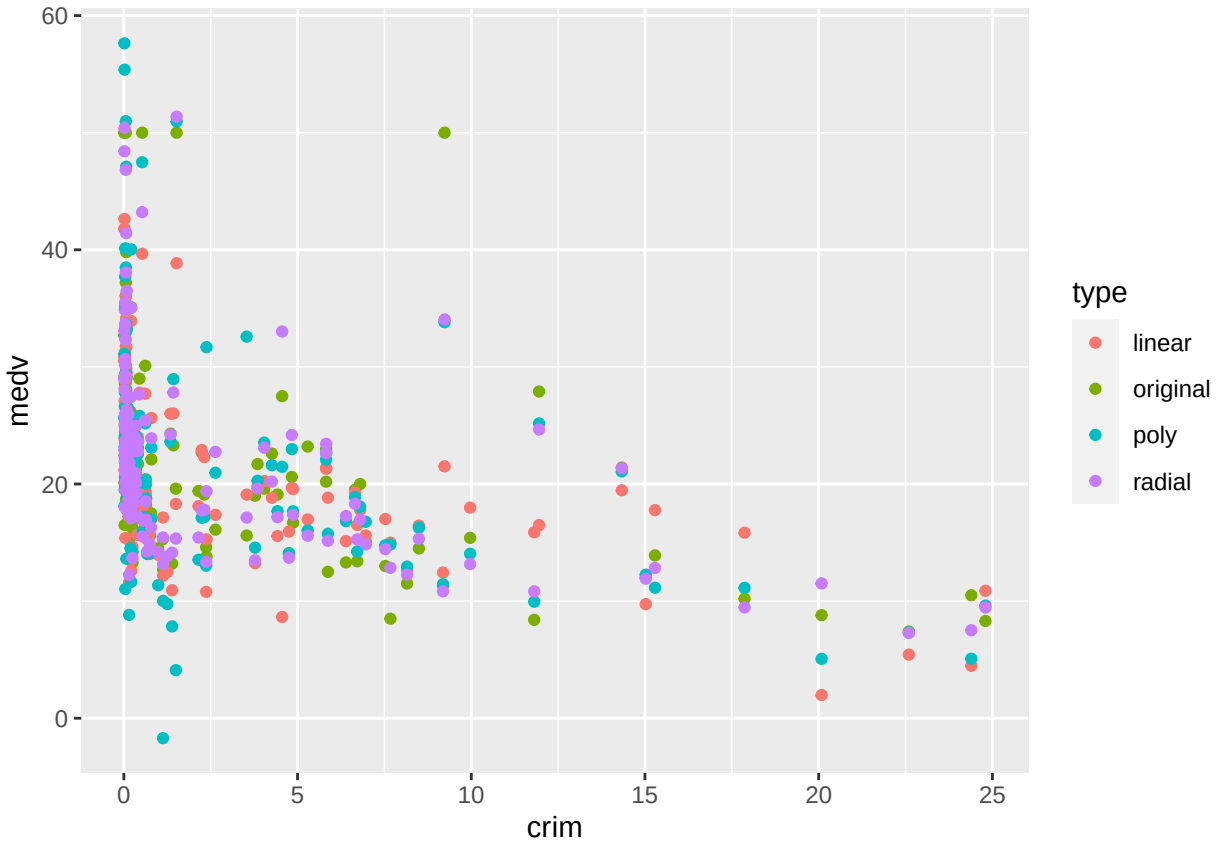
```
## [1] 4.30358
```

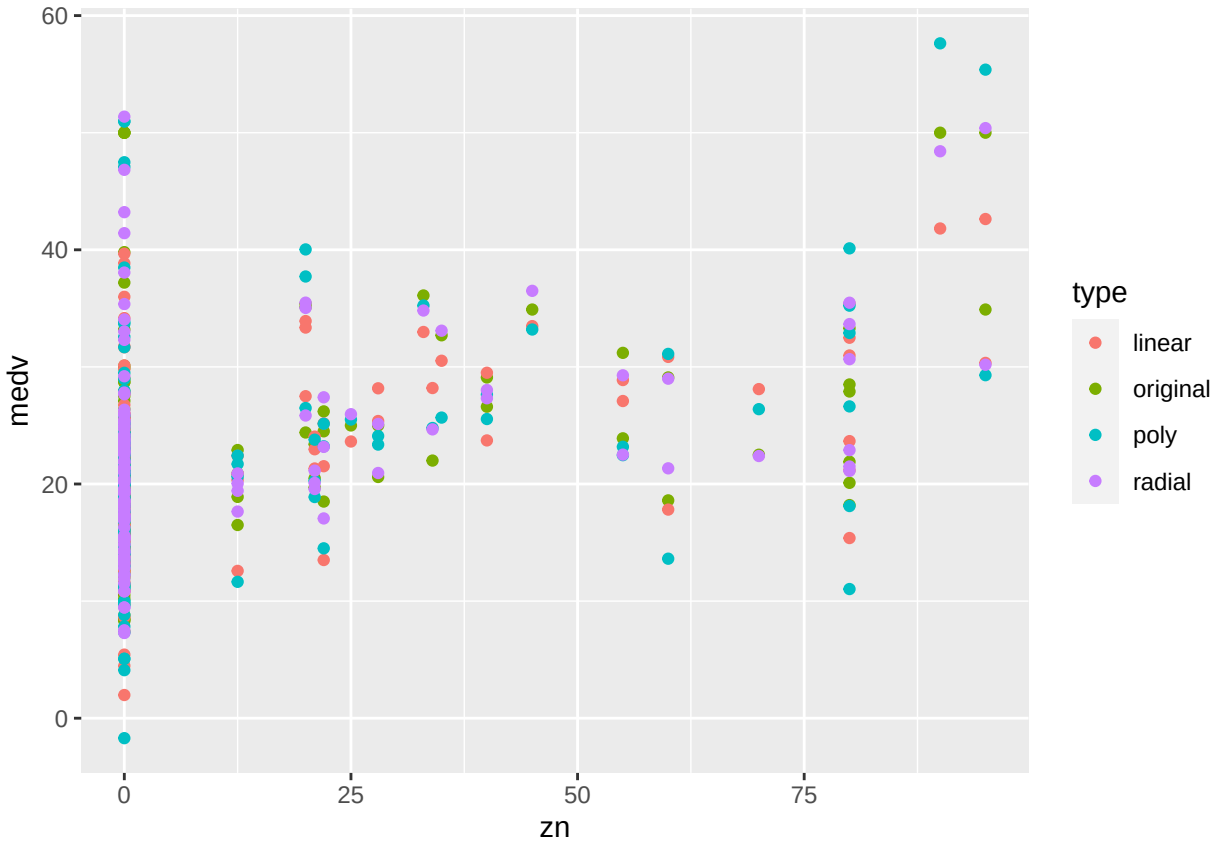
```
rmse(radial_prediction,test_target)
```

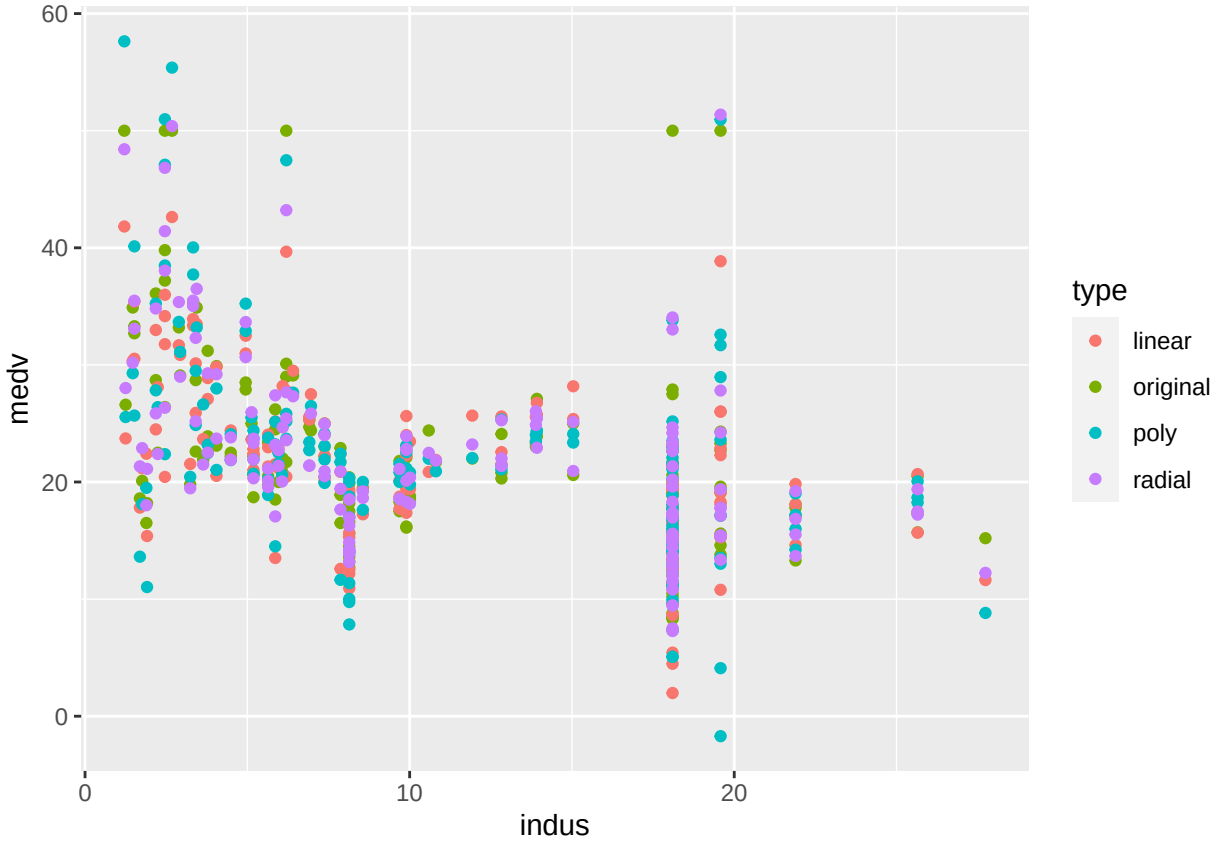
```
## [1] 2.652754
```

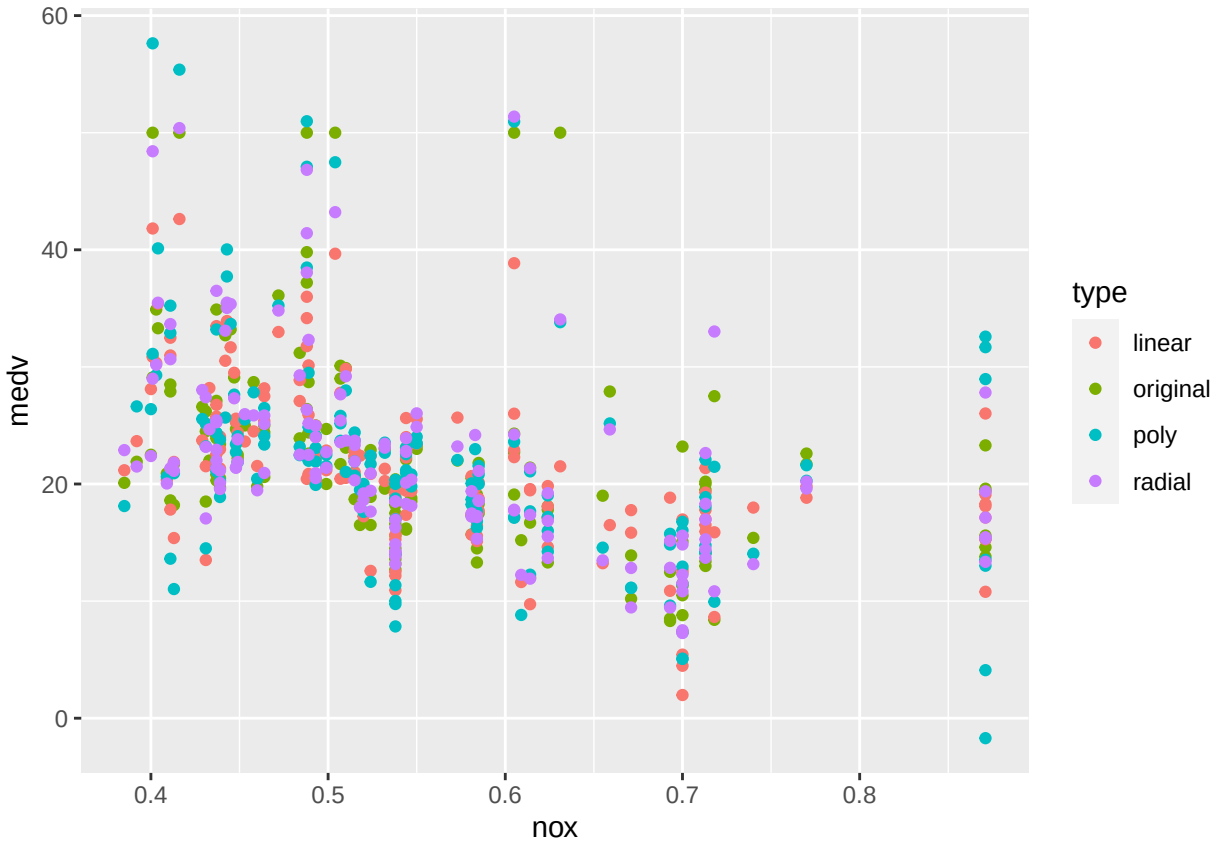
Plots

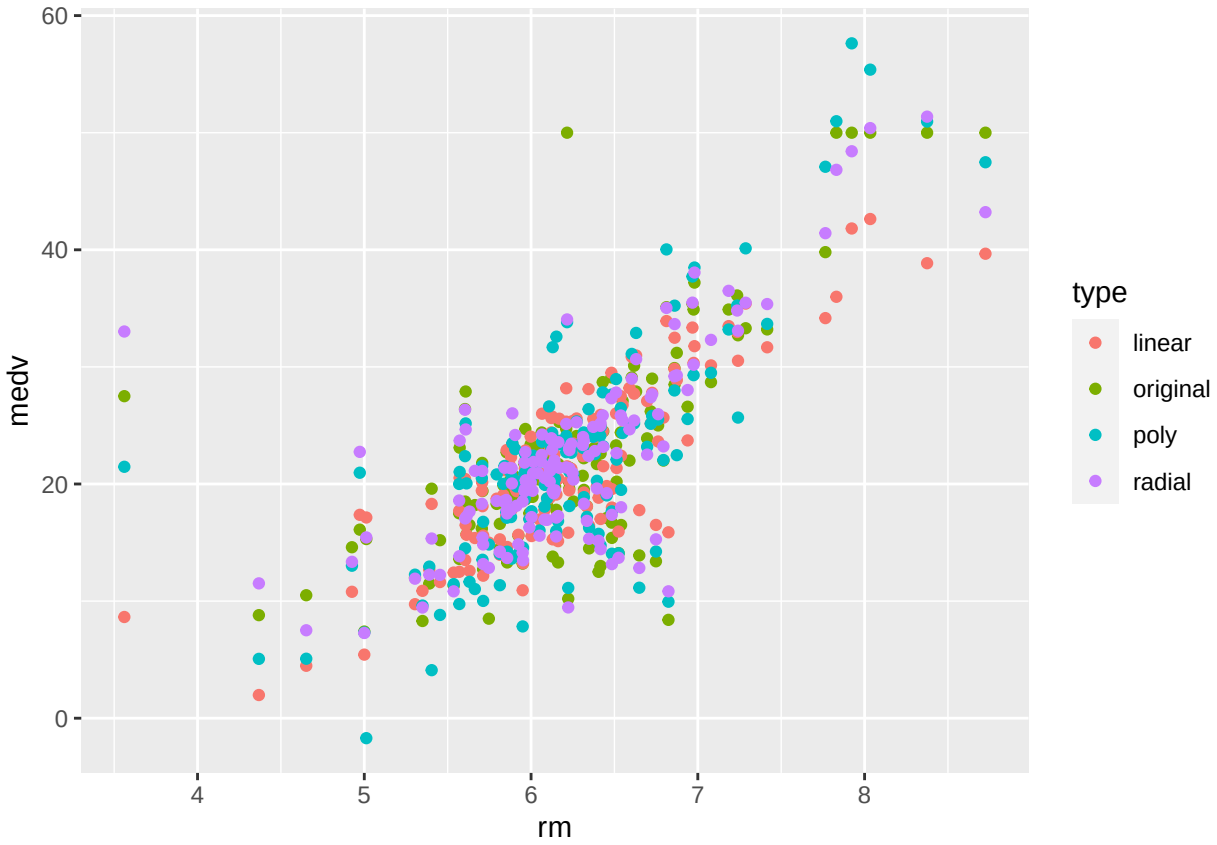
```
library(ggplot2)

o_data <- test %>%  mutate(type="original") %>%
  as_tibble

l_data <- test_feature %>% cbind(linear_prediction) %>%
  mutate(type="linear") %>%  as_tibble %>%
  mutate(medv=linear_prediction) %>% select(-linear_prediction)

p_data <- test_feature %>% cbind(poly_prediction) %>%
  mutate(type="poly") %>%  as_tibble %>%
  mutate(medv=poly_prediction) %>% select(-poly_prediction)

r_data <- test_feature %>% cbind(radial_prediction) %>%
  mutate(type="radial") %>%  as_tibble %>%
  mutate(medv=radial_prediction) %>% select(-radial_prediction)

x = o_data %>% bind_rows(l_data) %>%
  bind_rows(p_data) %>%
  bind_rows(r_data)

for(n in names(x)){
  p<-ggplot(x,aes_string(x=n,y="medv",color="type")) +
   geom_point()
  plot(p)
}
```

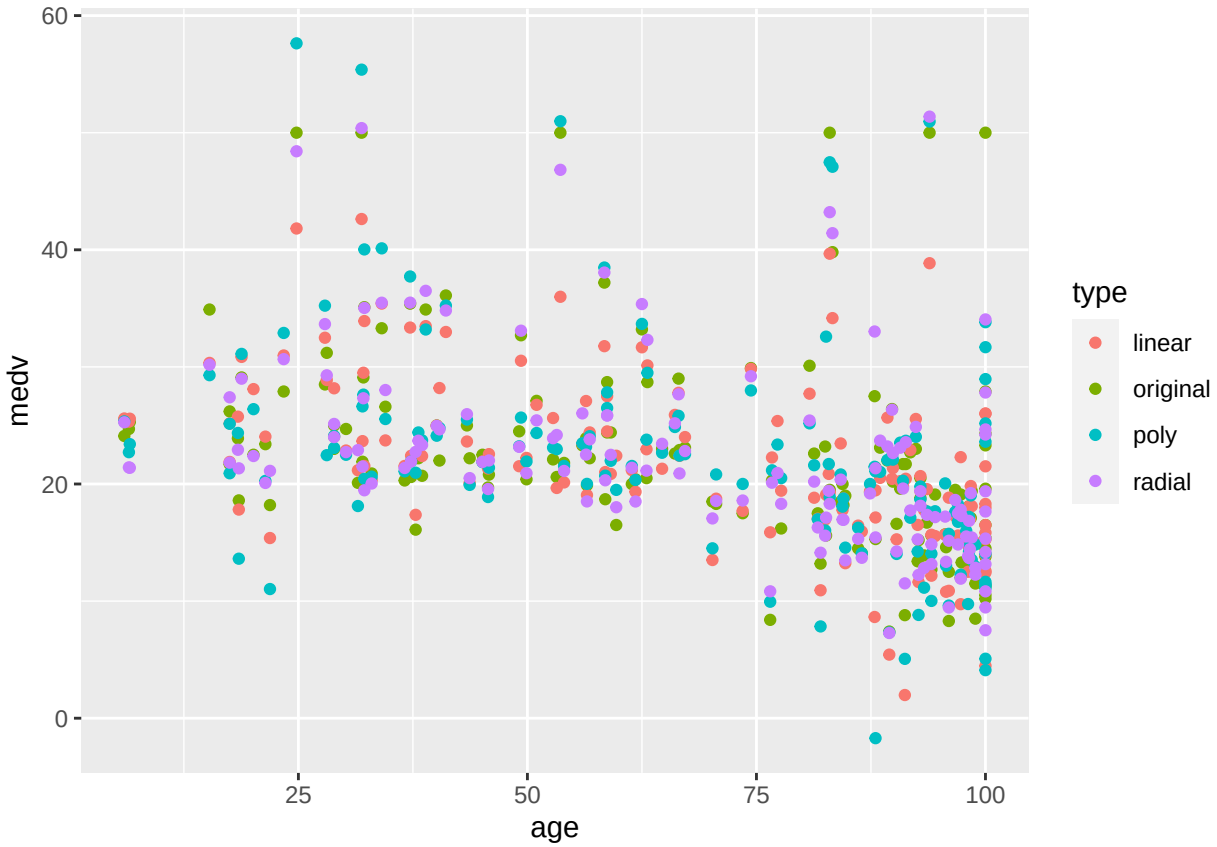

60
40
20
0
medv
0
5
10
15
20
25
crim
type
linear
original
poly
radial

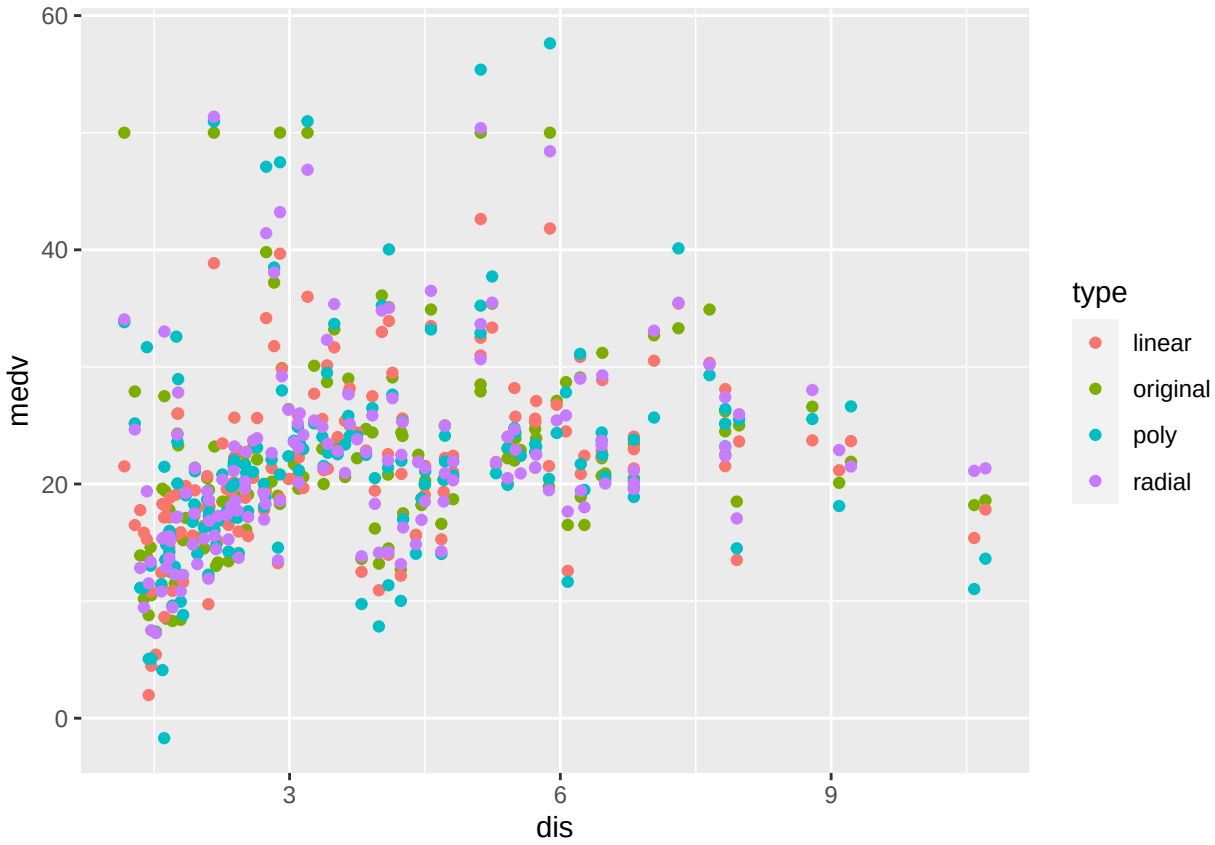

60
40
20
0
medv
0
25
50
75
zn
type
linear
original
poly
radial

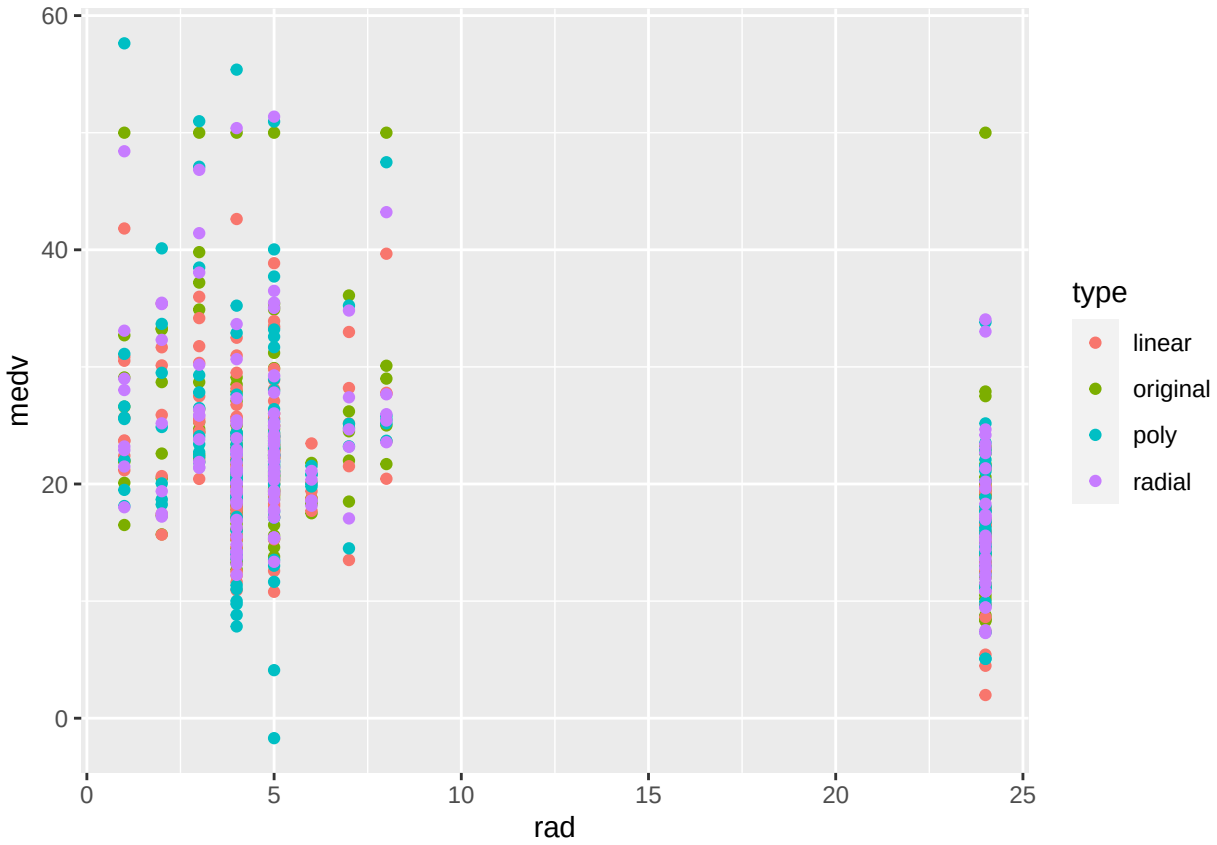

60
40
20
0
medv
0
10
20
indus
type
linear
original
poly
radial

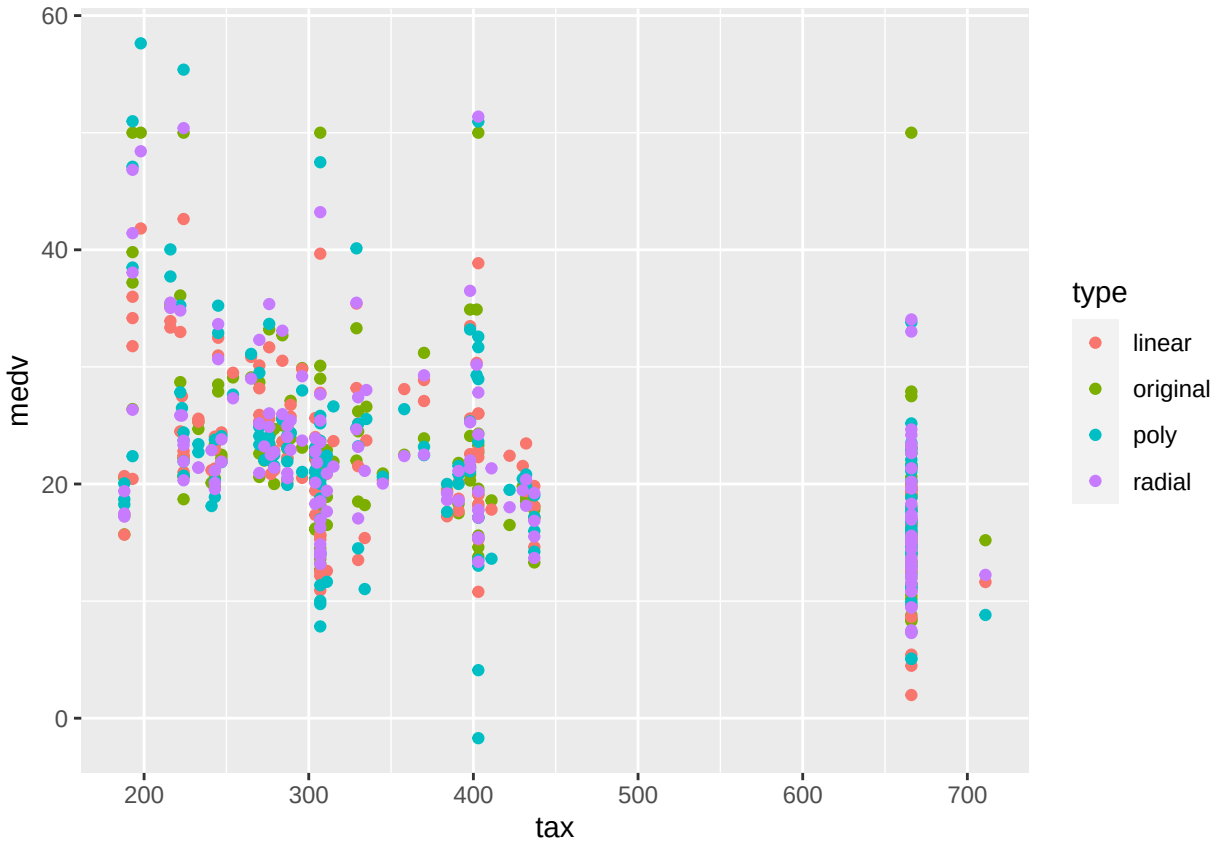

60
40
20
0
medv
0.4
0.5
0.6
0.7
0.8
nox
type
linear
original
poly
radial

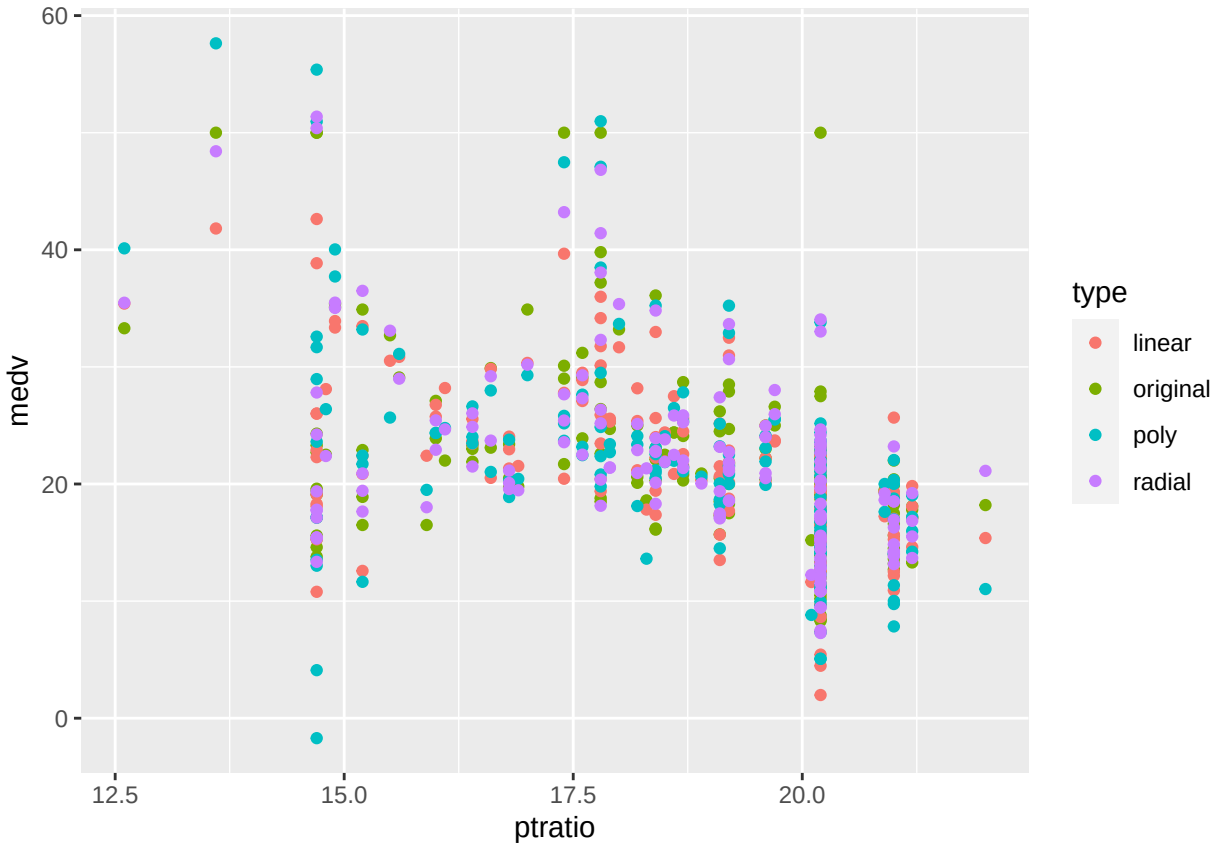

60
40
20
0
medv
4
5
6
7
8
rm
type
linear
original
poly
radial

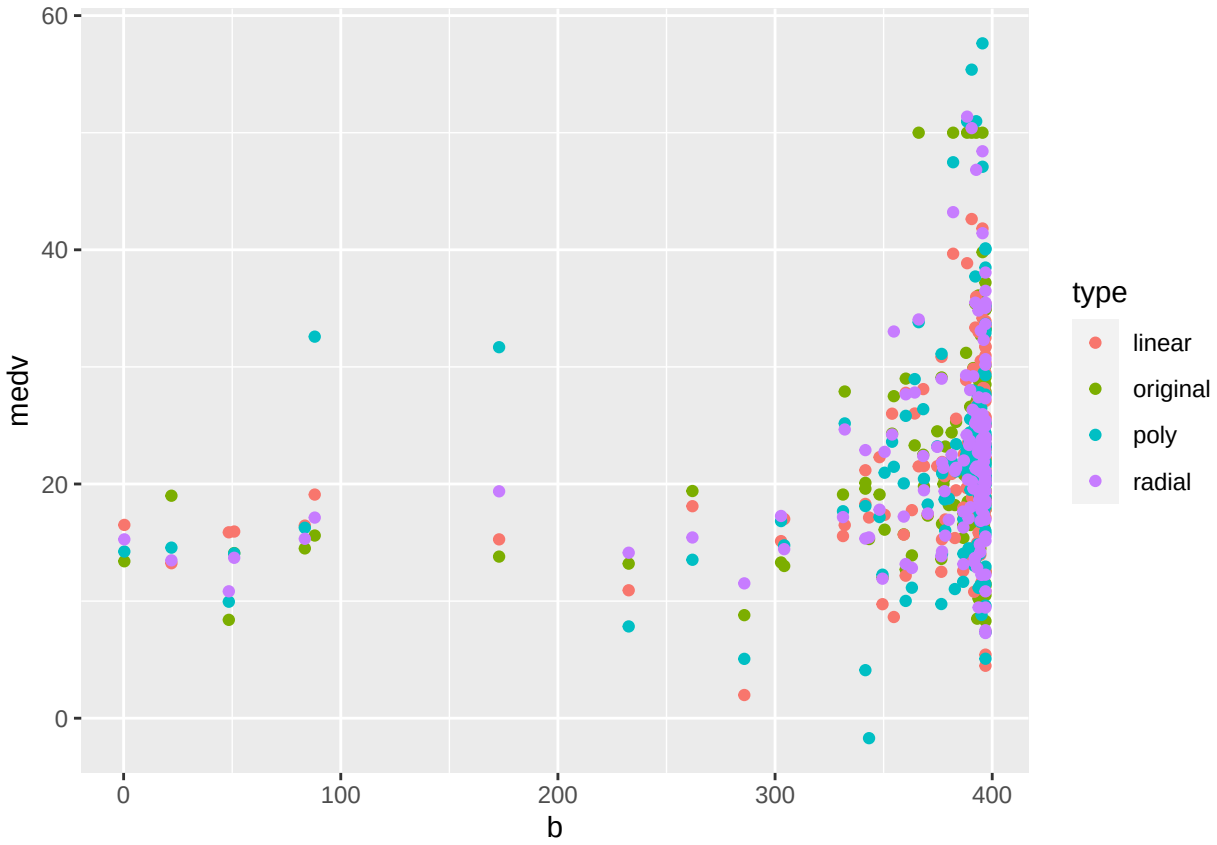

60
40
20
0
medv
25
50
75
100
age
type
linear
original
poly
radial

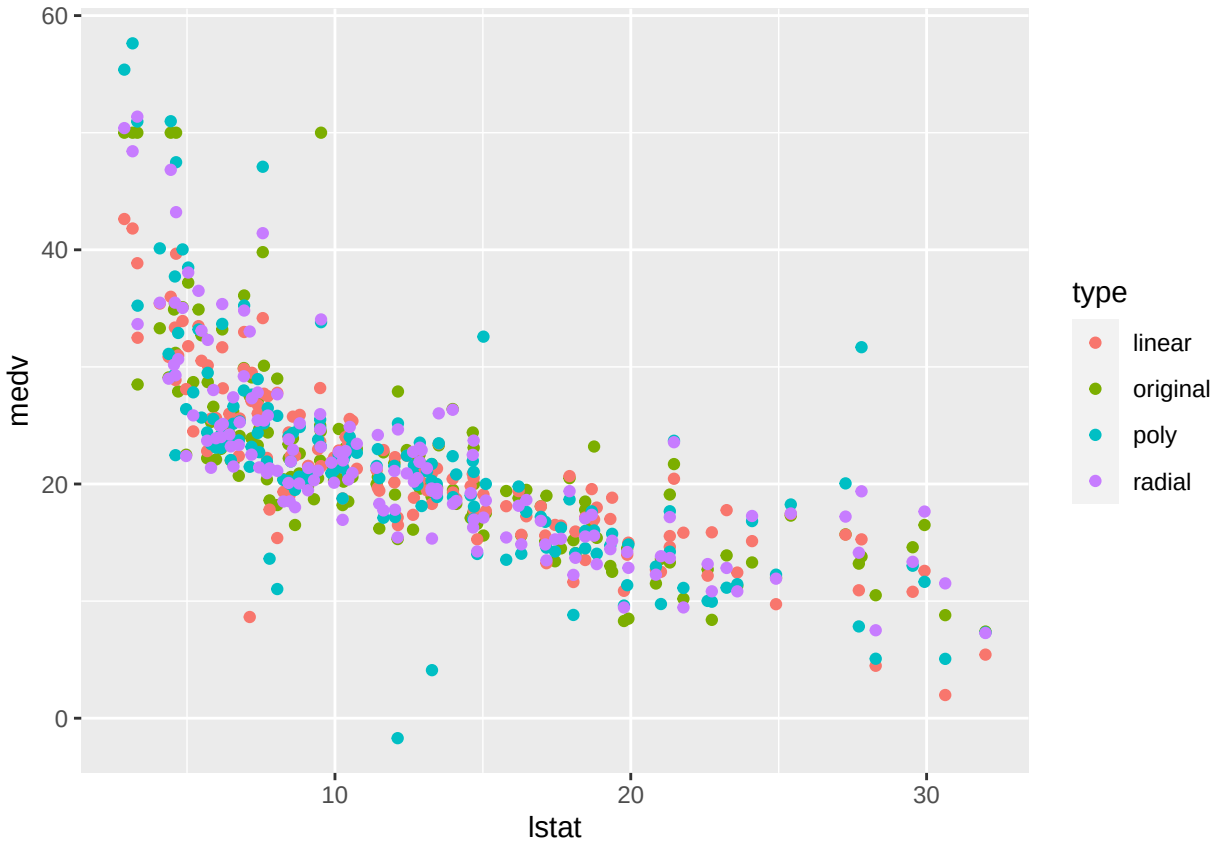

60
40
20
0
medv
3
6
9
dis
type
linear
original
poly
radial

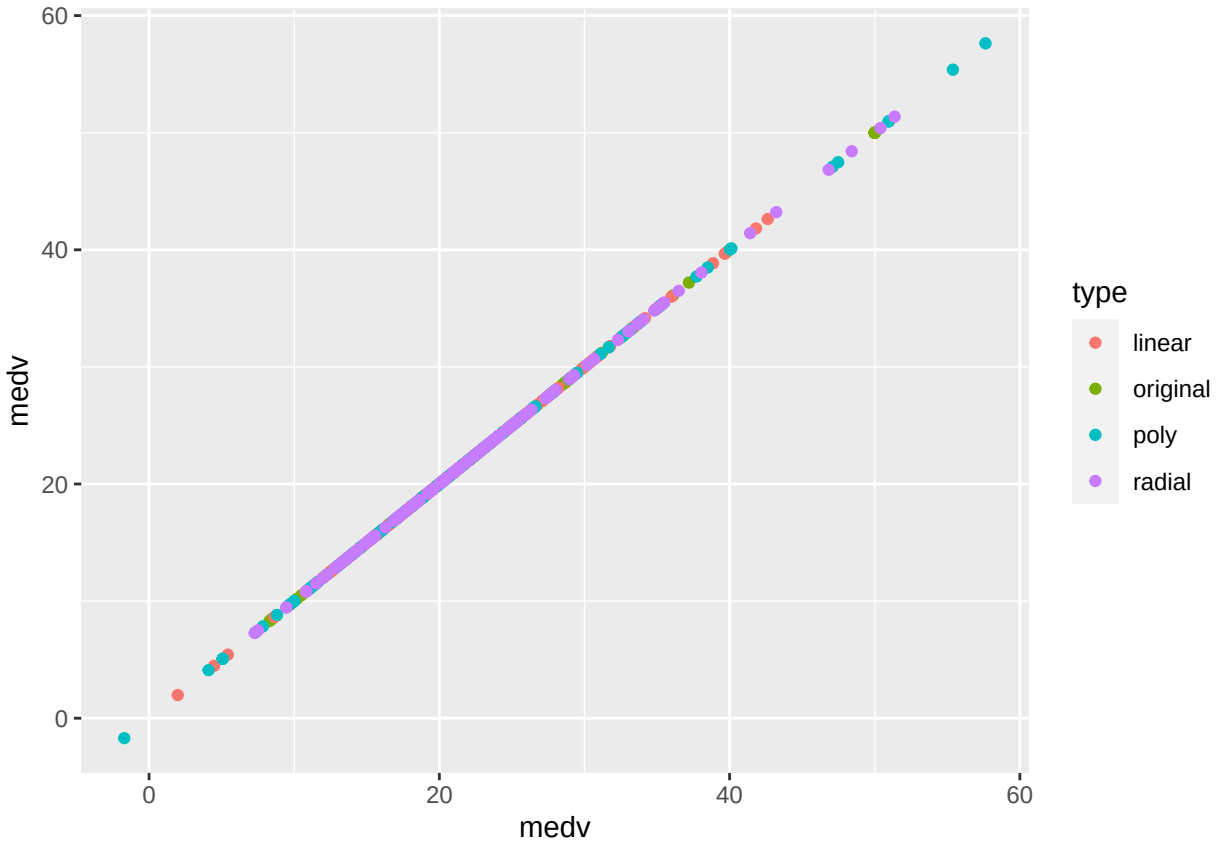

60
40
20
0
medv
0
5
10
15
20
25
rad
type
linear
original
poly
radial

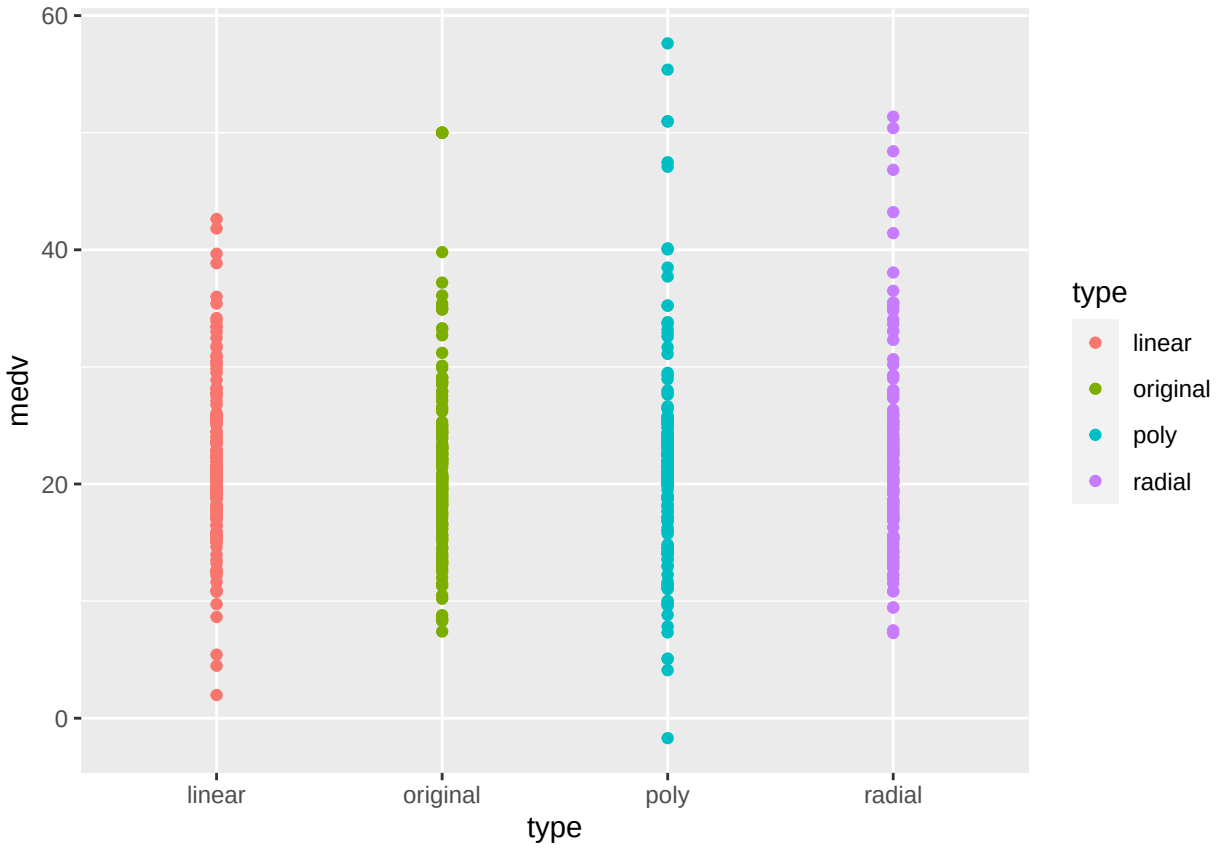

60
40
20
0
medv
200
300
400
500
600
700
tax
type
linear
original
poly
radial


60
40
20
0
medv
12.5
15.0
17.5
20.0
ptratio
type
linear
original
poly
radial


60
40
20
0
medv
0
100
200
300
400
b
type
linear
original
poly
radial


60
40
20
0
medv
10
20
30
lstat
type
linear
original
poly
radial


60
40
20
0
medv
0
20
40
60
medv
type
linear
original
poly
radial


60
40
20
0
medv
linear
original
poly
radial
type
type
linear
original
poly
radial