Loess, Polynomial Regression and Spline.rmd

Loess, Polynomial Regression and Spline

Goal1. Loess 모델 생성

Goal2. Polynomial Regression 모델 생성

Goal3. Spline 모델 생성

Goal4. 각 모델의 비교

Data Loading and Setting

데이터 로딩

```
DATA <- read.table("KMOOC_2_03_dataset_salary.txt", header=TRUE)
```

데이터 탐색

```
DATA[1:10,]
```

```
##    age    salary
## 1   18  75.04315
## 2   24  70.47602
## 3   45 130.98218
## 4   43 154.68529
## 5   50  75.04315
## 6   54 127.11574
## 7   44 169.52854
## 8   30 111.72085
## 9   41 118.88436
## 10  52 128.68049
```

```
dim(DATA)
```

```
## [1] 3000    2
```

데이터 분리

```
data.train <- DATA[1:2000,]
data.test <- DATA[2001:3000,]
```

Loess

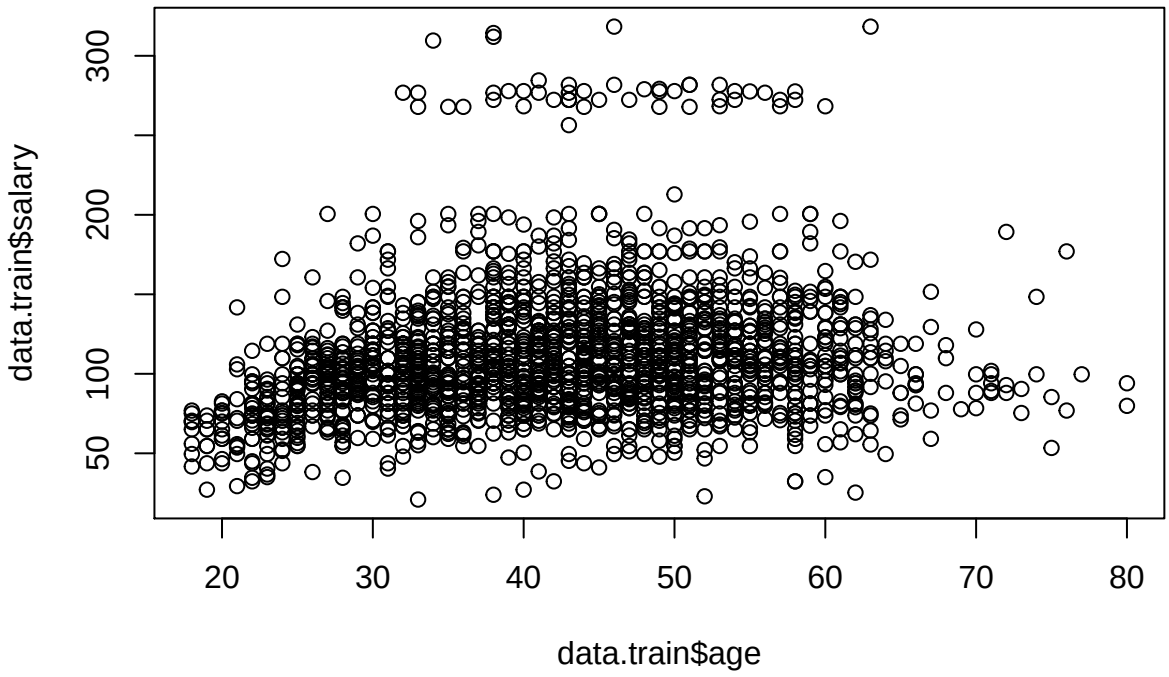
Loess 모델 생성

```
f.ls <- loess(salary ~ age, data=data.train)
f.ls
```

```
## Call:
## loess(formula = salary ~ age, data = data.train)
##
## Number of Observations: 2000
## Equivalent Number of Parameters: 4.98
## Residual Standard Error: 39.23
```

age와 salary의 산점도 시각화

```
plot(data.train$age, data.train$salary)
```

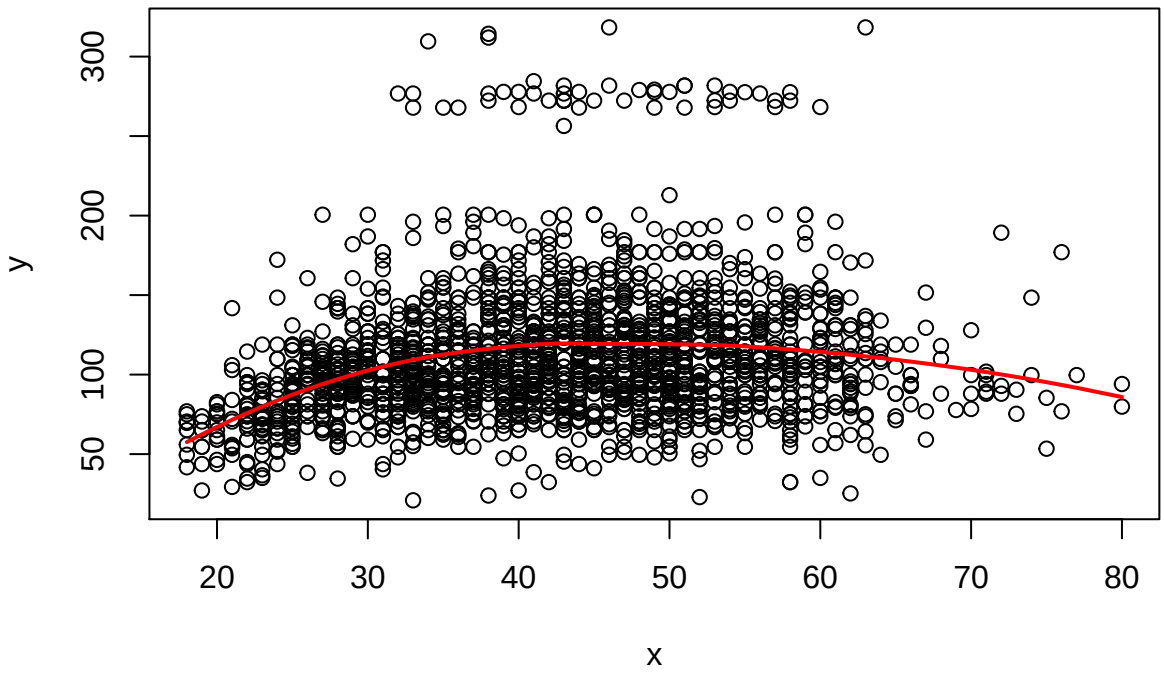


plot_fitted 함수 정의

```
plot_fitted <- function(x, y, yhat) {
  plot(x, y)
  o <- order(x)
  lines(x[o], yhat[o], type='l', col=2, lwd=2)
}
```

Loess 모델 시각화

```
plot_fitted(data.train$age, data.train$salary, f.ls$fitted)
```



Loess 모델의 RMSE 계산

```
y.train.ls <- predict(f.ls, newdata=data.train)
rmse.train.ls <- sqrt(mean((y.train.ls - data.train$salary)^2))
rmse.train.ls
```

```
## [1] 39.17384
```

```
y.test.ls <- predict(f.ls, newdata=data.test)
rmse.test.ls <- sqrt(mean((y.test.ls - data.test$salary)^2))
rmse.test.ls
```

```
## [1] 41.28488
```

Polynomial Regression

Polynomial Regression 모델 생성

```
f.pl <- lm(salary ~ poly(age, 5, raw=TRUE), data=data.train)
summary(f.pl)
```

```
##
## Call:
## lm(formula = salary ~ poly(age, 5, raw = TRUE), data = data.train)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -96.171 -24.058  -5.509  14.845 207.607
##
## Coefficients:
##                              Estimate Std. Error t value Pr(>|t|)
## (Intercept)                -1.862e+02  1.980e+02  -0.941    0.347
## poly(age, 5, raw = TRUE)1  2.311e+01  2.474e+01   0.934    0.350
## poly(age, 5, raw = TRUE)2 -7.368e-01  1.184e+00  -0.622    0.534
## poly(age, 5, raw = TRUE)3  1.277e-02  2.717e-02   0.470    0.638
## poly(age, 5, raw = TRUE)4 -1.197e-04  2.999e-04  -0.399    0.690
## poly(age, 5, raw = TRUE)5  4.608e-07  1.277e-06   0.361    0.718
##
## Residual standard error: 39.23 on 1994 degrees of freedom
## Multiple R-squared:  0.08786,    Adjusted R-squared:  0.08558
## F-statistic: 38.42 on 5 and 1994 DF,  p-value: < 2.2e-16
```

Polynomial Regression 모델의 RMSE 계산

```
y.train.pl <- predict(f.pl, newdata=data.train)
rmse.train.pl <- sqrt(mean((y.train.pl - data.train$salary)^2))
rmse.train.pl
```

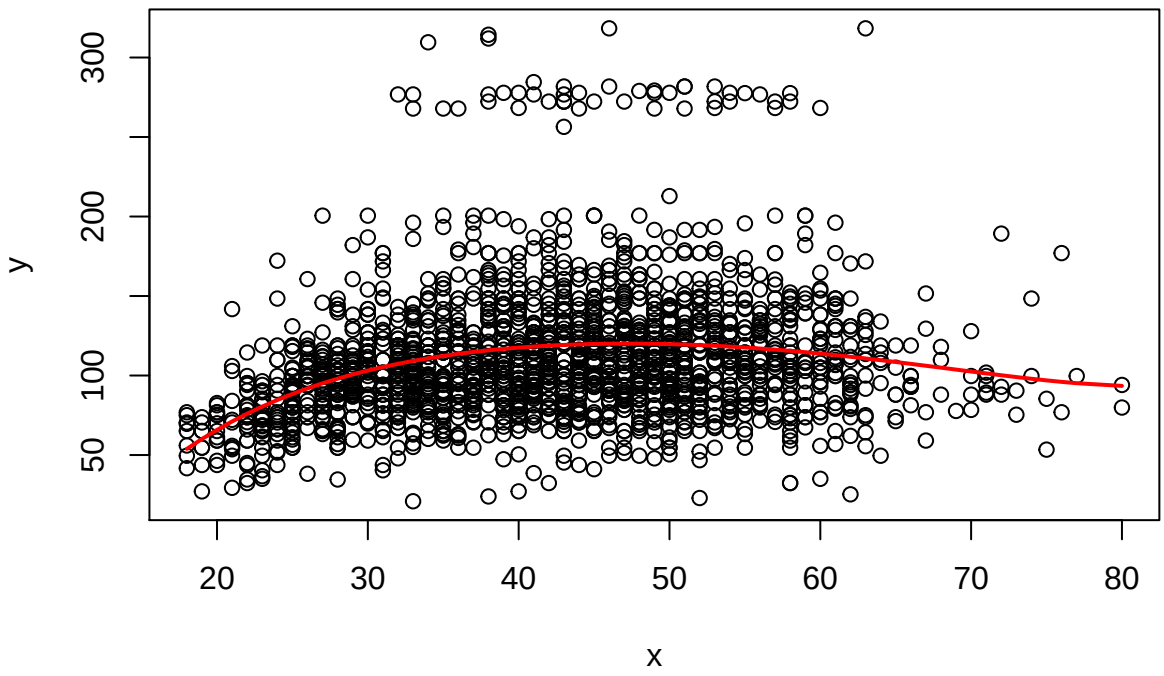
```
## [1] 39.1756
```

```
y.test.pl <- predict(f.pl, newdata=data.test)
rmse.test.pl <- sqrt(mean((y.test.pl - data.test$salary)^2))
rmse.test.pl
```

```
## [1] 41.3284
```

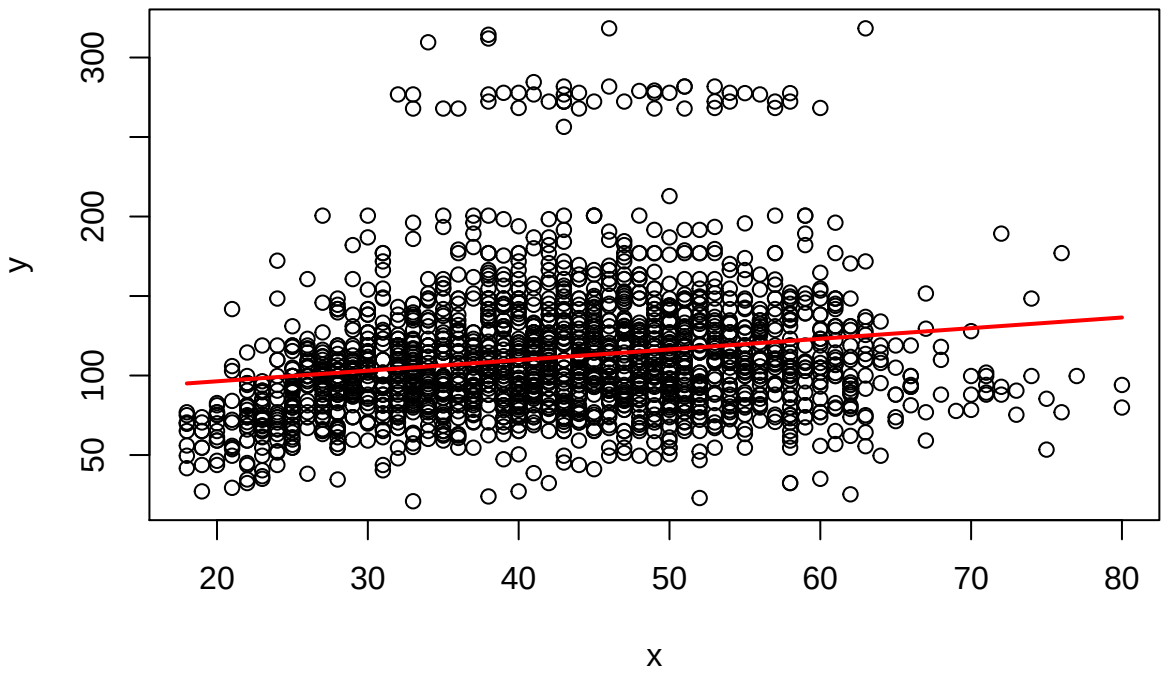
Polynomial Regression 모델 시각화

```
plot_fitted(data.train$age, data.train$salary, f.pl$fitted)
```



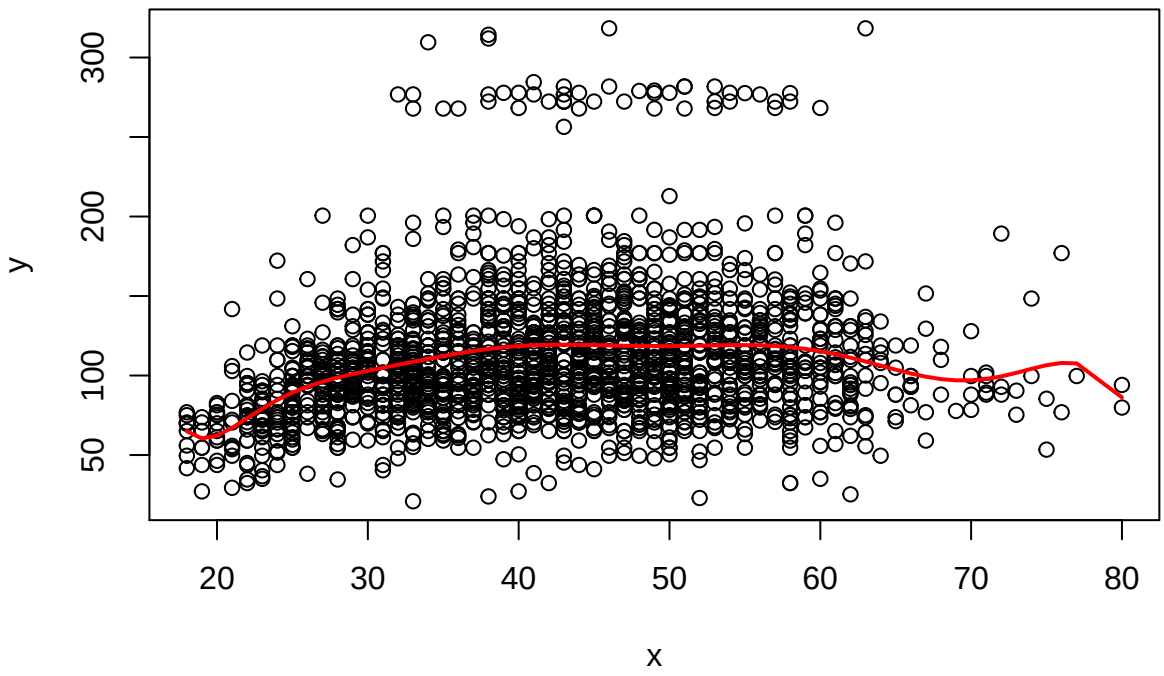
1차항 Polynomial Regression 모델 시각화

```
f.pl.1 <- lm(salary ~ poly(age, 1, raw=TRUE), data=data.train)
plot_fitted(data.train$age, data.train$salary, f.pl.1$fitted)
```



10차항 Polynomial Regression 모델 시각화

```
f.pl.10 <- lm(salary ~ poly(age, 10, raw=TRUE), data=data.train)
plot_fitted(data.train$age, data.train$salary, f.pl.10$fitted)
```



Spline

splines 패키지 attach

```
library(splines)
```

cut off point 생성

```
quantile(data.train$age, c(0, 0.25, 0.5, 0.75, 1))
```

```
##   0%  25%  50%  75% 100%
##   18   33   42   51   80
```

```
cutpt <- quantile(data.train$age, prob=seq(0, 1, by=0.25))
cutpt <- cutpt[2:(length(cutpt)-1)]
cutpt
```

```
## 25% 50% 75%
##  33  42  51
```

Spline 모델 생성

```
f.sp <- lm(salary ~ bs(age, knots=cutpt), data=data.train)
```

Spline 모델의 RMSE 계산

```
y.train.sp <- predict(f.sp, newdata=data.train)
rmse.train.sp <- sqrt(mean((y.train.sp - data.train$salary)^2))
rmse.train.sp
```

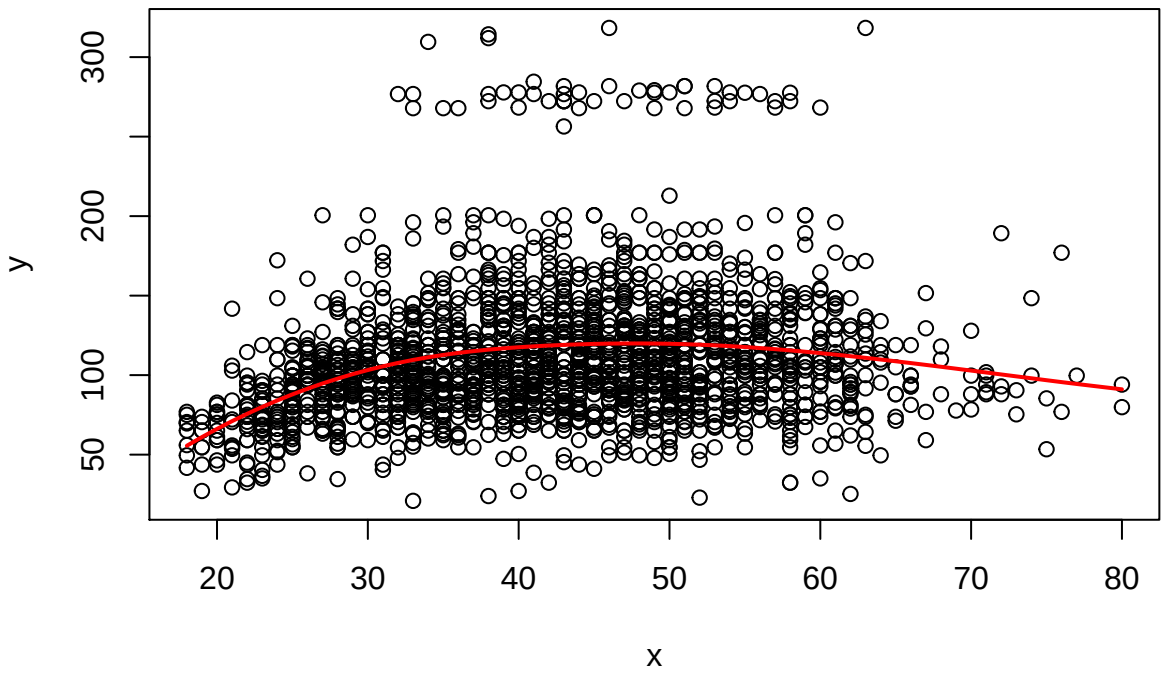
```
## [1] 39.1753
```

```
y.test.sp <- predict(f.sp, newdata=data.test)
rmse.test.sp <- sqrt(mean((y.test.sp - data.test$salary)^2))
rmse.test.sp
```

```
## [1] 41.32029
```

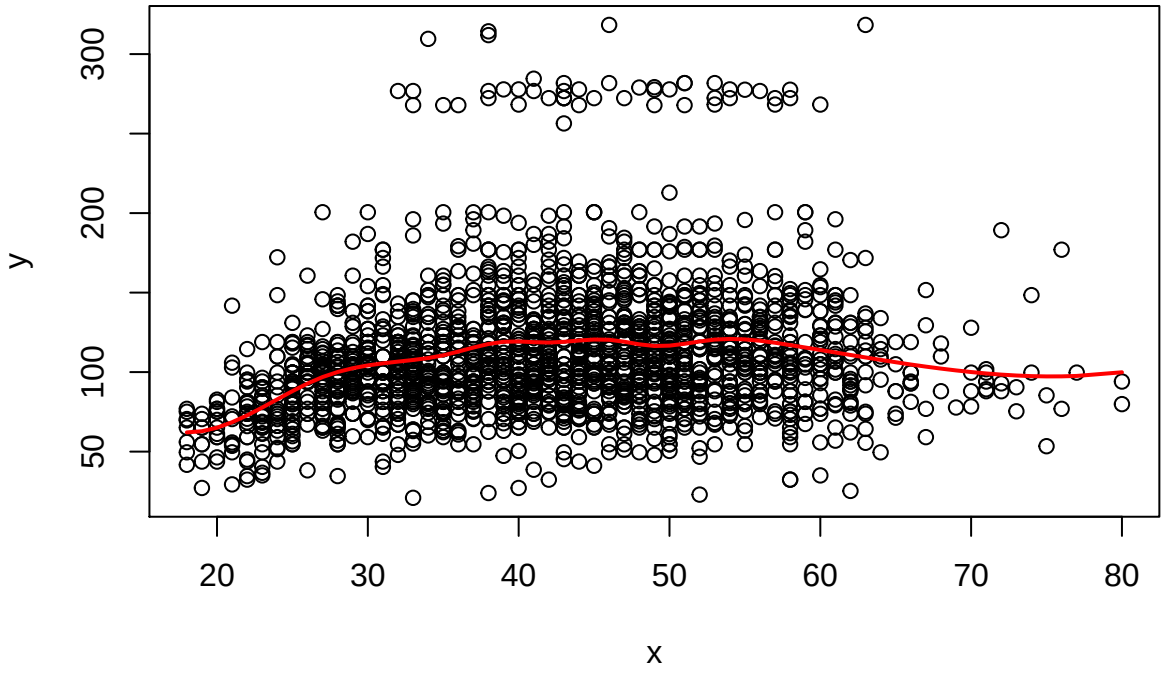
Spline 모델 시각화

```
plot_fitted(data.train$age, data.train$salary, f.sp$fitted)
```



10개의 cutpt를 갖는 Spline 모델 시각화

```
cutpt <- quantile(data.train$age, prob=seq(0, 1, by=0.1))
cutpt <- cutpt[2:(length(cutpt)-1)]
f.sp.10 <- lm(salary ~ bs(age, knots=cutpt), data=data.train)
plot_fitted(data.train$age, data.train$salary, f.sp.10$fitted)
```



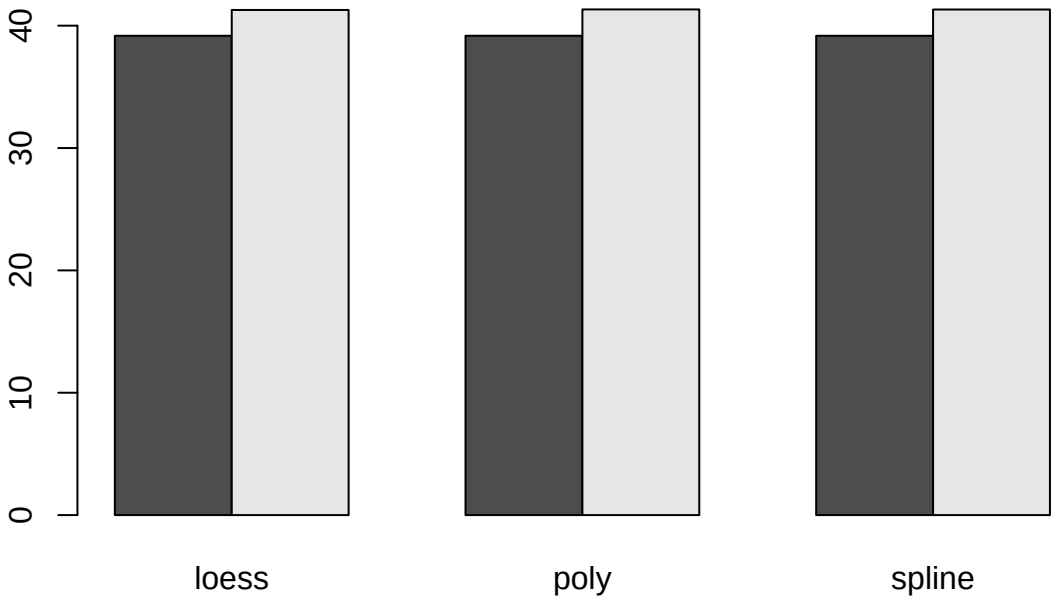
각 모델 비교

각 모델의 RMSE 비교 및 시각화

```
rmse.mat = matrix(c(rmse.train.ls, rmse.test.ls,
                    rmse.train.pl, rmse.test.pl,
                    rmse.train.sp, rmse.test.sp), nrow=2)
colnames(rmse.mat) <- c('loess', 'poly', 'spline')
rownames(rmse.mat) <- c('train', 'test')
rmse.mat
```

```
##          loess    poly   spline
## train 39.17384 39.1756 39.17530
## test  41.28488 41.3284 41.32029
```

```
barplot(rmse.mat, beside=TRUE)
```


0
10
20
30
40
loess
poly
spline