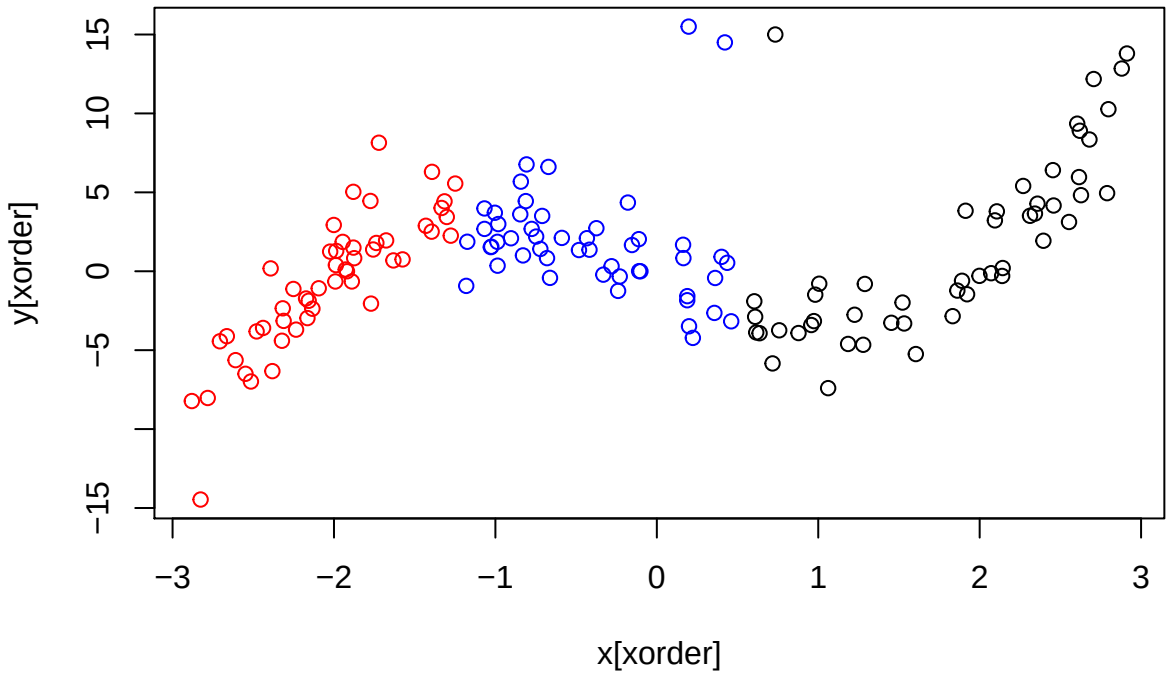
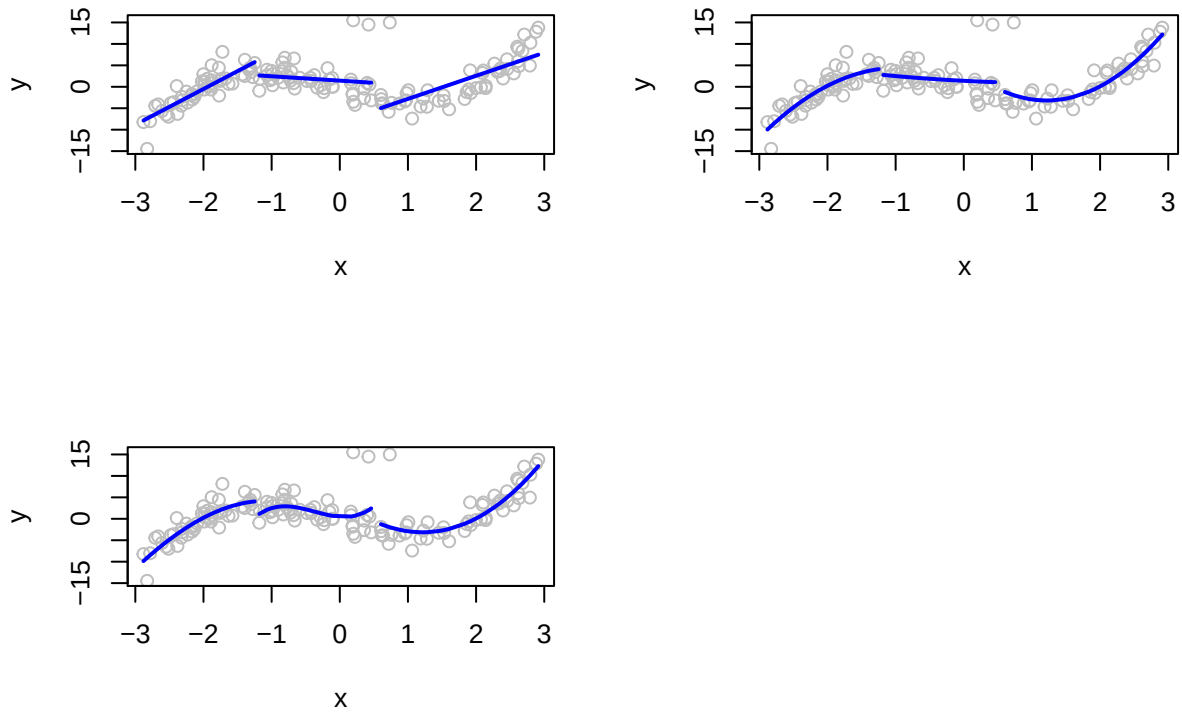
4.(a)

```
clr = c(rep('red', 50), rep('blue', 50), rep('black', 50))
plot(x[xorder], y[xorder], col = clr)
```



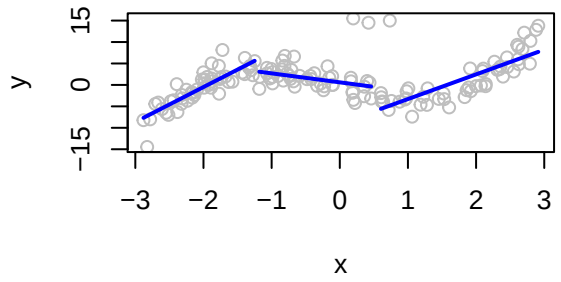
4.(b)

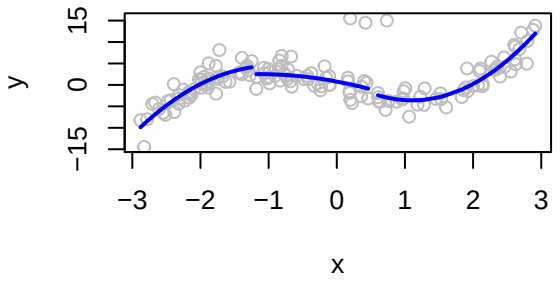
```
par(mfrow = c(2, 2))
complexity = c(1:3)
for(i in 1:length(complexity)){
  plot(x, y, col = 'grey')
  model1 = lm(y[xorder[1:50]] ~ poly(x[xorder[1:50]], i))
  model2 = lm(y[xorder[51:100]] ~ poly(x[xorder[51:100]], i))
  model3 = lm(y[xorder[101:150]] ~ poly(x[xorder[101:150]], i))
  lines(x[xorder[1:50]], predict(model1), col = 'blue', lwd = 2)
  lines(x[xorder[51:100]], predict(model2), col = 'blue', lwd = 2)
  lines(x[xorder[101:150]], predict(model3), col = 'blue', lwd = 2)
}
```

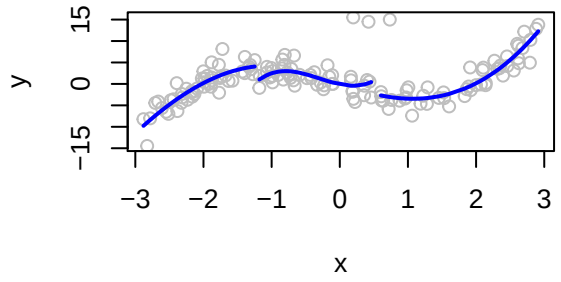


4.(c)

```
par(mfrow = c(2, 2))
for(i in 1:length(complexity)){
  plot(x, y, col = 'grey')
  model1 = rlm(y[xorder[1:50]] ~ poly(x[xorder[1:50]], i), psi = huberfn)
  model2 = rlm(y[xorder[51:100]] ~ poly(x[xorder[51:100]], i), psi = huberfn)
  model3 = rlm(y[xorder[101:150]] ~ poly(x[xorder[101:150]], i), psi = huberfn)
  lines(x[xorder[1:50]], predict(model1), col = 'blue', lwd = 2)
  lines(x[xorder[51:100]], predict(model2), col = 'blue', lwd = 2)
  lines(x[xorder[101:150]], predict(model3), col = 'blue', lwd = 2)
}
```

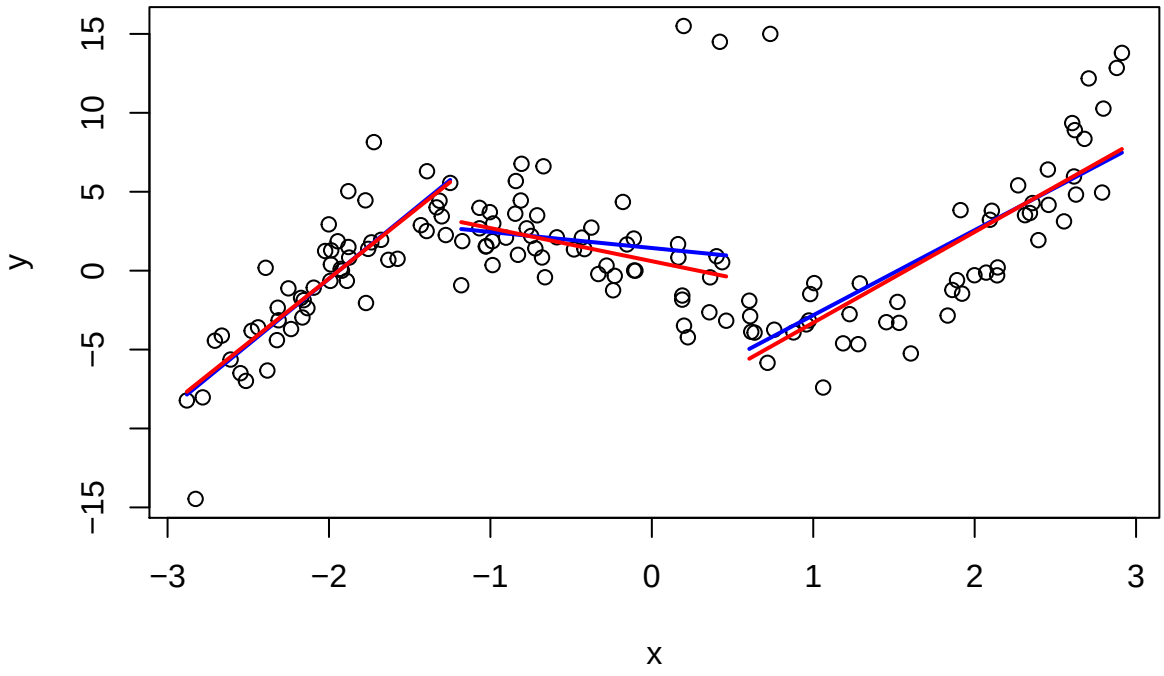


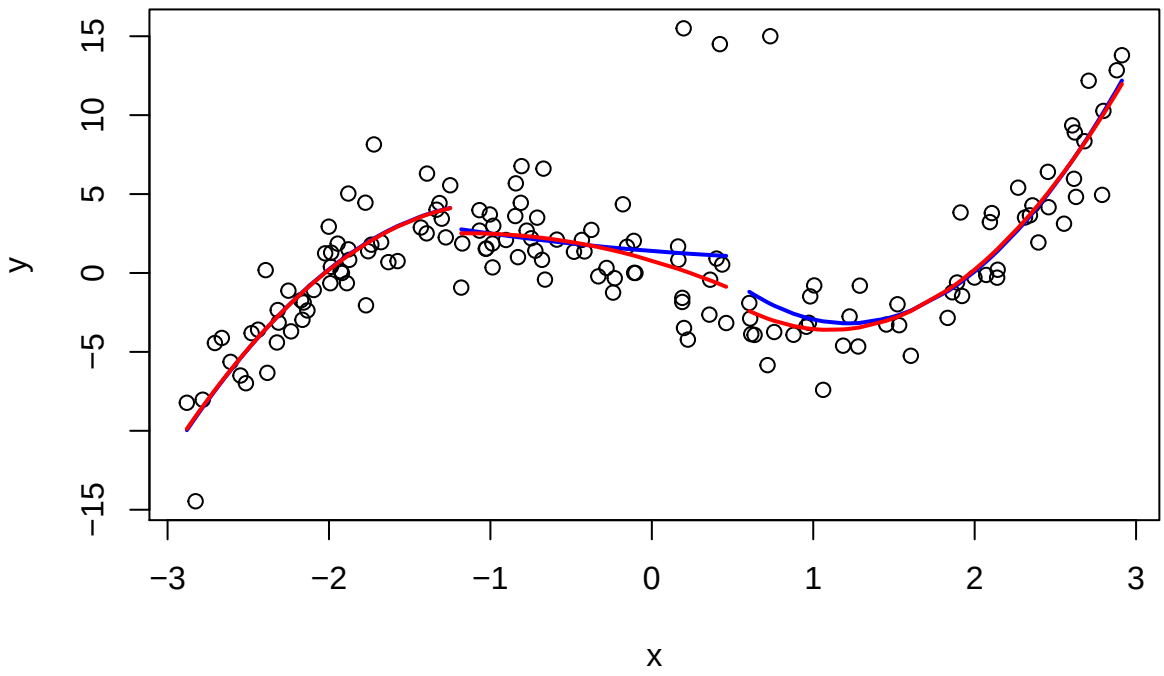


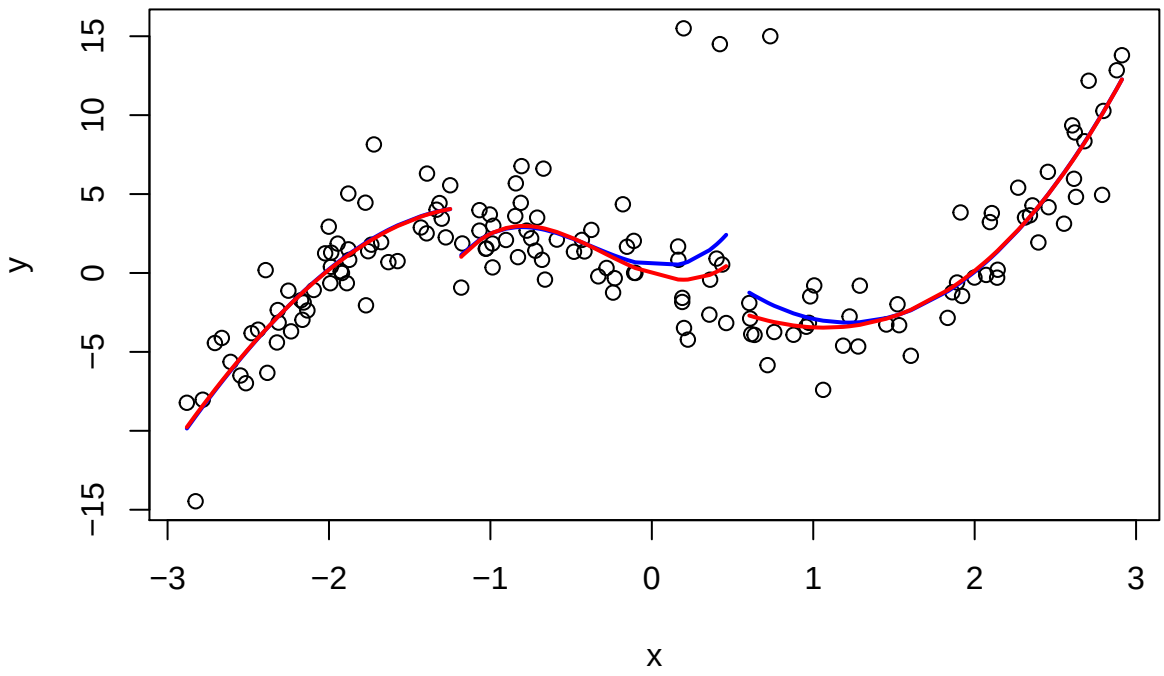


4.(d)

```
for(i in 1:length(complexity)){
  plot(x, y)
  model1 = lm(y[xorder[1:50]] ~ poly(x[xorder[1:50]], i))
  model2 = lm(y[xorder[51:100]] ~ poly(x[xorder[51:100]], i))
  model3 = lm(y[xorder[101:150]] ~ poly(x[xorder[101:150]], i))

  rbmodel1 = rlm(y[xorder[1:50]] ~ poly(x[xorder[1:50]], i), psi = huberfn)
  rbmodel2 = rlm(y[xorder[51:100]] ~ poly(x[xorder[51:100]], i), psi = huberfn)
  rbmodel3 = rlm(y[xorder[101:150]] ~ poly(x[xorder[101:150]], i), psi = huberfn)

  lines(x[xorder[1:50]], predict(model1), col = 'blue', lwd = 2)
  lines(x[xorder[51:100]], predict(model2), col = 'blue', lwd = 2)
  lines(x[xorder[101:150]], predict(model3), col = 'blue', lwd = 2)

  lines(x[xorder[1:50]], predict(rbmodel1), col = 'red', lwd = 2)
  lines(x[xorder[51:100]], predict(rbmodel2), col = 'red', lwd = 2)
  lines(x[xorder[101:150]], predict(rbmodel3), col = 'red', lwd = 2)
}
```




y
15
10
5
0
−5
−15
−3
−2
−1
0
1
2
3
x



There isn't much different between the Huber's robust loss function and least squares on the first polynomial. For the rest of the polynomials, since Huber's robust loss function is less sensitive to outliers, the Huber's robust loss fit is further away than the least squares fit.