

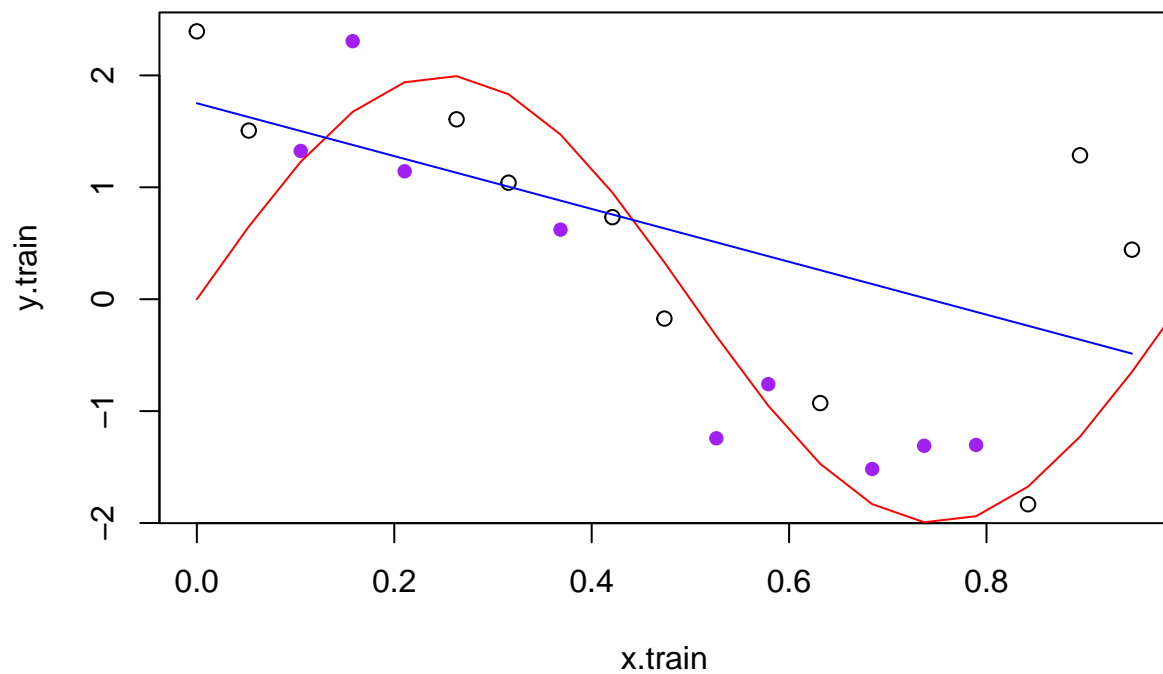
Michael_Ghattas_WP5

Michael Ghattas

2024-10-16

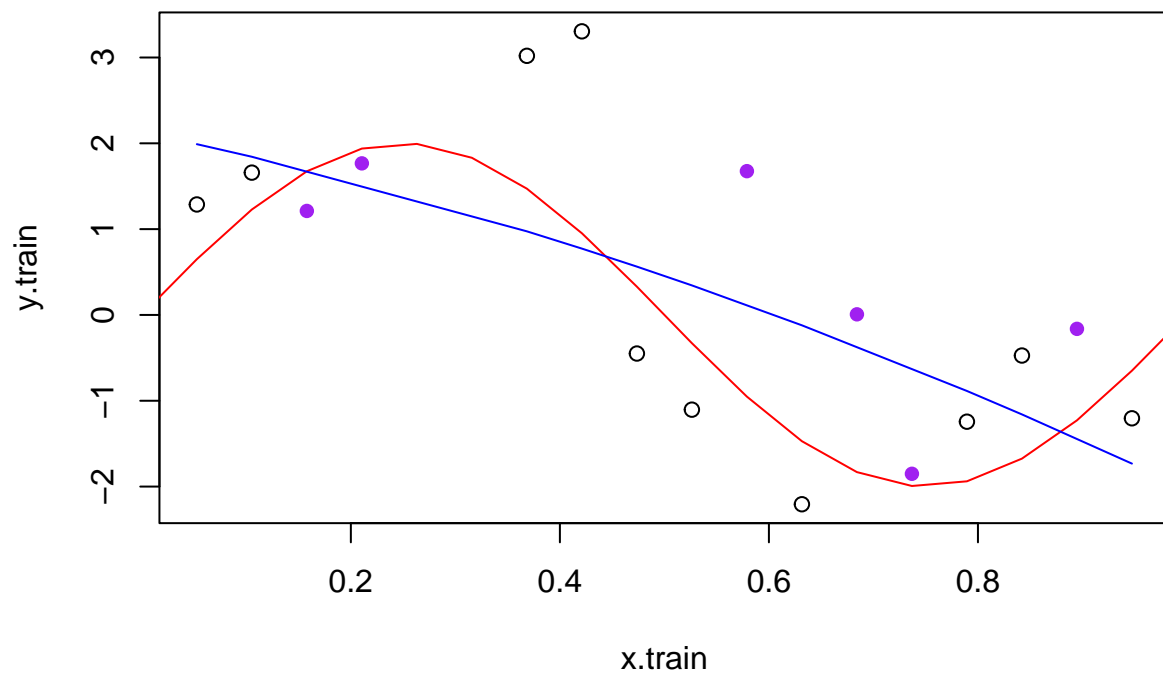
```
bv = function(n,p,noise){  
  
  #Training and testing data  
  x = seq(0,1,by = 1/(n-1) )  
  #Data generation  
  truey = 2*sin(2*pi*x)  
  y = 2*sin(2*pi*x)+ rnorm(length(x),0,noise)  
  
  #Training and testing  
  train.index = sample(c(1:length(x)),floor(.5*length(x)))  
  x.train = x[train.index]  
  y.train = y[train.index]  
  
  x.test = x[-train.index]  
  y.test = y[-train.index]  
  
  plot(x.train,y.train, col = 'black')  
  lines(x.test,y.test, type = 'p', pch = 16, col = 'purple')  
  lines(x,truey,type = 'l', col='red')  
  
  model = lm(y.train~poly(x.train,p))  
  ind = order(x.train)  
  lines(x.train[ind], model$fitted.values[ind], type = 'l',col = 'blue')  
  
  print('Average training error is')  
  print(sum(model$residuals^2)/length(model$residuals) )  
  
  print('Average test error is:')  
  data = data.frame(x.test)  
  names(data) = 'x.train'  
  pred = predict(model, newdata = data )  
  print(sum((y.test - pred)^2/length(y.test) ) )  
}
```

```
set.seed(123963)  
bv(20,1,1)#linear model
```



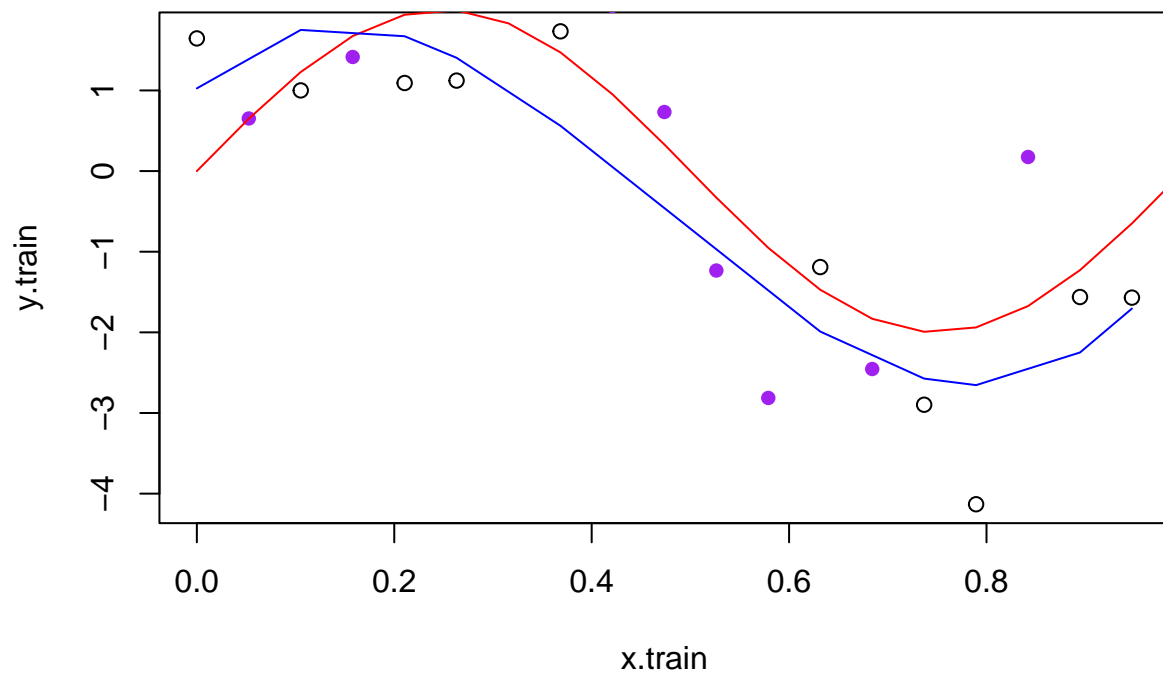
```
## [1] "Average training error is"
## [1] 0.8839068
## [1] "Average test error is:"
## [1] 1.140563
```

```
bv(20,2,1)#Quadratic
```



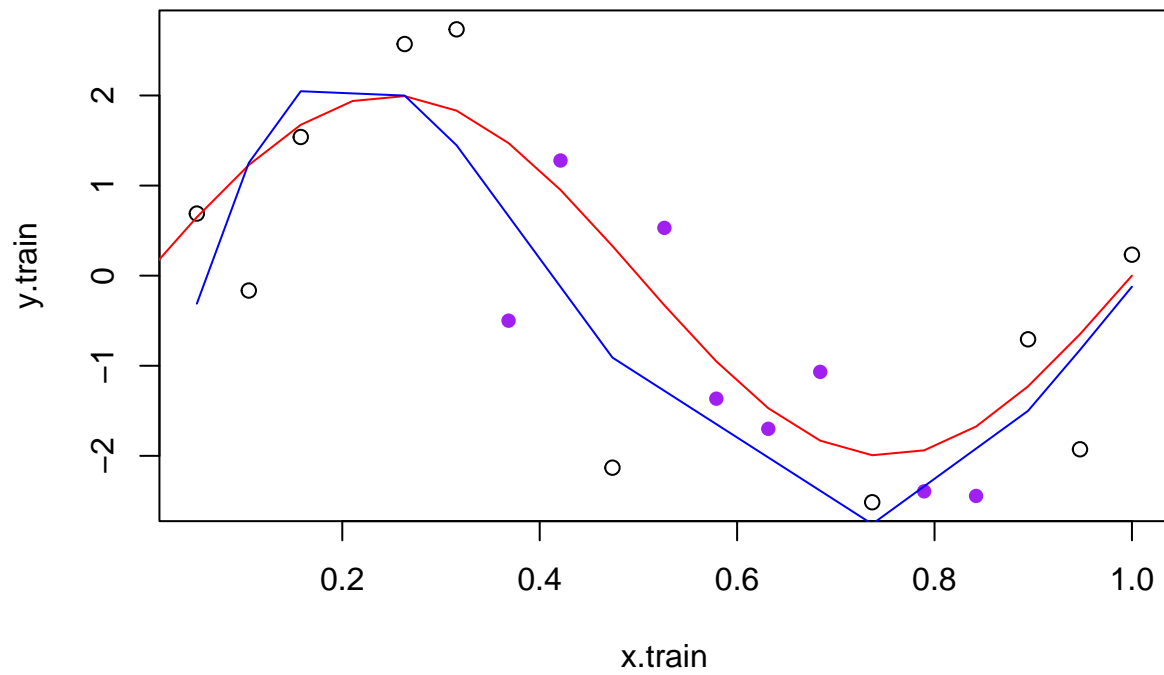
```
## [1] "Average training error is"
## [1] 1.946837
## [1] "Average test error is:"
## [1] 3.748626
```

```
bv(20,3,1)#Cubic
```



```
## [1] "Average training error is"  
## [1] 0.6150372  
## [1] "Average test error is:"  
## [1] 1.985554
```

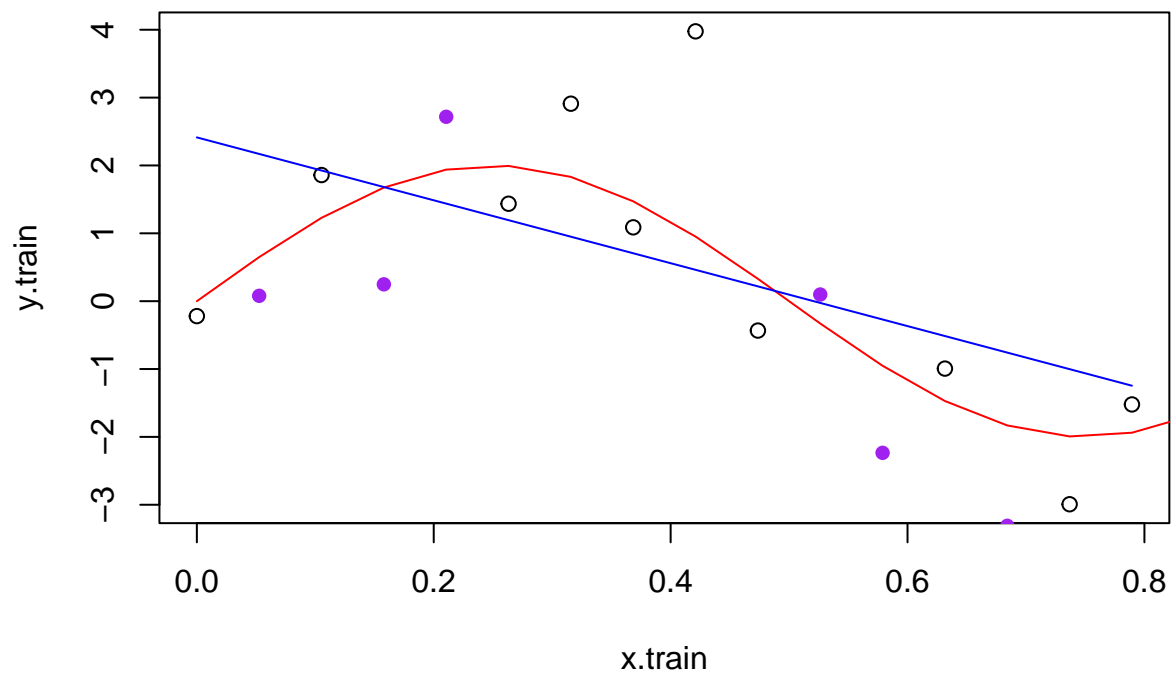
```
bv(20,4,1)
```



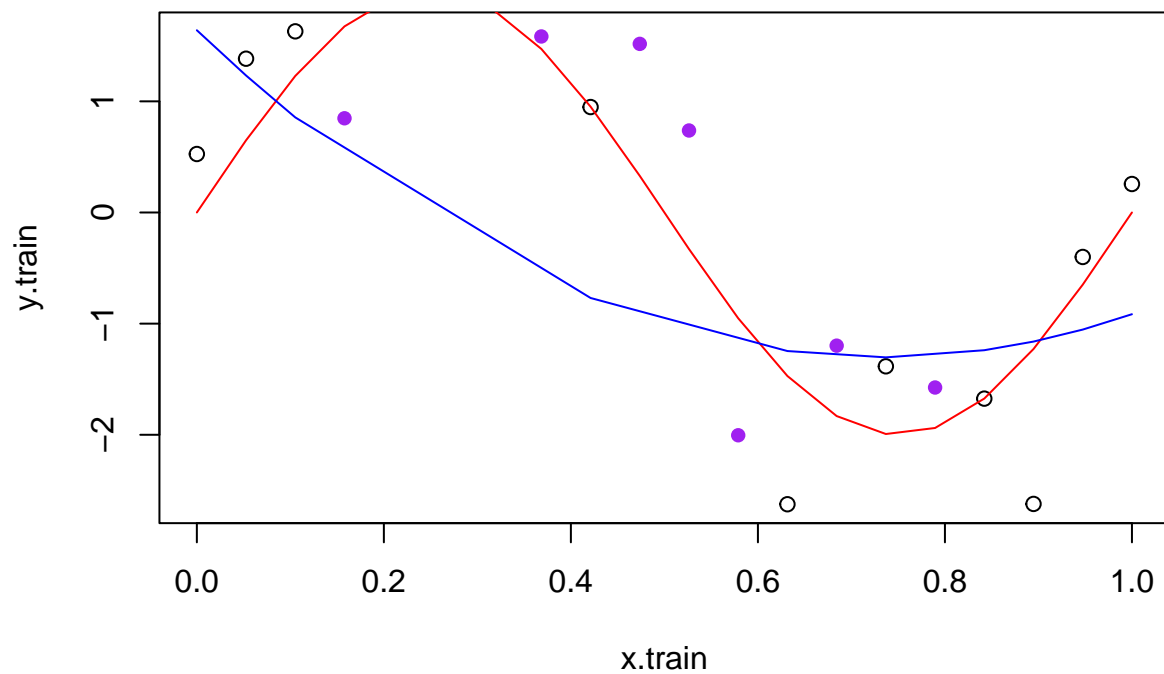
```
## [1] "Average training error is"
## [1] 0.8775785
## [1] "Average test error is:"
## [1] 1.940918
```

```
for (i in 1:9) {
  print(i)
  bv(20,i,1)
}
```

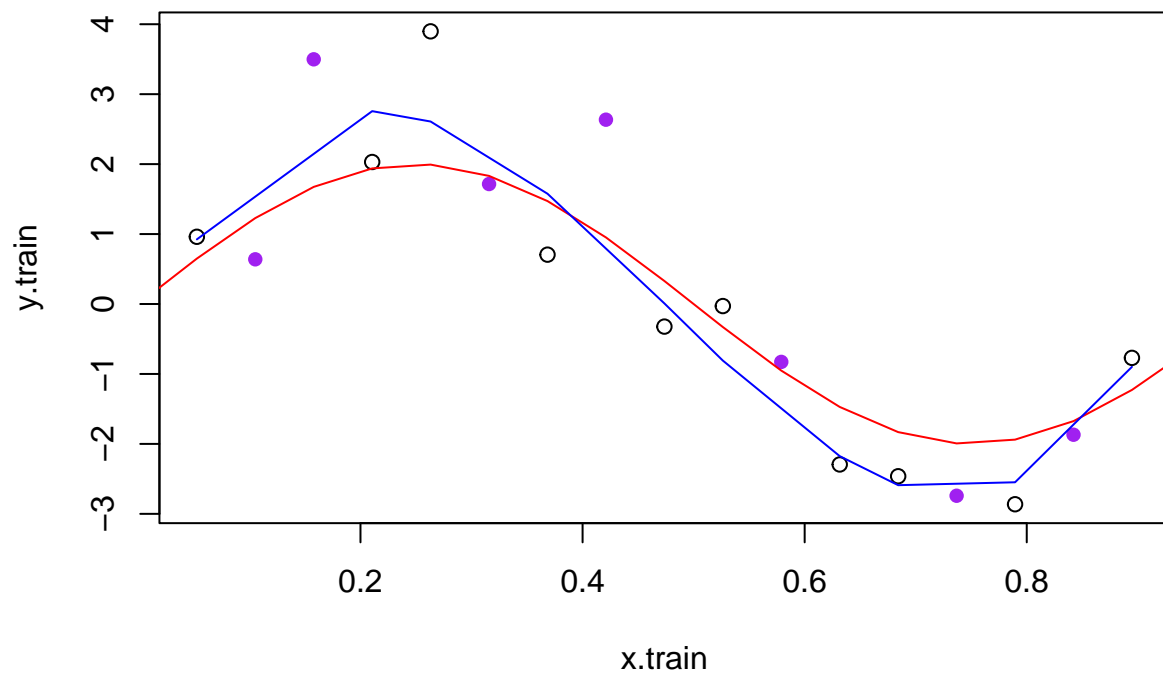
```
## [1] 1
```



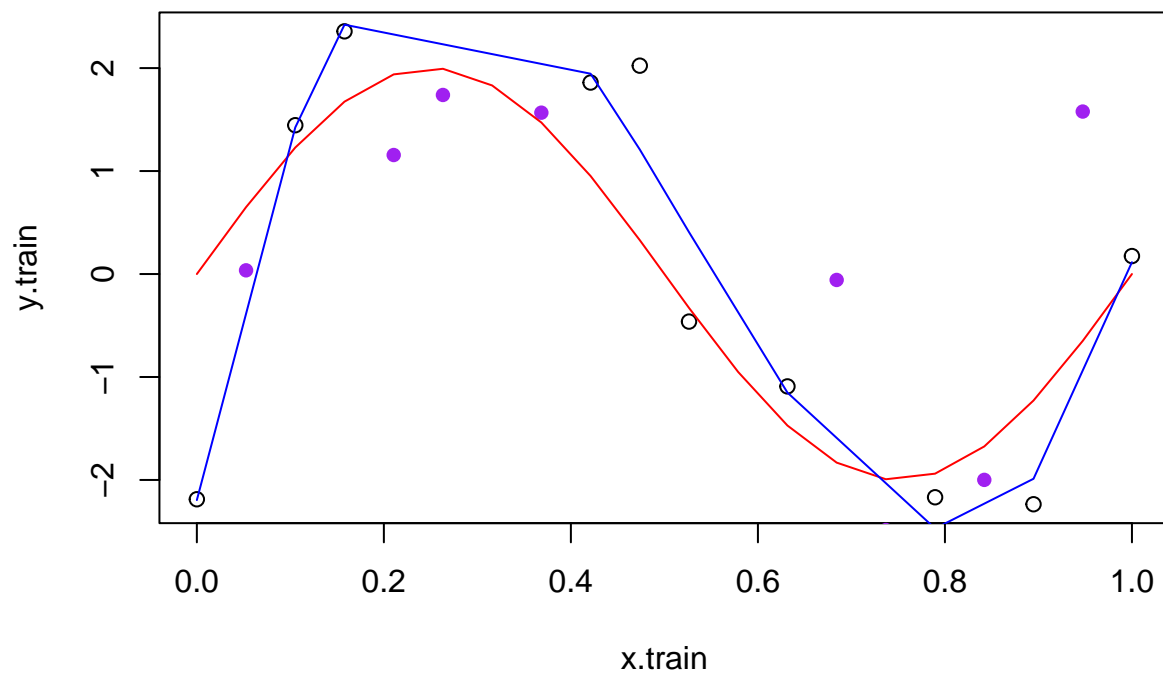
```
## [1] "Average training error is"
## [1] 2.803294
## [1] "Average test error is:"
## [1] 2.179192
## [1] 2
```



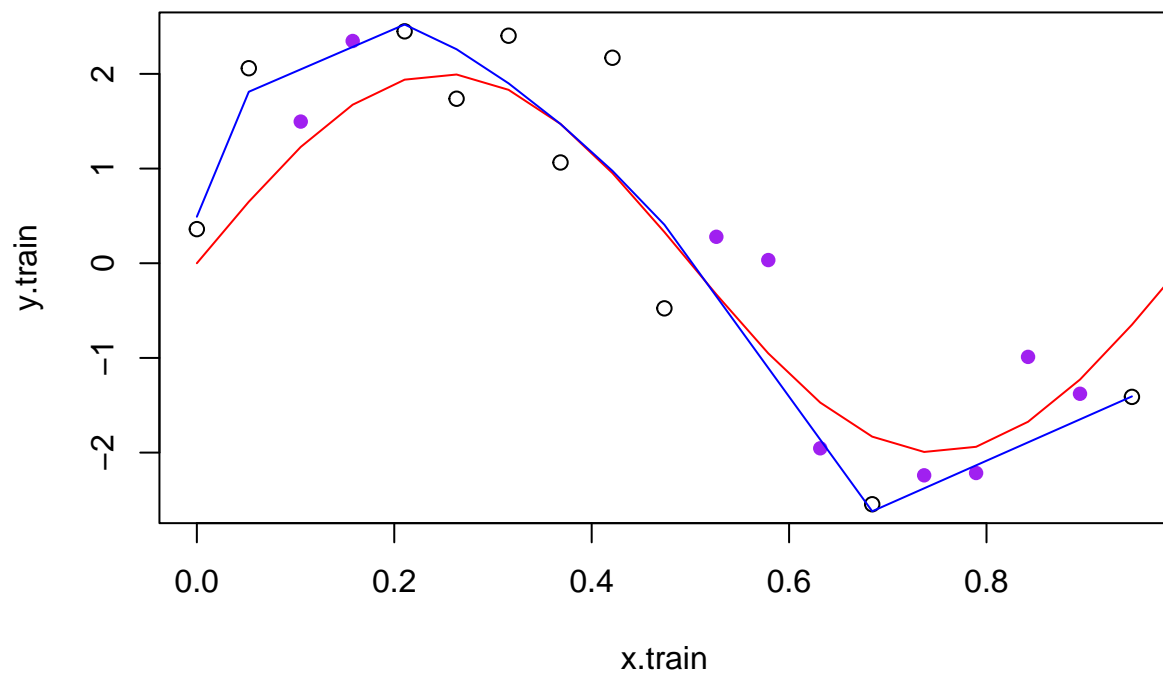
```
## [1] "Average training error is"
## [1] 1.084269
## [1] "Average test error is:"
## [1] 3.954304
## [1] 3
```



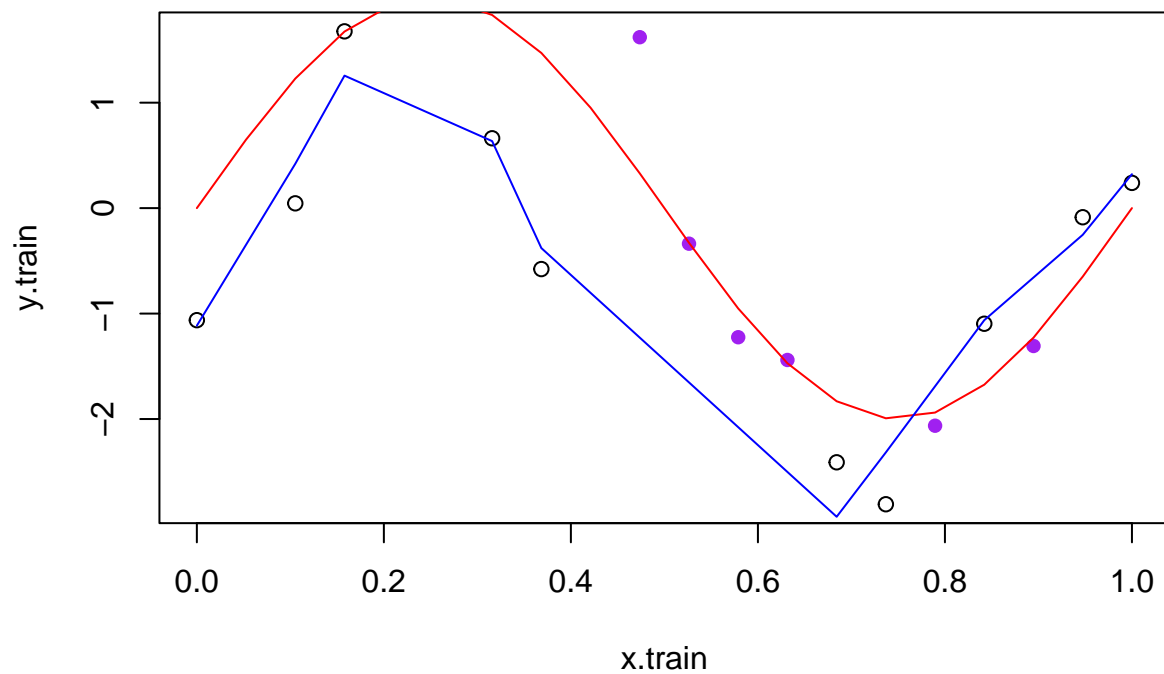
```
## [1] "Average training error is"  
## [1] 0.3813032  
## [1] "Average test error is:"  
## [1] 1.545597  
## [1] 4
```

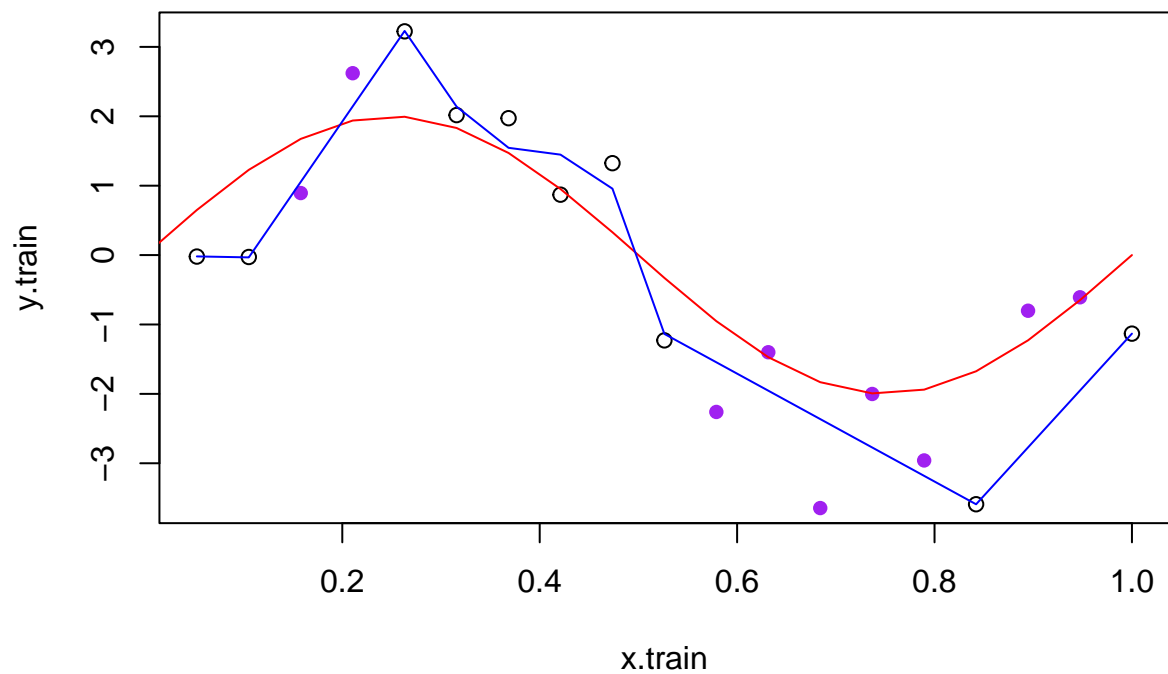
```
## [1] "Average training error is"
## [1] 0.1589865
## [1] "Average test error is:"
## [1] 2.158814
## [1] 5
```



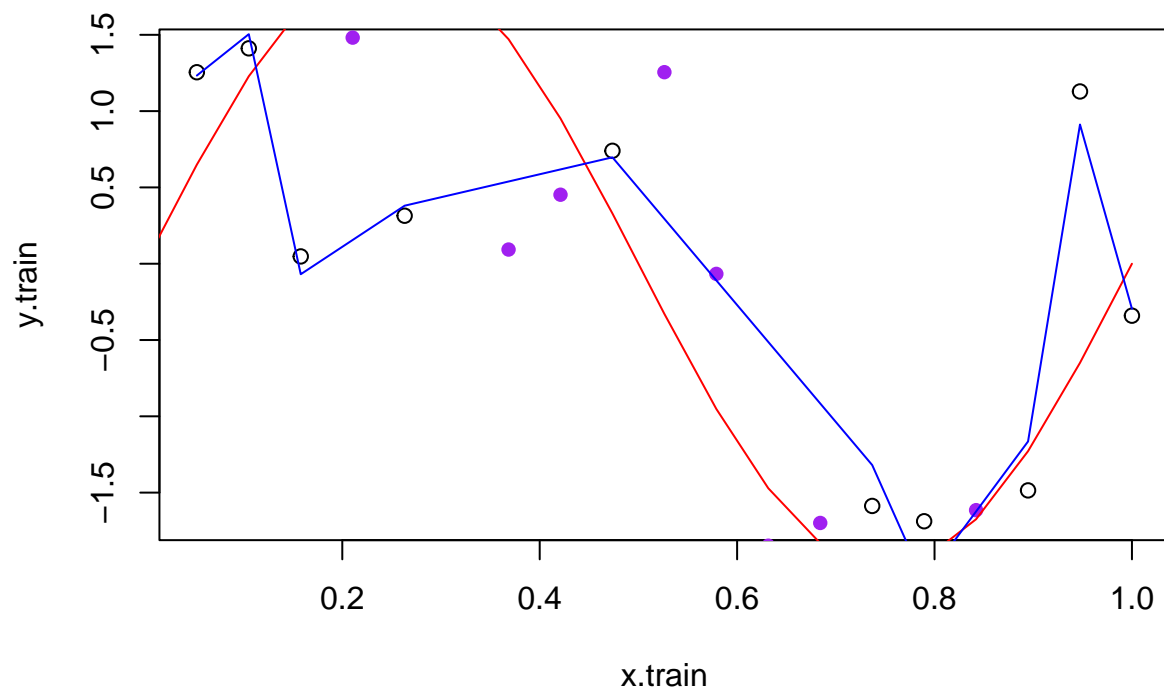
```
## [1] "Average training error is"
## [1] 0.2987909
## [1] "Average test error is:"
## [1] 1.965929
## [1] 6
```



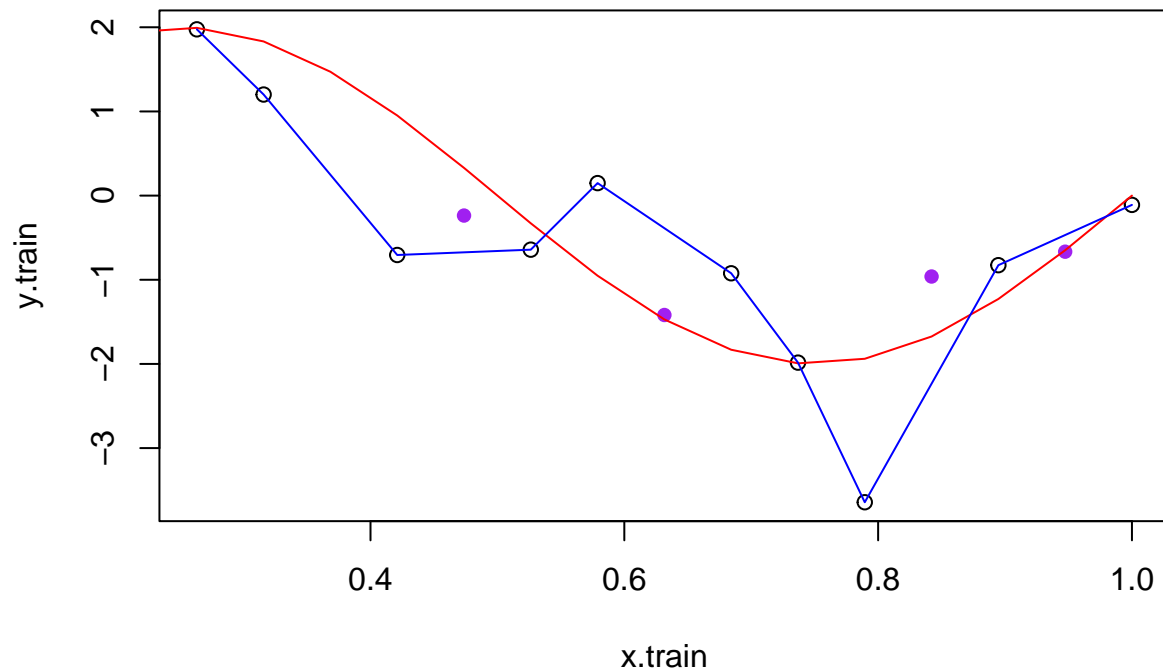
```
## [1] "Average training error is"  
## [1] 0.09042326  
## [1] "Average test error is:"  
## [1] 5.813831  
## [1] 7
```



```
## [1] "Average training error is"
## [1] 0.06733751
## [1] "Average test error is:"
## [1] 216.1037
## [1] 8
```



```
## [1] "Average training error is"
## [1] 0.04084667
## [1] "Average test error is:"
## [1] 12.57615
## [1] 9
```



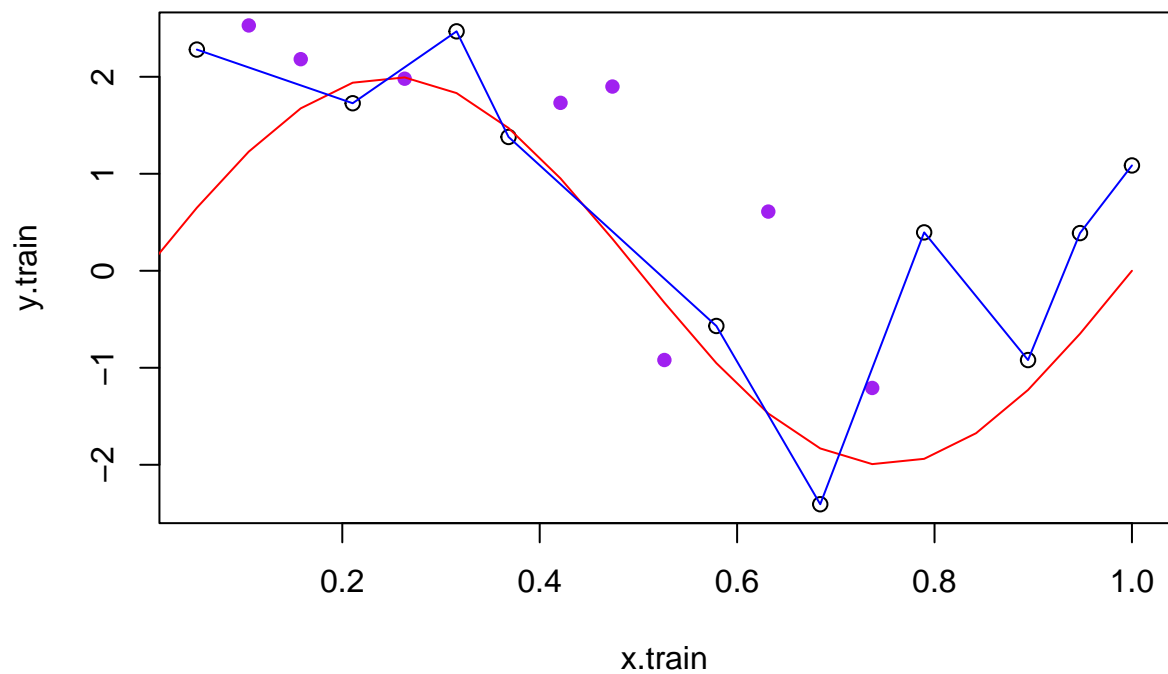
```
## [1] "Average training error is"
## [1] 0
## [1] "Average test error is:"
## [1] 26044018
```

Exercise 1:

Set: $n=20$, and noise $=1$. Questions:

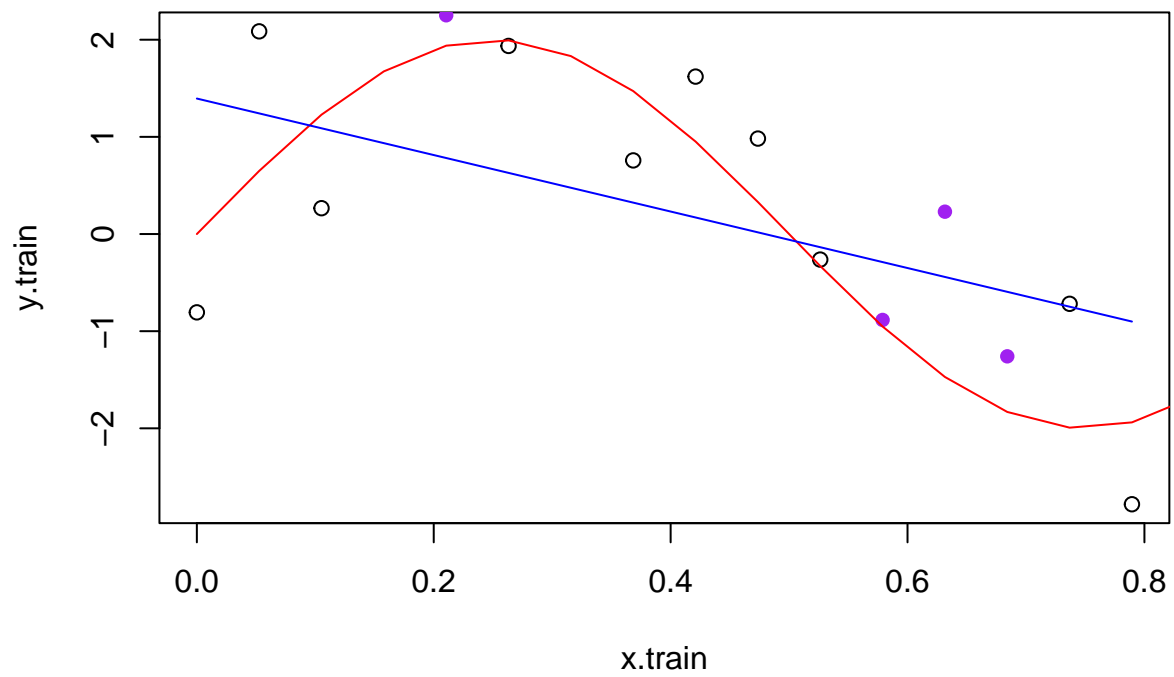
1. What happens when the degree of the polynomial approaches $n/2 - 1$? Does the training and testing error agree? - We can see that the training error is near zero, but the testing error has skyrocketed, which suggests overfitting. 2. What happens when the degree of the polynomial is 1. Does the training and testing error generally agree? What is true about the graph with the training and test data? - The training and test errors are relatively close, suggesting that the linear model is underfitting but generalizes decently to new data. The graph of training and test data likely shows that the linear model struggles to capture all the variations in the data but maintains a reasonable level of error for both training and testing sets. 3. What is the ideal degree of the polynomial? How did you decide? - The ideal degree of the polynomial is a polynomial of degree 3. This degree yields a relatively low training error while also keeping the test error reasonably small. The model at degree 3 seems to capture enough complexity of the underlying data without overfitting the noise.

```
set.seed(369321)
#1
bv(20,9,1)
```



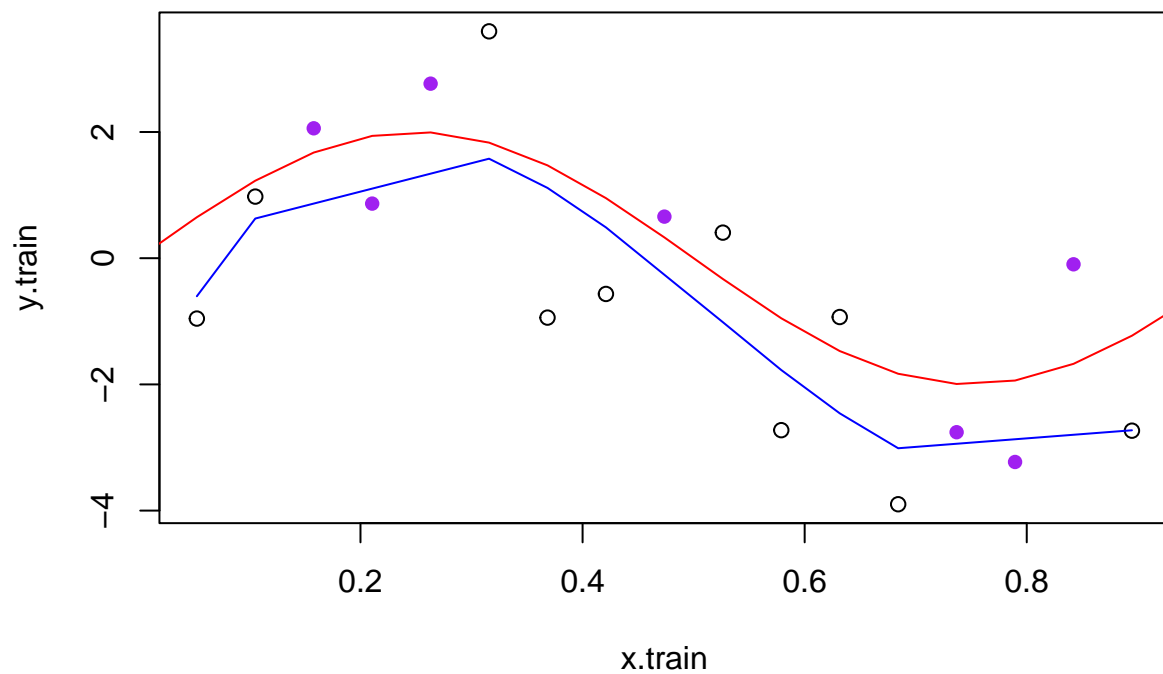
```
## [1] "Average training error is"
## [1] 0
## [1] "Average test error is:"
## [1] 8109.383
```

```
#2
bv(20,1,1)
```



```
## [1] "Average training error is"  
## [1] 1.471674  
## [1] "Average test error is:"  
## [1] 3.294294
```

```
#3  
bv(20,3,1)
```

```
## [1] "Average training error is"  
## [1] 1.569385  
## [1] "Average test error is:"  
## [1] 1.700662
```