Introduction to Statistical Learning

Dan Lizotte

2018-09-18

Advertising Data

A simulated dataset containing sales of child car seats at 400 different stores

```
Carseats
```

```
## # A tibble: 400 x 11
##    Sales CompPrice Income Advertising Population Price ShelveLoc   Age
##  * <dbl>     <dbl>  <dbl>       <dbl>      <dbl> <dbl> <fct>     <dbl>
##  1  9.5        138     73          11        276   120 Bad          42
##  2 11.2        111     48          16        260    83 Good         65
##  3 10.1        113     35          10        269    80 Medium       59
##  4  7.4        117    100           4        466    97 Medium       55
##  5  4.15       141     64           3        340   128 Bad          38
##  6 10.8        124    113          13        501    72 Bad          78
##  7  6.63       115    105           0         45   108 Medium       71
##  8 11.8        136     81          15        425   120 Good         67
##  9  6.54       132    110           0        108   124 Medium       76
## 10  4.69       132    113           0        131   124 Medium       76
## # ... with 390 more rows, and 3 more variables: Education <dbl>,
## #   Urban <fct>, US <fct>
```

Simple questions: Summaries of one variable

Data summaries

A "statistic" is a the result of applying a function (summary) to the data: `statistic <- function(data)`

E.g. ranks: Min, Quantiles, Median, Mean, Max

```
summary (Carseats$Sales)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   0.000   5.390   7.490   7.496   9.320  16.270
```

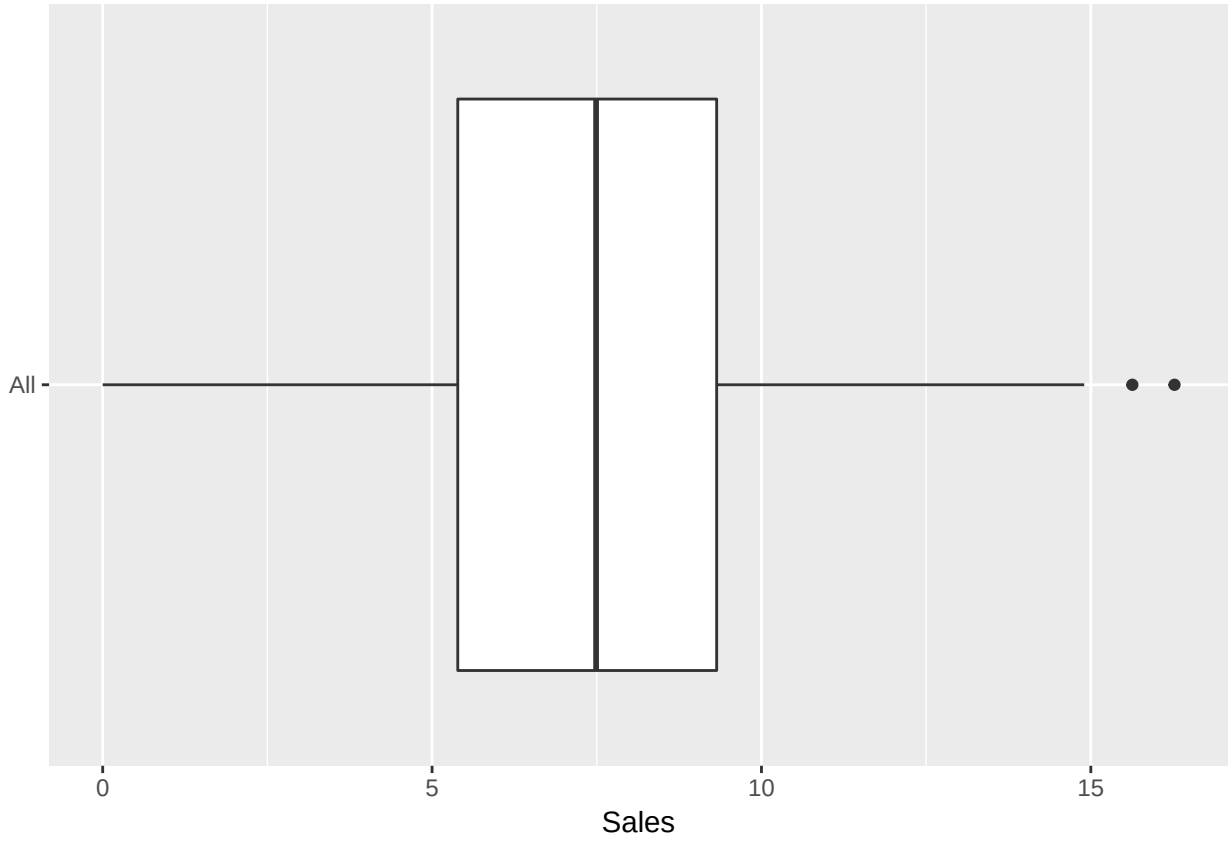
Roughly, a quantile for a proportion p is a value x for which p of the data are less than or equal to x. The first quartile, median, and third quartile are the quantiles for $p = 0.25$, $p = 0.5$, and $p = 0.75$, respectively.

Visual Summary 1: Box Plot, Jitter Plot

```
library(ggplot2);
summary(Carseats$Sales)
```

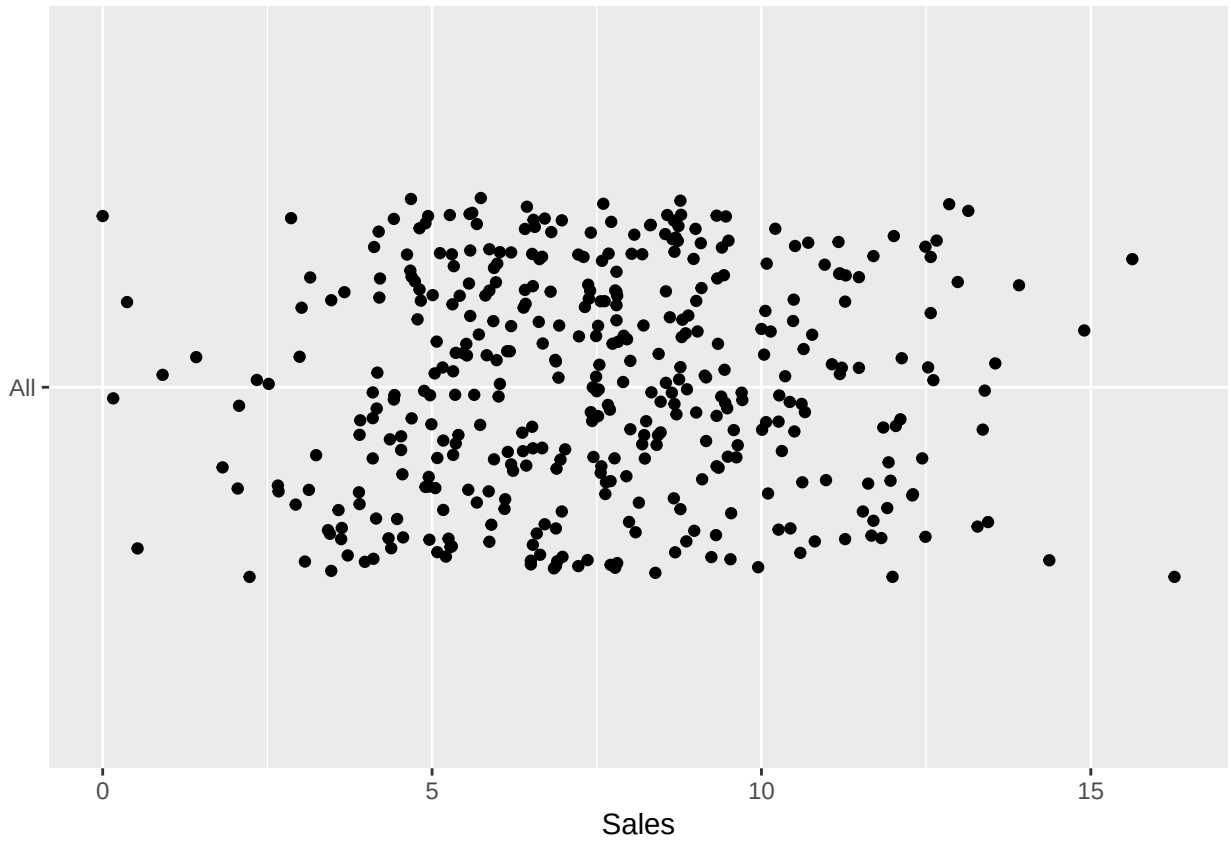
```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   0.000   5.390   7.490   7.496   9.320  16.270
```

```
ggplot(Carseats, aes(x="All",y=Sales)) + labs(x=NULL) + geom_boxplot() + coord_flip()
```



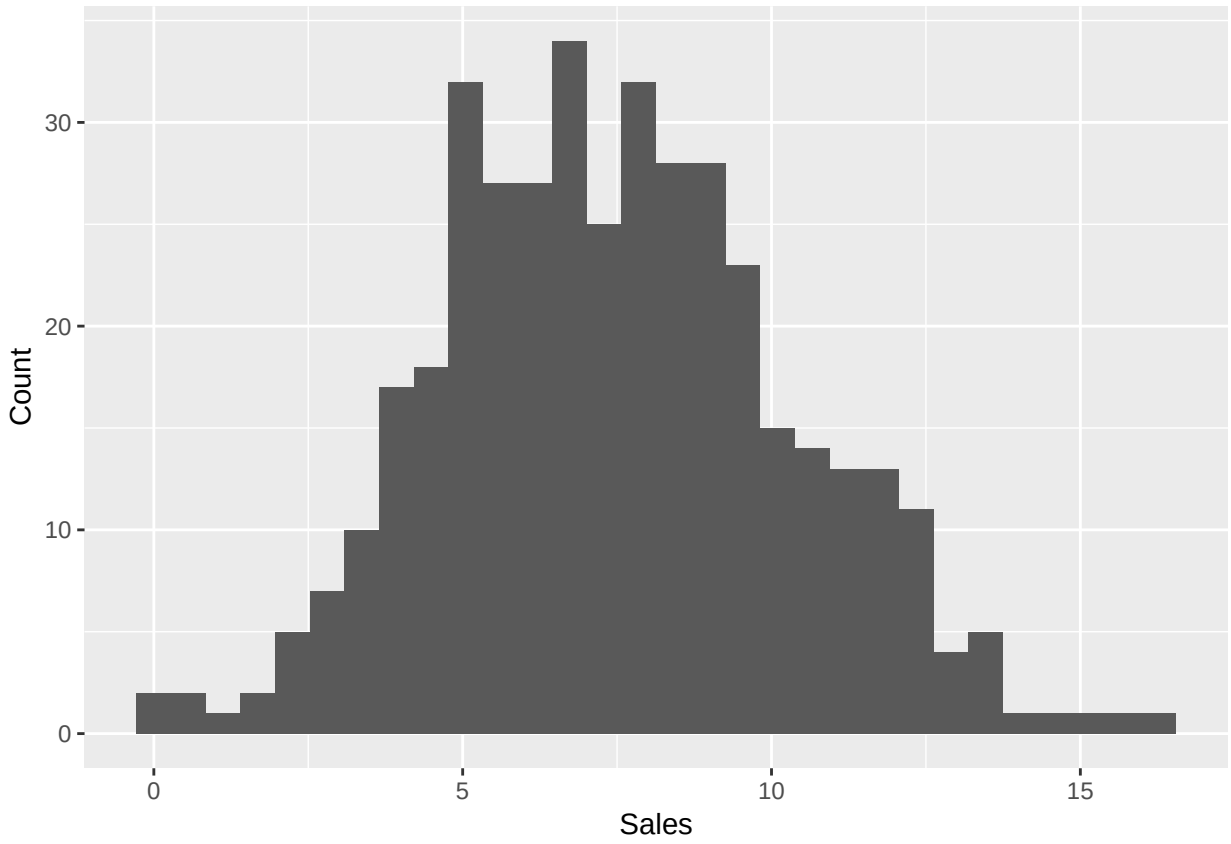
Visual Summary 2: Jitter Plot

```
library(ggplot2);library(gridExtra); #boxplot relatives
#jitter plot
ggplot(Carseats, aes(x="All",y=Sales)) + labs(x=NULL) +
    geom_jitter(position=position_jitter(height=0,width=0.25)) + coord_flip()
```



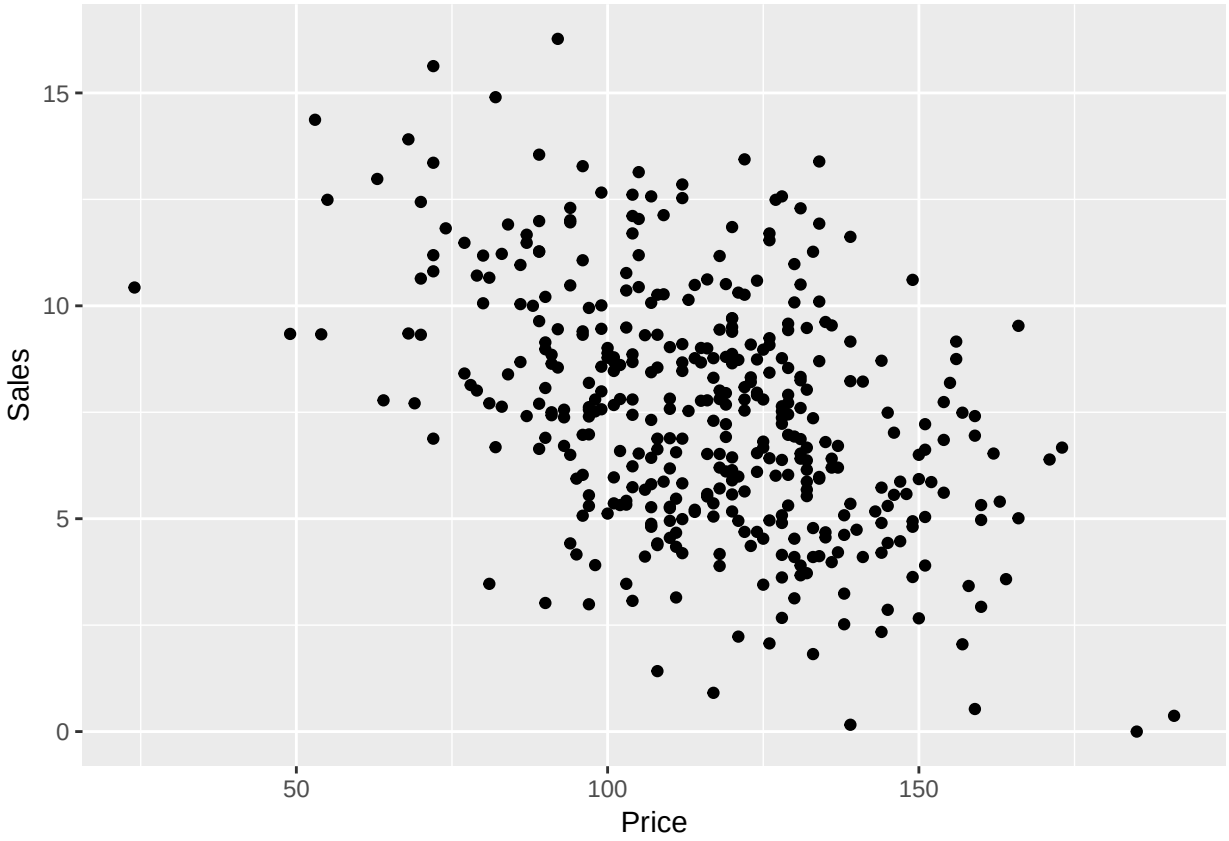
Visual Summary 3: Histogram

```
## Construct different histogram of eruption times
ggplot(Carseats, aes(x=Sales)) + labs(y="Count") + geom_histogram(aes(y = ..count..))
```


Count
30
20
10
0
0
5
10
15
Sales

Complex questions: Relationships

Relationships between variables



All of Supervised Learning

Proposal:

$$Y = f(X) + \epsilon$$

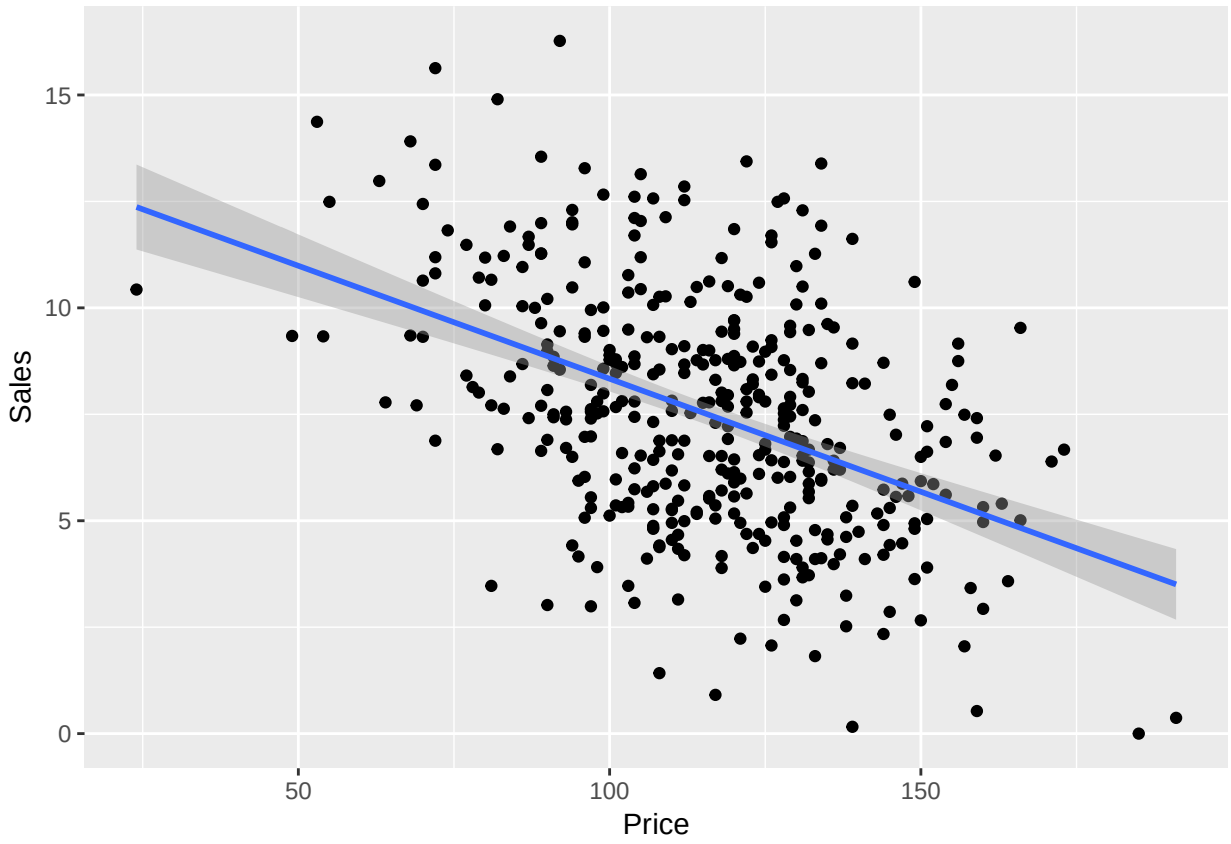
1. Here is some data
2. Tell me what f is

Example: linear fit

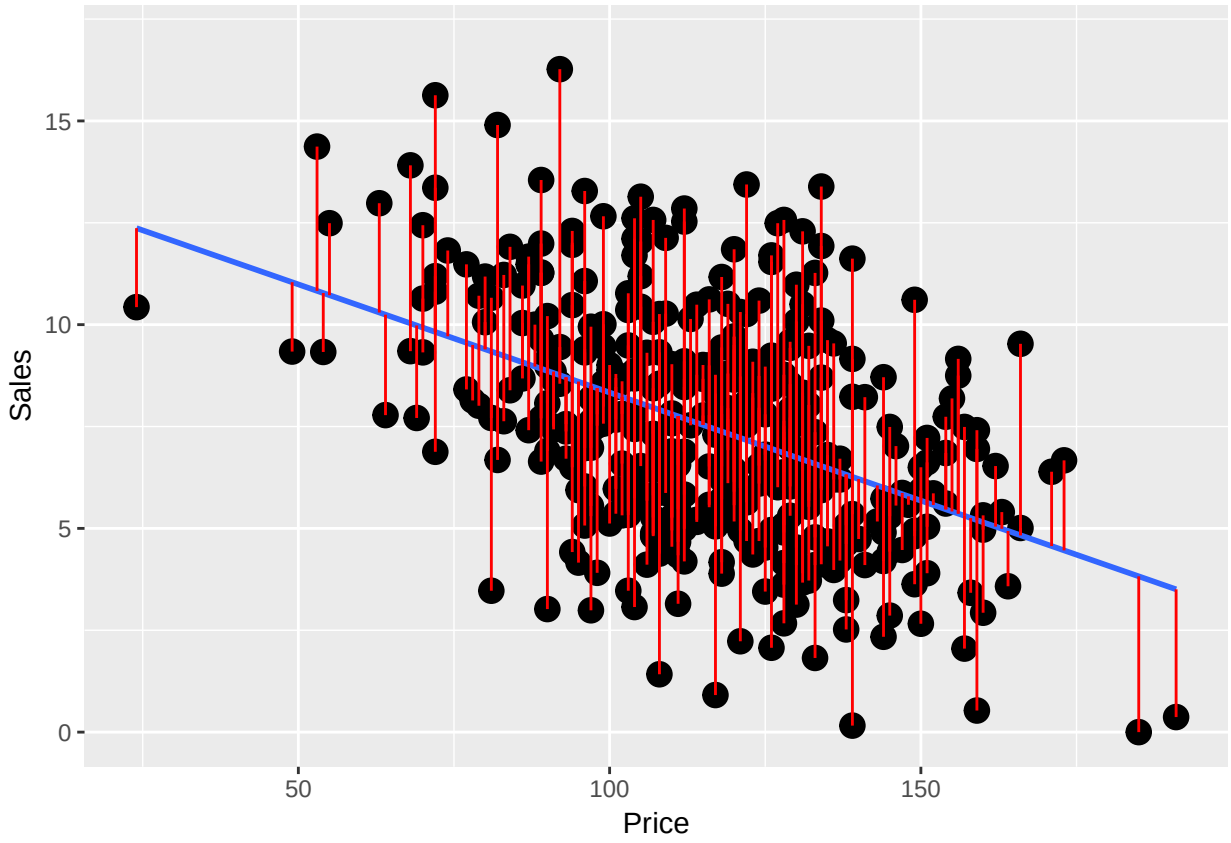
```
csform <- Sales ~ Price; csmod <- lm(csform, data=Carseats); print(csmod$coefficients)
```

```
## (Intercept)       Price
## 13.64191518 -0.05307302
```

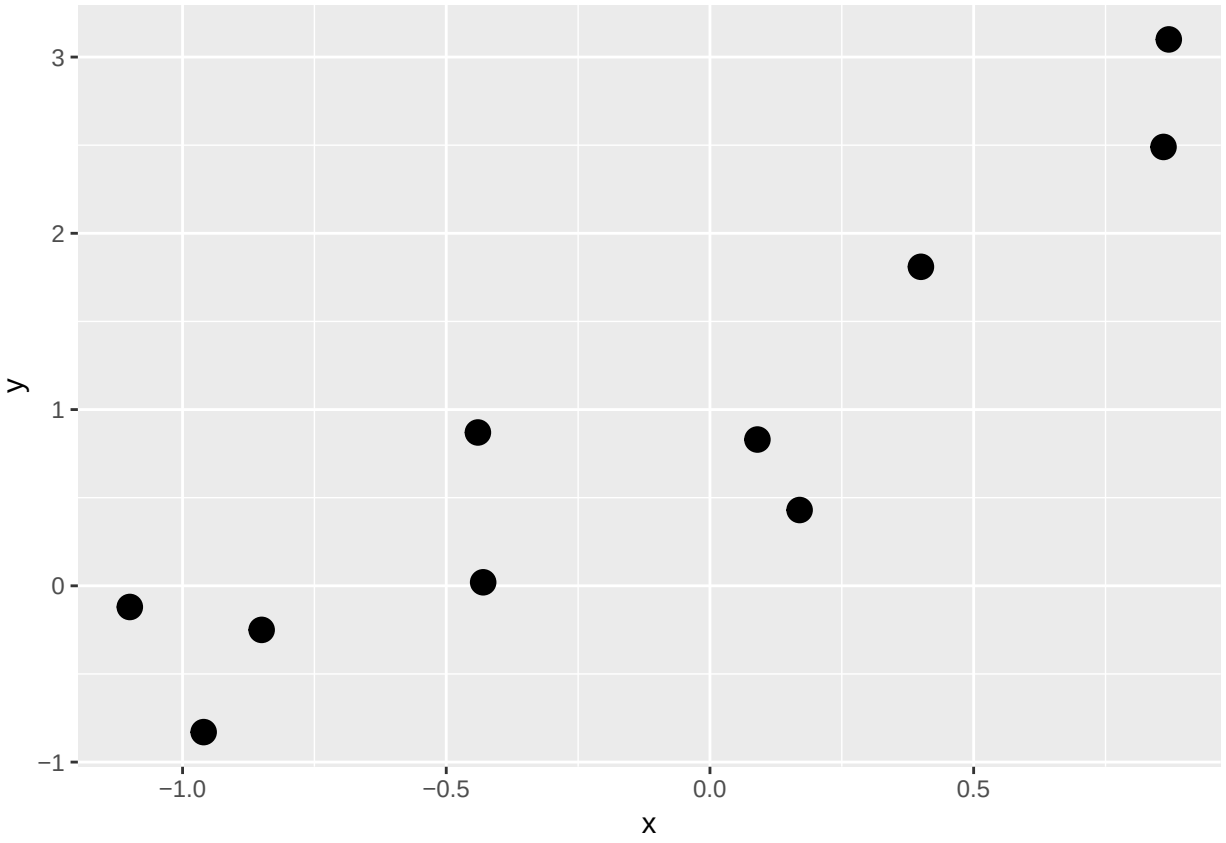
```
ggplot(Carseats, aes(x = Price, y = Sales)) + geom_point() + geom_smooth(method = lm)
```


15
10
5
0
Sales
50
100
150
Price

Fitting by Minimizing Error

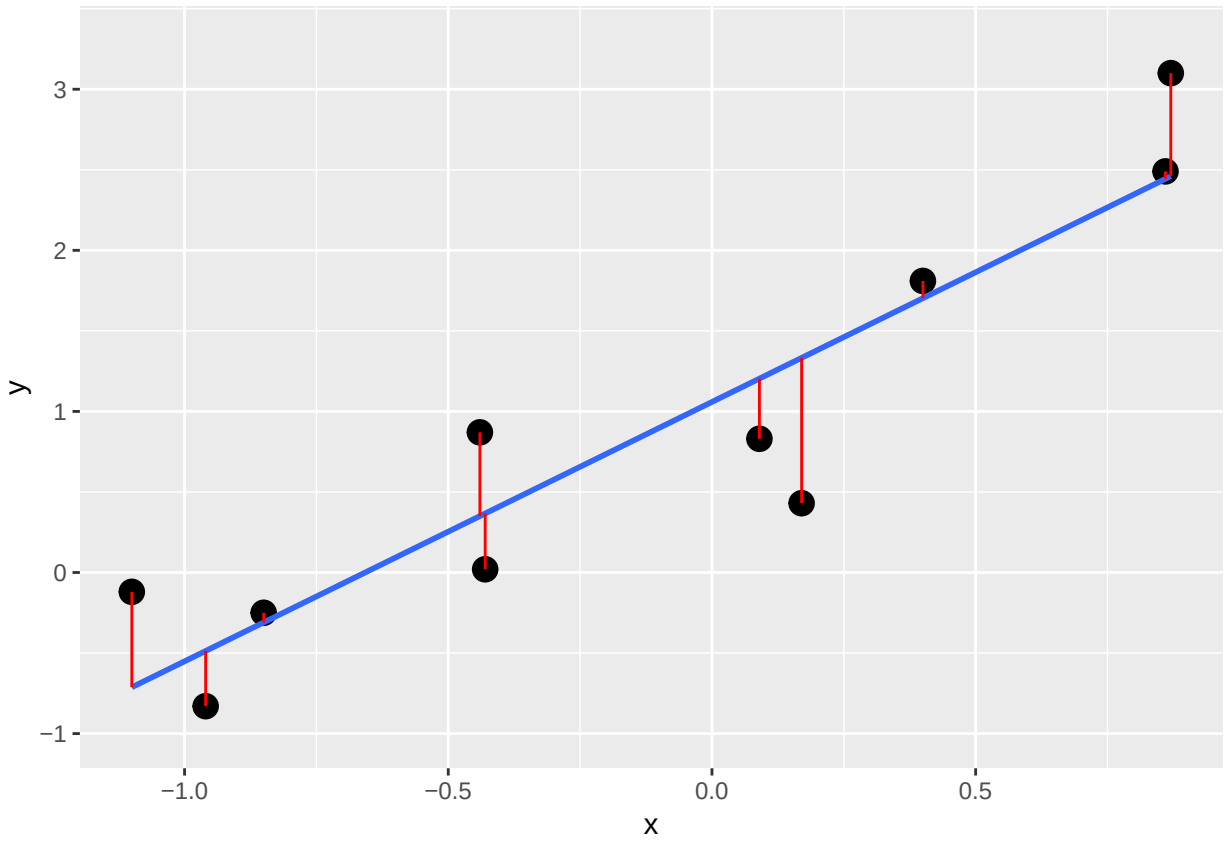


What kind of f are you looking for?



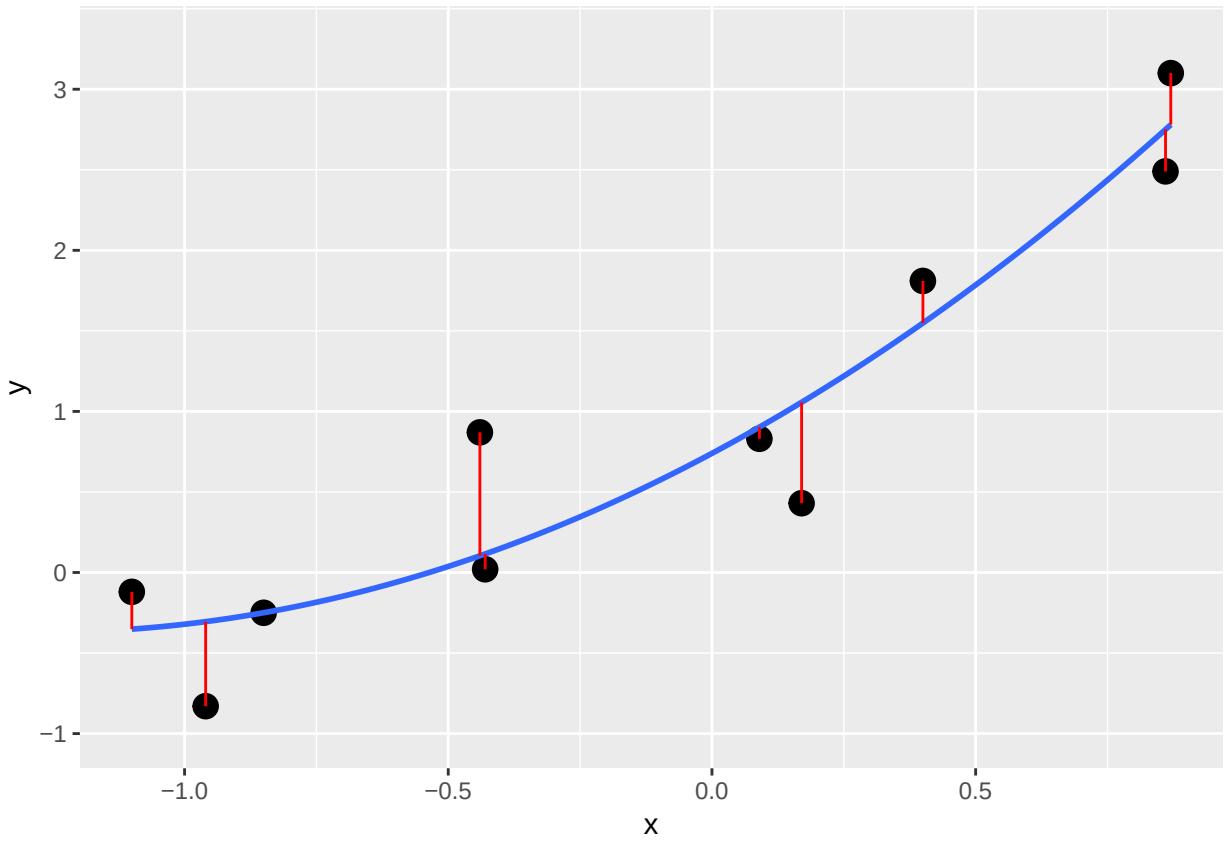
Data and linear fit

```
## (Intercept)           x
##         1.1         1.6
```



Data and quadratic fit

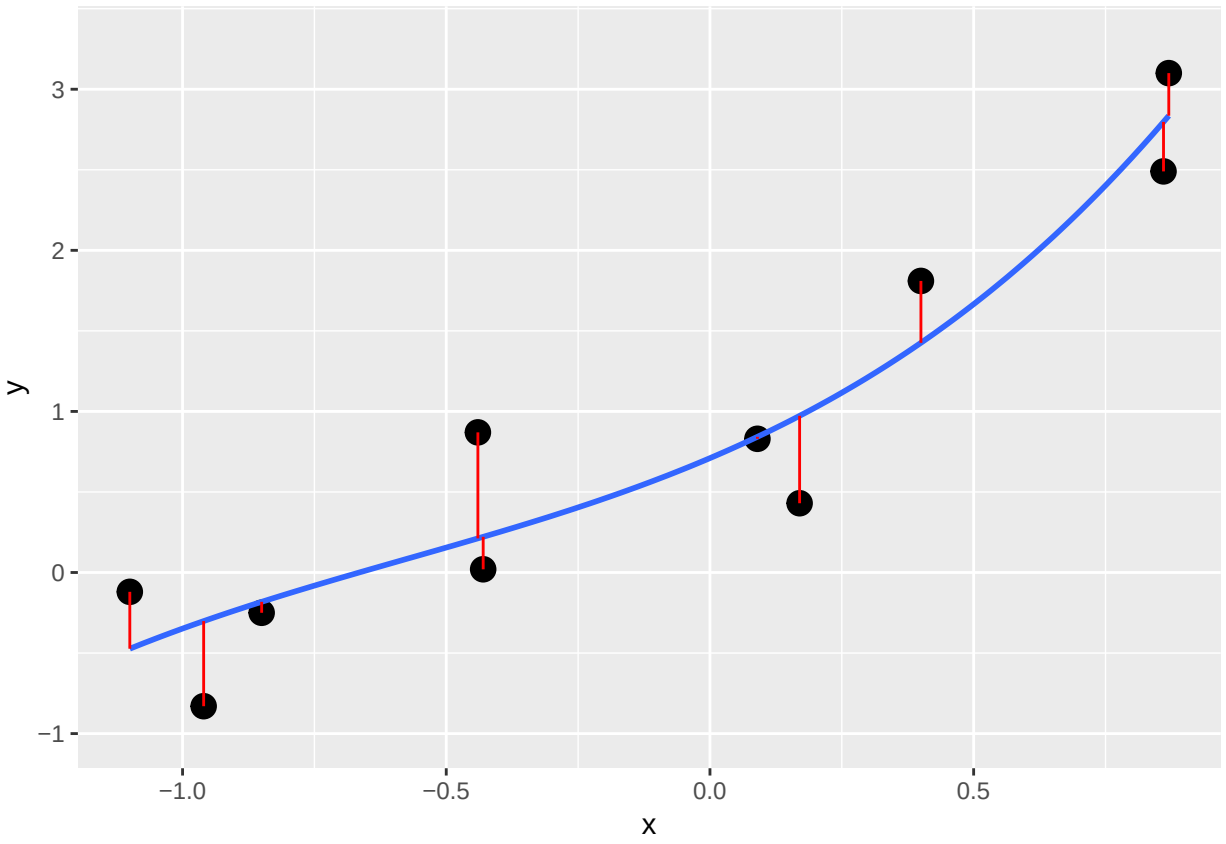
```
## (Intercept)           x      I(x^2)
##        0.74        1.75        0.69
```



Is this a better fit to the data?

Order-3 fit

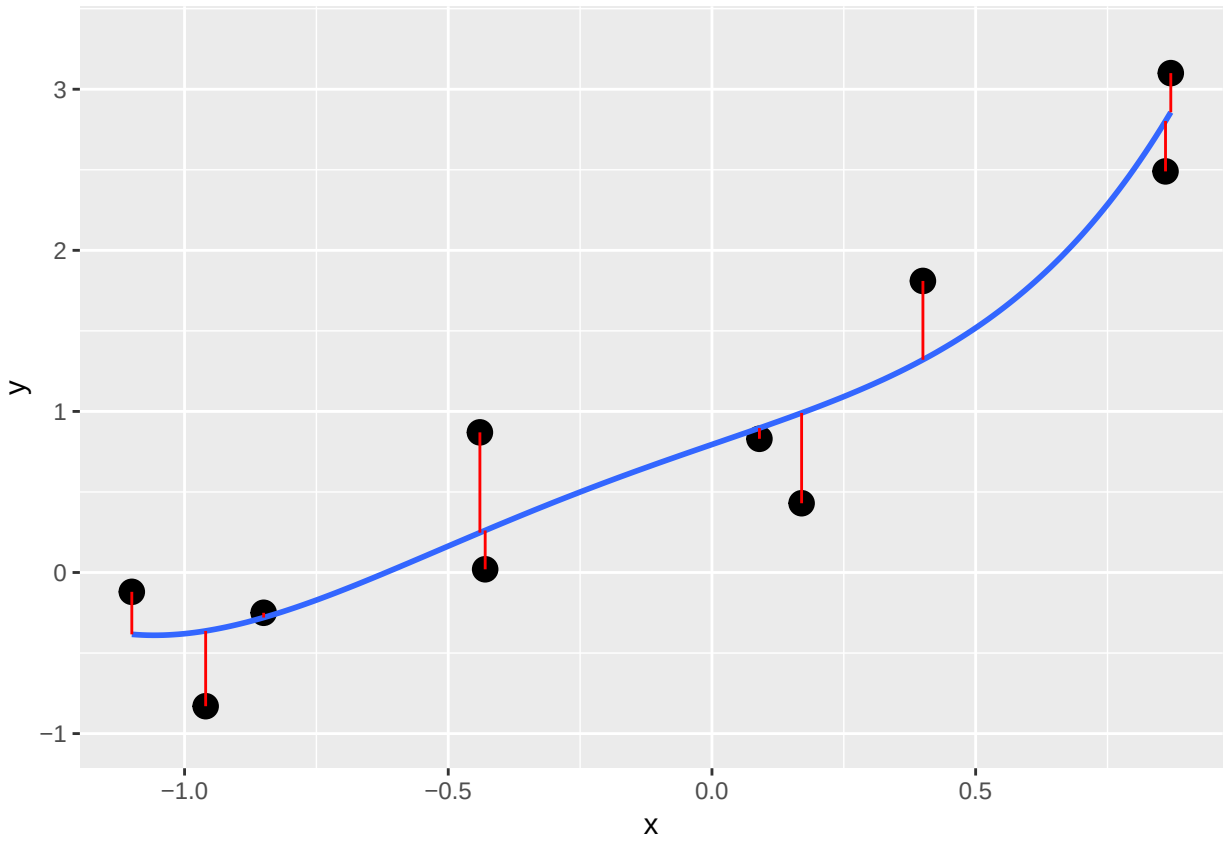
```
## (Intercept)           x      I(x^2)      I(x^3)
##        0.71        1.39        0.80        0.46
```



Is this a better fit to the data?

Order-4 fit

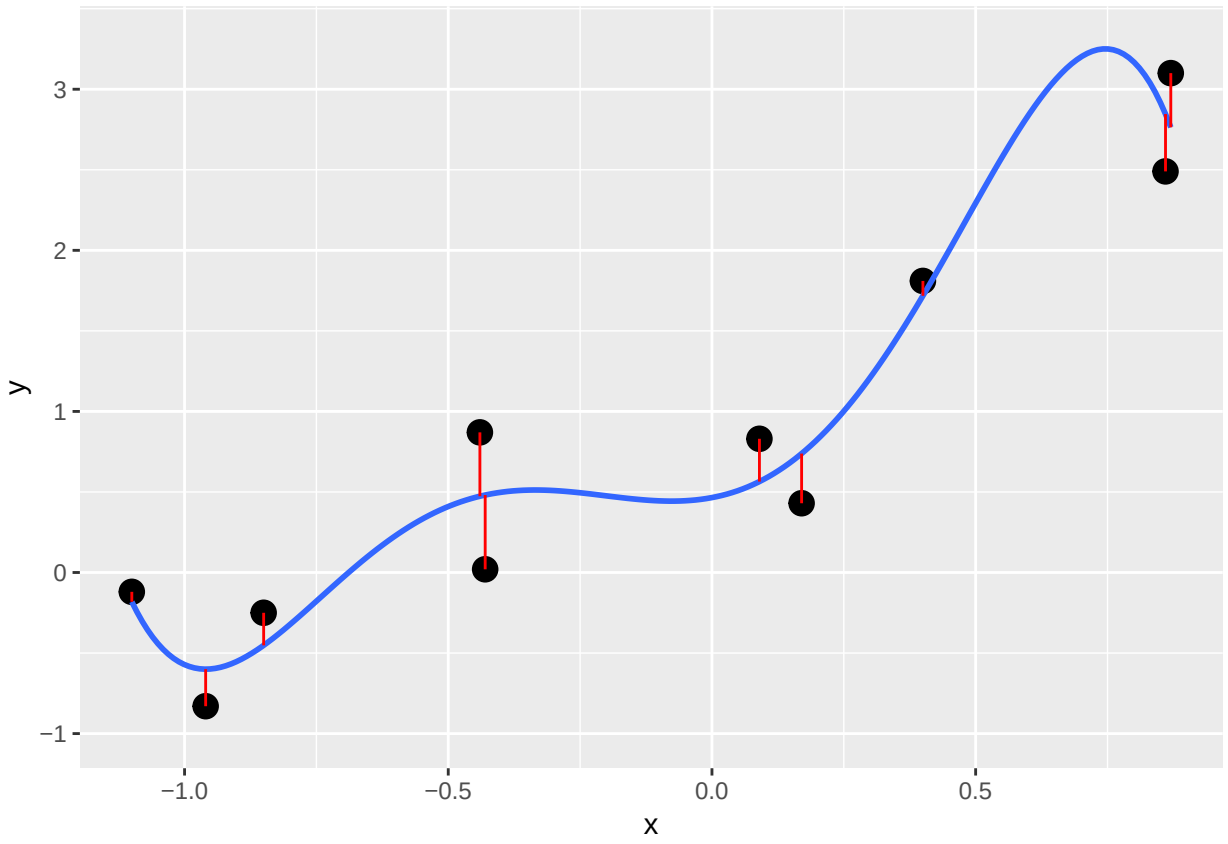
```
## (Intercept)           x      I(x^2)      I(x^3)      I(x^4)
##       0.795       1.128      -0.039       0.905       0.898
```



Is this a better fit to the data?

Order-5 fit

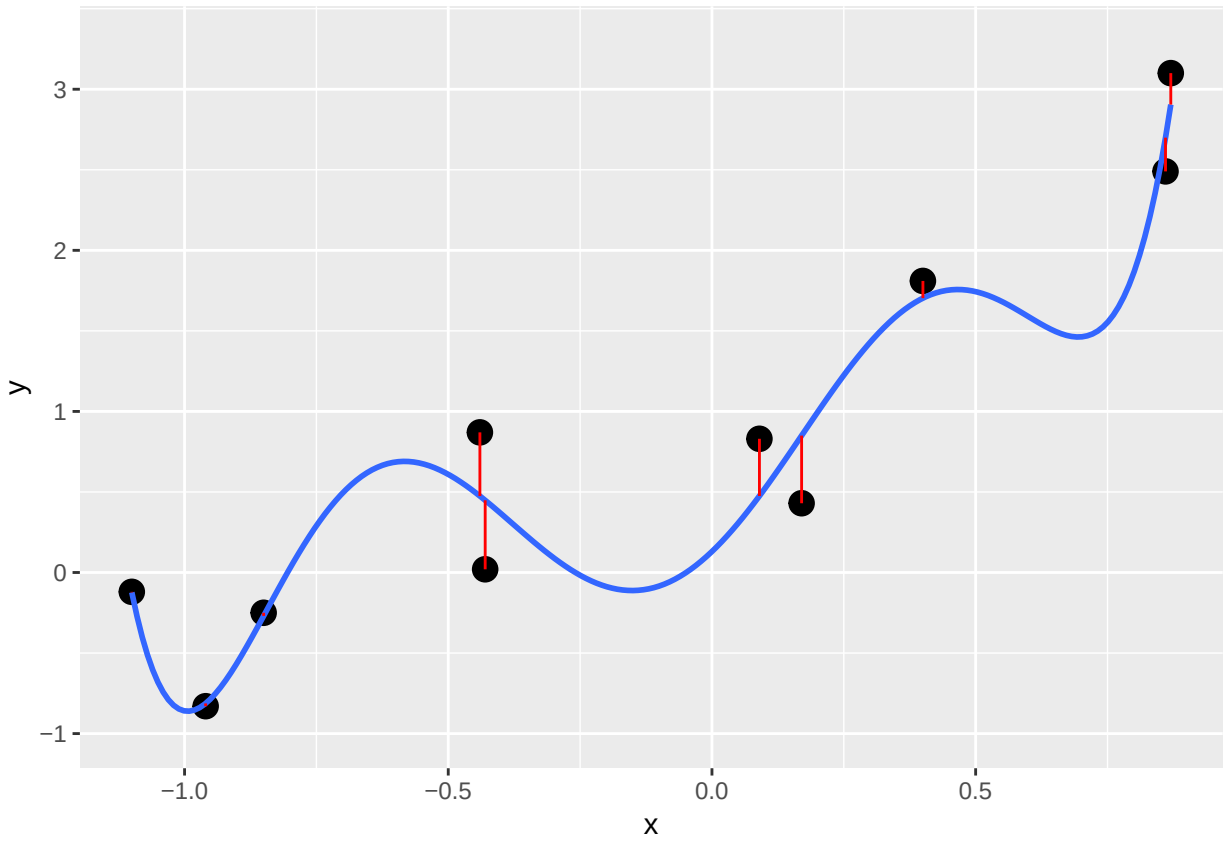
```
## (Intercept)           x      I(x^2)      I(x^3)      I(x^4)      I(x^5)
##        0.47        0.62        4.86        6.75       -5.25       -6.72
```



Is this a better fit to the data?

Order-6 fit

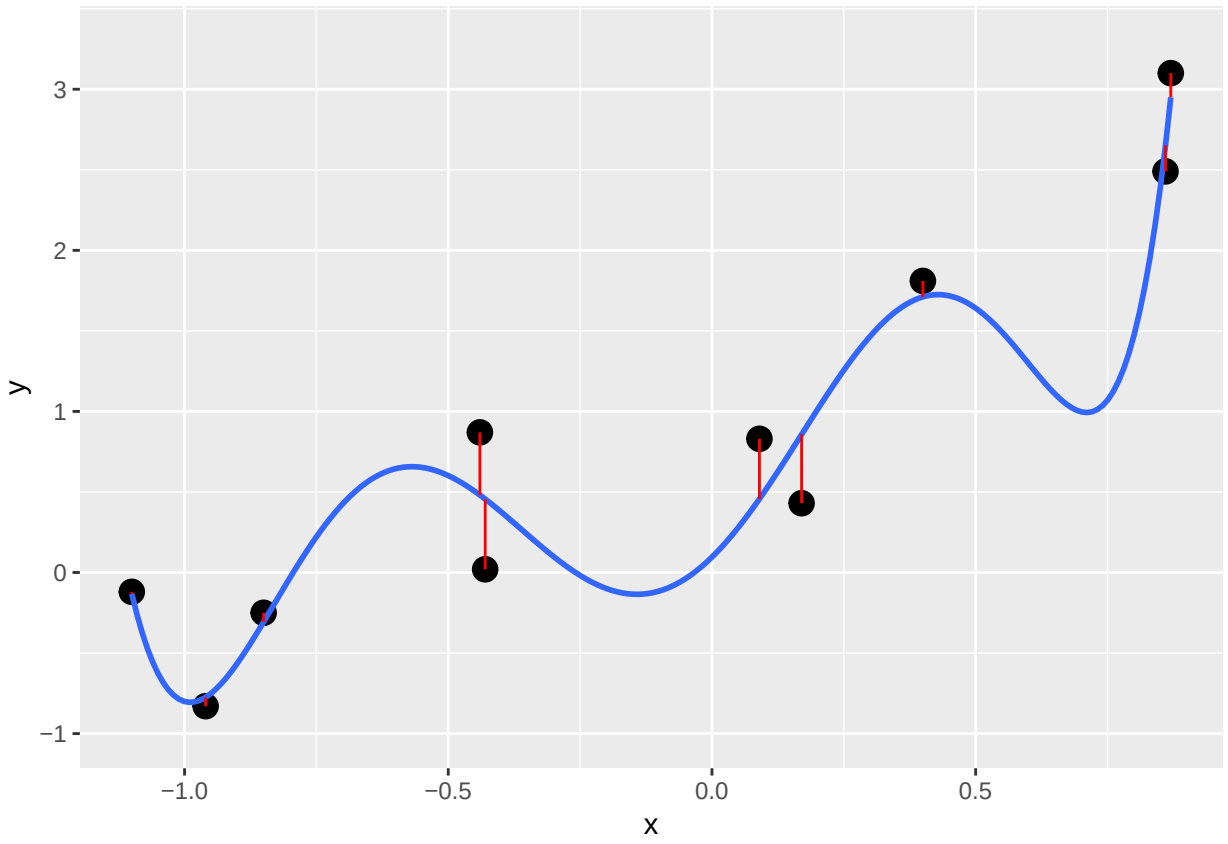
```
## (Intercept)           x      I(x^2)      I(x^3)      I(x^4)      I(x^5)      I(x^6)
##        0.13        3.13        8.99      -11.11      -23.83       12.52       18.38
```



Is this a better fit to the data?

Order-7 fit

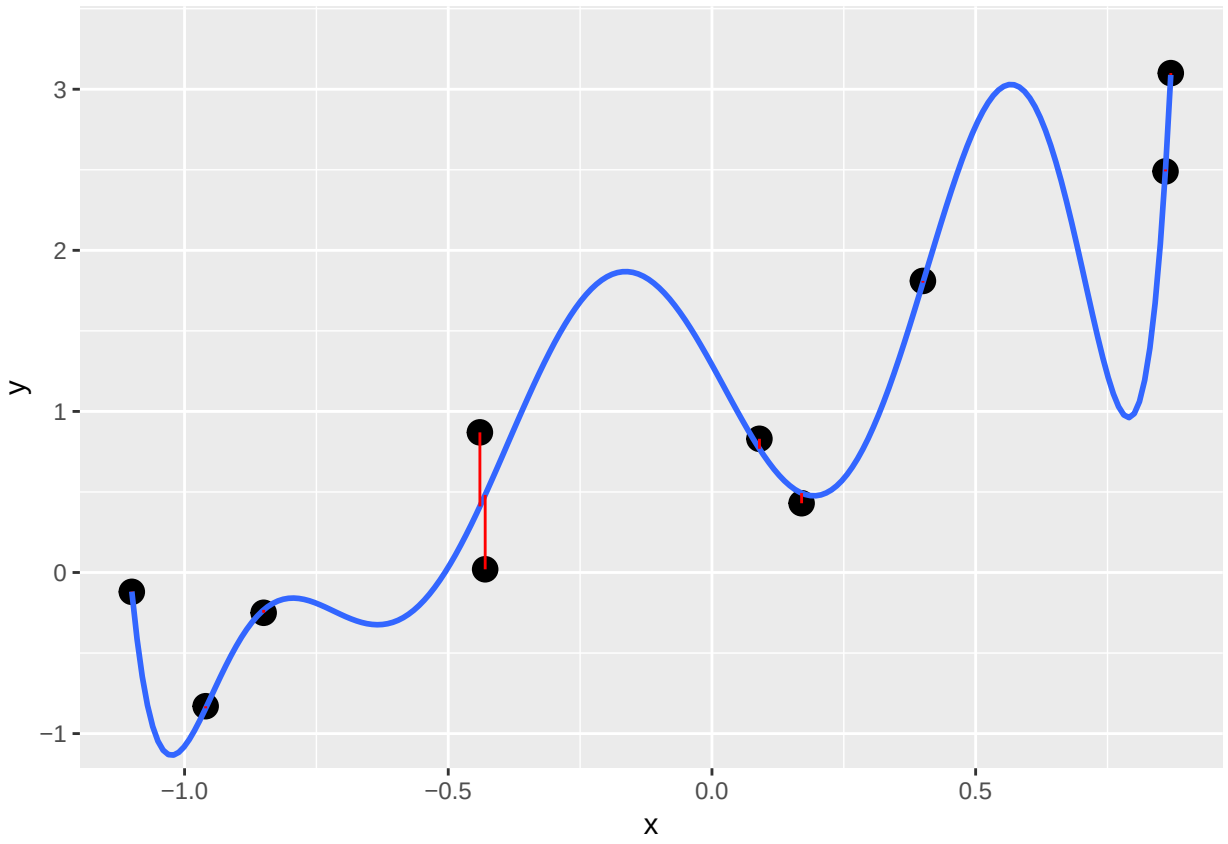
```
## (Intercept)           x      I(x^2)      I(x^3)      I(x^4)      I(x^5)      I(x^6)      I(x^7)
##       0.096       3.207      10.193     -11.078     -30.742       8.263      25.527       5.483
```



Is this a better fit to the data?

Order-8 fit

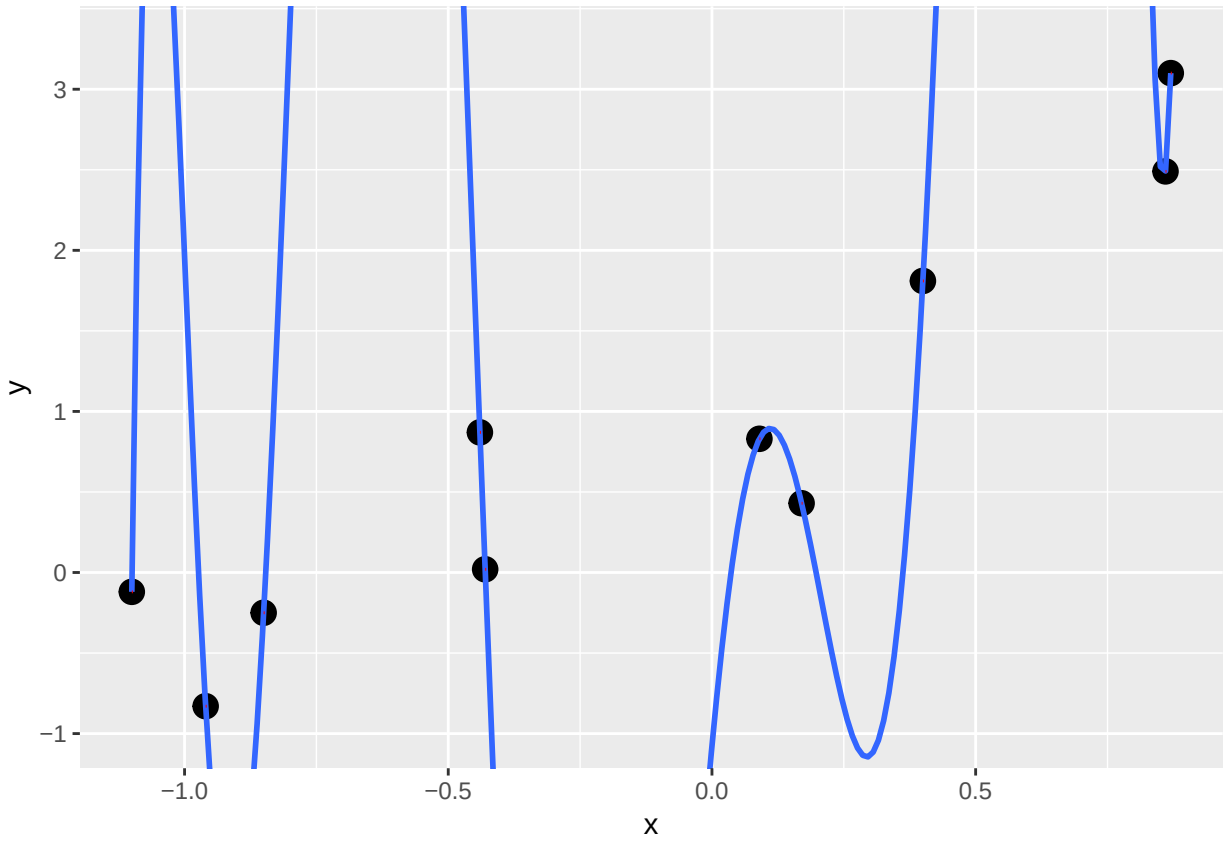
```
## (Intercept)           x      I(x^2)      I(x^3)      I(x^4)      I(x^5)      I(x^6)      I(x^7)
##         1.3        -5.9        -5.1        69.9        48.8      -172.0      -131.9       123.3
```



Is this a better fit to the data?

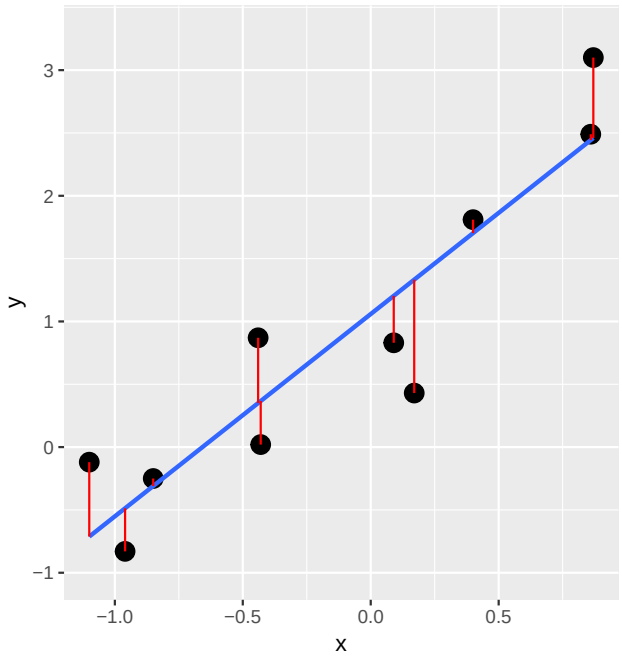
Order-9 fit

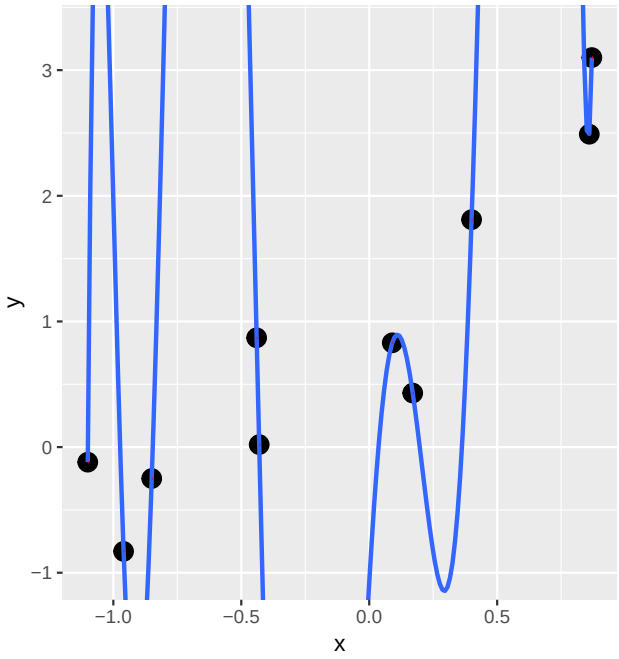
```
## (Intercept)           x      I(x^2)      I(x^3)      I(x^4)      I(x^5)      I(x^6)      I(x^7)
##        -1.1        34.8      -127.9      -379.9      1186.9      1604.8     -2475.4     -2627.6
```



Is this a better fit to the data?

Evaluating Performance





Which do you prefer and why?

Recommended exercises

- JWHT 2.3 Lab: Introduction to R

(Or, follow along and see if you can do it in python.)