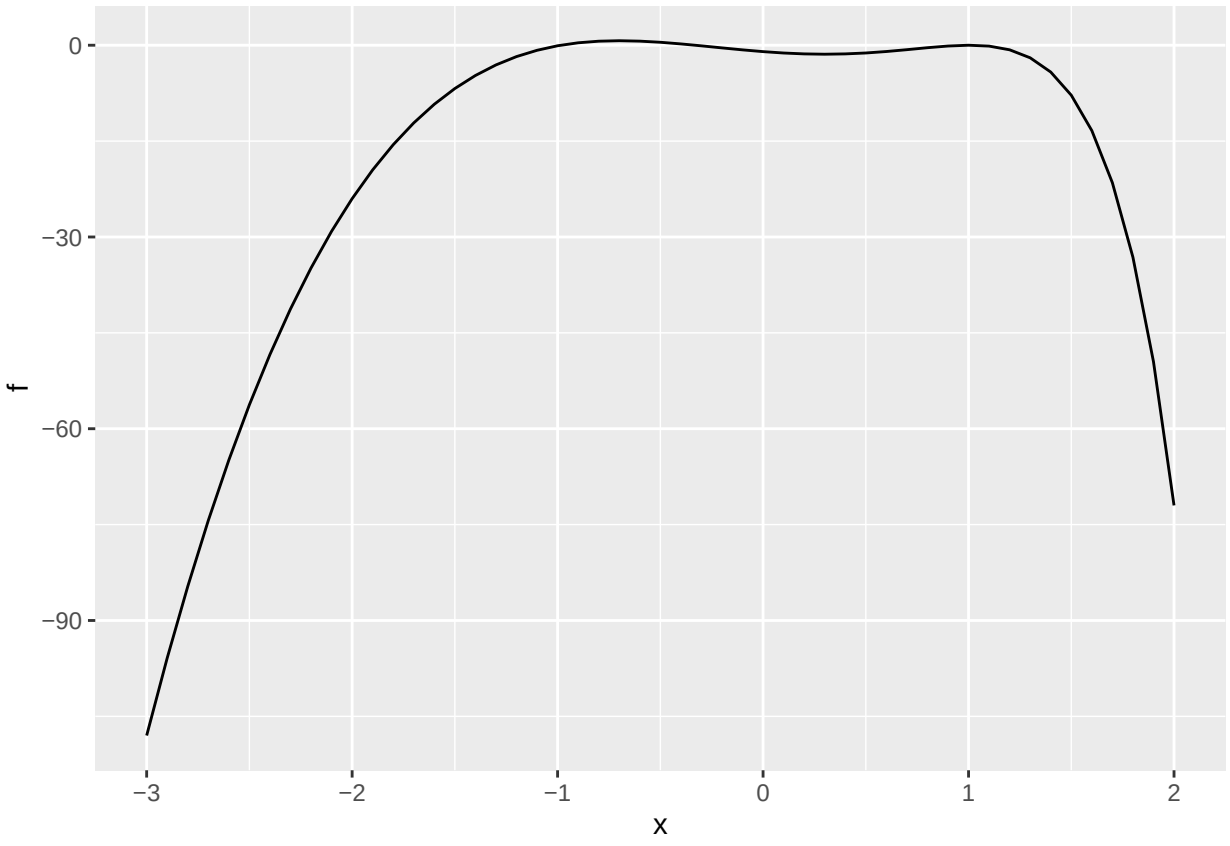
Week6_SimonsenHomework

Steven Simonsen

2024-10-06

```
set.seed(123)
library(ggplot2)

true_relationship <- function(x) { return(6*x^3 + 6*x^2 - 12^x) }

x <- seq(-3, 2, by = 0.1)
f <- true_relationship(x)
ggplot() + geom_line(aes(x = x, y = f), color = "black")
```

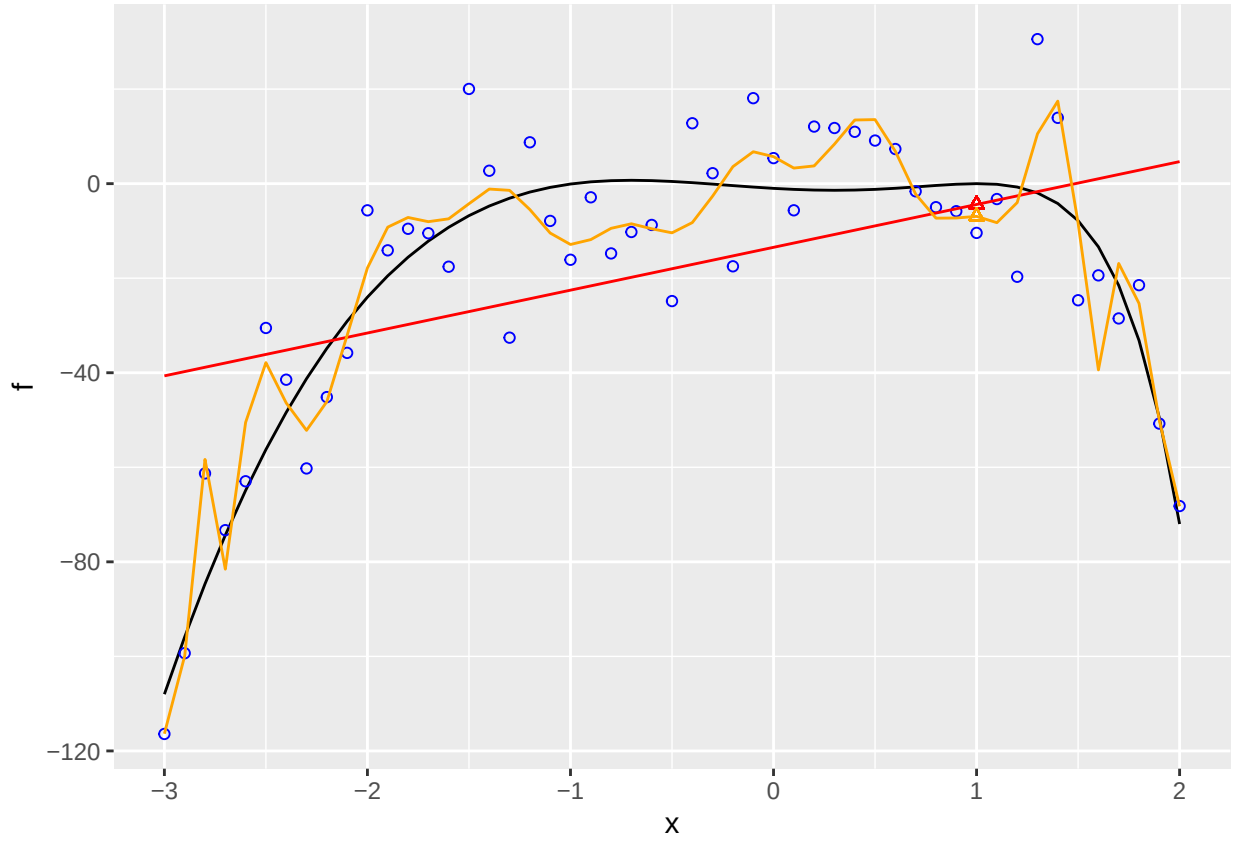



```
observations <- f + rnorm(length(x), mean = 0, sd = 15)


model1 <- lm(observations ~ poly(x, 1))
predictions1 <- predict(model1, newdata = data.frame(x = x))
```

```
model25 <- lm(observations ~ poly(x, 25))
predictions25 <- predict(model25, newdata = data.frame(x = x))

data <- data.frame(x = x,
                   f = f,
                   observations = observations,
                   lm = predictions1,
                   pm = predictions25)

ggplot(data, aes(x = x)) +
  geom_line(aes(y = f), color = "black") +
  geom_point(aes(y = observations), color = "blue", shape = 1) +
  geom_line(aes(y = lm), color = "red", linetype = "solid") +
  geom_line(aes(y = pm), color = "orange", linetype = "solid") +
  geom_point(aes(x = 1, y = data[x == 1, "lm"]), color = "red", shape=2) +
  geom_point(aes(x = 1, y = data[x == 1, "pm"]), color = "orange", shape=2)
```

```
## Warning in geom_point(aes(x = 1, y = data[x == 1, "lm"]), color = "red", : All aesthetics have lengt
## i Please consider using `annotate()` or provide this layer with data containing
##   a single row.
```

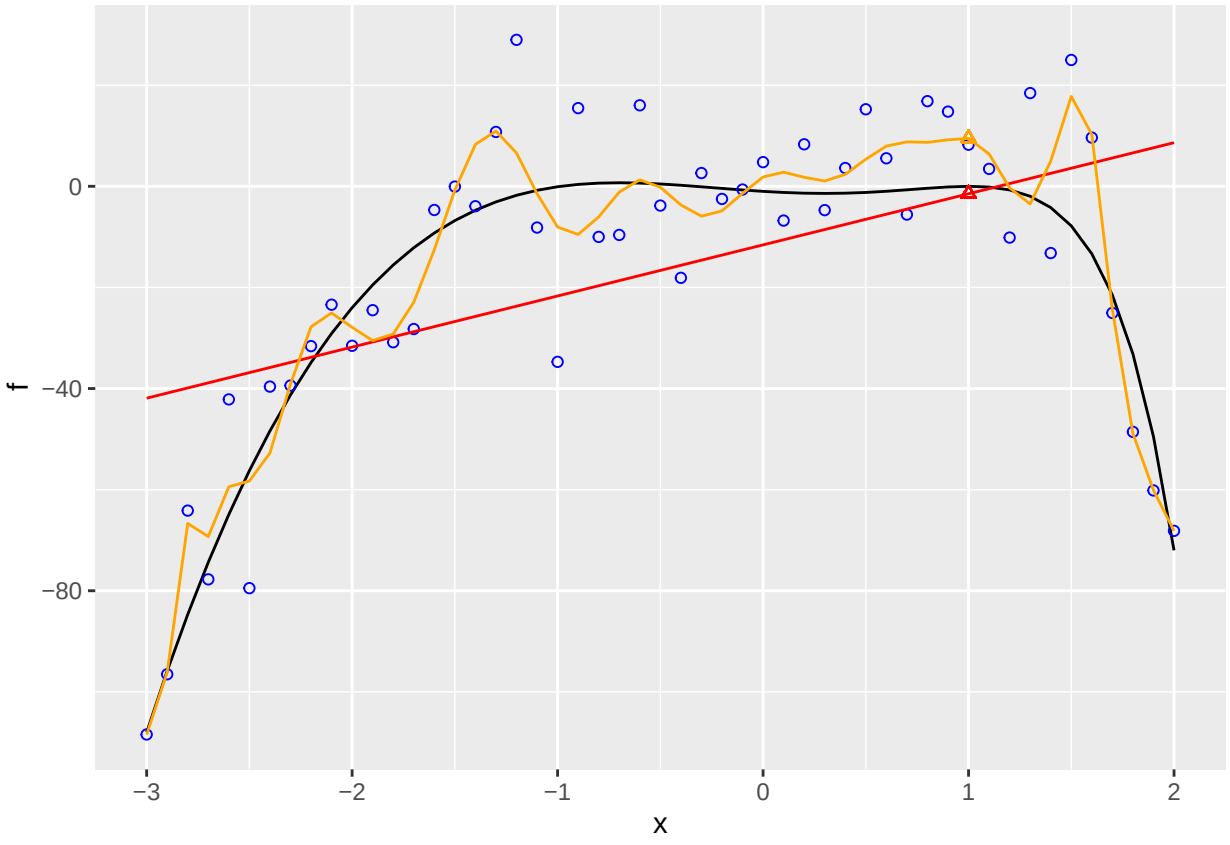
```
## Warning in geom_point(aes(x = 1, y = data[x == 1, "pm"]), color = "orange", : All aesthetics have le
## i Please consider using `annotate()` or provide this layer with data containing
##   a single row.
```



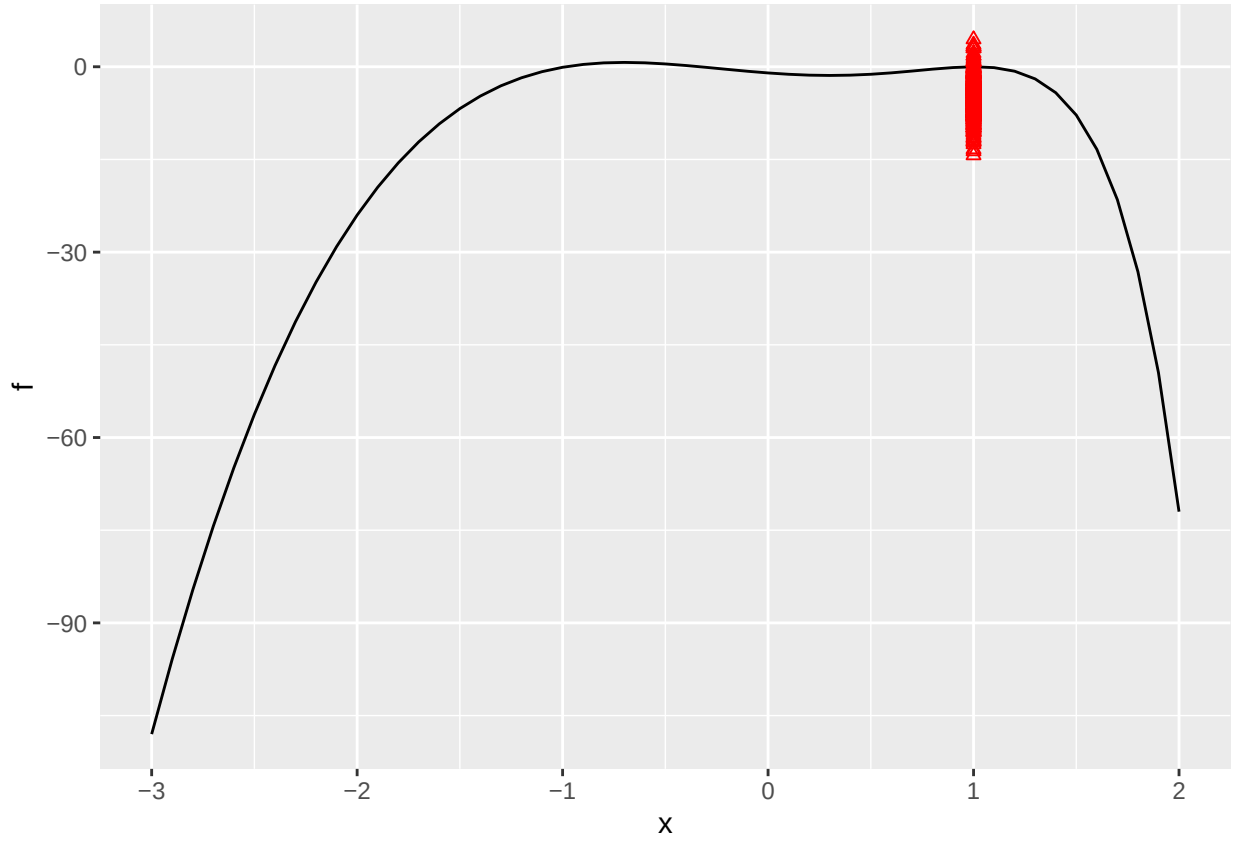

```
observations_new <- f + rnorm(length(x), mean = 0, sd = 15)
```

```r
model1 <- lm(observations_new ~ poly(x, 1))
predictions1 <- predict(model1, newdata = data.frame(x = x))

model25 <- lm(observations_new ~ poly(x, 25))
predictions25 <- predict(model25, newdata = data.frame(x = x))

data <- data.frame(x = x,
                   f = f,
                   observations = observations_new,
                   lm = predictions1,
                   pm = predictions25)

ggplot(data, aes(x = x)) +
  geom_line(aes(y = f), color = "black") +
  geom_point(aes(y = observations), color = "blue", shape = 1) +
  geom_line(aes(y = lm), color = "red", linetype = "solid") +
  geom_line(aes(y = pm), color = "orange", linetype = "solid") +
  geom_point(aes(x = 1, y = data[x == 1, "lm"]), color = "red", shape = 2) +
  geom_point(aes(x = 1, y = data[x == 1, "pm"]), color = "orange", shape = 2)
```

```
## Warning in geom_point(aes(x = 1, y = data[x == 1, "lm"]), color = "red", : All aesthetics have lengt
## i Please consider using `annotate()` or provide this layer with data containing
##   a single row.
## All aesthetics have length 1, but the data has 51 rows.
## i Please consider using `annotate()` or provide this layer with data containing
##   a single row.
```

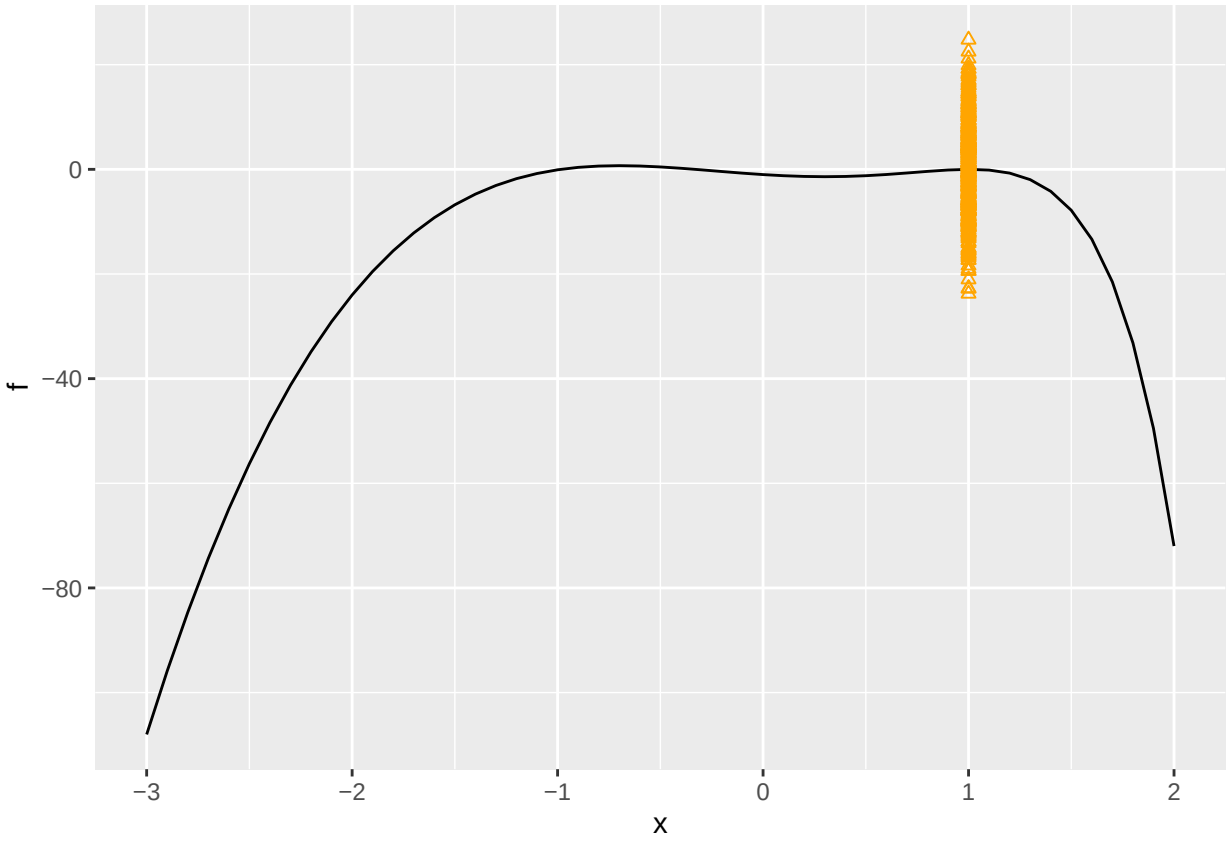


```r
results1 <- data.frame(x = 1, f_pred = 0)

for (i in 1:500) {
  x <- seq(-3, 2, by = 0.1)
  f <- true_relationship(x)
  temp_observations <- f + rnorm(length(x), mean=0, sd=15)
  model1 <- lm(temp_observations ~ poly(x, 1))
  results1[i, 1] <- 1
  results1[i, 2] <- predict(model1, newdata = data.frame(x=1))
}

ggplot() +
geom_line(data = data, aes(x = x, y = f), color = "black") +
geom_point(data = results1, aes(x = x, y = f_pred), color="red", shape=2)
```

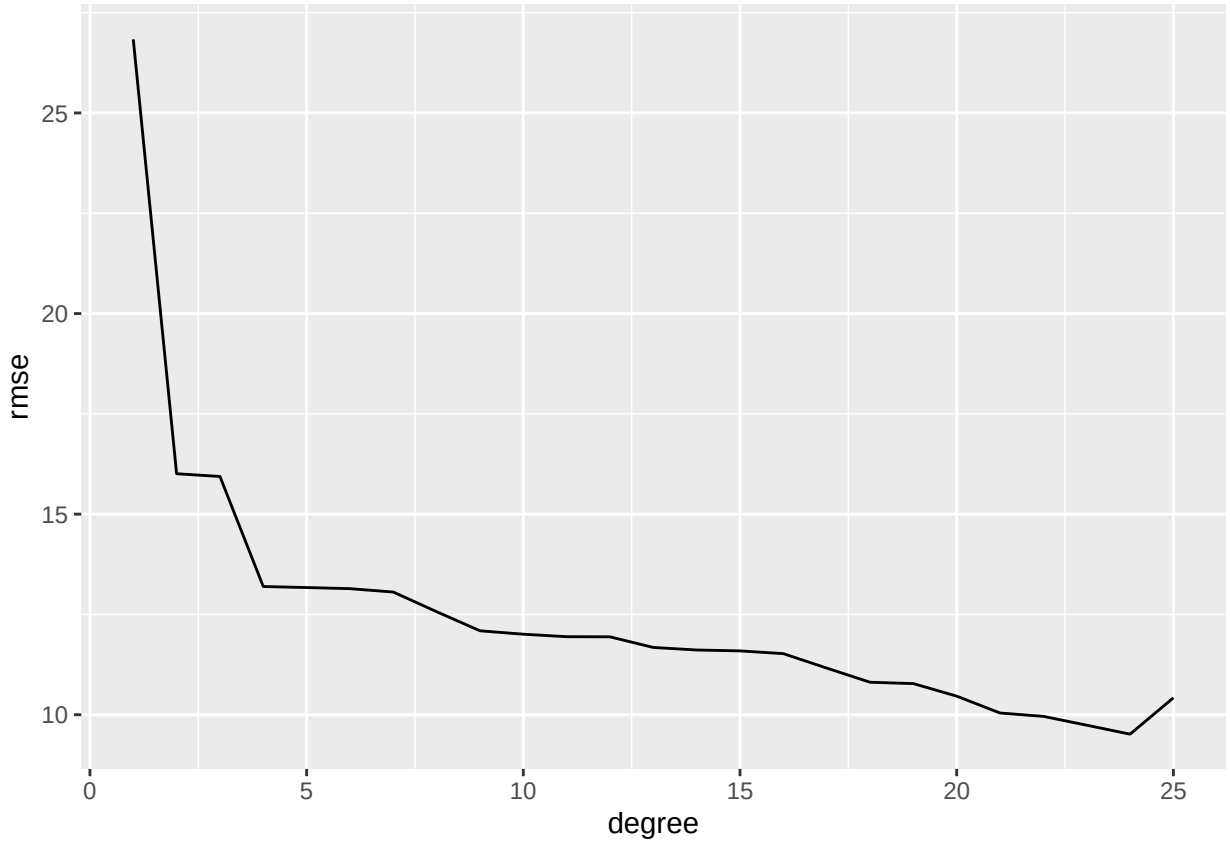



```r
results20 <- data.frame(x = 1, f_pred = 0)

for (i in 1:500) {
  x <- seq(-3, 2, by = 0.1)
  f <- true_relationship(x)
  temp_observations <- f + rnorm(length(x), mean=0, sd=15)
  model20 <- lm(temp_observations ~ poly(x, 20))
  results20[i, 1] <- 1
  results20[i, 2] <- predict(model20, newdata = data.frame(x=1))
}
```

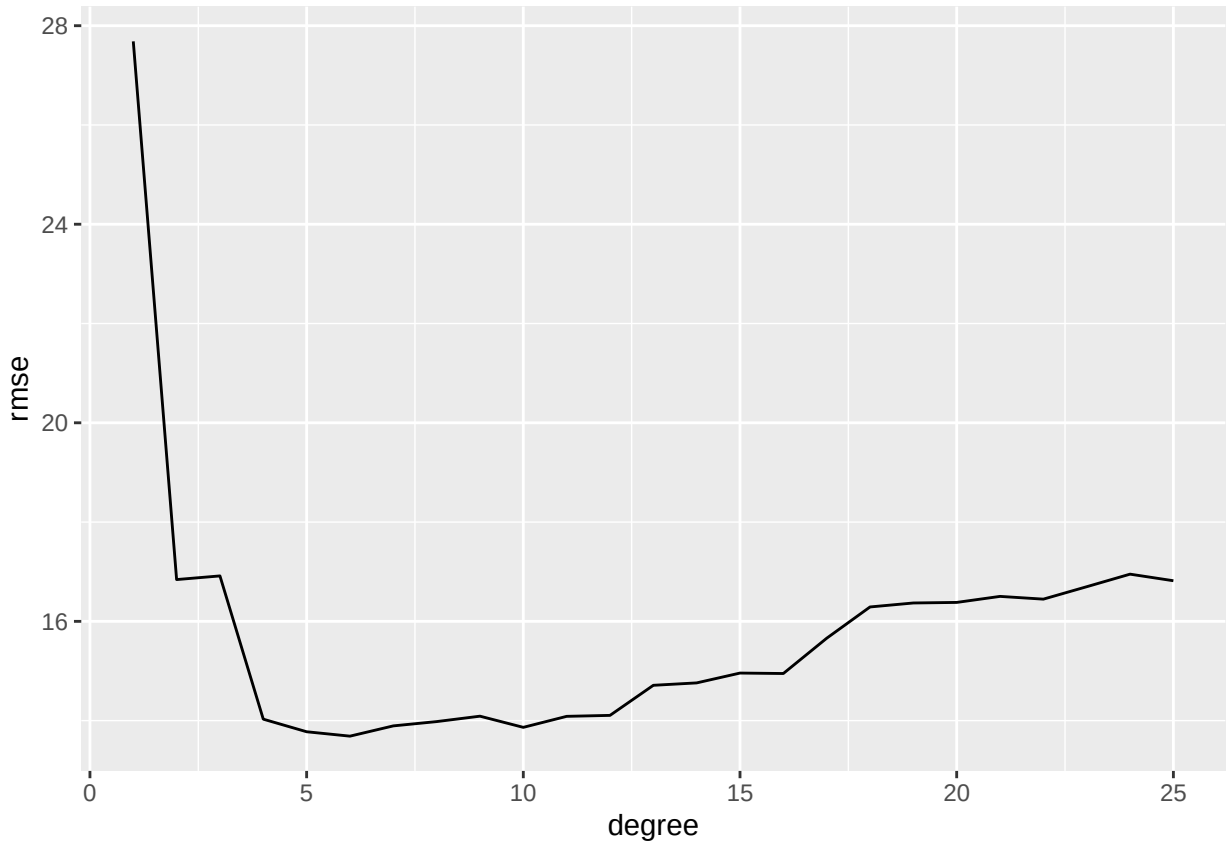
```
ggplot() +
geom_line(data = data, aes(x = x, y = f), color = "black") +
geom_point(data = results20, aes(x = x, y = f_pred), color="orange", shape=2)
```




```
models <- vector("list", 25)

for (degree in 1:25) {
  model <- lm(observations ~ poly(x, degree))
  models[[degree]] <- model
}

results <- data.frame(degree = 1:25, rmse = 0)

for (degree in 1:25) {
  predictions <- predict(models[[degree]], newdata = data.frame(x=x))
  results[results$degree==degree, "rmse"] <-
    sqrt((1/length(predictions))*sum((predictions-observations)^2))
}

ggplot() +
geom_line(data = results, aes(x = degree, y = rmse), color = "black")
```




```
results <- data.frame(degree = 1:25, rmse = 0)

for (degree in 1:25) {
  predictions <- predict(models[[degree]], newdata = data.frame(x=x))
  results[results$degree==degree, "rmse"] <-
    sqrt((1/length(predictions))*sum((predictions-observations_new)^2))
}

ggplot() +
geom_line(data = results, aes(x = degree, y = rmse), color = "black")
```



Exercises 1) Why do the two RMSE plots show very different behavior? Use underfitting and overfitting in your answer, as well as the bias-variance tradeoff.

In the first RMSE plot, the original observations were used both to fit the model and evaluate on the same dataset. In the evaluation of the model, the higher degree polynomials are simply trying to match the observations. This behavior leads to overfitting, since the higher the degree of polynomial, the higher the variance and lower the bias in the bias-variance tradeoff. Since the evaluation data (or validation data) is the same as the fitted data, the higher the degree, the better the model can memorize the data, and therefore perform very well in evaluation.

By contrast, the second RMSE plot fits the model on the original observations, and then evaluates the model on observations_new (this could be considered the validation dataset). Since the new observations are not the same as the original observations used to fit the model, we experience very different results in the plot. The model experiences performance improvement in terms of lowering the RMSE to a point, but after about 10 degrees begins to degrade in performance. This is due to the fact that the model is overfitting, or essentially memorizing the original observations and then using this memorization to evaluate on the new observations. Thus, the RMSE begins to rise again after 10 degrees.

It should be noted that in both graphs, the left side of the RMSE curve, up to about four degrees, is underfitting. This is because the model is not flexible enough to capture the true shape of the data, or in other words has a high bias, and low variance in the bias-variance tradeoff.

2) Using the last plot above, what degree of polynomial should we choose? After running all of the above code, substitute the degree you chose for the question mark in the code below and run it. Copy and paste the resulting plot.

We should choose a degree of about six, as this minimizes the rmse when evaluating the model on a new dataset.

```r
model <- lm(observations ~ poly(x, 6))

predictions=predict(model, newdata = data.frame(x=x))

data = data.frame(x=x, f=f, predictions=predictions)

ggplot(data, aes(x=x)) +
geom_line(aes(y = f), color = "black") +
geom_line(aes(y = predictions), color = "red", linetype="solid")
```