lecture06-penalized-regression-annotated.R

mac

2019-07-26

```
# STA303 S19 Lecture 6: penalized regression
#
# # Load required packages
#

library(dplyr)
```

```
##
## Attaching package: 'dplyr'
```

```
## The following objects are masked from 'package:stats':
##
##     filter, lag
```

```
## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```
library(ggplot2)
library(glmnet)
```

```
## Loading required package: Matrix
```

```
## Loading required package: foreach
```

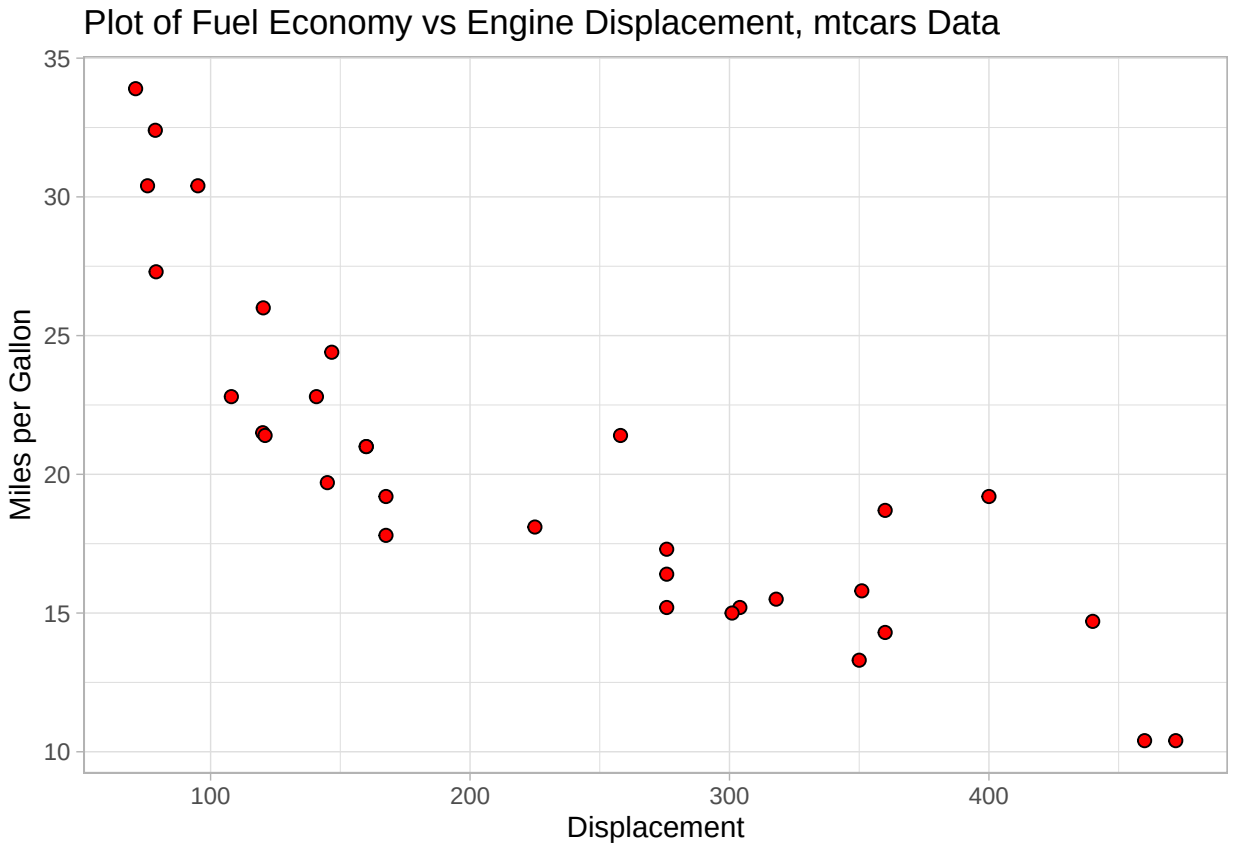
```
## Loaded glmnet 2.0-18
```

```
# Linear regression on the mtcars data.
mtcars <- mtcars %>%
  as_tibble() %>% # Removes rownames and pretty-prints
  select(disp,mpg)
mtcars
```

```
## # A tibble: 32 x 2
##     disp   mpg
##    <dbl> <dbl>
##  1  160   21
##  2  160   21
##  3  108   22.8
##  4  258   21.4
##  5  360   18.7
##  6  225   18.1
##  7  360   14.3
##  8  147.  24.4
##  9  141.  22.8
## 10  168.  19.2
## # ... with 22 more rows
```

```
# First thing to do: plot the data, see what type of relationship might be present.
# Plot
mt_plot <- mtcars %>%
```

```
  ggplot(aes(x=disp,y=mpg)) +
  theme_light() +
  geom_point(colour="black",fill="red",pch=21,size=2) +
  labs(title = "Plot of Fuel Economy vs Engine Displacement, mtcars Data",
       x="Displacement",
       y="Miles per Gallon")

mt_plot
```




```
# Looks curvy. Linear regression?
# Scale the data first so that y and x have mean 0 and variance 1
# This is because the response and covariate are on different scales.
# So coefficient estimates are not immediately interpretable.
# Rescaling the data mitigates this.
mtscaled <- mtcars %>%
  mutate_all(~(.x - mean(.x))/sd(.x))

mtscaled
```

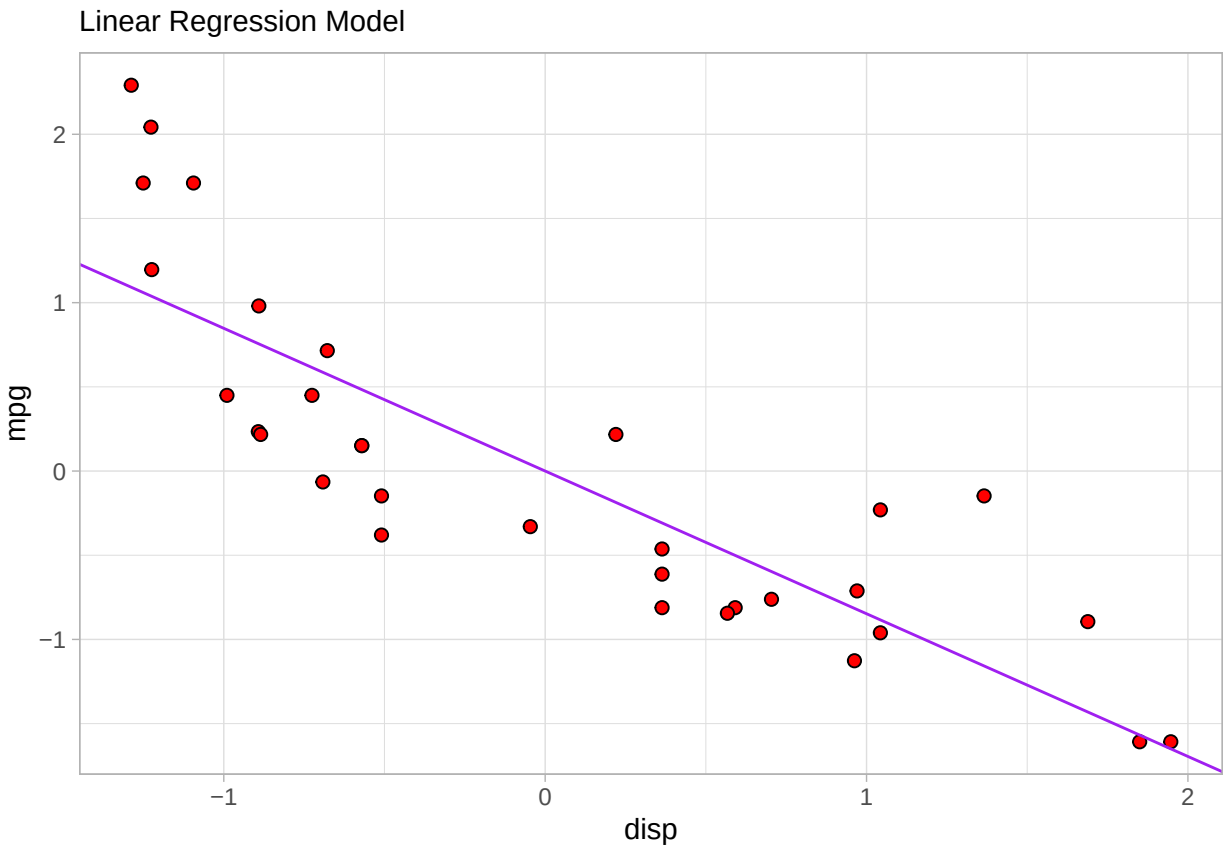
```
## # A tibble: 32 x 2
##       disp    mpg
##      <dbl>  <dbl>
##  1 -0.571   0.151
##  2 -0.571   0.151
##  3 -0.990   0.450
##  4  0.220   0.217
##  5  1.04   -0.231
```

```
##  6 -0.0462 -0.330
##  7  1.04   -0.961
##  8 -0.678   0.715
##  9 -0.726   0.450
## 10 -0.509  -0.148
## # ... with 22 more rows
```

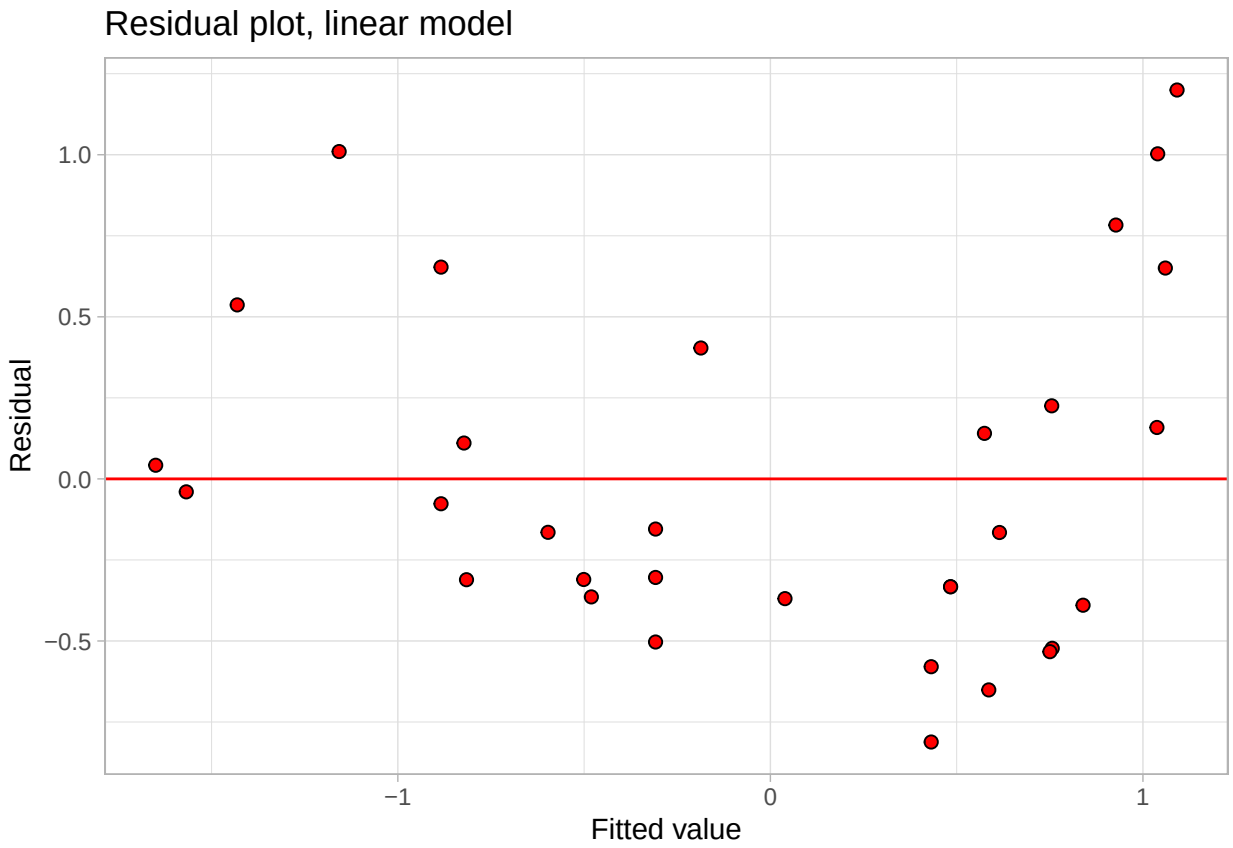
```
mod1 <- lm(mpg~disp,data=mtscaled)
summary(mod1)
```

```
##
## Call:
## lm(formula = mpg ~ disp, data = mtscaled)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -0.8117 -0.3654 -0.1598  0.2700  1.1997
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) -7.037e-17  9.537e-02   0.000        1
## disp        -8.476e-01  9.689e-02  -8.747 9.38e-10 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.5395 on 30 degrees of freedom
## Multiple R-squared:  0.7183, Adjusted R-squared:  0.709
## F-statistic: 76.51 on 1 and 30 DF,  p-value: 9.38e-10
```

```
# We have centred the data, so an intercept of 0 means that a car with average
# displacement (engine size) has average mpg (fuel economy).
# Further, beta(disp) = -.848. A one standard deviation increase in disp is associated with a .847
# standard deviation decrease in fuel economy, on average.
# Predicted values:
mtscaled %>%
  ggplot(aes(x=disp,y=mpg)) +
  theme_light() +
  geom_point(colour="black",fill="red",pch=21,size=2) +
  geom_abline(intercept = coef(mod1)[1],slope=coef(mod1)[2],colour = "purple") +
  labs(subtitle="Linear Regression Model")
```




```
# Definitely looks like a linear relationship is missing some of the structure.
# Can we pick this up in a residual plot?
tibble(x = mod1$fitted.values,
       y = residuals(mod1)) %>%
  ggplot(aes(x=x,y=y)) +
  theme_light() +
  geom_point(colour="black",fill="red",pch=21,size=2) +
  geom_hline(yintercept = 0,colour = "red") +
  labs(title = "Residual plot, linear model",
       x = "Fitted value",
       y = "Residual")
```




```
# The scale of the residuals looks pretty good- all fall
# between -2 and 2, which we expect since the data
# was scaled to have unit variance.
# But the pattern? Doesn't look evenly distributed about
# y = 0. Looks like... a quadratic?

# Model fit- R^2
summary(mod1)$r.squared # Fits pretty well.
```

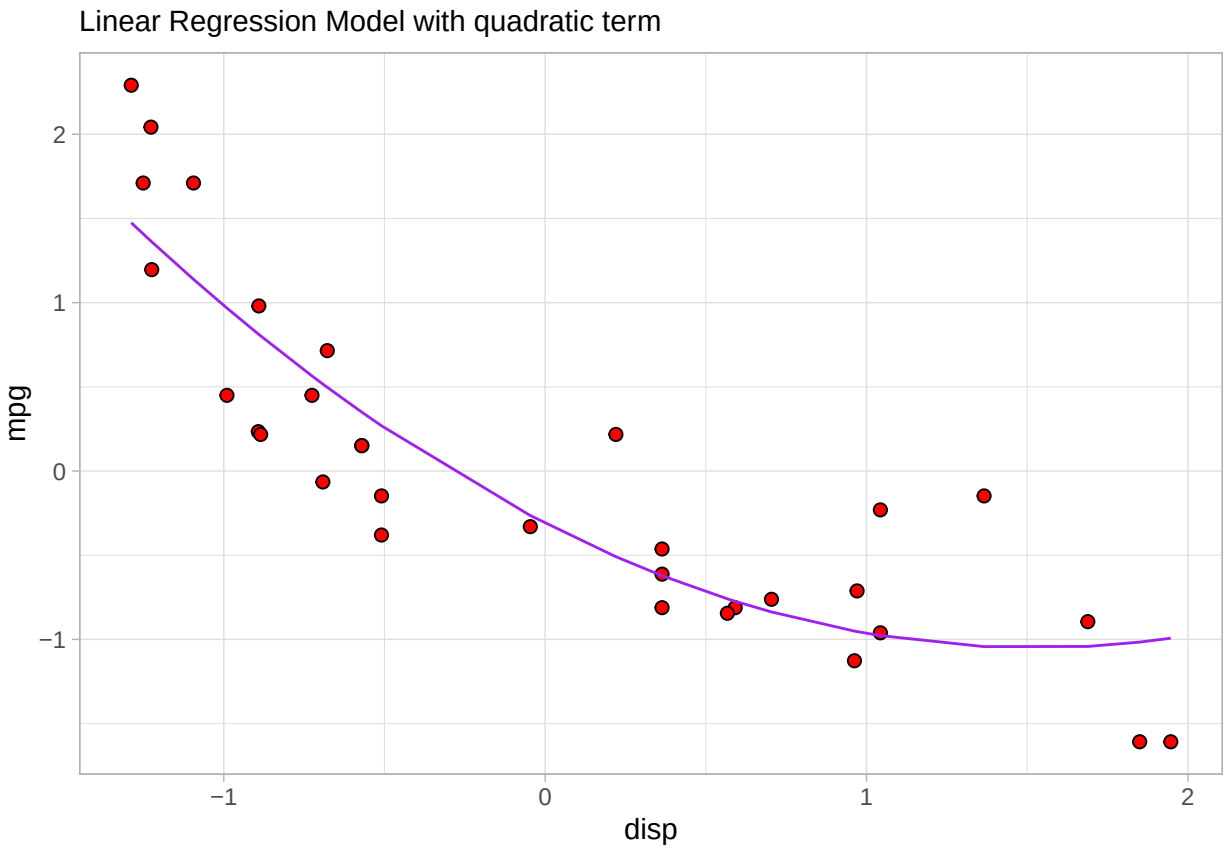
```
## [1] 0.7183433
```

```
# Now try a quadratic...
# The poly() function just creates a polynomial out of
# the input variable with the specified degree.
# By default it creates something called "orthogonal polynomials",
# but you can get regular ones using raw = TRUE.
mod2 <- lm(mpg~poly(disp,degree = 2,raw = TRUE),data=mtscaled)
summary(mod2)
```

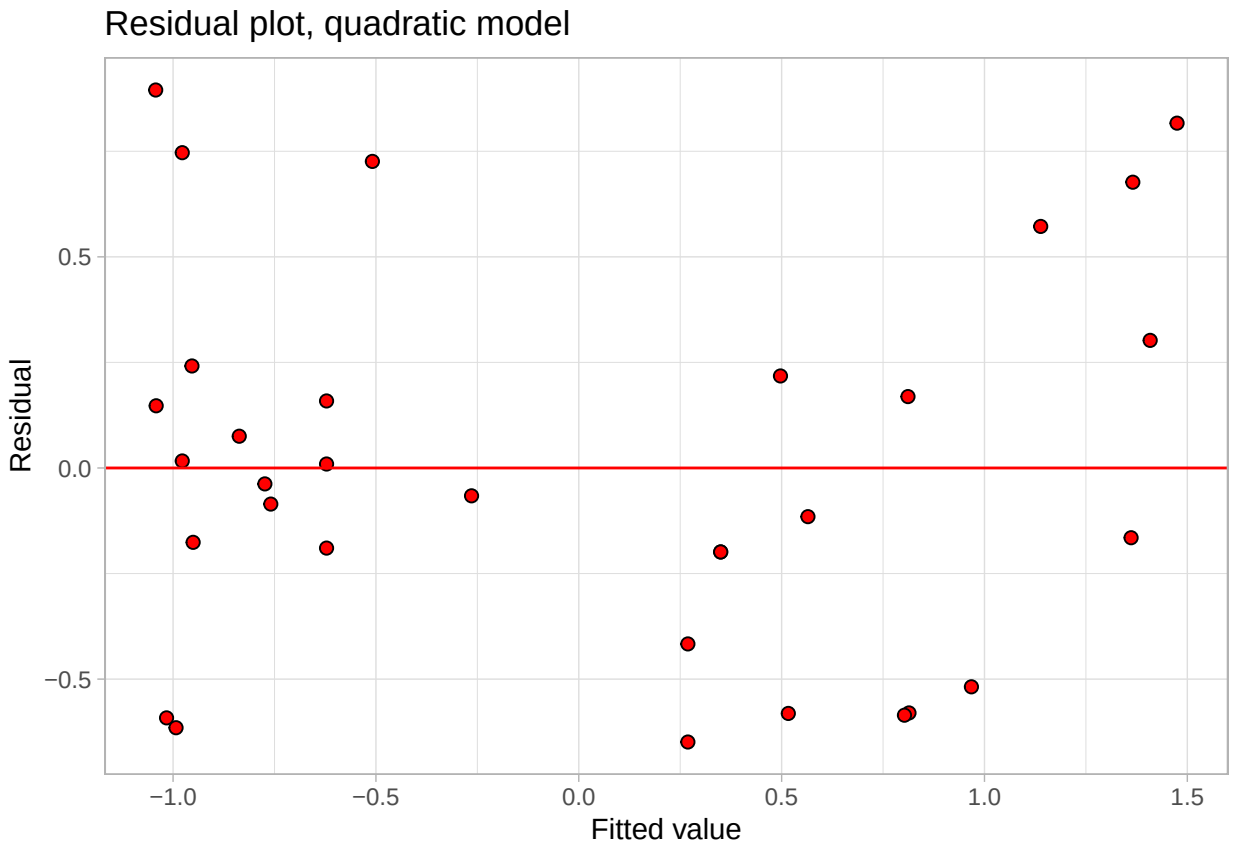
```
##
## Call:
## lm(formula = mpg ~ poly(disp, degree = 2, raw = TRUE), data = mtscaled)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.64895 -0.25334 -0.05184  0.22381  0.89509
##
## Coefficients:
```

```
##                                   Estimate Std. Error t value Pr(>|t|)
## (Intercept)                        -0.30996    0.12710  -2.439   0.0211
## poly(disp, degree = 2, raw = TRUE)1 -0.97360    0.09313 -10.454 2.39e-11
## poly(disp, degree = 2, raw = TRUE)2  0.31996    0.09917   3.226   0.0031
##
## (Intercept)                        *
## poly(disp, degree = 2, raw = TRUE)1 ***
## poly(disp, degree = 2, raw = TRUE)2 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.4707 on 29 degrees of freedom
## Multiple R-squared:  0.7927, Adjusted R-squared:  0.7784
## F-statistic: 55.46 on 2 and 29 DF,  p-value: 1.229e-10
```

```r
# Harder to directly interpret the betas in a polynomial model
# but the standard errors should be not huge compared to the point estimates
# if the fitting procedure was nice and stable.
# (remember: fitting y=beta0 + beta1 x + beta2 x^2) is STILL linear regression
# because it's a linear function of the betas).

mtscaled %>%
  mutate(predvals = predict(mod2)) %>%
  ggplot(aes(x=disp,y=mpg)) +
  theme_light() +
  geom_point(colour="black",fill="red",pch=21,size=2) +
  geom_line(aes(y = predvals),colour = "purple") +
  labs(subtitle="Linear Regression Model with quadratic term")
```




```
# A quadratic gives a much better fit, but it's not that flexible.
# We see it starting to curve back up at the right-most x value
# Would a more flexible (higher degree) polynomial be appropriate?

# Residuals:
tibble(x = mod2$fitted.values,
       y = residuals(mod2)) %>%
  ggplot(aes(x=x,y=y)) +
  theme_light() +
  geom_point(colour="black",fill="red",pch=21,size=2) +
  geom_hline(yintercept = 0,colour = "red") +
  labs(title = "Residual plot, quadratic model",
       x = "Fitted value",
       y = "Residual")
```




```
# I don't see anything that alarming in this plot. It looks good!

# Coefficients
coef(mod2)
```

```
##                          (Intercept) poly(disp, degree = 2, raw = TRUE)1
##                           -0.3099566                           -0.9736036
## poly(disp, degree = 2, raw = TRUE)2
##                            0.3199552
```

```
# Rsquared
summary(mod2)$r.squared # Fits a bit better than linear model.
```

```
## [1] 0.7927323
```

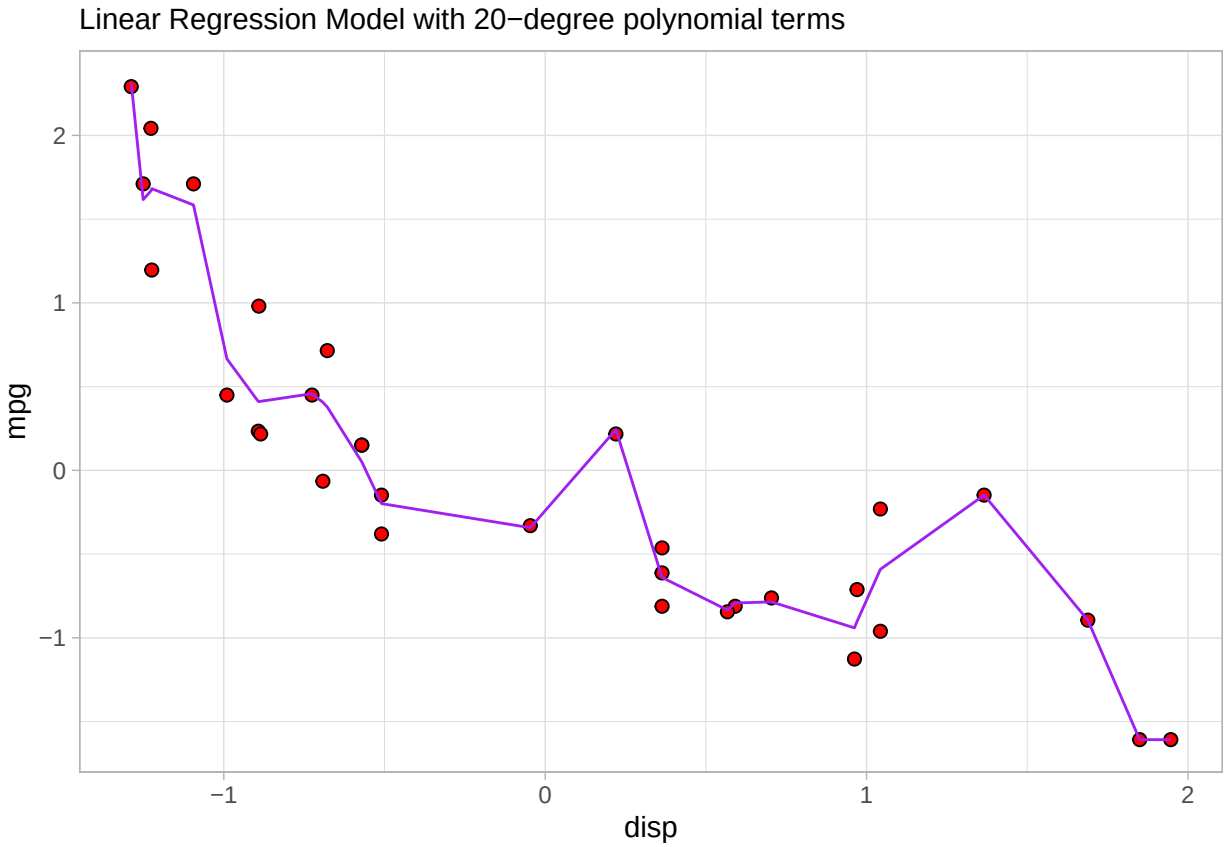
```
# Adding a quadratic term improved the fit a bit.
# Does adding more terms always improve the fit?

mod3 <- lm(mpg~poly(disp,degree = 20,raw = TRUE),data=mtscaled)
summary(mod3)
```

```
##
## Call:
## lm(formula = mpg ~ poly(disp, degree = 20, raw = TRUE), data = mtscaled)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -0.4857 -0.1734  0.0000  0.1001  0.5695
##
```

```
## Coefficients: (2 not defined because of singularities)
##                                       Estimate Std. Error t value
## (Intercept)                          -1.501e-02  3.904e-01  -0.038
## poly(disp, degree = 20, raw = TRUE)1  6.736e+00  4.874e+00   1.382
## poly(disp, degree = 20, raw = TRUE)2 -1.205e+01  2.765e+01  -0.436
## poly(disp, degree = 20, raw = TRUE)3 -9.245e+01  6.158e+01  -1.501
## poly(disp, degree = 20, raw = TRUE)4  6.054e+01  2.687e+02   0.225
## poly(disp, degree = 20, raw = TRUE)5  4.684e+02  4.223e+02   1.109
## poly(disp, degree = 20, raw = TRUE)6 -8.656e+01  1.071e+03  -0.081
## poly(disp, degree = 20, raw = TRUE)7 -1.262e+03  1.472e+03  -0.858
## poly(disp, degree = 20, raw = TRUE)8 -3.787e+01  2.229e+03  -0.017
## poly(disp, degree = 20, raw = TRUE)9  1.971e+03  2.780e+03   0.709
## poly(disp, degree = 20, raw = TRUE)10 2.029e+02  2.627e+03   0.077
## poly(disp, degree = 20, raw = TRUE)11 -1.843e+03  3.034e+03  -0.607
## poly(disp, degree = 20, raw = TRUE)12 -1.888e+02  1.762e+03  -0.107
## poly(disp, degree = 20, raw = TRUE)13  1.030e+03  1.950e+03   0.528
## poly(disp, degree = 20, raw = TRUE)14  7.138e+01  6.252e+02   0.114
## poly(disp, degree = 20, raw = TRUE)15 -3.302e+02  7.104e+02  -0.465
## poly(disp, degree = 20, raw = TRUE)16 -9.542e+00  9.093e+01  -0.105
## poly(disp, degree = 20, raw = TRUE)17  5.452e+01  1.276e+02   0.427
## poly(disp, degree = 20, raw = TRUE)18         NA         NA      NA
## poly(disp, degree = 20, raw = TRUE)19 -3.460e+00  7.245e+00  -0.478
## poly(disp, degree = 20, raw = TRUE)20         NA         NA      NA
##                                      Pr(>|t|)
## (Intercept)                             0.970
## poly(disp, degree = 20, raw = TRUE)1    0.190
## poly(disp, degree = 20, raw = TRUE)2    0.670
## poly(disp, degree = 20, raw = TRUE)3    0.157
## poly(disp, degree = 20, raw = TRUE)4    0.825
## poly(disp, degree = 20, raw = TRUE)5    0.287
## poly(disp, degree = 20, raw = TRUE)6    0.937
## poly(disp, degree = 20, raw = TRUE)7    0.407
## poly(disp, degree = 20, raw = TRUE)8    0.987
## poly(disp, degree = 20, raw = TRUE)9    0.491
## poly(disp, degree = 20, raw = TRUE)10   0.940
## poly(disp, degree = 20, raw = TRUE)11   0.554
## poly(disp, degree = 20, raw = TRUE)12   0.916
## poly(disp, degree = 20, raw = TRUE)13   0.606
## poly(disp, degree = 20, raw = TRUE)14   0.911
## poly(disp, degree = 20, raw = TRUE)15   0.650
## poly(disp, degree = 20, raw = TRUE)16   0.918
## poly(disp, degree = 20, raw = TRUE)17   0.676
## poly(disp, degree = 20, raw = TRUE)18      NA
## poly(disp, degree = 20, raw = TRUE)19   0.641
## poly(disp, degree = 20, raw = TRUE)20      NA
##
## Residual standard error: 0.3544 on 13 degrees of freedom
## Multiple R-squared:  0.9473, Adjusted R-squared:  0.8744
## F-statistic: 12.99 on 18 and 13 DF,  p-value: 1.496e-05
```

```
# Woah. The coefficients (point estimates) are all over the place in magnitude,
# and direction. Their standard errors are huge.
# This is the "instability" we talked about in lecture.
# Whether it's statistical (sampling) error or numerical error, something is wrong here.
```

```
# Rather than plot a linear/quadratic using geom_line(),
# to do predictions for 20 degree polynomial, I use the predict() function
# and then plot its output.
mtscaled %>%
  mutate(predvals = predict(mod3)) %>%
  ggplot(aes(x=disp,y=mpg)) +
  theme_light() +
  geom_point(colour="black",fill="red",pch=21,size=2) +
  geom_line(aes(y = predvals),colour = "purple") +
  labs(subtitle="Linear Regression Model with 20-degree polynomial terms")
```




```
# While this model fits THESE data well, it fits them TOO well.
# Sacrifices generalizability to new datasets.
# STatistically: point estimates have very high variance. Confidence intervals
# would be wide. We're asking too much of this small dataset.
# Coefficients
unname(coef(mod3))
```

```
##  [1] -1.501106e-02  6.736184e+00 -1.204588e+01 -9.244740e+01  6.053962e+01
##  [6]  4.683826e+02 -8.655610e+01 -1.262487e+03 -3.786733e+01  1.971390e+03
## [11]  2.028601e+02 -1.842881e+03 -1.887617e+02  1.029692e+03  7.137544e+01
## [16] -3.301832e+02 -9.542259e+00  5.451536e+01            NA -3.460233e+00
## [21]            NA
```

```
# Rsquared
summary(mod3)$r.squared # YEs, this model does fit THESE DATA very well.
```

```
## [1] 0.9473424
```

```
# The model fits the observed data better. But it gives ridiculous predictions.
# The coefficients are huge. The fitted curve hugs the observed data too closely.
# This model will not generalize well to new data, and the coefficients are highly
# sensitive/unstable (look at their standard errors/p-values)


### Penalized Regression ###

# glmnet() wants X and y to be provided separately (no formula)
# so get the design matrices from each of the above models.
# If this is unfamiliar, print out each of the below X's and see what's inside.
y <- mtscaled$mpg
y
```

```
##  [1]  0.15088482  0.15088482  0.44954345  0.21725341 -0.23073453
##  [6] -0.33028740 -0.96078893  0.71501778  0.44954345 -0.14777380
## [11] -0.38006384 -0.61235388 -0.46302456 -0.81145962 -1.60788262
## [16] -1.60788262 -0.89442035  2.04238943  1.71054652  2.29127162
## [21]  0.23384555 -0.76168319 -0.81145962 -1.12671039 -0.14777380
## [26]  1.19619000  0.98049211  1.71054652 -0.71190675 -0.06481307
## [31] -0.84464392  0.21725341
```

```
X1 <- model.matrix(mod1)
X1
```

```
##    (Intercept)        disp
## 1            1 -0.57061982
## 2            1 -0.57061982
## 3            1 -0.99018209
## 4            1  0.22009369
## 5            1  1.04308123
## 6            1 -0.04616698
## 7            1  1.04308123
## 8            1 -0.67793094
## 9            1 -0.72553512
## 10           1 -0.50929918
## 11           1 -0.50929918
## 12           1  0.36371309
## 13           1  0.36371309
## 14           1  0.36371309
## 15           1  1.94675381
## 16           1  1.84993175
## 17           1  1.68856165
## 18           1 -1.22658929
## 19           1 -1.25079481
## 20           1 -1.28790993
## 21           1 -0.89255318
## 22           1  0.70420401
## 23           1  0.59124494
## 24           1  0.96239618
## 25           1  1.36582144
## 26           1 -1.22416874
## 27           1 -0.89093948
## 28           1 -1.09426581
## 29           1  0.97046468
```

```
## 30           1 -0.69164740
## 31           1  0.56703942
## 32           1 -0.88529152
## attr(,"assign")
## [1] 0 1
```

```
X2 <- model.matrix(mod2)
X2
```

```
##    (Intercept) poly(disp, degree = 2, raw = TRUE)1
## 1            1                          -0.57061982
## 2            1                          -0.57061982
## 3            1                          -0.99018209
## 4            1                           0.22009369
## 5            1                           1.04308123
## 6            1                          -0.04616698
## 7            1                           1.04308123
## 8            1                          -0.67793094
## 9            1                          -0.72553512
## 10           1                          -0.50929918
## 11           1                          -0.50929918
## 12           1                           0.36371309
## 13           1                           0.36371309
## 14           1                           0.36371309
## 15           1                           1.94675381
## 16           1                           1.84993175
## 17           1                           1.68856165
## 18           1                          -1.22658929
## 19           1                          -1.25079481
## 20           1                          -1.28790993
## 21           1                          -0.89255318
## 22           1                           0.70420401
## 23           1                           0.59124494
## 24           1                           0.96239618
## 25           1                           1.36582144
## 26           1                          -1.22416874
## 27           1                          -0.89093948
## 28           1                          -1.09426581
## 29           1                           0.97046468
## 30           1                          -0.69164740
## 31           1                           0.56703942
## 32           1                          -0.88529152
##    poly(disp, degree = 2, raw = TRUE)2
## 1                           0.32560698
## 2                           0.32560698
## 3                           0.98046057
## 4                           0.04844123
## 5                           1.08801845
## 6                           0.00213139
## 7                           1.08801845
## 8                           0.45959036
## 9                           0.52640121
## 10                          0.25938565
## 11                          0.25938565
## 12                          0.13228721
```

```
## 13                          0.13228721
## 14                          0.13228721
## 15                          3.78985041
## 16                          3.42224749
## 17                          2.85124044
## 18                          1.50452130
## 19                          1.56448766
## 20                          1.65871200
## 21                          0.79665117
## 22                          0.49590329
## 23                          0.34957057
## 24                          0.92620640
## 25                          1.86546820
## 26                          1.49858911
## 27                          0.79377315
## 28                          1.19741766
## 29                          0.94180170
## 30                          0.47837612
## 31                          0.32153370
## 32                          0.78374108
## attr(,"assign")
## [1] 0 1 1
```

```
X3 <- model.matrix(mod3)
X3
```

```
##    (Intercept) poly(disp, degree = 20, raw = TRUE)1
## 1            1                          -0.57061982
## 2            1                          -0.57061982
## 3            1                          -0.99018209
## 4            1                           0.22009369
## 5            1                           1.04308123
## 6            1                          -0.04616698
## 7            1                           1.04308123
## 8            1                          -0.67793094
## 9            1                          -0.72553512
## 10           1                          -0.50929918
## 11           1                          -0.50929918
## 12           1                           0.36371309
## 13           1                           0.36371309
## 14           1                           0.36371309
## 15           1                           1.94675381
## 16           1                           1.84993175
## 17           1                           1.68856165
## 18           1                          -1.22658929
## 19           1                          -1.25079481
## 20           1                          -1.28790993
## 21           1                          -0.89255318
## 22           1                           0.70420401
## 23           1                           0.59124494
## 24           1                           0.96239618
## 25           1                           1.36582144
## 26           1                          -1.22416874
## 27           1                          -0.89093948
## 28           1                          -1.09426581
```

```
## 29           1                          0.97046468
## 30           1                         -0.69164740
## 31           1                          0.56703942
## 32           1                         -0.88529152
##    poly(disp, degree = 20, raw = TRUE)2
## 1                            0.32560698
## 2                            0.32560698
## 3                            0.98046057
## 4                            0.04844123
## 5                            1.08801845
## 6                            0.00213139
## 7                            1.08801845
## 8                            0.45959036
## 9                            0.52640121
## 10                           0.25938565
## 11                           0.25938565
## 12                           0.13228721
## 13                           0.13228721
## 14                           0.13228721
## 15                           3.78985041
## 16                           3.42224749
## 17                           2.85124044
## 18                           1.50452130
## 19                           1.56448766
## 20                           1.65871200
## 21                           0.79665117
## 22                           0.49590329
## 23                           0.34957057
## 24                           0.92620640
## 25                           1.86546820
## 26                           1.49858911
## 27                           0.79377315
## 28                           1.19741766
## 29                           0.94180170
## 30                           0.47837612
## 31                           0.32153370
## 32                           0.78374108
##    poly(disp, degree = 20, raw = TRUE)3
## 1                         -1.857978e-01
## 2                         -1.857978e-01
## 3                         -9.708345e-01
## 4                          1.066161e-02
## 5                          1.134892e+00
## 6                         -9.839983e-05
## 7                          1.134892e+00
## 8                         -3.115705e-01
## 9                         -3.819226e-01
## 10                        -1.321049e-01
## 11                        -1.321049e-01
## 12                         4.811459e-02
## 13                         4.811459e-02
## 14                         4.811459e-02
## 15                         7.377906e+00
## 16                         6.330924e+00
```

```
## 17                          4.814495e+00
## 18                         -1.845430e+00
## 19                         -1.956853e+00
## 20                         -2.136272e+00
## 21                         -7.110535e-01
## 22                          3.492171e-01
## 23                          2.066818e-01
## 24                          8.913775e-01
## 25                          2.547896e+00
## 26                         -1.834526e+00
## 27                         -7.072038e-01
## 28                         -1.310293e+00
## 29                          9.139853e-01
## 30                         -3.308676e-01
## 31                          1.823223e-01
## 32                         -6.938393e-01
##    poly(disp, degree = 20, raw = TRUE)4
## 1                           1.060199e-01
## 2                           1.060199e-01
## 3                           9.613029e-01
## 4                           2.346553e-03
## 5                           1.183784e+00
## 6                           4.542823e-06
## 7                           1.183784e+00
## 8                           2.112233e-01
## 9                           2.770982e-01
## 10                          6.728092e-02
## 11                          6.728092e-02
## 12                          1.749991e-02
## 13                          1.749991e-02
## 14                          1.749991e-02
## 15                          1.436297e+01
## 16                          1.171178e+01
## 17                          8.129572e+00
## 18                          2.263584e+00
## 19                          2.447622e+00
## 20                          2.751325e+00
## 21                          6.346531e-01
## 22                          2.459201e-01
## 23                          1.221996e-01
## 24                          8.578583e-01
## 25                          3.479972e+00
## 26                          2.245769e+00
## 27                          6.300758e-01
## 28                          1.433809e+00
## 29                          8.869904e-01
## 30                          2.288437e-01
## 31                          1.033839e-01
## 32                          6.142501e-01
##    poly(disp, degree = 20, raw = TRUE)5
## 1                          -6.049706e-02
## 2                          -6.049706e-02
## 3                          -9.518650e-01
## 4                           5.164616e-04
```

```
## 5                          1.234783e+00
## 6                         -2.097284e-07
## 7                          1.234783e+00
## 8                         -1.431948e-01
## 9                         -2.010445e-01
## 10                        -3.426612e-02
## 11                        -3.426612e-02
## 12                         6.364945e-03
## 13                         6.364945e-03
## 14                         6.364945e-03
## 15                         2.796116e+01
## 16                         2.166599e+01
## 17                         1.372728e+01
## 18                        -2.776488e+00
## 19                        -3.061472e+00
## 20                        -3.543459e+00
## 21                        -5.664616e-01
## 22                         1.731779e-01
## 23                         7.224989e-02
## 24                         8.255995e-01
## 25                         4.753020e+00
## 26                        -2.749201e+00
## 27                        -5.613594e-01
## 28                        -1.568968e+00
## 29                         8.607929e-01
## 30                        -1.582792e-01
## 31                         5.862276e-02
## 32                        -5.437904e-01
##    poly(disp, degree = 20, raw = TRUE)6
## 1                          3.452082e-02
## 2                          3.452082e-02
## 3                          9.425196e-01
## 4                          1.136699e-04
## 5                          1.287979e+00
## 6                          9.682527e-09
## 7                          1.287979e+00
## 8                          9.707619e-02
## 9                          1.458648e-01
## 10                         1.745170e-02
## 11                         1.745170e-02
## 12                         2.315014e-03
## 13                         2.315014e-03
## 14                         2.315014e-03
## 15                         5.443349e+01
## 16                         4.008060e+01
## 17                         2.317936e+01
## 18                         3.405611e+00
## 19                         3.829274e+00
## 20                         4.563657e+00
## 21                         5.055971e-01
## 22                         1.219526e-01
## 23                         4.271738e-02
## 24                         7.945538e-01
## 25                         6.491776e+00
```

```
## 26                          3.365485e+00
## 27                          5.001373e-01
## 28                          1.716868e+00
## 29                          8.353691e-01
## 30                          1.094734e-01
## 31                          3.324142e-02
## 32                          4.814130e-01
##    poly(disp, degree = 20, raw = TRUE)7
## 1                          -1.969826e-02
## 2                          -1.969826e-02
## 3                          -9.332661e-01
## 4                           2.501804e-05
## 5                           1.343467e+00
## 6                          -4.470130e-10
## 7                           1.343467e+00
## 8                          -6.581095e-02
## 9                          -1.058301e-01
## 10                         -8.888139e-03
## 11                         -8.888139e-03
## 12                          8.420008e-04
## 13                          8.420008e-04
## 14                          8.420008e-04
## 15                          1.059686e+02
## 16                          7.414638e+01
## 17                          3.913979e+01
## 18                         -4.177286e+00
## 19                         -4.789636e+00
## 20                         -5.877579e+00
## 21                         -4.512723e-01
## 22                          8.587949e-02
## 23                          2.525643e-02
## 24                          7.646756e-01
## 25                          8.866607e+00
## 26                         -4.119922e+00
## 27                         -4.455920e-01
## 28                         -1.878710e+00
## 29                          8.106962e-01
## 30                         -7.571697e-02
## 31                          1.884919e-02
## 32                         -4.261909e-01
##    poly(disp, degree = 20, raw = TRUE)8
## 1                           1.124022e-02
## 2                           1.124022e-02
## 3                           9.241033e-01
## 4                           5.506312e-06
## 5                           1.401345e+00
## 6                           2.063724e-11
## 7                           1.401345e+00
## 8                           4.461528e-02
## 9                           7.678343e-02
## 10                          4.526722e-03
## 11                          4.526722e-03
## 12                          3.062467e-04
## 13                          3.062467e-04
```

```
## 14                          3.062467e-04
## 15                          2.062948e+02
## 16                          1.371657e+02
## 17                          6.608994e+01
## 18                          5.123814e+00
## 19                          5.990852e+00
## 20                          7.569792e+00
## 21                          4.027846e-01
## 22                          6.047668e-02
## 23                          1.493274e-02
## 24                          7.359209e-01
## 25                          1.211020e+01
## 26                          5.043480e+00
## 27                          3.969955e-01
## 28                          2.055808e+00
## 29                          7.867520e-01
## 30                          5.236945e-02
## 31                          1.068824e-02
## 32                          3.773032e-01
##    poly(disp, degree = 20, raw = TRUE)9
## 1                          -6.413892e-03
## 2                          -6.413892e-03
## 3                          -9.150306e-01
## 4                           1.211905e-06
## 5                           1.461717e+00
## 6                          -9.527590e-13
## 7                           1.461717e+00
## 8                          -3.024608e-02
## 9                          -5.570908e-02
## 10                         -2.305456e-03
## 11                         -2.305456e-03
## 12                          1.113859e-04
## 13                          1.113859e-04
## 14                          1.113859e-04
## 15                          4.016052e+02
## 16                          2.537473e+02
## 17                          1.115969e+02
## 18                         -6.284815e+00
## 19                         -7.493326e+00
## 20                         -9.749210e+00
## 21                         -3.595066e-01
## 22                          4.258792e-02
## 23                          8.828906e-03
## 24                          7.082474e-01
## 25                          1.654037e+01
## 26                         -6.174070e+00
## 27                         -3.536990e-01
## 28                         -2.249601e+00
## 29                          7.635151e-01
## 30                         -3.622119e-02
## 31                          6.060651e-03
## 32                         -3.340233e-01
##    poly(disp, degree = 20, raw = TRUE)10
## 1                           3.659894e-03
```

```
## 2                            3.659894e-03
## 3                            9.060469e-01
## 4                            2.667325e-07
## 5                            1.524689e+00
## 6                            4.398600e-14
## 7                            1.524689e+00
## 8                            2.050475e-02
## 9                            4.041889e-02
## 10                           1.174167e-03
## 11                           1.174167e-03
## 12                           4.051252e-05
## 13                           4.051252e-05
## 14                           4.051252e-05
## 15                           7.818264e+02
## 16                           4.694151e+02
## 17                           1.884383e+02
## 18                           7.708887e+00
## 19                           9.372613e+00
## 20                           1.255610e+01
## 21                           3.208788e-01
## 22                           2.999058e-02
## 23                           5.220046e-03
## 24                           6.816146e-01
## 25                           2.259120e+01
## 26                           7.558104e+00
## 27                           3.151244e-01
## 28                           2.461661e+00
## 29                           7.409644e-01
## 30                           2.505229e-02
## 31                           3.436628e-03
## 32                           2.957080e-01
##    poly(disp, degree = 20, raw = TRUE)11
## 1                           -2.088408e-03
## 2                           -2.088408e-03
## 3                           -8.971514e-01
## 4                            5.870615e-08
## 5                            1.590375e+00
## 6                           -2.030701e-15
## 7                            1.590375e+00
## 8                           -1.390081e-02
## 9                           -2.932532e-02
## 10                          -5.980021e-04
## 11                          -5.980021e-04
## 12                           1.473493e-05
## 13                           1.473493e-05
## 14                           1.473493e-05
## 15                           1.522024e+03
## 16                           8.683859e+02
## 17                           3.181897e+02
## 18                          -9.455639e+00
## 19                          -1.172322e+01
## 20                          -1.617113e+01
## 21                          -2.864014e-01
## 22                           2.111949e-02
```

```
## 23                           3.086326e-03
## 24                           6.559833e-01
## 25                           3.085554e+01
## 26                          -9.252395e+00
## 27                          -2.807568e-01
## 28                          -2.693712e+00
## 29                           7.190798e-01
## 30                          -1.732735e-02
## 31                           1.948703e-03
## 32                          -2.617878e-01
##    poly(disp, degree = 20, raw = TRUE)12
## 1                            1.191687e-03
## 2                            1.191687e-03
## 3                            8.883432e-01
## 4                            1.292085e-08
## 5                            1.658890e+00
## 6                            9.375132e-17
## 7                            1.658890e+00
## 8                            9.423787e-03
## 9                            2.127655e-02
## 10                           3.045620e-04
## 11                           3.045620e-04
## 12                           5.359289e-06
## 13                           5.359289e-06
## 14                           5.359289e-06
## 15                           2.963005e+03
## 16                           1.606455e+03
## 17                           5.372829e+02
## 18                           1.159819e+01
## 19                           1.466334e+01
## 20                           2.082696e+01
## 21                           2.556285e-01
## 22                           1.487243e-02
## 23                           1.824774e-03
## 24                           6.313158e-01
## 25                           4.214316e+01
## 26                           1.132649e+01
## 27                           2.501373e-01
## 28                           2.947637e+00
## 29                           6.978415e-01
## 30                           1.198442e-02
## 31                           1.104992e-03
## 32                           2.317585e-01
##    poly(disp, degree = 20, raw = TRUE)13
## 1                           -6.800002e-04
## 2                           -6.800002e-04
## 3                           -8.796216e-01
## 4                            2.843798e-09
## 5                            1.730357e+00
## 6                           -4.328215e-18
## 7                            1.730357e+00
## 8                           -6.388677e-03
## 9                           -1.543689e-02
## 10                          -1.551132e-04
```

```
## 11                          -1.551132e-04
## 12                           1.949243e-06
## 13                           1.949243e-06
## 14                           1.949243e-06
## 15                           5.768242e+03
## 16                           2.971832e+03
## 17                           9.072354e+02
## 18                          -1.422621e+01
## 19                          -1.834083e+01
## 20                          -2.682325e+01
## 21                          -2.281620e-01
## 22                           1.047322e-02
## 23                           1.078889e-03
## 24                           6.075759e-01
## 25                           5.756003e+01
## 26                          -1.386554e+01
## 27                          -2.228572e-01
## 28                          -3.225498e+00
## 29                           6.772306e-01
## 30                          -8.288992e-03
## 31                           6.265739e-04
## 32                          -2.051738e-01
##    poly(disp, degree = 20, raw = TRUE)14
## 1                            3.880216e-04
## 2                            3.880216e-04
## 3                            8.709855e-01
## 4                            6.259021e-10
## 5                            1.804903e+00
## 6                            1.998206e-19
## 7                            1.804903e+00
## 8                            4.331081e-03
## 9                            1.120000e-02
## 10                           7.899901e-05
## 11                           7.899901e-05
## 12                           7.089653e-07
## 13                           7.089653e-07
## 14                           7.089653e-07
## 15                           1.122935e+04
## 16                           5.497686e+03
## 17                           1.531923e+03
## 18                           1.744972e+01
## 19                           2.294061e+01
## 20                           3.454593e+01
## 21                           2.036467e-01
## 22                           7.375287e-03
## 23                           6.378875e-04
## 24                           5.847287e-01
## 25                           7.861673e+01
## 26                           1.697376e+01
## 27                           1.985523e-01
## 28                           3.529552e+00
## 29                           6.572283e-01
## 30                           5.733060e-03
## 31                           3.552921e-04
```

```
## 32                          1.816387e-01
##    poly(disp, degree = 20, raw = TRUE)15
## 1                          -2.214128e-04
## 2                          -2.214128e-04
## 3                          -8.624343e-01
## 4                           1.377571e-10
## 5                           1.882660e+00
## 6                          -9.225115e-21
## 7                           1.882660e+00
## 8                          -2.936174e-03
## 9                          -8.125996e-03
## 10                         -4.023413e-05
## 11                         -4.023413e-05
## 12                          2.578600e-07
## 13                          2.578600e-07
## 14                          2.578600e-07
## 15                          2.186077e+04
## 16                          1.017034e+04
## 17                          2.586746e+03
## 18                         -2.140364e+01
## 19                         -2.869400e+01
## 20                         -4.449205e+01
## 21                         -1.817655e-01
## 22                          5.193706e-03
## 23                          3.771477e-04
## 24                          5.627407e-01
## 25                          1.073764e+02
## 26                         -2.077874e+01
## 27                         -1.768980e-01
## 28                         -3.862268e+00
## 29                          6.378169e-01
## 30                         -3.965256e-03
## 31                          2.014646e-04
## 32                         -1.608032e-01
##    poly(disp, degree = 20, raw = TRUE)16
## 1                           1.263425e-04
## 2                           1.263425e-04
## 3                           8.539670e-01
## 4                           3.031947e-11
## 5                           1.963768e+00
## 6                           4.258957e-22
## 7                           1.963768e+00
## 8                           1.990523e-03
## 9                           5.895695e-03
## 10                          2.049121e-05
## 11                          2.049121e-05
## 12                          9.378705e-08
## 13                          9.378705e-08
## 14                          9.378705e-08
## 15                          4.255754e+04
## 16                          1.881444e+04
## 17                          4.367880e+03
## 18                          2.625347e+01
## 19                          3.589030e+01
```

```
## 20                          5.730175e+01
## 21                          1.622354e-01
## 22                          3.657429e-03
## 23                          2.229867e-04
## 24                          5.415795e-01
## 25                          1.466570e+02
## 26                          2.543669e+01
## 27                          1.576055e-01
## 28                          4.226348e+00
## 29                          6.189788e-01
## 30                          2.742559e-03
## 31                          1.142384e-04
## 32                          1.423577e-01
##    poly(disp, degree = 20, raw = TRUE)17
## 1                          -7.209356e-05
## 2                          -7.209356e-05
## 3                          -8.455828e-01
## 4                           6.673124e-12
## 5                           2.048369e+00
## 6                          -1.966232e-23
## 7                           2.048369e+00
## 8                          -1.349437e-03
## 9                          -4.277534e-03
## 10                         -1.043616e-05
## 11                         -1.043616e-05
## 12                          3.411158e-08
## 13                          3.411158e-08
## 14                          3.411158e-08
## 15                          8.284906e+04
## 16                          3.480543e+04
## 17                          7.375435e+03
## 18                         -3.220223e+01
## 19                         -4.489141e+01
## 20                         -7.379949e+01
## 21                         -1.448037e-01
## 22                          2.575576e-03
## 23                          1.318397e-04
## 24                          5.212140e-01
## 25                          2.003073e+02
## 26                         -3.113880e+01
## 27                         -1.404169e-01
## 28                         -4.624748e+00
## 29                          6.006970e-01
## 30                         -1.896884e-03
## 31                          6.477766e-05
## 32                         -1.260280e-01
##    poly(disp, degree = 20, raw = TRUE)18
## 1                           4.113801e-05
## 2                           4.113801e-05
## 3                           8.372809e-01
## 4                           1.468713e-12
## 5                           2.136615e+00
## 6                           9.077497e-25
## 7                           2.136615e+00
```

```
## 8                            9.148253e-04
## 9                            3.103501e-03
## 10                           5.315126e-06
## 11                           5.315126e-06
## 12                           1.240683e-08
## 13                           1.240683e-08
## 14                           1.240683e-08
## 15                           1.612867e+05
## 16                           6.438767e+04
## 17                           1.245388e+04
## 18                           3.949891e+01
## 19                           5.614994e+01
## 20                           9.504710e+01
## 21                           1.292450e-01
## 22                           1.813731e-03
## 23                           7.794958e-05
## 24                           5.016144e-01
## 25                           2.735840e+02
## 26                           3.811915e+01
## 27                           1.251030e-01
## 28                           5.060704e+00
## 29                           5.829553e-01
## 30                           1.311975e-03
## 31                           3.673149e-05
## 32                           1.115716e-01
##    poly(disp, degree = 20, raw = TRUE)19
## 1                           -2.347417e-05
## 2                           -2.347417e-05
## 3                           -8.290606e-01
## 4                            3.232544e-13
## 5                            2.228663e+00
## 6                           -4.190806e-26
## 7                            2.228663e+00
## 8                           -6.201884e-04
## 9                           -2.251699e-03
## 10                          -2.706989e-06
## 11                          -2.706989e-06
## 12                           4.512525e-09
## 13                           4.512525e-09
## 14                           4.512525e-09
## 15                           3.139855e+05
## 16                           1.191128e+05
## 17                           2.102914e+04
## 18                          -4.844893e+01
## 19                          -7.023205e+01
## 20                          -1.224121e+02
## 21                          -1.153581e-01
## 22                           1.277237e-03
## 23                           4.608730e-05
## 24                           4.827518e-01
## 25                           3.736669e+02
## 26                          -4.666427e+01
## 27                          -1.114592e-01
## 28                          -5.537755e+00
```

```
## 29                          5.657375e-01
## 30                         -9.074239e-04
## 31                          2.082820e-05
## 32                         -9.877336e-02
##    poly(disp, degree = 20, raw = TRUE)20
## 1                           1.339482e-05
## 2                           1.339482e-05
## 3                           8.209210e-01
## 4                           7.114625e-14
## 5                           2.324677e+00
## 6                           1.934769e-27
## 7                           2.324677e+00
## 8                           4.204449e-04
## 9                           1.633687e-03
## 10                          1.378667e-06
## 11                          1.378667e-06
## 12                          1.641264e-09
## 13                          1.641264e-09
## 14                          1.641264e-09
## 15                          6.112526e+05
## 16                          2.203505e+05
## 17                          3.550900e+04
## 18                          5.942694e+01
## 19                          8.784588e+01
## 20                          1.576558e+02
## 21                          1.029632e-01
## 22                          8.994352e-04
## 23                          2.724888e-05
## 24                          4.645985e-01
## 25                          5.103622e+02
## 26                          5.712494e+01
## 27                          9.930338e-02
## 28                          6.059776e+00
## 29                          5.490282e-01
## 30                          6.276173e-04
## 31                          1.181041e-05
## 32                          8.744321e-02
## attr(,"assign")
##  [1] 0 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
```

```
# The first model fit well so the answer here shouldn't be too different
# glmnet() picks a bunch of lambda values and tries them all
# We don't really have a good way of choosing lambda... yet.
# We will tackle this in later lectures.
penalized_mod1 <- glmnet(x = X1,y = y,alpha = 0,nlambda = 100)
penalized_mod1
```

```
##
## Call:  glmnet(x = X1, y = y, alpha = 0, nlambda = 100)
##
##        Df      %Dev    Lambda
##   [1,]  1 1.451e-36 834.20000
##   [2,]  1 1.857e-03 760.10000
##   [3,]  1 2.037e-03 692.60000
##   [4,]  1 2.236e-03 631.00000
```

```
##   [5,]  1 2.453e-03 575.00000
##   [6,]  1 2.691e-03 523.90000
##   [7,]  1 2.953e-03 477.40000
##   [8,]  1 3.240e-03 435.00000
##   [9,]  1 3.555e-03 396.30000
##  [10,]  1 3.900e-03 361.10000
##  [11,]  1 4.278e-03 329.00000
##  [12,]  1 4.694e-03 299.80000
##  [13,]  1 5.149e-03 273.20000
##  [14,]  1 5.648e-03 248.90000
##  [15,]  1 6.195e-03 226.80000
##  [16,]  1 6.795e-03 206.60000
##  [17,]  1 7.452e-03 188.30000
##  [18,]  1 8.172e-03 171.60000
##  [19,]  1 8.962e-03 156.30000
##  [20,]  1 9.826e-03 142.40000
##  [21,]  1 1.077e-02 129.80000
##  [22,]  1 1.181e-02 118.20000
##  [23,]  1 1.295e-02 107.70000
##  [24,]  1 1.419e-02  98.17000
##  [25,]  1 1.555e-02  89.45000
##  [26,]  1 1.704e-02  81.50000
##  [27,]  1 1.867e-02  74.26000
##  [28,]  1 2.045e-02  67.66000
##  [29,]  1 2.240e-02  61.65000
##  [30,]  1 2.453e-02  56.18000
##  [31,]  1 2.685e-02  51.19000
##  [32,]  1 2.939e-02  46.64000
##  [33,]  1 3.215e-02  42.50000
##  [34,]  1 3.517e-02  38.72000
##  [35,]  1 3.846e-02  35.28000
##  [36,]  1 4.205e-02  32.15000
##  [37,]  1 4.595e-02  29.29000
##  [38,]  1 5.019e-02  26.69000
##  [39,]  1 5.480e-02  24.32000
##  [40,]  1 5.981e-02  22.16000
##  [41,]  1 6.523e-02  20.19000
##  [42,]  1 7.111e-02  18.40000
##  [43,]  1 7.748e-02  16.76000
##  [44,]  1 8.435e-02  15.27000
##  [45,]  1 9.177e-02  13.92000
##  [46,]  1 9.976e-02  12.68000
##  [47,]  1 1.084e-01  11.55000
##  [48,]  1 1.176e-01  10.53000
##  [49,]  1 1.275e-01   9.59100
##  [50,]  1 1.381e-01   8.73900
##  [51,]  1 1.494e-01   7.96300
##  [52,]  1 1.614e-01   7.25500
##  [53,]  1 1.741e-01   6.61100
##  [54,]  1 1.876e-01   6.02400
##  [55,]  1 2.019e-01   5.48900
##  [56,]  1 2.168e-01   5.00100
##  [57,]  1 2.325e-01   4.55700
##  [58,]  1 2.489e-01   4.15200
```

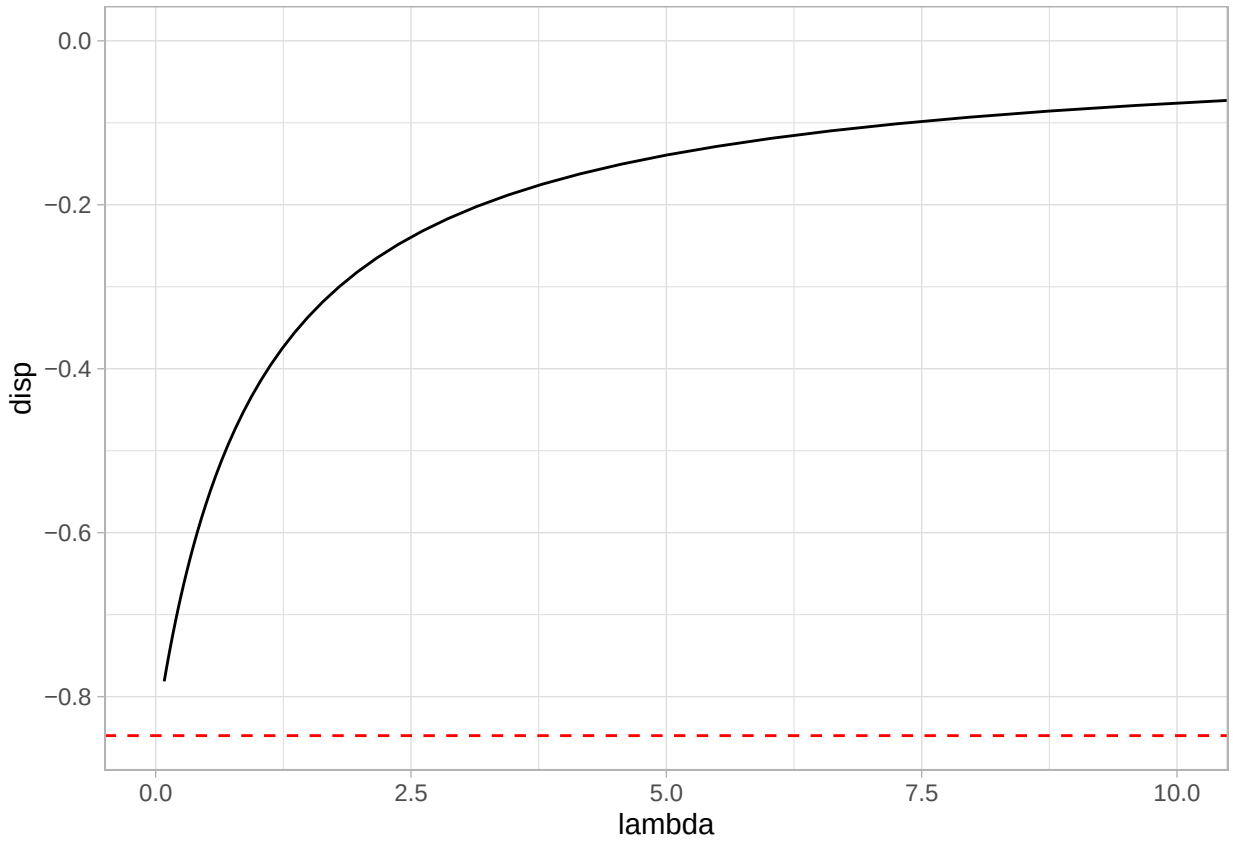
```
##  [59,]  1 2.660e-01   3.78300
##  [60,]  1 2.837e-01   3.44700
##  [61,]  1 3.019e-01   3.14100
##  [62,]  1 3.206e-01   2.86200
##  [63,]  1 3.398e-01   2.60700
##  [64,]  1 3.592e-01   2.37600
##  [65,]  1 3.789e-01   2.16500
##  [66,]  1 3.987e-01   1.97200
##  [67,]  1 4.184e-01   1.79700
##  [68,]  1 4.381e-01   1.63800
##  [69,]  1 4.575e-01   1.49200
##  [70,]  1 4.766e-01   1.36000
##  [71,]  1 4.953e-01   1.23900
##  [72,]  1 5.134e-01   1.12900
##  [73,]  1 5.308e-01   1.02800
##  [74,]  1 5.475e-01   0.93710
##  [75,]  1 5.633e-01   0.85380
##  [76,]  1 5.783e-01   0.77800
##  [77,]  1 5.924e-01   0.70890
##  [78,]  1 6.056e-01   0.64590
##  [79,]  1 6.178e-01   0.58850
##  [80,]  1 6.290e-01   0.53620
##  [81,]  1 6.393e-01   0.48860
##  [82,]  1 6.487e-01   0.44520
##  [83,]  1 6.572e-01   0.40560
##  [84,]  1 6.648e-01   0.36960
##  [85,]  1 6.717e-01   0.33680
##  [86,]  1 6.778e-01   0.30690
##  [87,]  1 6.832e-01   0.27960
##  [88,]  1 6.880e-01   0.25480
##  [89,]  1 6.922e-01   0.23210
##  [90,]  1 6.959e-01   0.21150
##  [91,]  1 6.991e-01   0.19270
##  [92,]  1 7.019e-01   0.17560
##  [93,]  1 7.043e-01   0.16000
##  [94,]  1 7.064e-01   0.14580
##  [95,]  1 7.082e-01   0.13280
##  [96,]  1 7.097e-01   0.12100
##  [97,]  1 7.111e-01   0.11030
##  [98,]  1 7.122e-01   0.10050
##  [99,]  1 7.131e-01   0.09155
## [100,]  1 7.140e-01   0.08342
```

```
coef_pen_mod1 <- tibble(
  lambda = penalized_mod1$lambda,
  intercept = coef(penalized_mod1)[1, ],
  disp = coef(penalized_mod1)[3, ]
)
coef_pen_mod1 # Gives estimated coefficients for each value of lambda
```

```
## # A tibble: 100 x 3
##    lambda intercept      disp
##     <dbl>     <dbl>     <dbl>
##  1   834.  4.60e-17 -8.56e-37
##  2   760.  4.59e-17 -1.10e- 3
```

```
##  3   693.  4.59e-17 -1.20e- 3
##  4   631.  4.59e-17 -1.32e- 3
##  5   575.  4.58e-17 -1.45e- 3
##  6   524.  4.58e-17 -1.59e- 3
##  7   477.  4.58e-17 -1.74e- 3
##  8   435.  4.58e-17 -1.91e- 3
##  9   396.  4.58e-17 -2.10e- 3
## 10   361.  4.58e-17 -2.30e- 3
## # ... with 90 more rows
```

```r
# Higher lambda ==> stronger penalty ==> smaller estimates.
View(coef_pen_mod1)
# As lambda gets smaller, the estimated coefficient approaches
# what it was in the unpenalized model.

# Plot the disp coefficient as a function of lambda and compare to
# our unpenalized model
coef_pen_mod1 %>%
  ggplot(aes(x = lambda,y = disp)) +
  theme_light() +
  geom_line() +
  geom_hline(yintercept = coef(mod1)[2],linetype = "dashed",colour = "red") +
  coord_cartesian(xlim = c(0,10))
```

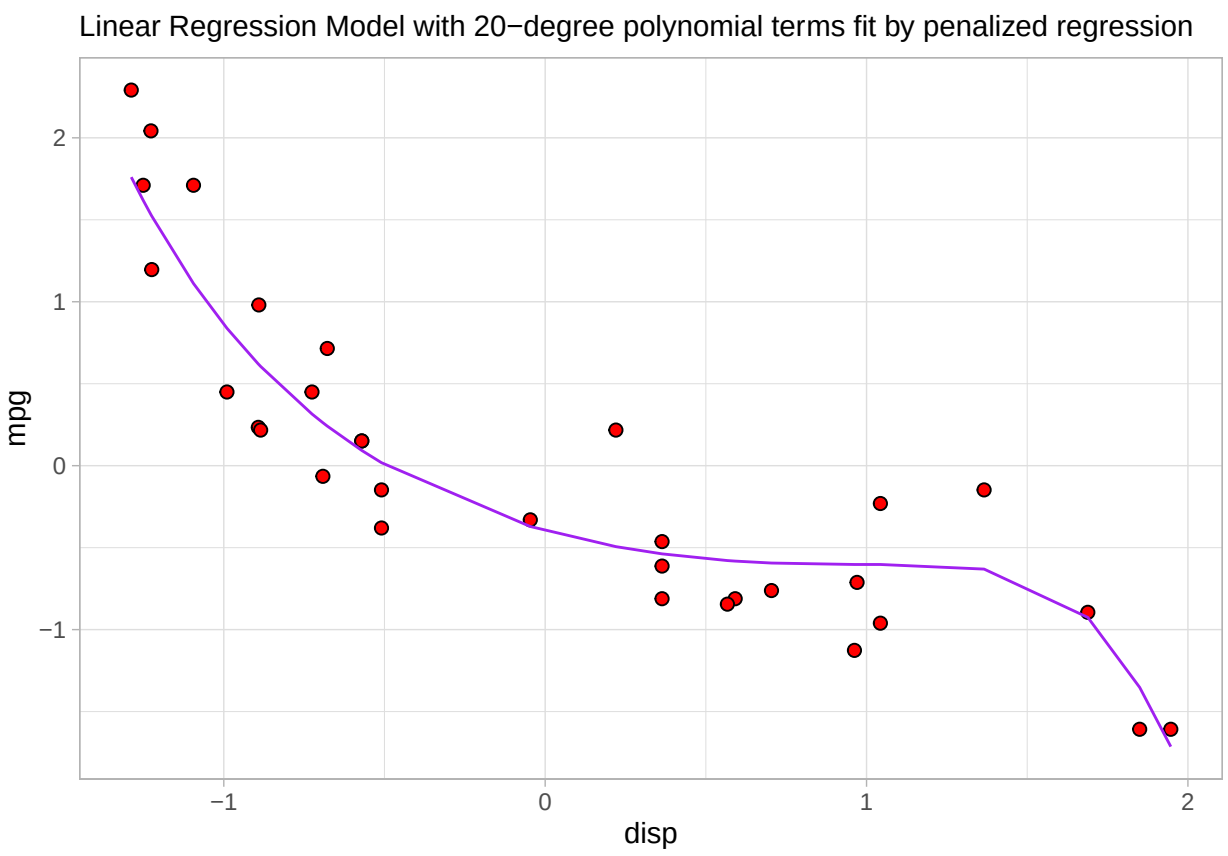



```r
# Low lambda ==> limited penalization ==> answer is close to unpenalized
# High lambda ==> a lot of penalization ==> answer doesn't want to move far from zero

# Try the quadratic one yourself.
```

```
# Try the big one:

penalized_mod3 <- glmnet(x = X3,y = y,alpha = 0,nlambda = 100)
coef_pen_mod3 <- tibble(lambda = penalized_mod3$lambda) %>%
  bind_cols(as_tibble(as.matrix(t(coef(penalized_mod3)))))
```

```
## Warning: `as_tibble.matrix()` requires a matrix with column names or a `.name_repair` argument. Using
## This warning is displayed once per session.
```

```
options(scipen = 999) # Turn off scientific notation
View(coef_pen_mod3) # Opens in a spreadsheet

# Pick any lambda- I chose the smallest one. You can try others!
whichlambda <- penalized_mod3$lambda[length(penalized_mod3$lambda)]

# Plot predictions
mtscaled %>%
  mutate(predvals = predict(penalized_mod3,newx = X3,s = whichlambda)) %>%
  ggplot(aes(x=disp,y=mpg)) +
  theme_light() +
  geom_point(colour="black",fill="red",pch=21,size=2) +
  geom_line(aes(y = predvals),colour = "purple") +
  labs(subtitle="Linear Regression Model with 20-degree polynomial terms fit by penalized regression")
```




```
# Looks a lot like the quadratic, except it's fine at the
# right end point.
# Very little effort on our part to build this very
```

```
# sophisticated model.
# The betas here should have higher variance than if we had
# just fit a linear/quadratic model. We pay a price for
# generality. However, they have MUCH lower variance than
# the original 20-degree polynomial we tried.
# "No such thing as a free lunch". We pay a small price for
# being able to let the data tell us what model to use.
# I think it's worth it...

# Advanced: plot it for ALL the lambda values together! Woah!
# Requires the purrr package
# THIS WILL NOT BE TESTED
library(purrr)
```

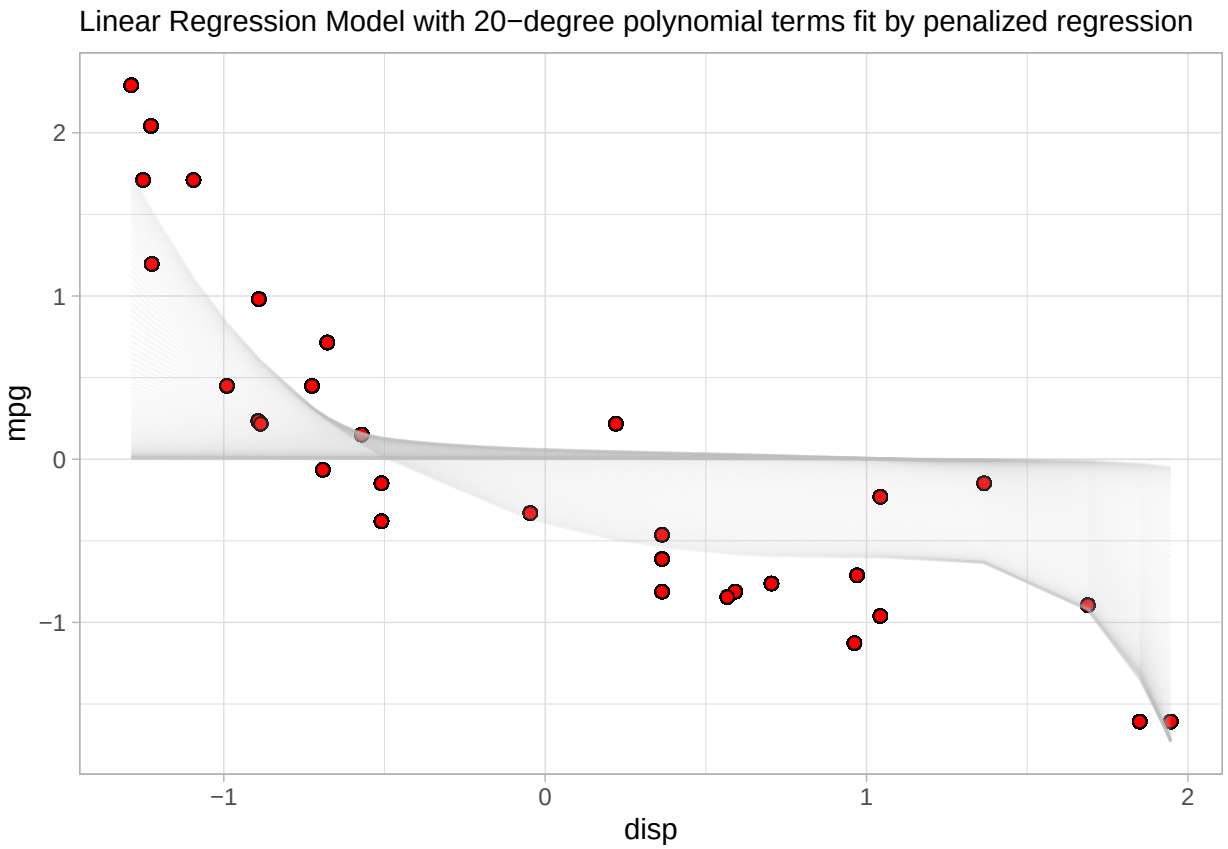
```
##
## Attaching package: 'purrr'
```

```
## The following objects are masked from 'package:foreach':
##
##     accumulate, when
```

```
predframe <- purrr::map(penalized_mod3$lambda,
                        ~mtscaled %>%
                          mutate(predvals = predict(penalized_mod3,newx = X3,s = .x)[,1],
                                 lambda = .x)) %>%
  purrr::reduce(rbind)

predframe
```

```
## # A tibble: 3,200 x 4
##       disp    mpg predvals lambda
##      <dbl>  <dbl>    <dbl>  <dbl>
##  1 -0.571   0.151 4.60e-17   834.
##  2 -0.571   0.151 4.60e-17   834.
##  3 -0.990   0.450 4.60e-17   834.
##  4  0.220   0.217 4.60e-17   834.
##  5  1.04   -0.231 4.60e-17   834.
##  6 -0.0462 -0.330 4.60e-17   834.
##  7  1.04   -0.961 4.60e-17   834.
##  8 -0.678   0.715 4.60e-17   834.
##  9 -0.726   0.450 4.60e-17   834.
## 10 -0.509  -0.148 4.60e-17   834.
## # ... with 3,190 more rows
```

```
predframe %>%
  ggplot(aes(x=disp,y=mpg,group = lambda)) +
  theme_light() +
  geom_point(colour="black",fill="red",pch=21,size=2) +
  geom_line(aes(y = predvals),colour = "grey",alpha = .1) +
  labs(subtitle="Linear Regression Model with 20-degree polynomial terms fit by penalized regression")
```




```
# That's a comparison of all the curves fit using different
# values of lambda.
```