Kentucky Derby Winning Times

CDT I. M. Smrt

```
# Load packages
library(tidyverse)
library(ggResidpanel)

### Data management
ky.dat<-read.table("https://raw.githubusercontent.com/jkstarling/MA376/main/KYDerby18.txt",
                   header=T,
                   stringsAsFactors = TRUE)

# setwd("C:/Users/james.starling/OneDrive - West Point/Teaching/MA376/JimsLessons/Block III/LSN20/")
# ky.dat <- read.table("KYDerby18.txt", header = TRUE, stringsAsFactors = TRUE)
```

Background Recall that the general form for a linear regression model is:

$$y_i = \beta_0 + \beta_1 x_{1,i} + \beta_2 x_{2,i} + \cdots + \beta_n x_{n,i} + \epsilon_i$$

We can think about this model as having two components, a linear predictor $\beta_0 + \beta_1 x_{1,i} + \beta_2 x_{2,i} + \cdots + \beta_n x_{n,i}$ and a random mechanism that perturbs us from the plane, ϵ_i.

Quadratic Models However, sometimes, we might believe that the relationship between our explanatory and response variable is not linear. This may be for a few different reasons, perhaps the best reason is that we have some previous knowledge that suggests they have a nonlinear relationship. Therefore it might make sense to talk about a general form for statistical models.

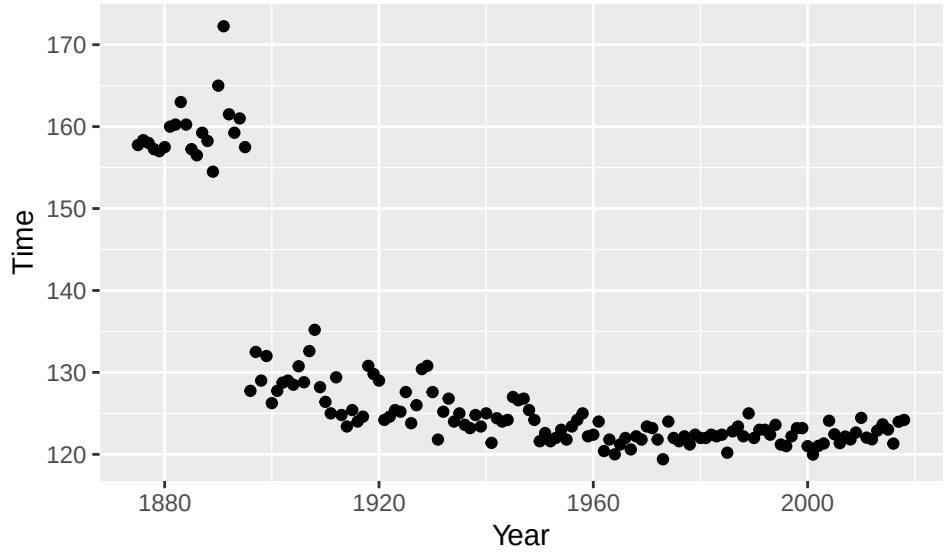
The simplest function outside of a linear relationship is if we assume a quadratic model

$$y_i = \beta_0 + \beta_1 x_{1,i} + \beta_2 x_{1,i}^2$$

This is, what our text calls, a polynomial statistical model. Fitting the model can be achieved in the exact same way as a linear regression model.

(1) Create a scatterplot of the winning times vs. the years. Describe any patterns you see. Are there any unusual observations? How do you suggest proceeding with the analysis of how the performance of the horses has changed over the years?To see this, let's consider the Kentucky Derby data. For question 1 we create the scatter plot of time vs years.

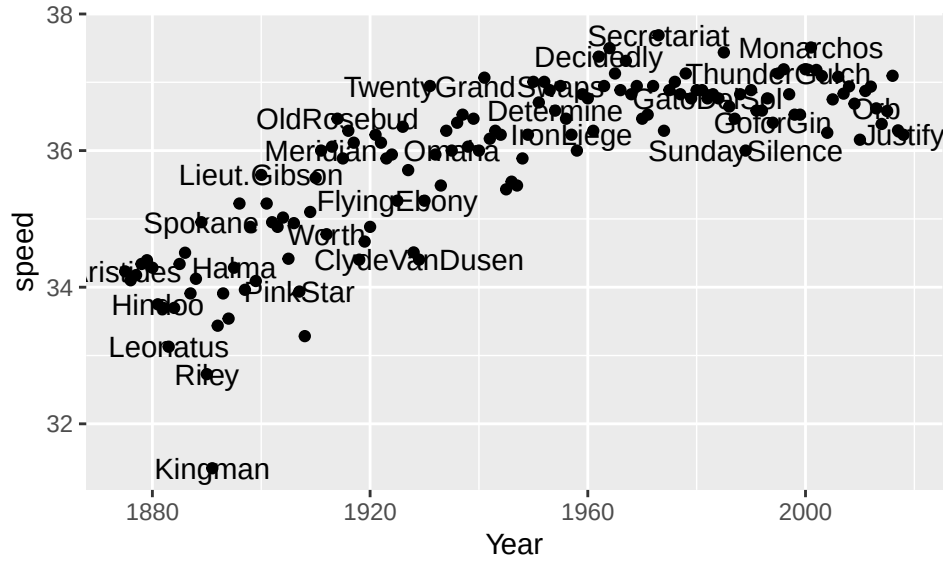
```
ky.dat %>% ggplot(aes(x=Year,y=Time)) + geom_point()
```



(2) Looks kinda weird... But as it turns out, the distance changed in 1896, so we're comparing apples to oranges. The Derby was first run at a distance of 1.5 miles but in 1896 the distance changed to its current 1.25 miles (2 kilometers).

Produce a scatterplot of the speeds of the winning horse (in miles per hour) vs. year. What is the overall pattern in these speeds over the years? Why does this pattern make sense in this context? Are there any unusual observations now? (If so, identify by name.) Does it make sense to fit a linear model to these data?

```
ky.dat %>% ggplot(aes(x=Year,y=speed)) +
  geom_point() +
  geom_text(aes(label = winner), check_overlap = T)
```



(3) What type of model might we want to fit? Why?

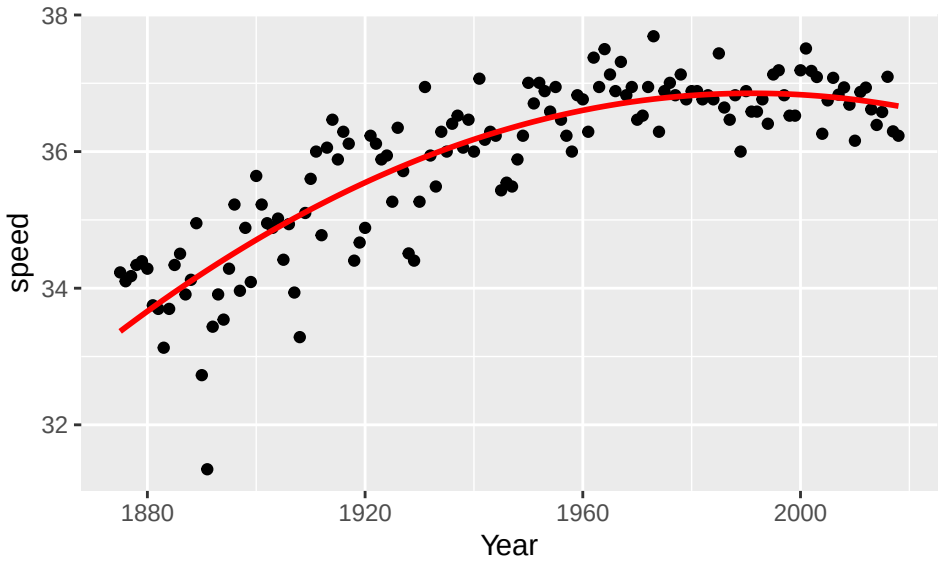
(4) Fit a quadratic model to predict speed from year and year^2.

```
ky.dat <- ky.dat %>% mutate(Year.sq=Year^2)
poly.lm<-lm(speed ~ Year + Year.sq,data=ky.dat)
summary(poly.lm)
```

```
##
## Call:
## lm(formula = speed ~ Year + Year.sq, data = ky.dat)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -2.9131 -0.3167  0.0314  0.3986  1.1499
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept) -9.888e+02  1.230e+02  -8.036 3.37e-13 ***
## Year         1.030e+00  1.265e-01   8.146 1.81e-13 ***
## Year.sq     -2.587e-04  3.249e-05  -7.964 5.02e-13 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.6024 on 141 degrees of freedom
## Multiple R-squared:  0.7522, Adjusted R-squared:  0.7487
## F-statistic:   214 on 2 and 141 DF,  p-value: < 2.2e-16
```

(5) To check the fit plot the model on your scatterplot. You can add the model by adding `geom_line()` to your plot. The `data` for this geom is the model you made above and `.fitted` will provide the model values for your aesthetics.

```
ky.dat %>% ggplot(aes(x=Year,y=speed)) + geom_point() +
  geom_line(aes(x=Year,y=.fitted),data=poly.lm,lwd=1,color="red")
```

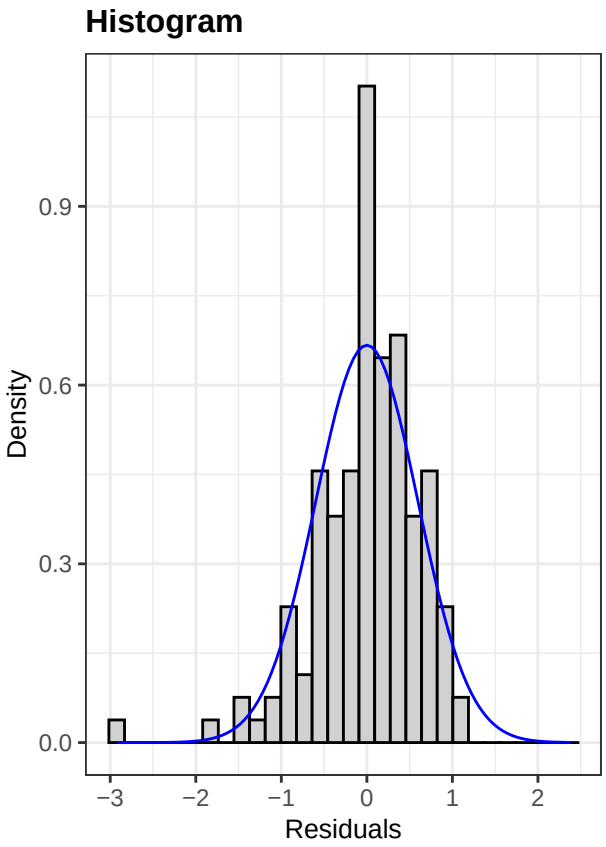


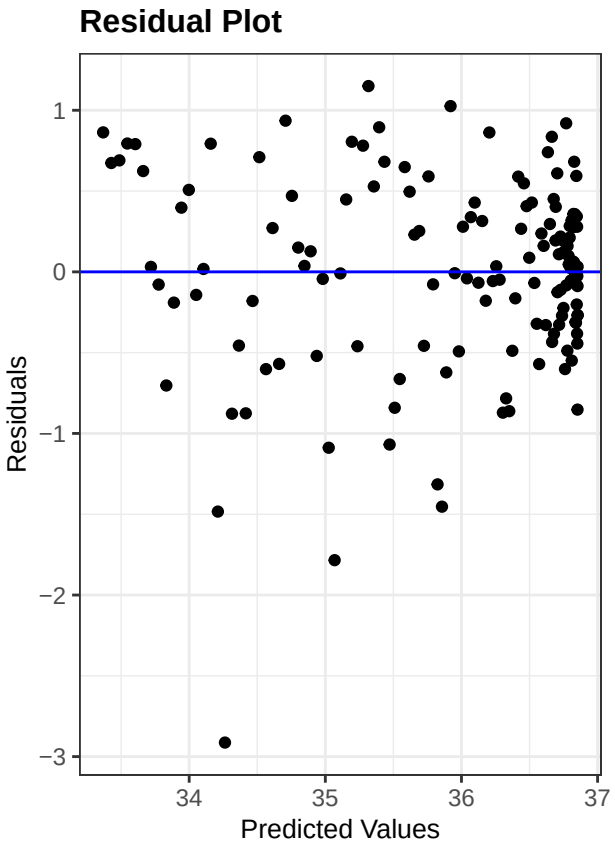
(6) Interpret the intercept of this model.

(7) Write the fitted model or predicted equation. Is the coefficient of year^2 positive or negative, and what does that imply about the relationship between speed and year as year increases? (Hint: You can think of this as an interaction between year and year!)

(8) Does this model meet our validity conditions?

```
resid_panel(poly.lm, plots = c('hist', 'resid'))
```





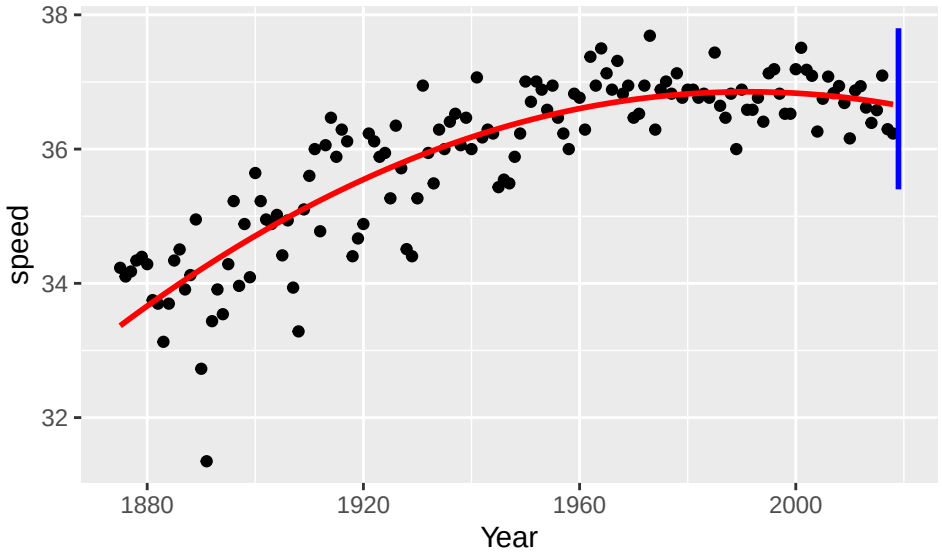
(9) If we want to use this model to predict, we could use the `predict()` function. How do we interpret this output?

```
predict(poly.lm,data.frame(Year=2019,Year.sq=2019^2),interval="prediction")
```

```
##        fit      lwr      upr
## 1 36.65162 35.42299 37.88026
```

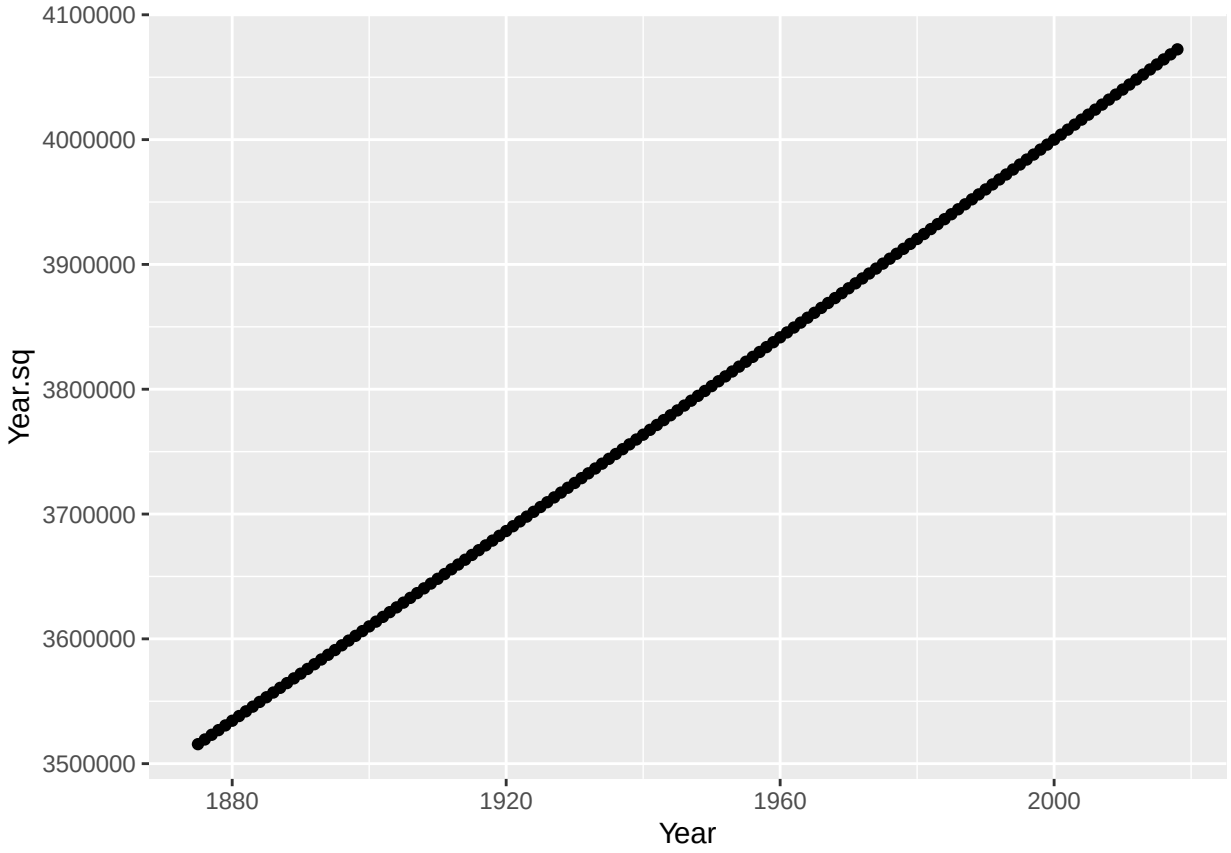
(10) Look at the following code. What does the blue line represent? Does our model perform well? What are some potential issues we may have with our explanatory variables?

```
pred.df<-data.frame(y1=35.4,y2=37.8,x1=2019,x2=2019)
ky.dat %>% ggplot(aes(x=Year,y=speed)) + geom_point() +
  geom_line(aes(x=Year,y=.fitted),data=poly.lm,lwd=1,color="red") +
  geom_segment(aes(x = x1, y = y1, xend = x2, yend = y2),lwd=1,colour = "blue", data = pred.df)
```



(11) Create a scatterplot of year^2 and year. What does this tell you?

```
ky.dat %>% ggplot(aes(x = Year, y = Year.sq)) + geom_point()
```



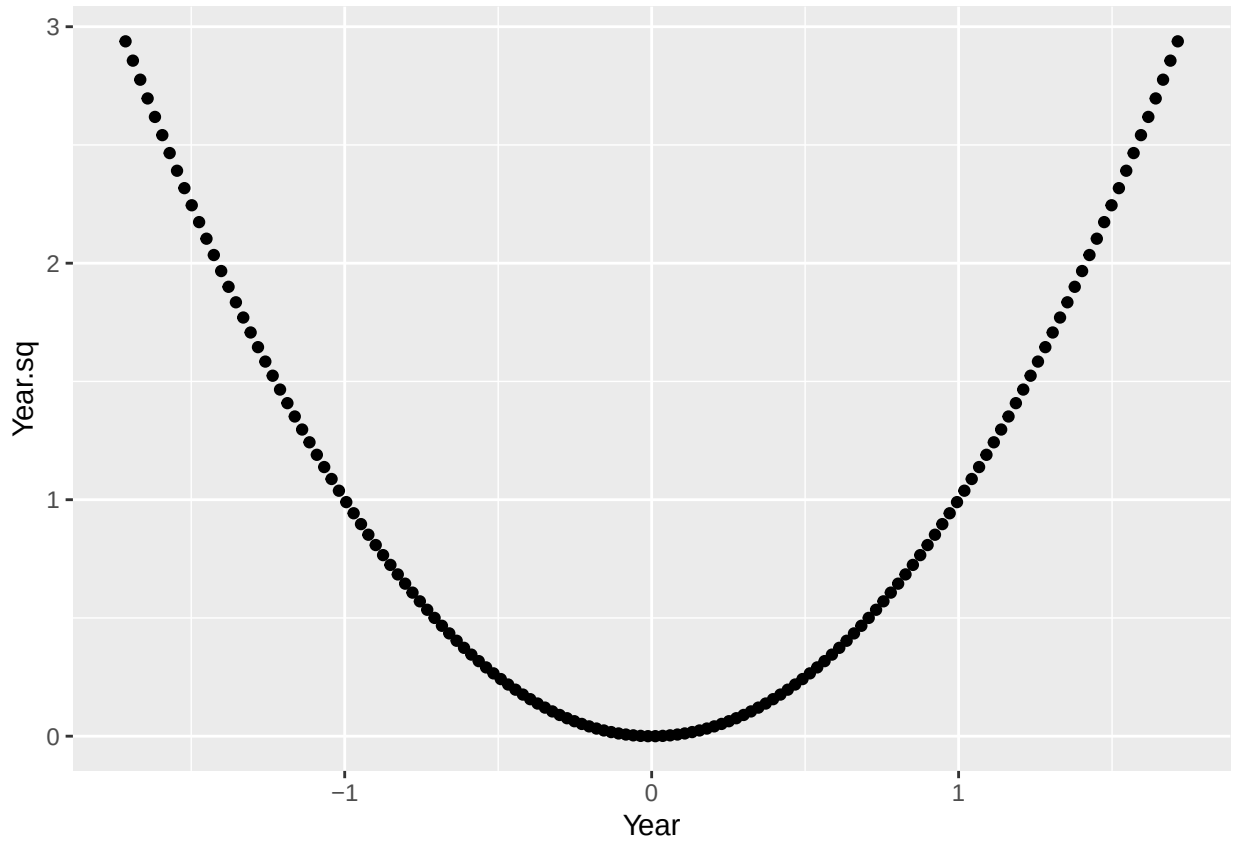
(12) How can we adjust for this?

(13) Standardize the year and year^2 variables and refit your model.

```
ky.dat.std = ky.dat %>% mutate_at(list(~scale(.)), .vars = vars(Year)) %>%
  mutate(Year.sq = Year^2)
poly.lm.std<-lm(speed ~ Year + Year.sq,data=ky.dat.std)
summary(poly.lm.std)
```

```
##
## Call:
## lm(formula = speed ~ Year + Year.sq, data = ky.dat.std)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -2.9131 -0.3167  0.0314  0.3986  1.1499
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 36.33966    0.07531 482.557  < 2e-16 ***
## Year         0.96192    0.05038  19.094  < 2e-16 ***
## Year.sq     -0.45017    0.05652  -7.964 5.02e-13 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.6024 on 141 degrees of freedom
## Multiple R-squared:  0.7522, Adjusted R-squared:  0.7487
## F-statistic:   214 on 2 and 141 DF,  p-value: < 2.2e-16
```

```
ky.dat.std %>% ggplot(aes(x = Year, y = Year.sq))  +  geom_point()
```



(14) Provide an interpretation of the intercept of this model (include units!).

Adjusting for track conditions

(15) If we want to we can adjust for track conditions. Create a model that adjusts for track conditions. To see if the added variables are important we can compare it to the smaller (nested) model via which test? What are the hypotheses for this test?

```
levels(factor(ky.dat$condition))
```

```
## [1] "dusty"   "fast"    "good"    "heavy"   "muddy"   "sloppy"  "slow"
## [8] "wetfast"
```

```
condition.lm<-lm(speed ~ Year  +  Year.sq + condition,data=ky.dat)
summary(condition.lm)
```

```
##
## Call:
## lm(formula = speed ~ Year + Year.sq + condition, data = ky.dat)
```

```
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -2.0476 -0.2359  0.0044  0.2583  0.8571
##
## Coefficients:
##                    Estimate Std. Error t value Pr(>|t|)
## (Intercept)      -1.060e+03  9.181e+01 -11.541  < 2e-16 ***
## Year              1.106e+00  9.446e-02  11.704  < 2e-16 ***
## Year.sq          -2.786e-04  2.427e-05 -11.476  < 2e-16 ***
## conditionfast    -3.988e-01  2.590e-01  -1.540  0.12600
## conditiongood    -7.694e-01  2.800e-01  -2.748  0.00682 **
## conditionheavy   -1.676e+00  2.883e-01  -5.813 4.26e-08 ***
## conditionmuddy   -1.636e+00  3.006e-01  -5.442 2.43e-07 ***
## conditionsloppy  -9.222e-01  3.017e-01  -3.057  0.00270 **
## conditionslow    -1.579e+00  3.068e-01  -5.145 9.32e-07 ***
## conditionwetfast -8.175e-01  5.013e-01  -1.631  0.10527
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.4192 on 134 degrees of freedom
## Multiple R-squared:  0.8859, Adjusted R-squared:  0.8783
## F-statistic: 115.7 on 9 and 134 DF,  p-value: < 2.2e-16
```

H_0 :

H_a :

(16) Conduct the partial-F test between the condition model and the original quadratic model. What are your conclusions?

```
anova(poly.lm, condition.lm)
```

```
## Analysis of Variance Table
##
## Model 1: speed ~ Year + Year.sq
## Model 2: speed ~ Year + Year.sq + condition
##   Res.Df    RSS Df Sum of Sq      F    Pr(>F)
## 1    141 51.172
## 2    134 23.553  7    27.619 22.448 < 2.2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

(17) Would the model you have created be useful in predicting this year's Kentucky Derby winner? Explain why or why not. (Hint: Consider issues beyond validity conditions.)

(18) Now let's take a look at the cubic model. Create a model predicting speed with year, year^2 and year^3. Does this appear to significantly improve the fit?

```
ky.dat.std <- ky.dat.std %>% mutate(Year.3=Year^3)
ky.dat <- ky.dat %>% mutate(Year.3=Year^3)
poly3.lm<-lm(speed ~ Year + Year.sq + Year.3,data=ky.dat.std)
# poly3.lm<-lm(speed ~ Year + Year.sq + Year.3,data=ky.dat)
summary(poly3.lm)
```

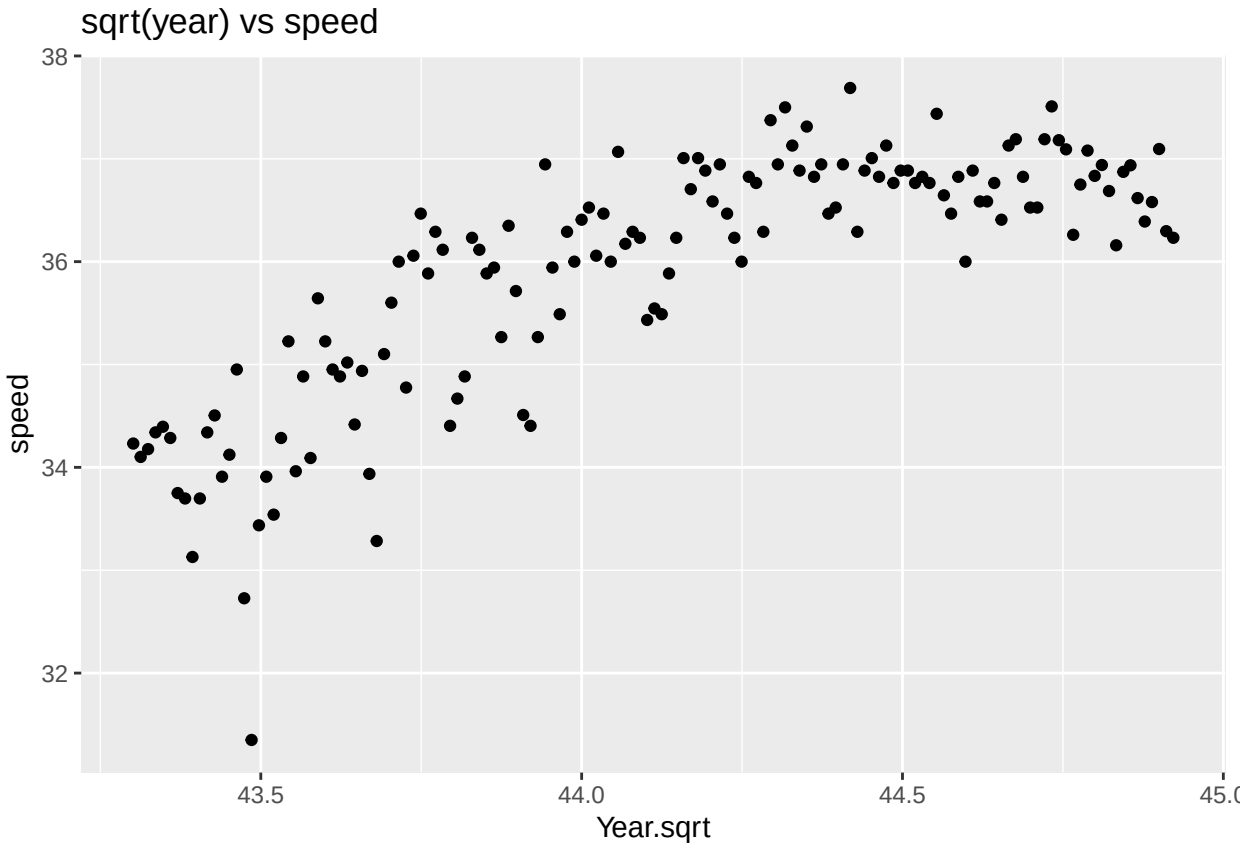
```
##
## Call:
## lm(formula = speed ~ Year + Year.sq + Year.3, data = ky.dat.std)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -2.91024 -0.34659  0.02968  0.36472  1.26550
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 36.33966    0.07455 487.439  < 2e-16 ***
## Year         1.18670    0.12470   9.516  < 2e-16 ***
## Year.sq     -0.45017    0.05596  -8.045 3.31e-13 ***
## Year.3      -0.12576    0.06395  -1.967   0.0512 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.5964 on 140 degrees of freedom
## Multiple R-squared:  0.7589, Adjusted R-squared:  0.7537
## F-statistic: 146.9 on 3 and 140 DF,  p-value: < 2.2e-16
```

```
anova(poly.lm, poly3.lm)
```

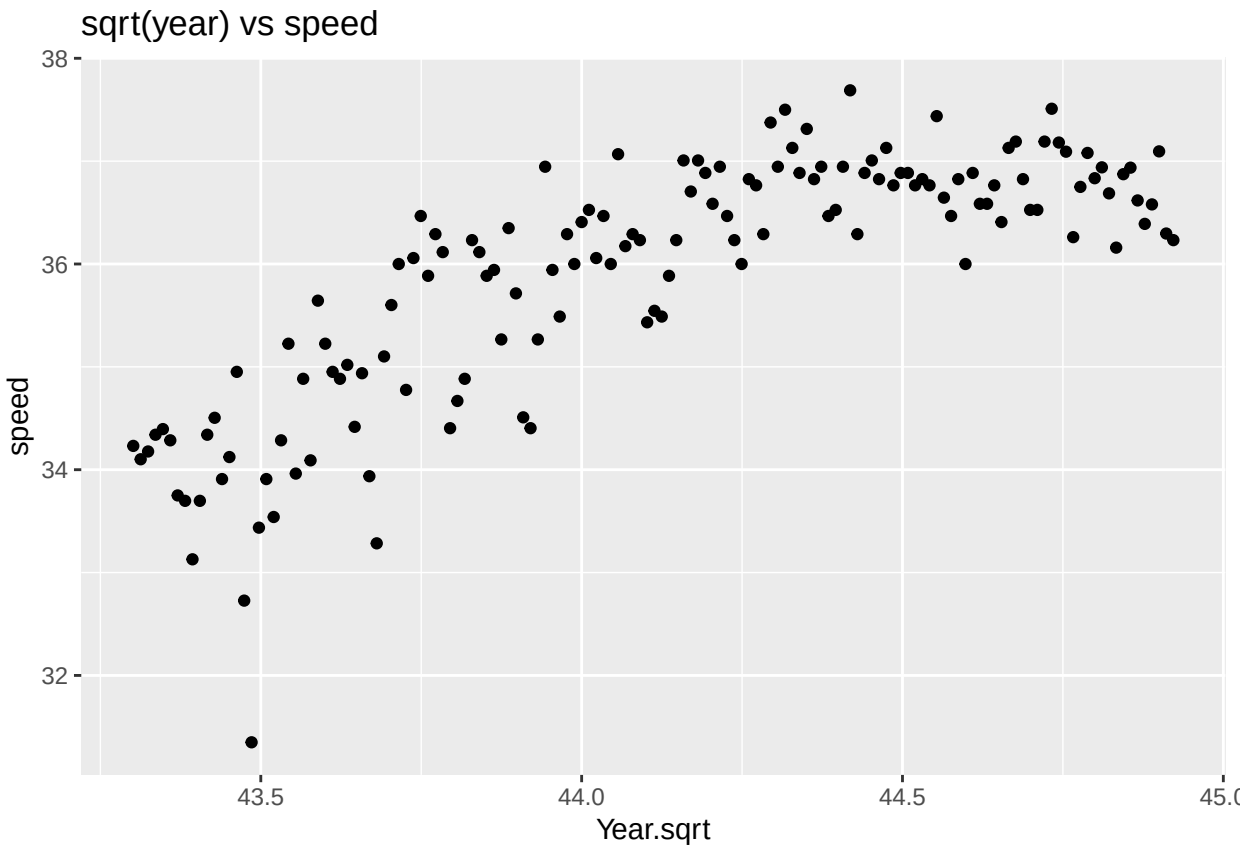
```
## Analysis of Variance Table
##
## Model 1: speed ~ Year + Year.sq
## Model 2: speed ~ Year + Year.sq + Year.3
##   Res.Df    RSS Df Sum of Sq      F Pr(>F)
## 1    141 51.172
## 2    140 49.796  1    1.3757 3.8677 0.0512 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

(19) Changing gears... try performing transformations on the data. Create scatterplots of 1) log(year) vs. speed; 2) sqrt(year) vs. speed; 3) year-squared vs. speed. Which transformation looks like it has the best fit? Create a linear model

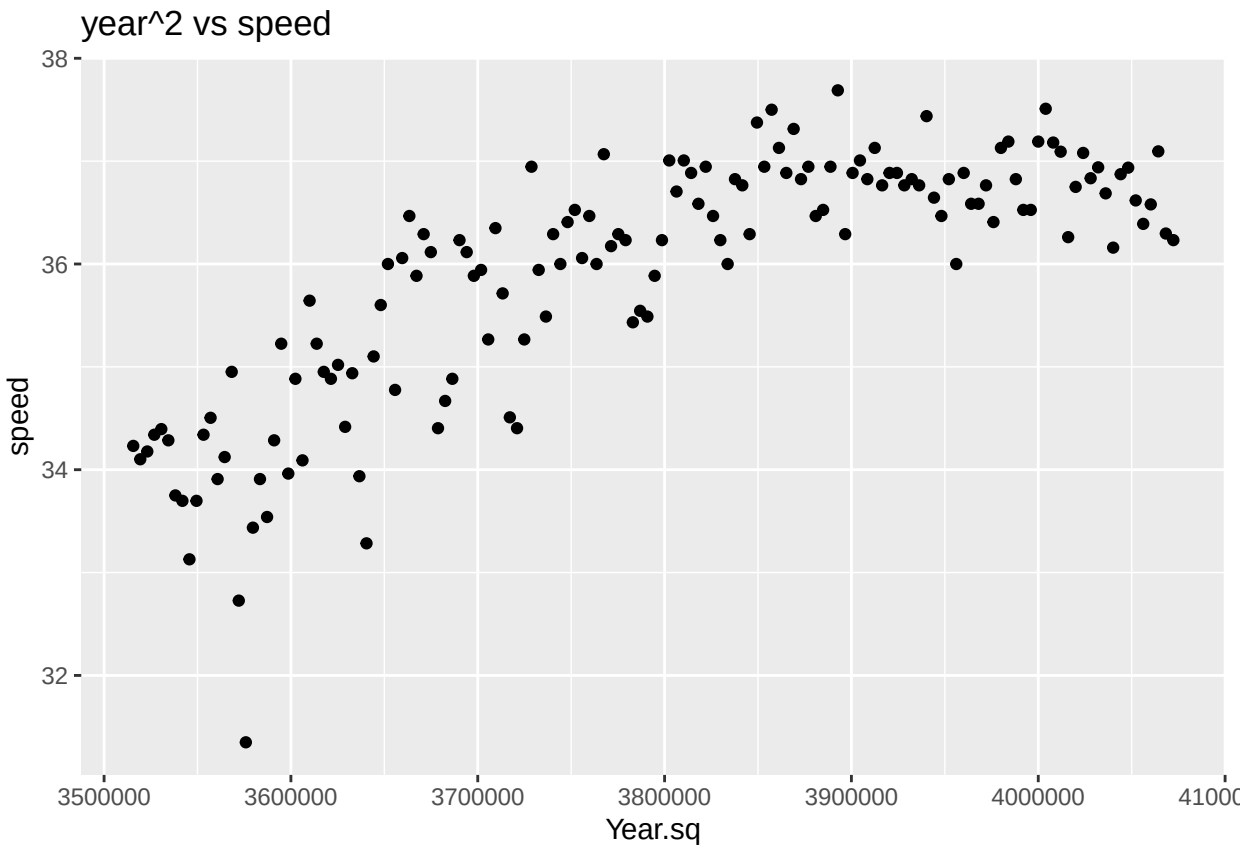
```
ky.dat <- ky.dat %>% mutate(Year.sqrt = sqrt(Year), Year.log = log(Year))
ky.dat %>% ggplot(aes(x=Year.sqrt, y=speed)) + geom_point() + labs(title="sqrt(year) vs speed")
```




```
ky.dat %>% ggplot(aes(x=Year.sqrt, y=speed)) + geom_point() + labs(title="sqrt(year) vs speed")
```




```
ky.dat %>% ggplot(aes(x=Year.sq, y=speed)) + geom_point() + labs(title="year^2 vs speed")
```

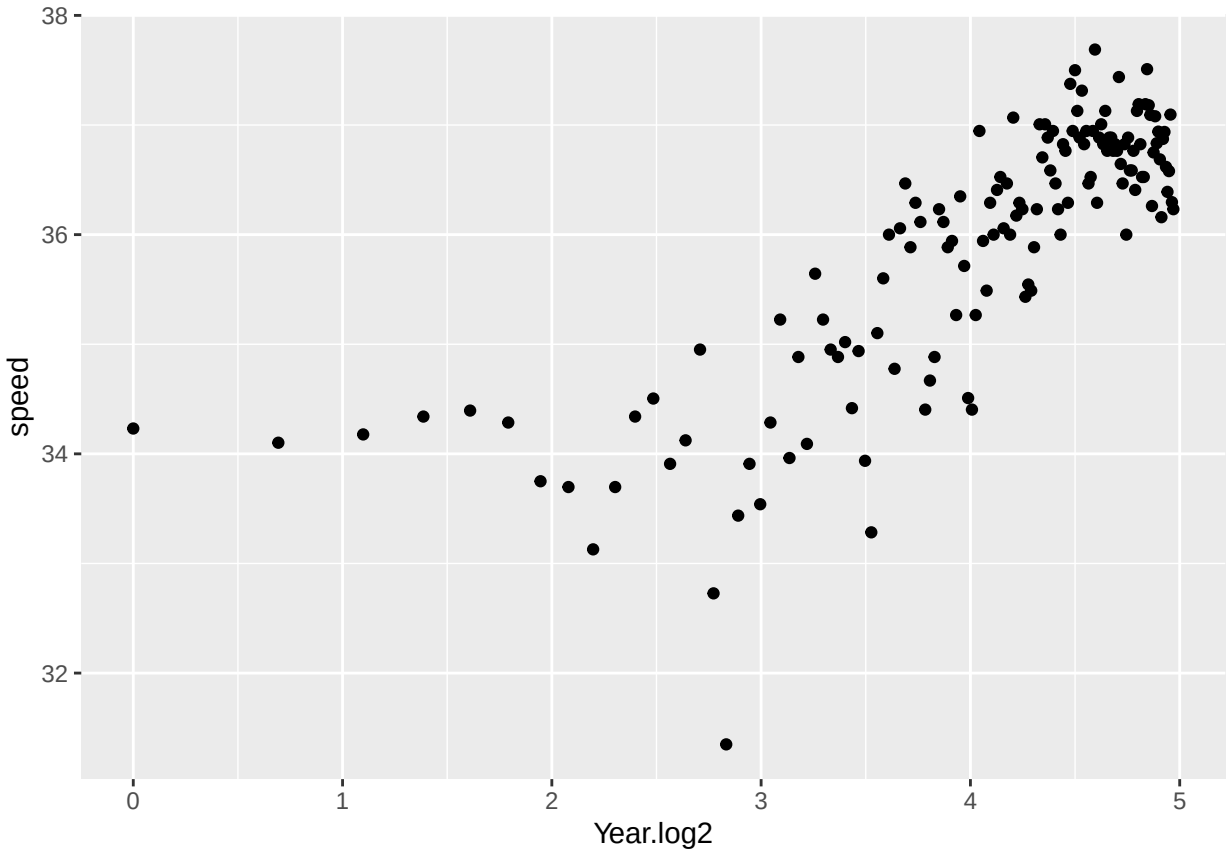



```
log.lm <- ky.dat %>% lm(speed ~ Year.log, data = .)
summary(log.lm)
```

```
##
## Call:
## lm(formula = speed ~ Year.log, data = .)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -3.2500 -0.4279  0.0494  0.5071  1.4031
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  -305.29      21.20  -14.40   <2e-16 ***
## Year.log       45.05       2.80   16.09   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.7177 on 142 degrees of freedom
## Multiple R-squared:  0.6458, Adjusted R-squared:  0.6433
## F-statistic: 258.9 on 1 and 142 DF,  p-value: < 2.2e-16
```

(20) An alternative is to use the log transformation but shifted by a value. Create a new variable `log(Year.log - 1874)`. Create a scatterplot of this transformation vs. speed. Does it appear to be more successful? Justify your answer.

```
my_year <- 1874
ky.dat <- ky.dat %>% mutate(Year.log2 = log(Year - my_year))
ky.dat %>% ggplot(aes(x=Year.log2, y=speed)) + geom_point()
```



(21) Create a linear model to predict speed using the updated transformation in (20). What is a 95% prediction interval in 2019?

```
log.lm2 <- ky.dat %>% lm(speed ~ Year.log2, data = .)
summary(log.lm2)
```

```
##
## Call:
## lm(formula = speed ~ Year.log2, data = .)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
```

```
## -3.3631 -0.3394  0.0873  0.4527  2.3992
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 31.83219    0.26426  120.46   <2e-16 ***
## Year.log2    1.01677    0.06441   15.79   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.7266 on 142 degrees of freedom
## Multiple R-squared:  0.637,  Adjusted R-squared:  0.6344
## F-statistic: 249.2 on 1 and 142 DF,  p-value: < 2.2e-16
```

```
predict(log.lm2,data.frame(Year.log2=log(2019-my_year)),interval="prediction")
```

```
##        fit      lwr      upr
## 1 36.89238 35.44564 38.33911
```