```
library(ggplot2)
```

```
## Warning: package ’ggplot2’ was built under R version 4.1.2
```

```
## Warning in as.POSIXlt.POSIXct(Sys.time()): unable to identify current timezone ’H’:
## please set environment variable ’TZ’
```

```
library(tidyverse)
```

```
## Warning: package ’tidyverse’ was built under R version 4.1.2
```

```
## -- Attaching packages --------------------------------------- tidyverse 1.3.1 --
```

```
## v tibble  3.1.6     v dplyr   1.0.7
## v tidyr   1.1.4     v stringr 1.4.0
## v readr   2.1.1     v forcats 0.5.1
## v purrr   0.3.4
```

```
## Warning: package ’tibble’ was built under R version 4.1.2
```

```
## Warning: package ’tidyr’ was built under R version 4.1.2
```

```
## Warning: package ’readr’ was built under R version 4.1.2
```

```
## Warning: package ’purrr’ was built under R version 4.1.2
```

```
## Warning: package ’dplyr’ was built under R version 4.1.2
```

```
## Warning: package ’stringr’ was built under R version 4.1.2
```

```
## Warning: package ’forcats’ was built under R version 4.1.2
```

```
## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

```
library(broom)
```

```
## Warning: package ’broom’ was built under R version 4.1.2
```

```
library(viridis)
```

```
## Warning: package ’viridis’ was built under R version 4.1.2
```

```
## Loading required package: viridisLite
```

```
## Warning: package ’viridisLite’ was built under R version 4.1.2
```

```
movie_budgets = read.table("https://jfukuyama.github.io/teaching/stat670/assignments/movie_budgets.txt"
```

```
view(movie_budgets)
```

```
movie_budgets$log10budget = log10(movie_budgets$budget)
```

```
#ggplot(data = movie_budgets,aes(x = year,y=log10budget)) + geom_point() +
  #facet_wrap(~cut_number(length,n=8)) + geom_smooth(method = "loess",span=1,method.args = list(degree=
```

Question 1

For the following set of answers refer the attached R file for plots.

1. In the above plot (see .r file for the plots), we plotted log10(budget) against years according to length of the movie in minutes. We see in the above graph, that as the year increases, the log10(budget) decreases when length is less than 72. The log10(budget) then remains almost constant when across all years when length is between 72 to 86 minutes. For length greater than 86 minutes, thelog10(budget) increases as year increases and we see that the slope is positive. So the slope also increases as length increases. Hence, we use should use a curved function to fit the year according to length.

```
#ggplot(data = movie_budgets,aes(x = length,y=log10budget)) + geom_point() +
  #facet_wrap(~cut_number(year,n=8)) + geom_smooth(method = "loess",span=1,method.args = list(degree=1,
```

2. In the above plot, we plotted log10(budget) against length according to the years. We see that log10(budget) increases initially as the length increases but then it remains constant or decreases after length of 200 minutes. Hence we will use a curved model for fitting length in terms of year.
3. From given plot for log10(budget) against length according to the years, we see that curve for log10(budget) is initially similar for almost all time intervals except for [1906,1955] and once the length of movies exceeds 200 then the fitted curve changes for all time frames. This is because more outliers are present after the movie length of 200. However when we look at the plot for log10(budget) against years according to length, we see that the thelog10(budget) increases with increase in year according to length. So we say the slope of curve shifts from negative to positive. Due to this relationship, we need an interaction between year and length.
4. For both the plots, we have used a span of 1 for fitting our loess model on our data. We see that the curve is unaffected by the presence of outliers as seen in the above plot. There are many outliers when length of movies exceeds 200 minutes. So as span of 1 gives an ideal fit for the data.

```
cor(movie_budgets$year,movie_budgets$log10budget)
```

```
## [1] 0.1887752
```

```
cor(movie_budgets$length,movie_budgets$log10budget)
```

```
## [1] 0.6342756
```

5. We have used a symmetric family in our loess model. This is defined as a robust fit for the data.

We then check the correlation for the year and length with the log10(budget). We see that year shows a weak positive correlation of 0.188 and length shows a moderately positive correlation with log10(budget). Therefore we use length to predict the log10(budget) according to years.

Question 2

Plotting the fit for the model on length on the basis of year requires a grid which is dense in length and sparse in year.

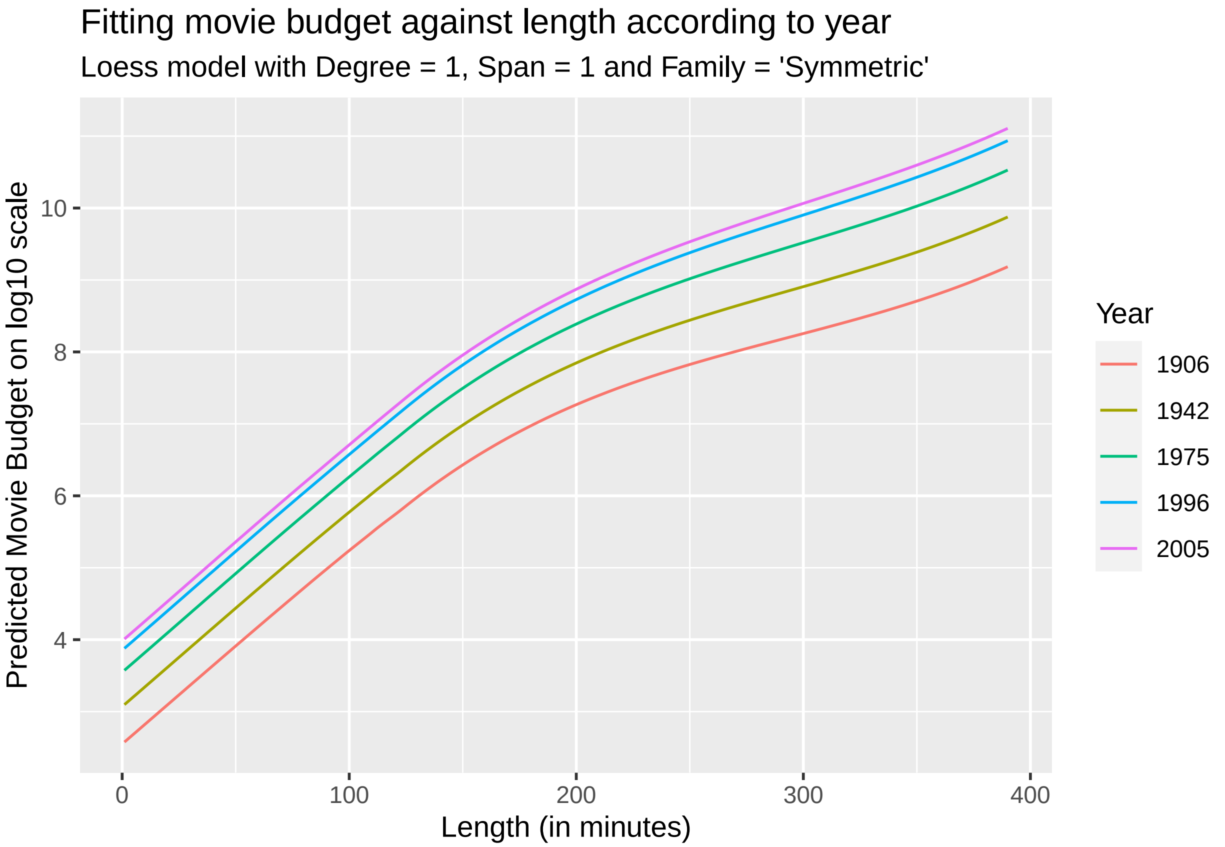
```
budget_loess = loess(log10budget~length*year, data = movie_budgets,span=1, degree = 1, family = "symmetr

movies_grid = expand.grid(year = c(1906, 1942, 1975, 1996, 2005),length=seq(1,390,1))

movies_pred = predict(budget_loess, newdata=movies_grid)

movies_pred = data.frame(movies_grid, predicted_log10_budget = as.vector(movies_pred))

ggplot(movies_pred,aes(x = length, y = predicted_log10_budget, group = year, color = factor(year))) +
 geom_line()  + labs(color = "Year") + labs(title = "Fitting movie budget against length according to y
```



From the above plot, we see that the relationship between log10(budget) and length is non-monotonic. The log10(budget) increases initially and then decreases for movies having length more than 200 minutes. The curves do not seem to converge which shows that year of release is also important when we are predicting the movie budget on log10 scale using length.

Question 3

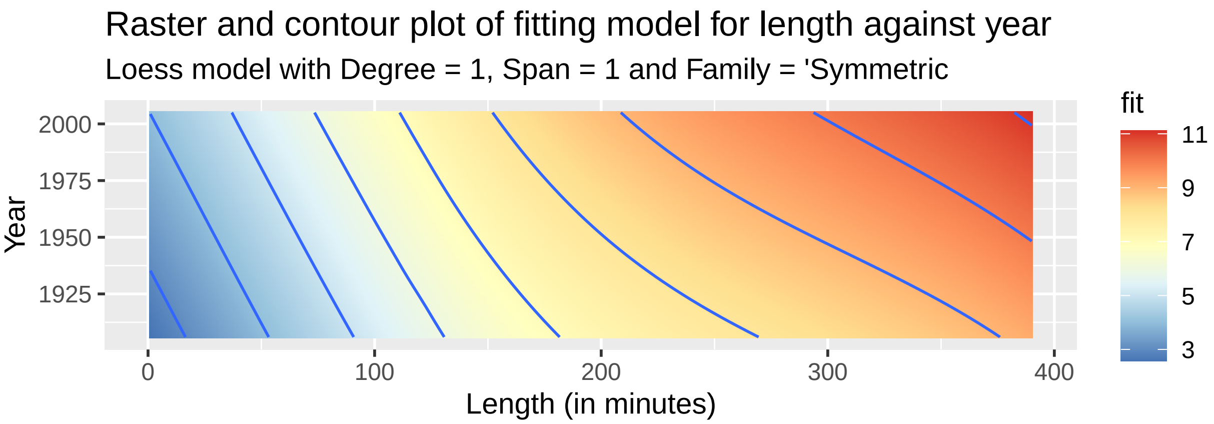
```
budget_loess = loess(log10budget ~ length * year, data = movie_budgets, span = 1, degree = 1, family =

movies_grid = expand.grid(year =seq(1906, 2005, 1), length=seq(1,390,1))

movies_pred = predict(budget_loess, newdata = movies_grid)

movies_df = data.frame(movies_grid, fit = as.vector(movies_pred))

ggplot(movies_df, aes(x = length, y = year, z = fit)) + geom_raster(aes(fill = fit)) +
coord_fixed() + scale_fill_distiller(palette = "RdYlBu") + geom_contour() + labs(title = "Raster and co
```



From the above plot, We see that model shows the fact that the budget on log10 scale of a movie doesn't increase a lot when the length increases beyond 200.