prediction model for Churn\_out\_rate

library(readr)  
library(ggplot2)  
data<-read.csv("E:\\assignments\\3.simple linear regression\\emp\_data.csv")  
data

## Salary\_hike Churn\_out\_rate  
## 1 1580 92  
## 2 1600 85  
## 3 1610 80  
## 4 1640 75  
## 5 1660 72  
## 6 1690 70  
## 7 1706 68  
## 8 1730 65  
## 9 1800 62  
## 10 1870 60

attach(data)  
SH<-Salary\_hike  
COR<-Churn\_out\_rate  
  
# Exploratory data analysis  
# structure of data  
str(data)

## 'data.frame': 10 obs. of 2 variables:  
## $ Salary\_hike : int 1580 1600 1610 1640 1660 1690 1706 1730 1800 1870  
## $ Churn\_out\_rate: int 92 85 80 75 72 70 68 65 62 60

# Descriptive statistics  
library(psych)

##   
## Attaching package: 'psych'

## The following objects are masked from 'package:ggplot2':  
##   
## %+%, alpha

describe(data)

## vars n mean sd median trimmed mad min max range skew  
## Salary\_hike 1 10 1688.6 92.10 1675 1679.50 88.96 1580 1870 290 0.62  
## Churn\_out\_rate 2 10 72.9 10.26 71 72.12 11.12 60 92 32 0.47  
## kurtosis se  
## Salary\_hike -0.94 29.12  
## Churn\_out\_rate -1.16 3.24

class(data)

## [1] "data.frame"

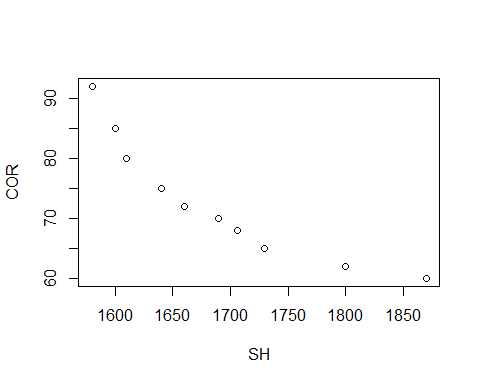
dim(data)

## [1] 10 2

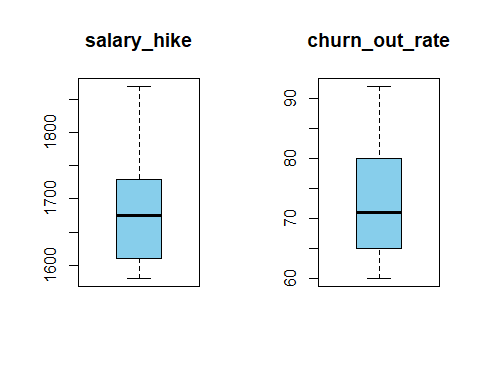
#correlation  
cor(SH, COR)

## [1] -0.9117216

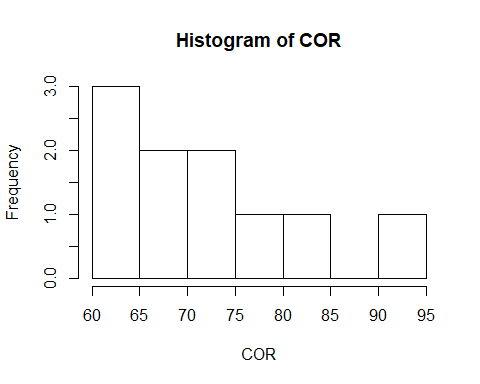
#data is heighly negatively correlated  
  
plot(SH, COR)



# from the above plot data is negatively correlated  
  
par(mfrow=c(1,2))  
boxplot(SH, main="salary\_hike", col="skyblue")  
boxplot(COR, main="churn\_out\_rate", col="skyblue")



# from above boxplots there is no outliers  
hist(COR)



# the data is right skewed  
  
#Regression analysis  
reg<-lm(COR ~ SH)  
summary(reg)

##   
## Call:  
## lm(formula = COR ~ SH)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -3.804 -3.059 -1.819 2.430 8.072   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 244.36491 27.35194 8.934 1.96e-05 \*\*\*  
## SH -0.10154 0.01618 -6.277 0.000239 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 4.469 on 8 degrees of freedom  
## Multiple R-squared: 0.8312, Adjusted R-squared: 0.8101   
## F-statistic: 39.4 on 1 and 8 DF, p-value: 0.0002386

confint(reg, level=0.95)

## 2.5 % 97.5 %  
## (Intercept) 181.2912317 307.4385905  
## SH -0.1388454 -0.0642399

predict(reg, interval="predict")

## Warning in predict.lm(reg, interval = "predict"): predictions on current data refer to \_future\_ responses

## fit lwr upr  
## 1 83.92753 72.38391 95.47115  
## 2 81.89668 70.59327 93.20009  
## 3 80.88125 69.68123 92.08127  
## 4 77.83497 66.87456 88.79538  
## 5 75.80412 64.94216 86.66607  
## 6 72.75784 61.94828 83.56740  
## 7 71.13316 60.30425 81.96206  
## 8 68.69613 57.77694 79.61533  
## 9 61.58815 50.00746 73.16884  
## 10 54.48016 41.72742 67.23290

rmse1<-sqrt(mean(reg$residuals^2))  
rmse1

## [1] 3.997528

# multiple R-squared value is 0.8312  
# for better R-squared value using transformation  
# Logarthmic transformation  
reg\_log<-lm(COR ~ log(SH))  
summary(reg\_log)

##   
## Call:  
## lm(formula = COR ~ log(SH))  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -3.678 -2.851 -1.794 2.275 7.624   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 1381.5 195.4 7.070 0.000105 \*\*\*  
## log(SH) -176.1 26.3 -6.697 0.000153 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 4.233 on 8 degrees of freedom  
## Multiple R-squared: 0.8486, Adjusted R-squared: 0.8297   
## F-statistic: 44.85 on 1 and 8 DF, p-value: 0.0001532

confint(reg\_log, level=0.95)

## 2.5 % 97.5 %  
## (Intercept) 930.8584 1832.0540  
## log(SH) -236.7512 -115.4682

predict(reg\_log, interval="predict")

## Warning in predict.lm(reg\_log, interval = "predict"): predictions on current data refer to \_future\_ responses

## fit lwr upr  
## 1 84.37627 73.40258 95.34996  
## 2 82.16102 71.43838 92.88366  
## 3 81.06376 70.44736 91.68017  
## 4 77.81241 67.43614 88.18869  
## 5 75.67773 65.39568 85.95978  
## 6 72.52344 62.28515 82.76172  
## 7 70.86397 60.60253 81.12541  
## 8 68.40372 58.04985 78.75760  
## 9 61.41829 50.44392 72.39265  
## 10 54.69939 42.69592 66.70286

rmse2<-sqrt(mean(reg\_log$residuals^2))  
rmse2

## [1] 3.786004

# p value<0.05 and multiple R-squared value is 0.8486  
#Exponential transformation  
reg\_exp<-lm(log(COR) ~ SH)  
summary(reg\_exp)

##   
## Call:  
## lm(formula = log(COR) ~ SH)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -0.04825 -0.03519 -0.01909 0.02942 0.08970   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 6.6383000 0.3175983 20.902 2.88e-08 \*\*\*  
## SH -0.0013963 0.0001878 -7.434 7.38e-05 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 0.0519 on 8 degrees of freedom  
## Multiple R-squared: 0.8735, Adjusted R-squared: 0.8577   
## F-statistic: 55.26 on 1 and 8 DF, p-value: 7.377e-05

confint(reg\_exp, level=0.95)

## 2.5 % 97.5 %  
## (Intercept) 5.905917079 7.3706828388  
## SH -0.001829477 -0.0009631923

exp(predict(reg\_exp, interval="predict"))

## Warning in predict.lm(reg\_exp, interval = "predict"): predictions on current data refer to \_future\_ responses

## fit lwr upr  
## 1 84.10710 73.55634 96.17122  
## 2 81.79076 71.73037 93.26215  
## 3 80.65662 70.82070 91.85860  
## 4 77.34770 68.10452 87.84537  
## 5 75.21752 66.30465 85.32848  
## 6 72.13174 63.62321 81.77813  
## 7 70.53808 62.20357 79.98932  
## 8 68.21338 60.09051 77.43427  
## 9 61.86146 54.07801 70.76517  
## 10 56.10101 48.37944 65.05499

rmse3<-sqrt(mean(reg\_exp$residuals^2))  
rmse3

## [1] 0.04641748

#p values <0.05 and multiple R-squared value is 0.8735  
#polynomial of second degree transformation  
reg\_poly<-lm(COR ~ SH+I(SH^2))   
summary(reg\_poly)

##   
## Call:  
## lm(formula = COR ~ SH + I(SH^2))  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -2.5523 -1.3280 0.3497 0.9029 2.8296   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 1.647e+03 2.281e+02 7.222 0.000174 \*\*\*  
## SH -1.737e+00 2.657e-01 -6.538 0.000322 \*\*\*  
## I(SH^2) 4.754e-04 7.720e-05 6.158 0.000464 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 1.886 on 7 degrees of freedom  
## Multiple R-squared: 0.9737, Adjusted R-squared: 0.9662   
## F-statistic: 129.6 on 2 and 7 DF, p-value: 2.949e-06

confint(reg\_poly, level=0.95)

## 2.5 % 97.5 %  
## (Intercept) 1.107738e+03 2.186285e+03  
## SH -2.365306e+00 -1.108872e+00  
## I(SH^2) 2.928508e-04 6.579259e-04

predict(reg\_poly, interval="predict")

## Warning in predict.lm(reg\_poly, interval = "predict"): predictions on current data refer to \_future\_ responses

## fit lwr upr  
## 1 89.17035 83.78480 94.55590  
## 2 84.66327 79.65803 89.66851  
## 3 82.55234 77.66358 87.44111  
## 4 76.79003 72.03031 81.54976  
## 5 73.42388 68.63567 78.21210  
## 6 69.08774 64.20255 73.97292  
## 7 67.12501 62.19286 72.05716  
## 8 64.63730 59.66195 69.61265  
## 9 60.50952 55.48124 65.53780  
## 10 61.04055 54.97443 67.10667

rmse4<-sqrt(mean(reg\_poly$residuals^2))  
rmse4

## [1] 1.577975

#p values>0.05 and multiple R-squared value is 0.9737  
#polynomial of three degree transformation  
reg\_poly1<-lm(COR ~ SH+I(SH^2)+I(SH^3))  
summary(reg\_poly1)

##   
## Call:  
## lm(formula = COR ~ SH + I(SH^2) + I(SH^3))  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -2.06811 -0.49848 0.04253 0.76434 1.49050   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 1.320e+04 3.900e+03 3.384 0.0148 \*  
## SH -2.194e+01 6.817e+00 -3.218 0.0182 \*  
## I(SH^2) 1.223e-02 3.966e-03 3.084 0.0216 \*  
## I(SH^3) -2.276e-06 7.679e-07 -2.964 0.0251 \*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 1.298 on 6 degrees of freedom  
## Multiple R-squared: 0.9893, Adjusted R-squared: 0.984   
## F-statistic: 185.4 on 3 and 6 DF, p-value: 2.647e-06

##p values<0.05 and multiple R-square valuesis 0.9893  
#Adjusted R-squared value is 0.984  
#polynomial of three degree gives the best R-squared values  
  
confint(reg\_poly1, level=0.95)

## 2.5 % 97.5 %  
## (Intercept) 3.654531e+03 2.273912e+04  
## SH -3.861930e+01 -5.255724e+00  
## I(SH^2) 2.525628e-03 2.193585e-02  
## I(SH^3) -4.154996e-06 -3.971350e-07

#predicted values  
predict(reg\_poly1, interval="predict")

## Warning in predict.lm(reg\_poly1, interval = "predict"): predictions on current data refer to \_future\_ responses

## fit lwr upr  
## 1 90.86026 86.77985 94.94067  
## 2 84.73315 81.16894 88.29736  
## 3 82.06811 78.56441 85.57180  
## 4 75.50317 71.95163 79.05471  
## 5 72.18180 68.62172 75.74188  
## 6 68.50950 64.99864 72.02037  
## 7 67.08304 63.57116 70.59491  
## 8 65.48442 61.87360 69.09525  
## 9 62.88304 58.80179 66.96430  
## 10 59.69351 55.23357 64.15346

#residuals  
residuals<-reg\_poly1$residuals  
residuals

## 1 2 3 4 5 6 7   
## 1.1397407 0.2668535 -2.0681055 -0.5031676 -0.1817993 1.4904957 0.9169608   
## 8 9 10   
## -0.4844233 -0.8830420 0.3064871

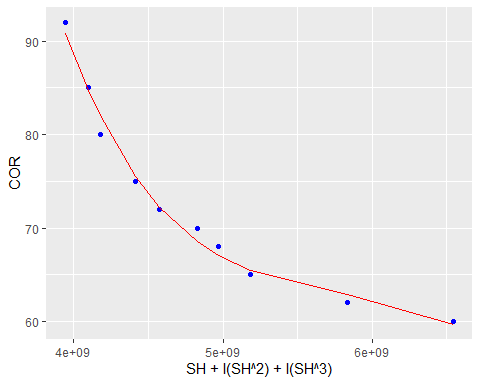
#residual mean square error  
rmse<-sqrt(mean(residuals^2))  
rmse

## [1] 1.0052

predicted\_values<-predict(reg\_poly1)  
predicted\_values

## 1 2 3 4 5 6 7 8   
## 90.86026 84.73315 82.06811 75.50317 72.18180 68.50950 67.08304 65.48442   
## 9 10   
## 62.88304 59.69351

ggplot(data=data, aes(x=SH+I(SH^2)+I(SH^3), y=COR))+geom\_point(color="blue")+geom\_line(color="red", data=data, aes(x=SH+I(SH^2)+I(SH^3), y=predicted\_values))



par(mfrow=c(2,2))  
plot(reg\_poly1)

## Warning in sqrt(crit \* p \* (1 - hh)/hh): NaNs produced  
  
## Warning in sqrt(crit \* p \* (1 - hh)/hh): NaNs produced

