Rmarkdown - Understanding Driver Performance

AshleshBilladyshetty

August 15, 2017

Importing the data

```
setwd("C:\\Users\\Ashlesh B Shetty\\OneDrive\\6120 IntroStatdataScientists\\Project_Stats_F1\\Mo
del_TeamIteration3_15Aug2017\\")
library(readxl)
```

```
## Warning: package 'readxl' was built under R version 3.4.1
```

```
df <- read_excel("Updated_DataFinal_15Aug2017.xlsx", col_names = TRUE)
```

Data understanding

```
#variable distribution analysis of the cleaned, collated, and missing value and outlier treated
 data
# h <- hist(df$Speed)
# h <- hist(df$Height)
# h <- hist(df$Dependents)
# h <- hist(df$Age)
# h <- hist(df$TeamPrevYrScore)
# h <- hist(df$TotalTurns)

summary( df$Speed,df$TotalTurns , df$Height , df$Age , df$Grid , df$TeamPrevYrScore)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   100.6   170.2   184.2   182.2   198.8   237.6
```

```
#write.csv( summary(df), file ="MyData.csv",row.names=FALSE)

library(dplyr)
```

```
## Warning: package 'dplyr' was built under R version 3.4.1
```

```
##
## 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
```

```
op <- df %>%
  group_by(Race) %>%
  summarise(avg_speed = mean(Speed), count = n())%>%
  arrange(avg_speed)
#write.csv( op, file ="MyData.csv",row.names=FALSE)

op <- df %>%
  group_by(Dependents) %>%
  summarise(avg_speed = mean(Speed), count = n())%>%
  arrange(avg_speed)
#write.csv( op, file ="MyData.csv",row.names=FALSE)

op <- df %>%
  group_by(RaceContinent) %>%
  summarise(avg_speed = mean(Speed), count = n())%>%
  arrange(avg_speed)
#write.csv( op, file ="MyData.csv",row.names=FALSE)

data.frame(cor(df[,c('Height','Age','Speed','TotalTurns','TeamPrevYrScore','Grid')]))
```

```
##                      Height         Age       Speed   TotalTurns
## Height           1.00000000 -0.30645616 -0.07791845  0.021188002
## Age             -0.30645616  1.00000000  0.09783172 -0.059002038
## Speed           -0.07791845  0.09783172  1.00000000 -0.659882909
## TotalTurns       0.02118800 -0.05900204 -0.65988291  1.000000000
## TeamPrevYrScore -0.57516708  0.12496590  0.07155435  0.005543305
## Grid             0.05748949 -0.16907953 -0.15561718  0.063831939
##                 TeamPrevYrScore        Grid
## Height             -0.575167076  0.05748949
## Age                 0.124965896 -0.16907953
## Speed               0.071554354 -0.15561718
## TotalTurns          0.005543305  0.06383194
## TeamPrevYrScore     1.000000000 -0.04220617
## Grid               -0.042206168  1.00000000
```

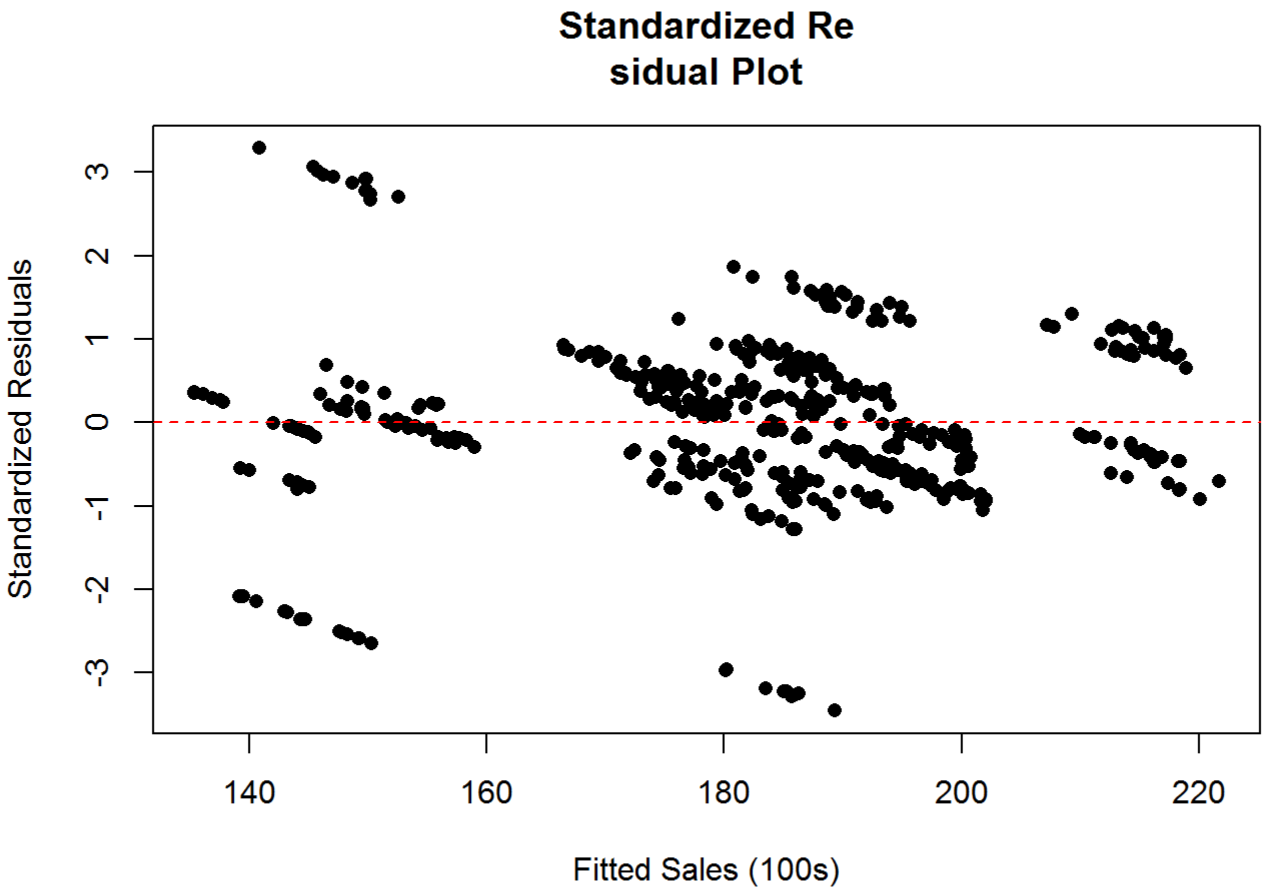
```
#write.csv( op, file ="MyData.csv",row.names=FALSE)
```

Modeling exercise

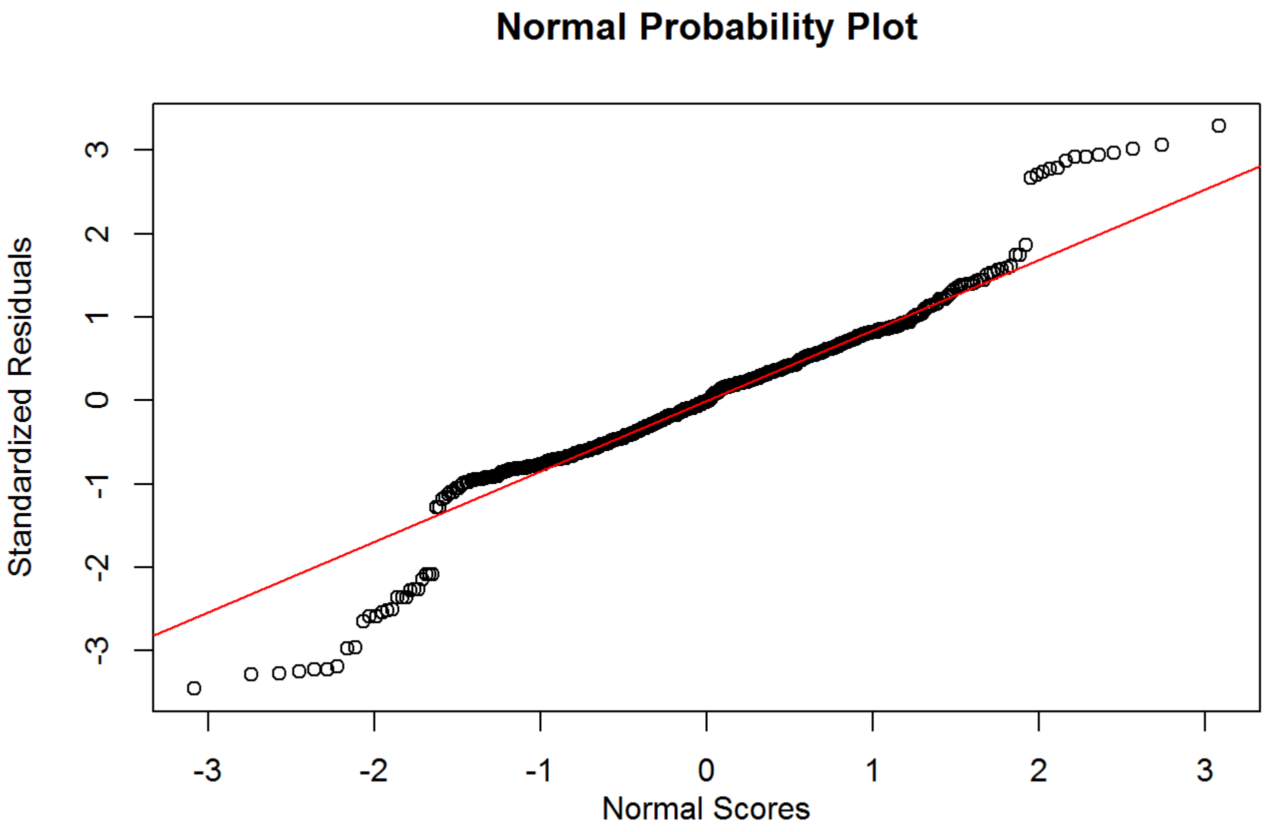
```
df$RaceContinent_F<-factor(df$RaceContinent)
df$RaceContinent_F <- relevel(df$RaceContinent_F, ref ='SA')

fit1 <- lm(df$Speed ~ df$TotalTurns + df$Dependents + df$Height + df$Age + df$Grid +df$TeamPrevY
rScore +df$RaceContinent_F)
fit1 <- lm(df$Speed ~ df$TotalTurns + df$Dependents +  df$Age + df$Grid +df$TeamPrevYrScore
+df$RaceContinent_F)
fit1 <- lm(df$Speed ~ df$TotalTurns +  df$Age + df$Grid +df$TeamPrevYrScore +df$RaceContinent_F)
fit1 <- lm(df$Speed ~ df$TotalTurns + df$Grid +df$TeamPrevYrScore +df$RaceContinent_F)

summary(fit1)
```

```
## 
## Call:
## lm(formula = df$Speed ~ df$TotalTurns + df$Grid + df$TeamPrevYrScore + 
##     df$RaceContinent_F)
## 
## Residuals:
##     Min      1Q  Median      3Q     Max 
## -64.924 -10.742  -0.248  10.634  60.642 
## 
## Coefficients:
##                         Estimate Std. Error t value Pr(>|t|)    
## (Intercept)           230.658271   6.206607  37.163  < 2e-16 ***
## df$TotalTurns          -0.080289   0.003935 -20.404  < 2e-16 ***
## df$Grid                -0.518158   0.157036  -3.300  0.00104 ** 
## df$TeamPrevYrScore      0.067299   0.033213   2.026  0.04328 *  
## df$RaceContinent_FASI  33.271074   3.768709   8.828  < 2e-16 ***
## df$RaceContinent_FAUS  25.976880   5.035344   5.159 3.62e-07 ***
## df$RaceContinent_FEU   28.225923   3.802533   7.423 5.17e-13 ***
## df$RaceContinent_FNA   36.437673   4.378860   8.321 8.86e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## 
## Residual standard error: 18.87 on 486 degrees of freedom
## Multiple R-squared:  0.5363, Adjusted R-squared:  0.5296 
## F-statistic: 80.29 on 7 and 486 DF,  p-value: < 2.2e-16
```

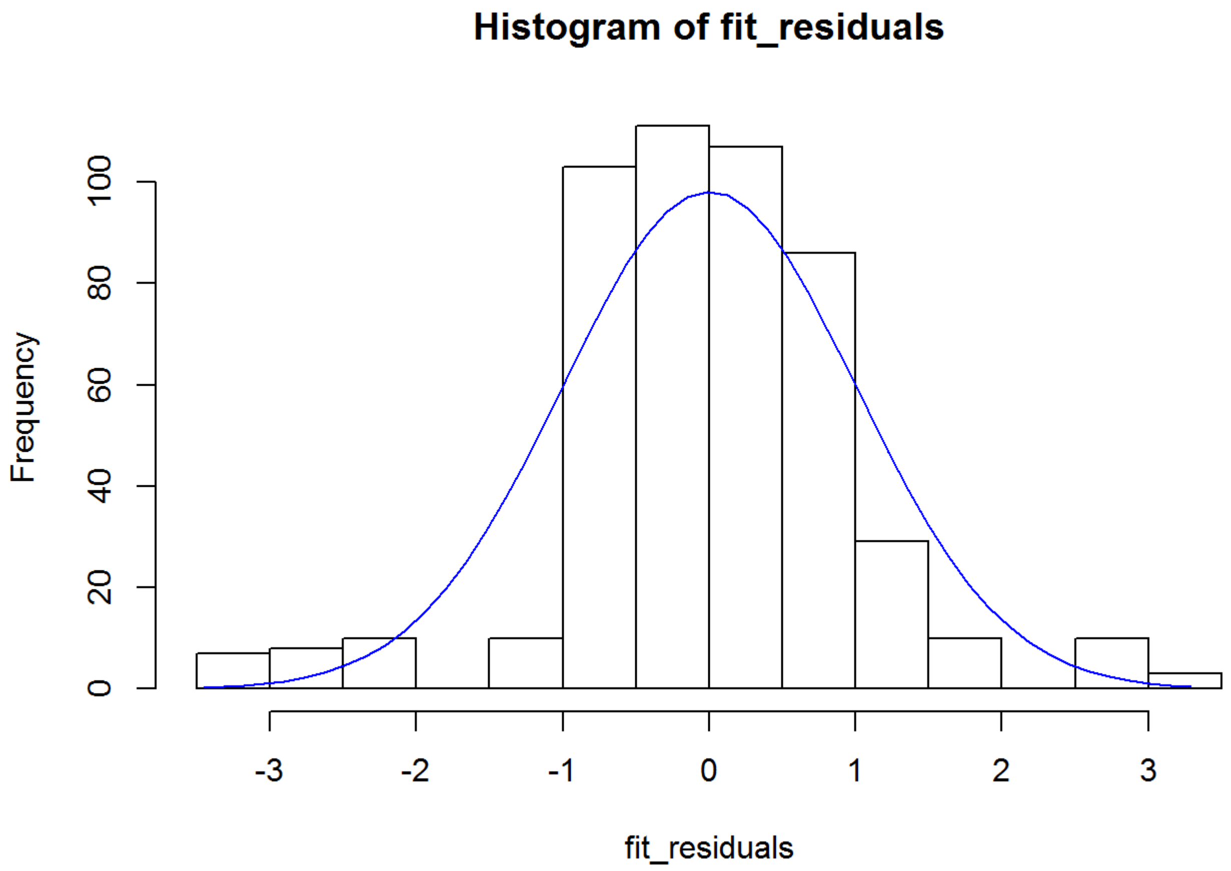
```
#assumptions test
fit_residuals <- rstandard(fit1)
plot(fit1$fitted.values, fit_residuals, pch = 16, main = "Standardized Re
sidual Plot", xlab = "Fitted Sales (100s)", ylab = "Standardized Residuals")
abline(0,0, lty=2, col="red")
```



```
qqnorm(fit_residuals, main = "Normal Probability Plot", xlab = "Normal Scores
", ylab = "Standardized Residuals")
qqline(fit_residuals, col = "red")
```



```
h <- hist(fit_residuals)
x <- fit_residuals
xfit <- seq(min(x), max(x), length = 50)
yfit <- dnorm(xfit, mean = mean(x), sd = sd(x))
yfit <- yfit*diff(h$mids[1:2])*length(x)
lines(xfit, yfit, col="blue")
```



```
shapiro.test(fit_residuals)
```

```
## 
##  Shapiro-Wilk normality test
## 
## data:  fit_residuals
## W = 0.94451, p-value = 1.231e-12
```