---
title: "Regression"
output: html_notebook

The data set used in this notebook is named "Bike Sharing Dataset" and was acquired from the UCI Machine Learning Repository.

The data set can be found here: https://archive.ics.uci.edu/ml/datasets/Bike+Sharing+Dataset

It contains information on the renting of bicycles, broken down by hours of the day and including information pertaining to the hour.

The objective of using this data set is to create a linear regression model to predict total number of bikes rented in a given hour.

```{r}

bikeData <- read.csv("hour.csv", header=TRUE)

sapply(bikeData, function(x) sum(is.na(x)))

bikeData <- bikeData[(complete.cases(bikeData)),]
sum(is.na(bikeData))

set.seed(12345)

sample <- sample(c(TRUE,FALSE), nrow(bikeData), replace=TRUE, prob=c(0.8,0.2))
train <- bikeData[sample, ]
test <- bikeData[!sample, ]

print("num rows: ")
nrow(train)
print("num cols: ")
ncol(train)
summary(train)
print("Correlation between hour and total count: ")
cor(train$hr, train$cnt)
print("Correlation between season and total count: ")
cor(train$season, train$cnt)

boxplot(train$cnt~train$hr)
boxplot(train$cnt~train$season)

lm1 <- lm(cnt ~ hr, data = train)
par(mfrow=c(2,2))
plot(lm1)
par(mfrow=c(1,1))
summary(lm1)

```
Simple Linear Regression:

This simple model is inadequate. While the variable hr has a good p value for cnt it woefully under performs in terms of R-Squared. This means that, while hr may be a decent predictor of cnt, it can not be used exclusively to predict the value of cnt. In summation, this model is too simple to be seriously considered.

Simple Linear Model Residuals:
Analyzing the four graphs created by plotting the residuals we can surmise this model is an ill fit for the data. Looking at the Q-Q graph we can observe that the data points form a line that curves significantly towards the end. This implies that the data contains values that are too extreme and the model was unable to handle them. The scale-location graph shows that the residuals are not evenly spread. However, according to the final graph, there appears to be no outliers affecting the regression model.

```{R}

lm2 <- lm(cnt ~ .- dteday - registered - casual - instant, data = train)
par(mfrow=c(2,2))
plot(lm2)
par(mfrow=c(1,1))
summary(lm2)

lm3 <- lm(cnt ~ poly(yr, season, hr, atemp, holiday, weekday, hum, windspeed),
data = train)
par(mfrow=c(2,2))
plot(lm3)
par(mfrow=c(1,1))
summary(lm3)

```
Comparing all three models:
Of the three models, the multiple linear regression has the most promising results with the highest R squared value. While the third model with polynomial regression has a comparable R-squared value its residual plots are disconcerting, in particular the residuals vs leverage graph, whith values heavily skewed. However, both models are superior to the simple linear regression model.

```{R}

pred1 <- predict(lm1, newdata=test)
pred2 <- predict(lm2, newdata=test)
pred3 <- predict(lm3, newdata=test)

correlation1 <- cor(pred1, test$cnt)
print("Model 1: ")
print(paste("Correlation: ", correlation1))
mse1 <- mean((pred1 - test$cnt)^2)
print(paste("MSE: ", mse1))
rmse1 <- sqrt(mse1)
print(paste("RMSE: ", rmse1))
```

```
correlation2 <- cor(pred2, test$cnt)
print("Model 2: ")
print(paste("Correlation: ", correlation2))
mse2 <- mean((pred2 - test$cnt)^2)
print(paste("MSE: ", mse2))
rmse2 <- sqrt(mse2)
print(paste("RMSE: ", rmse2))

correlation3 <- cor(pred3, test$cnt)
print("Model 3: ")
print(paste("Correlation: ", correlation3))
mse3 <- mean((pred3 - test$cnt)^2)
print(paste("MSE: ", mse3))
rmse3 <- sqrt(mse3)
print(paste("RMSE: ", rmse3))
```

Analyzing Prediction Results:
After analyzing the results of all three models only the second two models are suitable for prediction. Of the three, the highest correlation between model and data is only 0.62 which is just barely on the cusp of strong correlation. Obviously, the simple regression model has the weakest correlation and highest MSE, likely due to its simplicity. The third model with polynomial regression marginally beat the second model.