Question 8.1

Describe a situation or problem from your job, everyday life, current events, etc., for which a linear regression model would be appropriate. List some (up to 5) predictors that you might use.

I think that linear regression would be quite valuable when it comes to running a business and for looking at trends and making forecasts. A more specific example would be that if I were running a company and I had sales growth data over time, then I could try and forecast future sales using linear regression. When it comes to actually implementing the model, some of the predictors that I might use would be: the number of product sales, product pricing, product performance, the risk associated with the product, and advertising spending. By using all of these predictors, I might be able to forecast future revenue.

Question 8.2

Using crime data from http://www.statsci.org/data/general/uscrime.txt (file uscrime.txt, description at http://www.statsci.org/data/general/uscrime.html), use regression (a useful R function is lm or glm) to predict the observed crime rate in a city with the following data:

M = 14.0
So = 0
Ed = 10.0
Po1 = 12.0
Po2 = 15.5
LF = 0.640
M.F = 94.0
Pop = 150
NW = 1.1
U1 = 0.120
U2 = 3.6
Wealth = 3200
Ineq = 20.1
Prob = 0.04
Time = 39.0

Show your model (factors used and their coefficients), the software output, and the quality of fit.

Note that because there are only 47 data points and 15 predictors, you'll probably notice some overfitting. We'll see ways of dealing with this sort of problem later in the course.

All the code and output for this question can be found in appendix 8.2. Additionally, for this question, I decided to use the lm() function, and I began by using that function to fit a linear regression model with all 15 predictors. When I used this model to predict the crime value for the new data point provided above, I got a value of 155. This was suspicious because it is less than half of the lowest value for crime from the dataset (342). Because of this result, I thought that there might be a lot of "noise" being generated from some insignificant predictors in the 15 used.

When I examined the p-values, I was able to confirm this as a lot of them were high (greater than 0.10).

I wanted to see if the crime value I obtained would be different if I were to create a model using only the factors whose p-values were less than 0.10. When I refitted using only the select few predictors (M, Ed, Po1, U2, Ineq, and Prob), I observed that they all had p-values of less than 0.05. The new predicted value for crime changed as well from 342 to 1304.245. The models used and software output for both approaches are as follows:

The Original Model (15 predictors)

```
## Call:
## lm(formula = Crime ~ ., data = data)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -395.74  -98.09   -6.69  112.99  512.67

## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept) -5.984e+03  1.628e+03  -3.675 0.000893 ***
## M            8.783e+01  4.171e+01   2.106 0.043443 *
## So          -3.803e+00  1.488e+02  -0.026 0.979765
## Ed           1.883e+02  6.209e+01   3.033 0.004861 **
## Po1          1.928e+02  1.061e+02   1.817 0.078892 .
## Po2         -1.094e+02  1.175e+02  -0.931 0.358830
## LF          -6.638e+02  1.470e+03  -0.452 0.654654
## M.F          1.741e+01  2.035e+01   0.855 0.398995
## Pop         -7.330e-01  1.290e+00  -0.568 0.573845
## NW           4.204e+00  6.481e+00   0.649 0.521279
## U1          -5.827e+03  4.210e+03  -1.384 0.176238
## U2           1.678e+02  8.234e+01   2.038 0.050161 .
## Wealth       9.617e-02  1.037e-01   0.928 0.360754
## Ineq         7.067e+01  2.272e+01   3.111 0.003983 **
## Prob        -4.855e+03  2.272e+03  -2.137 0.040627 *
## Time        -3.479e+00  7.165e+00  -0.486 0.630708
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

The New Model (6 Predictors)

```
## Call:
## lm(formula = Crime ~ M + Ed + Po1 + U2 + Ineq + Prob, data = data)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -470.68  -78.41  -19.68  133.12  556.23

## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) -5040.50     899.84  -5.602 1.72e-06 ***
```

```
## M             105.02      33.30   3.154  0.00305 **
## Ed            196.47      44.75   4.390 8.07e-05 ***
## Po1           115.02      13.75   8.363 2.56e-10 ***
## U2             89.37      40.91   2.185  0.03483 *
## Ineq           67.65      13.94   4.855 1.88e-05 ***
## Prob        -3801.84    1528.10  -2.488  0.01711 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Now, when it comes to assessing the quality of the fit, I first looked at the R^2 value for the model with 15 predictors. The value was 0.8031. In comparison, the value for the refitted model was 0.7659. To me, this indicates that having the insignificant factors in the model causes some overfitting. Since we have learned that taking measurements on training data can lead to overfitting too, I also decided to utilize a 10-fold cross-validation approach. I first calculated the R^2 value on the 15-predictor model with cross-validation, and I obtained a value of 0.6087746. This is a significant drop off from the 0.8031 value that was obtained without cross-validation, which indicates that there might be a lot of overfitting going on. Furthermore, I also obtained the R^2 value with cross-validation for the refitted model. There, I obtained a value of 0.7286619. Because it is lower than the original value of 0.7659, it indicates that there might be some overfitting going on in this situation too.

Appendix

Question 8.2

```
#Load libary
library(stats); library(caret)

## Loading required package: ggplot2

## Loading required package: lattice

#Load Data
data <- read.table("C:\\Users\\User\\OneDrive\\Desktop\\Data 8.2\\uscrime.txt
", stringsAsFactors = F, header = T)

#Baseline Model
crime_bm <- lm(Crime ~., data)
summary(crime_bm)

##
## Call:
## lm(formula = Crime ~ ., data = data)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -395.74  -98.09   -6.69  112.99  512.67
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept) -5.984e+03  1.628e+03  -3.675 0.000893 ***
## M            8.783e+01  4.171e+01   2.106 0.043443 *
## So          -3.803e+00  1.488e+02  -0.026 0.979765
## Ed           1.883e+02  6.209e+01   3.033 0.004861 **
## Po1          1.928e+02  1.061e+02   1.817 0.078892 .
## Po2         -1.094e+02  1.175e+02  -0.931 0.358830
## LF          -6.638e+02  1.470e+03  -0.452 0.654654
## M.F          1.741e+01  2.035e+01   0.855 0.398995
## Pop         -7.330e-01  1.290e+00  -0.568 0.573845
## NW           4.204e+00  6.481e+00   0.649 0.521279
## U1          -5.827e+03  4.210e+03  -1.384 0.176238
## U2           1.678e+02  8.234e+01   2.038 0.050161 .
## Wealth       9.617e-02  1.037e-01   0.928 0.360754
## Ineq         7.067e+01  2.272e+01   3.111 0.003983 **
## Prob        -4.855e+03  2.272e+03  -2.137 0.040627 *
## Time        -3.479e+00  7.165e+00  -0.486 0.630708
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 209.1 on 31 degrees of freedom
## Multiple R-squared:  0.8031, Adjusted R-squared:  0.7078
## F-statistic: 8.429 on 15 and 31 DF,  p-value: 3.539e-07
```

```r
#Test data
test <- data.frame(M = 14.0, So = 0, Ed = 10.0, Po1 = 12.0, Po2 = 15.5,
                       LF = 0.640,
                       M.F = 94.0,
                       Pop = 150,
                       NW = 1.1,
                       U1 = 0.120,
                       U2 = 3.6,
                       Wealth = 3200,
                       Ineq = 20.1,
                       Prob = 0.04,
                       Time = 39.0)

model_pred1 <- predict(crime_bm, test)
model_pred1

##        1
## 155.4349

#Adjust the crime_bm model and keep predictors with p < 0.1
crime_adj <- lm(Crime ~ M + Ed + Po1 + U2 + Ineq + Prob, data)
summary(crime_adj)

##
## Call:
## lm(formula = Crime ~ M + Ed + Po1 + U2 + Ineq + Prob, data = data)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -470.68  -78.41  -19.68  133.12  556.23
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) -5040.50     899.84  -5.602 1.72e-06 ***
## M             105.02      33.30   3.154  0.00305 **
## Ed            196.47      44.75   4.390 8.07e-05 ***
## Po1           115.02      13.75   8.363 2.56e-10 ***
## U2             89.37      40.91   2.185  0.03483 *
## Ineq           67.65      13.94   4.855 1.88e-05 ***
## Prob        -3801.84    1528.10  -2.488  0.01711 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 200.7 on 40 degrees of freedom
## Multiple R-squared:  0.7659, Adjusted R-squared:  0.7307
## F-statistic: 21.81 on 6 and 40 DF,  p-value: 3.418e-11

model_pred2 <- predict(crime_adj, test)
model_pred2
```

```
##        1
## 1304.245

#Model quality checks
model1_ctrl <- trainControl(method="cv", number = 10)
model1_cv <- train(Crime ~., data, trControl = model1_ctrl, method = "lm",
                   na.action = na.pass)
model1_cv

## Linear Regression
##
## 47 samples
## 15 predictors
##
## No pre-processing
## Resampling: Cross-Validated (10 fold)
## Summary of sample sizes: 43, 43, 42, 41, 42, 44, ...
## Resampling results:
##
##   RMSE      Rsquared   MAE
##   259.0246  0.6087746  208.3733
##
## Tuning parameter 'intercept' was held constant at a value of TRUE

model2_ctrl <- trainControl(method="cv", number = 10)
model2_cv <- train(Crime ~ M + Ed + Po1 + U2 + Ineq + Prob, data,
                   trControl = model2_ctrl, method = "lm",
                   na.action = na.pass)
model2_cv

## Linear Regression
##
## 47 samples
##  6 predictor
##
## No pre-processing
## Resampling: Cross-Validated (10 fold)
## Summary of sample sizes: 42, 41, 44, 43, 42, 42, ...
## Resampling results:
##
##   RMSE      Rsquared   MAE
##   208.6978  0.7286619  168.2955
##
## Tuning parameter 'intercept' was held constant at a value of TRUE
```