AnalyticsModeling_HW5

Fall 2024

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 you might use.

A linear regression model would be appropriate to predict the earning potential / future salary of a person based off their educational background. Some predictors for this could include their highest level of education, what major they studied (if they attended college), and the year they graduated.

Question 8.2

Using crime data, use regression (a useful R function is `lm` or `glm`) to predict the observed crime rate in a city with the below data. Show your model (the factors and coefficients), the software output, and the quality of fit. Note that because there are only 47 data points and 15 predictors, there will be some overfitting and we'll learn how to handle that later in the course.

- `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

Below are the descriptions for each of the columns in the dataset.

- `M`: percentage of males aged 14-24 in total state population
- `So`: indicator for a southern state
- `Ed`: mean years of schooling of the population aged 25 and up
- `Po1`: per capita expenditure on police protection in 1960
- `Po2`: per capita expenditure on police protection in 1959
- `LF`: labor force rate of civilian urban males aged 14-24
- `M.F`: the number of males per 100 females
- `Pop`: the state population in 1960 in hundreds of thousands
- `U1`: the unemployment rate of urban males aged 14-24
- `U2`: the unemployment rate of urban males aged 35-39
- `Wealth`: the median value of transferable assets of family income
- `Ineq`: income inequality; the percentage of families earning below half the median income
- `Prob`: the probability of imprisonment; the ratio of number of commitments to number of offenses
- `Time`: the average time in months served by offenders in state prisons before their first release
- `Crime`: the crime rate; number of offenses per 100k population in 1960

```
# Load data
crime_df <- read.table("uscrime.txt",  stringsAsFactors = FALSE, header = TRUE)
head(crime_df, 5)
```

```
##      M So   Ed  Po1  Po2    LF  M.F Pop   NW    U1  U2 Wealth Ineq     Prob
## 1 15.1  1  9.1  5.8  5.6 0.510  95.0  33 30.1 0.108 4.1   3940 26.1 0.084602
## 2 14.3  0 11.3 10.3  9.5 0.583 101.2  13 10.2 0.096 3.6   5570 19.4 0.029599
## 3 14.2  1  8.9  4.5  4.4 0.533  96.9  18 21.9 0.094 3.3   3180 25.0 0.083401
## 4 13.6  0 12.1 14.9 14.1 0.577  99.4 157  8.0 0.102 3.9   6730 16.7 0.015801
## 5 14.1  0 12.1 10.9 10.1 0.591  98.5  18  3.0 0.091 2.0   5780 17.4 0.041399
##       Time Crime
## 1 26.2011   791
## 2 25.2999  1635
## 3 24.3006   578
## 4 29.9012  1969
## 5 21.2998  1234
```

```
summary(crime_df$Crime)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   342.0   658.5   831.0   905.1  1057.5  1993.0
```

First I need to load the data, train the initial linear model, and make a prediction of the crime rate for the test city:

```
# create the test city for which I want to predict the crime rate:
city <- 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
                   )

# train the linear regression model on all attributes
lm_model <- lm(formula = Crime ~.
               , data = crime_df
               )
summary(lm_model)
```

```
##
## Call:
## lm(formula = Crime ~ ., data = crime_df)
##
## 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
```

```
# predict the crime rate for the city using the model trained on all attributes
crime_preds_lm_model <- predict(lm_model, city)
crime_preds_lm_model
```

```
##        1
## 155.4349
```

The initial model predicted a crime rate of 155.4349. This value is suspicious since the minimum value of Crime in the data set is 342.0. This means there are likely some features in the model that are irrelevant.

Looking at the summary of the model, the output includes some information on how significant each attribute is to the model.

- Three asterisks represent a significance at the p < 0.001 level, a highly significant p-value
- Two asterisks represent significance at the p < 0.01 level
- one asterisk indicates significance at the p < 0.05 level
- One period indicates a significance at the p < 0.10 level

If we only include attributes that are significant, will the model perform better?

```
# train  new model with only significant attributes
new_lm_model <-  lm(formula = Crime ~ M+Ed+Po1+U2+Ineq+Prob
                , data = crime_df
                )
summary(new_lm_model)
```

```
##
## Call:
## lm(formula = Crime ~ M + Ed + Po1 + U2 + Ineq + Prob, data = crime_df)
##
## 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
```

```
# predict the crime rate for the city using the model trained on significant attributes
crime_preds_new_lm_model <- predict(new_lm_model, city)
crime_preds_new_lm_model
```

```
##        1
## 1304.245
```

This model made a crime prediction of 1304.245, which is much more in line with what I would expect given this data set. Additionally, the evaluation metrics have improved too:

- Adjusted R-squared grew from 0.7078 to 0.7307
- The p-value shrunk from 3.539e-07 to 3.418e-11

So it seems safe to say that the model performed better when I trimmed out some of the insignificant features.