Problem Set 1

API222: Big Data & Machine Learning

Brendan Hellweg

9/29/2021

Problem 1

Problem 1A Regression, Inference. This is a regression problem because we are measuring continuous, quantifiable variables. It is an inference problem because we are seeking to understand current conditions to make a decision, not to predict future values.

Problem 1B Classification, Prediction. This is a classification problem because the output of homelessness is a binary (yes, no). It is a prediction problem because we are estimating the likelihood of a future event.

Problem 1C Regression, Inference. This is a regression problem because we are dealing with quantifiable attributes (including dummy variables with 0 or 1 values). It is inference because it is measuring relative risk in existing populations.

Problem 2

Problem 2A False. Low flexibility optimizes for low variance.

Problem 2B True. Low flexibility models will have higher bias in these cases.

Problem 2C False. The inflexible model will fit the data best.

Problem 2D False. A model is parametric if f() can be finitely modeled, not if it has a decision rule. KNN models have decision rules but cannot necessarily be finitelty modeled.

Problem 3

As flexibility increases, variance generally increases due to overfitting (meaning that the model is overly influenced by sampling error in the training set), while bias decreases because the model is better able to reflect the pattern of the training data (and therefore minimize MSE_TR). As flexibility decreases, variance also decreases because there is less overfitting, but bias will increase because the model is less able to represent patterns in the training data. As such, there is an inverse relationship between bias and variance, and the best models will optimize for the "sweet spot" in the middle that minimizes MSE_TE.

Data Questions

Problem 1

```
nrow(schools)
```

```
## [1] 220
```

Problem 2

```
ncol(schools)
```

```
## [1] 16
```

Problem 3

```
sum(is.na(schools))
```

```
## [1] 74
```

Problem 4

The categorical variables are `district` and `municipality`.

Problem 5

```
mean(schools$score8,na.rm = T) %>%
  round(.,2)
```

```
## [1] 698.41
```

Problem 6

```
sd(schools$stratio) %>%
  round(.,2)
```

```
## [1] 2.28
```

Setup for Regression & KNN Section

For the next section, I will filter the dataset to remove rows with NA values and create treatment and test bins.

```
schools2 <- schools %>%
  drop_na() %>%
  select(-c(1,2))

schools_tr <- schools2 %>%
  head(.,n = (nrow(schools2)-35))

schools_te <- schools2 %>%
  tail(.,n = 35)
```

Problem 7

```
school_lm <- lm(score8~.,schools_tr)

mean((schools_te$score8 - predict.lm(school_lm, schools_te)) ^ 2) %>%
  round(.,2)
```

```
## [1] 78.21
```

Problem 8

```
school_lm2 <- lm(score8~stratio+lunch+exptot,schools_tr)

mean((schools_te$score8 - predict.lm(school_lm2, schools_te)) ^ 2) %>%
  round(.,2)
```

```
## [1] 146.63
```

Below is the analysis for parts 2-5 of this question.

```
print(school_lm)
```

```
##
## Call:
## lm(formula = score8 ~ ., data = schools_tr)
##
## Coefficients:
## (Intercept)       expreg   expspecial       expbil       expocc       exptot
##   3.909e+02   -1.433e-02   -5.952e-05    3.517e-06   -4.299e-04    1.175e-02
##     scratio      special        lunch      stratio       income       score4
##  -5.114e-01   -6.701e-01   -5.222e-01   -5.457e-01    1.227e+00    4.493e-01
##      salary      english
##   7.989e-02    1.542e-01
```

```
print(school_lm2)
```

```
##
## Call:
## lm(formula = score8 ~ stratio + lunch + exptot, data = schools_tr)
##
## Coefficients:
## (Intercept)      stratio        lunch       exptot
##  680.960976     0.408270    -1.151764     0.005313
```

```
cor(schools2$stratio,schools2$score8)
```

```
## [1] -0.3404961
```

The coefficient with the largest change in effect is `stratio`, which in the first model is associated with a moderate decline in 8th grade test performance, while in the second model it is associated with a moderate improvement in 8th grade test performance. In comparison, an increase in `lunch` continues to be associated with a decline in `score8`, though the coefficient roughly doubles in magnitude. The smallest coefficient change is for `exptot`, which changes from slightly negative to slightly positive.

In the first model, `stratio` has a coefficient of `-0.54`, while in the second model its coefficient is `0.41`. This gap results in a difference of `0.95` between the first and second model. The correlation between `stratio` and `score8` in the full dataset is `-0.34`, so it appears that the first model is directionally accurate.

One reason this change would occur is interaction effects and collinearities that skew the three-variable model. For instance, `stratio` is inversely correlated with `income` and `score4`, both of which are positively correlated with `score8`. It is positively correlated with `english`, which is also positively correlated with `score8`. As a result of their omission in the second model, these variables and their interaction effects will influence the relationships described in `school_lm2`.

One way to think about this result is that, if you hold the total expense and subsidized lunch populations of a school relatively constant, the larger classrooms tend to do better for reasons outside the scope of this analysis. Perhaps those larger classrooms tend to be populated by students who speak English more comfortably, making it easier to score on a test in English. Any definitive claim will take much more rigorous analysis.

Problem 9

```
pr <- knn(schools_tr,schools_te,schools_tr$score8,k = 2)
tb <- as.data.frame(pr) %>%
  mutate(pr = as.numeric(as.character(pr))) %>%
  mutate(scores = schools_te$score8) %>%
  mutate(sqerror = as.numeric(as.character((.$pr-.$scores)^2)))

mean(as.numeric(as.character(tb$sqerror))) %>%
  round(.,2)
```

```
## [1] 502.4
```

Note: This value has a random element due to the process of generating MSE_TE of a KNN model. Over several instances running this code block, I found values ranging from approx. `450` and `650`.

Problem 10

```r
pr <- knn(schools_tr,schools_te,schools_tr$score8,k = 10)
tb <- as.data.frame(pr) %>%
  mutate(pr = as.numeric(as.character(pr))) %>%
  mutate(scores = schools_te$score8) %>%
  mutate(sqerror = as.numeric(as.character((.$pr-.$scores)^2)))

mean(as.numeric(as.character(tb$sqerror))) %>%
  round(.,2)
```

```
## [1] 442.66
```

Note: This value has a random element due to the process of generating MSE_TE of a KNN model. Over several instances running this code block, I found values ranging from approx. 350 and 550.