Feature Engineering and Selection: A Practical Approach for Predictive Models

Max Kuhn and Kjell Johnson

2019-06-10

Contents

Preface

The goal of our previous work, *Applied Predictive Modeling*, was to elucidate a framework for constructing models that generate accurate predictions for future, yet-to-be-seen data. This framework includes pre-processing the data, splitting the data into training and testing sets, selecting an approach for identifying optimal tuning parameters, building models, and estimating predictive performance.

Chapter 1

Introduction

Statistical models have gained importance as they have become ubiquitous in modern society. They enable us by generating various types of predictions in our daily lives. For example, doctors rely on general rules derived from models that tell them which specific cohorts of patients have an increased risk of a particular ailment or event. A numeric prediction of a flight's arrival time can help understand if our airplane is likely to be delayed. In other cases, models are effective at telling us what is important or concrete. For example, a lawyer might utilize a statistical model to quantify the likelihood that potential hiring bias is occurring by chance or whether it is likely to be a systematic problem.

In each of these cases, models are created by taking existing data and finding a mathematical representation that has acceptable fidelity to the data. From such a model, important statistics can be estimated. In the case of airline delays, a prediction of the outcome (arrival time) is the quantity of interest while the estimate of a possible hiring bias might be revealed through a specific model parameter. In the latter case, the hiring bias estimate is usually compared to the estimated uncertainty (i.e., noise) in the data and a determination is made based on how uncommon such a result would be relative to the noise - a concept usually referred to as "statistical significance." This type of model is generally thought of as being *inferential*: a conclusion is reached for the purpose of understanding the state of nature. In contrast, the prediction of a particular value (such as arrival time) reflects an *estimation problem* where our goal is not necessarily to understand if a trend or fact is genuine but is focused on having the most accurate determination of

that value. The uncertainty in the prediction is another important quantity, especially to gauge the trustworthiness of the value generated by the model.

Some math, just for kicks:

One facet of sensitivity, specificity, and precision that is worth understanding is that they are *conditional* statistics. For example, sensitivity reflects the probability that an event is correctly predicted *given that a sample is truly an event*. The latter part of this sentence shows the conditional nature of the metric. Of course, the true class is usually unknown and, if it were known, a model would not be needed. In any case, if Y denotes the true class and P denotes the prediction, we could write sensitivity as $\Pr[P = \text{STEM}|\, Y = \text{STEM}]$.

The question that one really wants to know is: "if my value was predicted to be an event, what are the chances that it is truly is an event?" or $\Pr[\, Y = \text{STEM}|P = \text{STEM}]$. Thankfully, the field of Bayesian analysis (McElreath, 2015) has an answer to this question. In this context, Bayes' Rule states that

$$Pr[Y|P] = \frac{Pr[Y] \times Pr[P|Y]}{Pr[P]} = \frac{Prior \times Likelihood}{Evidence}.$$

Sensitivity (or specificity, depending on one's point of view) is the "likelihood" part of this equation. The *prior probability*, or *prevalence*, is the overall rate that we see events in the wild (which may be different from what was observed in our training set). Usually, one would specify the overall event rate before data are collected and use it in the computations to determine the unconditional statistics. For sensitivity, its unconditional analog is called the *positive predictive value* (PPV):

$$PPV = \frac{sensitivity \times prevalence}{(sensitivity \times prevalence) + ((1 - specificity) \times (1 - prevalence))}.$$

The *negative predictive value* (NPV) is the analog to specificity and can be computed as

$$NPV = \frac{specificity \times (1 - prevalence)}{((1 - sensitivity) \times prevalence) + (specificity \times (1 - prevalence))}.$$

```
## - Session info ---------------------------------------------------------
##  setting  value
##  version  R version 3.6.0 (2019-04-26)
##  os       macOS High Sierra 10.13.6
##  system   x86_64, darwin15.6.0
##  ui       X11
##  language (EN)
##  collate  en_US.UTF-8
##  ctype    en_US.UTF-8
##  tz       America/New_York
##  date     2019-06-10
##
## - Packages -------------------------------------------------------------
##  package     * version    date       lib
##  assertthat    0.2.1      2019-03-21 [1]
##  bookdown      0.9.2      2019-04-14 [1]
##  cli           1.1.0      2019-03-19 [1]
##  colorspace    1.4-1      2019-03-18 [1]
##  crayon        1.3.4      2017-09-16 [1]
##  digest        0.6.19     2019-05-20 [1]
##  dplyr         0.8.0.1    2019-02-15 [1]
##  evaluate      0.13       2019-02-12 [1]
##  ggplot2     * 3.1.1      2019-04-07 [1]
##  glue          1.3.1      2019-03-12 [1]
##  gtable        0.3.0      2019-03-25 [1]
##  htmltools     0.3.6      2017-04-28 [1]
##  knitr         1.23       2019-05-18 [1]
##  lazyeval      0.2.2      2019-03-15 [1]
##  magrittr      1.5        2014-11-22 [1]
##  munsell       0.5.0      2018-06-12 [1]
##  pillar        1.4.1      2019-05-28 [1]
##  pkgconfig     2.0.2      2018-08-16 [1]
##  plyr          1.8.4      2016-06-08 [1]
##  purrr         0.3.2      2019-03-15 [1]
##  R6            2.4.0      2019-02-14 [1]
##  Rcpp          1.0.1      2019-03-17 [1]
##  rlang         0.3.4.9003 2019-06-04 [1]
##  rmarkdown     1.12       2019-03-14 [1]
##  scales        1.0.0      2018-08-09 [1]
##  sessioninfo * 1.1.1.9000 2019-03-26 [1]
```

```
## stringi       1.4.3     2019-03-12 [1]
## stringr       1.4.0     2019-02-10 [1]
## tibble        2.1.1     2019-03-16 [1]
## tidyselect    0.2.5     2018-10-11 [1]
## withr         2.1.2     2018-03-15 [1]
## xfun          0.7       2019-05-14 [1]
## yaml          2.2.0     2018-07-25 [1]
## source
## CRAN (R 3.6.0)
## Github (rstudio/bookdown@5998ab4)
## CRAN (R 3.6.0)
## CRAN (R 3.6.0)
## CRAN (R 3.6.0)
## CRAN (R 3.6.0)
## CRAN (R 3.6.0)
## CRAN (R 3.6.0)
## CRAN (R 3.6.0)
## CRAN (R 3.6.0)
## CRAN (R 3.6.0)
## CRAN (R 3.6.0)
## CRAN (R 3.6.0)
## CRAN (R 3.6.0)
## CRAN (R 3.6.0)
## CRAN (R 3.6.0)
## CRAN (R 3.6.0)
## CRAN (R 3.6.0)
## CRAN (R 3.6.0)
## CRAN (R 3.6.0)
## CRAN (R 3.6.0)
## CRAN (R 3.6.0)
## Github (r-lib/rlang@6a232c0)
## CRAN (R 3.6.0)
## CRAN (R 3.6.0)
## Github (r-lib/sessioninfo@dfb3ea8)
## CRAN (R 3.6.0)
## CRAN (R 3.6.0)
## CRAN (R 3.6.0)
## CRAN (R 3.6.0)
## CRAN (R 3.6.0)
## CRAN (R 3.6.0)
```

```
##  CRAN (R 3.6.0)
##
## [1] /Library/Frameworks/R.framework/Versions/3.6/Resources/library
```

Errata and Version History

2019-04-xx

- blah blah
- yada yada

Bibliography

McElreath, R. (2015). *Statistical Rethinking: A Bayesian Course with Examples in R and Stan*. Chapman and Hall/CRC.