Predicting programs sold - new games

Eric Chang

3/8/2017

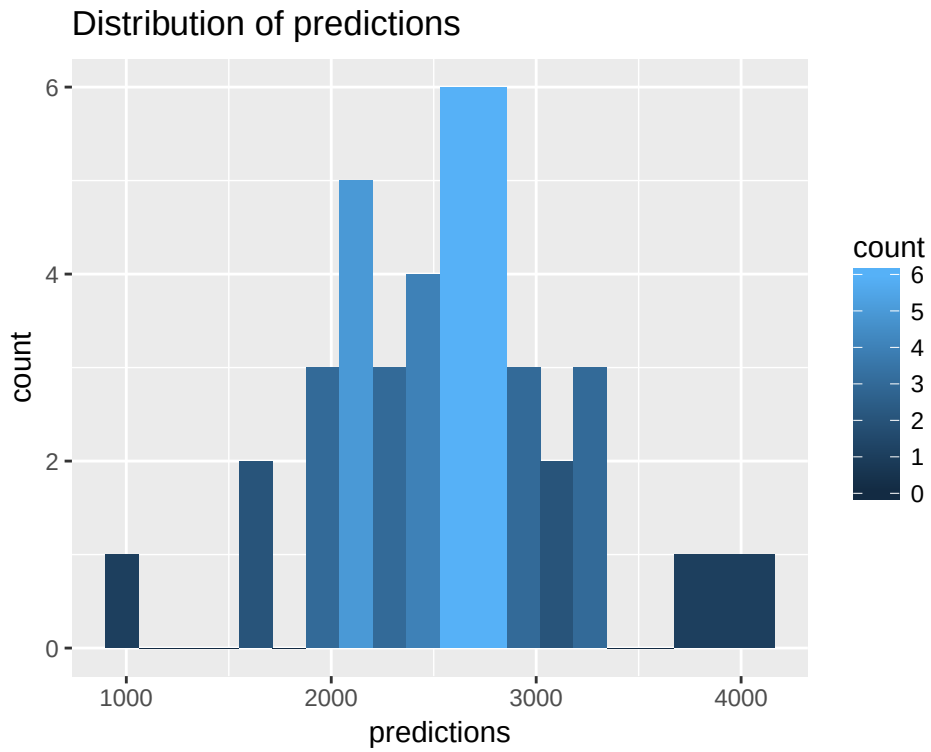
```
require(dplyr)
require(ggplot2)

# prepare data
source("featurize_data.R")

dat_lm <- filter(dat, !(programs_ordered == programs_sold)) %>%
          select(-programs_ordered, -time_kickoff, -percent_male) %>%
          filter(coach_pokey != 1) %>%  # take out old coach game
          select(-coach_pokey) %>%
          filter(!(month == "August" & day_of_month == 30 & opponent == "Nebraska")) %>%
          select(-opponent) %>%
          mutate(year = as.integer(as.character(year))) %>%
          filter(!month %in% c("August", "December"))

dat_lm$month <- droplevels(dat_lm$month)

#
fit_dat_lm <- lm(programs_sold ~ . + month*day_of_month,
                 select(dat_lm, -morning_kickoff) %>% slice(c(-17, -21)))

summary(fit_dat_lm)
```

```
##
## Call:
## lm(formula = programs_sold ~ . + month * day_of_month, data = select(dat_lm,
##     -morning_kickoff) %>% slice(c(-17, -21)))
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -626.97 -189.13  -41.58  224.19  602.68
##
## Coefficients:
##                                Estimate Std. Error t value Pr(>|t|)
## (Intercept)                  -2.490e+05  5.037e+04  -4.945 2.52e-05 ***
## monthOctober                  7.315e+02  2.710e+02   2.699 0.011148 *
## monthSeptember                1.954e+03  2.501e+02   7.816 8.06e-09 ***
## day_of_month                 -5.383e+00  1.518e+01  -0.355 0.725192
## year                          1.238e+02  2.506e+01   4.939 2.56e-05 ***
## homecoming                   -2.096e+01  1.717e+02  -0.122 0.903633
## hour                          1.256e+02  3.002e+01   4.185 0.000218 ***
## big_ten                       8.413e+02  1.314e+02   6.405 3.90e-07 ***
## monthOctober:day_of_month    -1.112e+01  1.798e+01  -0.619 0.540571
## monthSeptember:day_of_month -5.203e+01  1.923e+01  -2.706 0.010972 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
```

```
## Residual standard error: 318.3 on 31 degrees of freedom
## Multiple R-squared:  0.8247, Adjusted R-squared:  0.7738
## F-statistic:  16.2 on 9 and 31 DF,  p-value: 2.04e-09
```

Quick check - these are the same regression coefficients we shared yesterday.

```
preds <-
  fit_dat_lm$fitted.values %>%
    data.frame()
preds <- rename(preds, predictions = .)
ggplot(preds) +
  geom_histogram(aes(x=predictions, fill=..count..), bins = 20) +
  ggtitle("Distribution of predictions")
```



Fitted vals look good too, all between 1000 and 4000.

Make predictions with new data

```
source("create_test_data.R")
schedule2017_lm
```

```
##       month day_of_month year homecoming morning_kickoff hour big_ten
## 1 September            9 2017          0               1   11       0
## 2 September           16 2017          0               0   14       1
## 3   October            7 2017          0               0   12       1
## 4   October           14 2017          0               1   11       1
## 5  November            4 2017          0               0   17       0
## 6  November           11 2017          0               1   11       1
## 7  November           18 2017          0               0   14       1
```

```
predict(fit_dat_lm, schedule2017_lm)
```

```
##        1        2        3        4        5        6        7
## 3414.130 4230.532 3559.355 3318.170 2708.774 2758.565 3097.820
```