##Title & Authors Upworthy dataset evaluation of whether the length of magazines' titles affect the article click-rate.

##Keywords A/B test, ## ##Introduction —data source: While the Upworthy research archive from https://upworthy.natematias.com/index (https://upworthy.natematias.com/index) is a large A/B tests' open online dataset, and the dataset used for this assignment is requested from this Upworthy company.

—goals:

—methods involved:

variable analzing, mentioning the variable to be used:

For this project specifically, the focus will be on investigating the influence of different news headlines on article exposure with the use of various regression model. The dataset was first cleaned into a tidier version, followed by models analyzing the various result of naming the magazine with different title. The variables to be used(approximately) in this study are as follows: headline, eyecatcher_id, impression, clicks and significant. Headline as result variable, and the rest of the variable as predictors(Matias et al.).

##Data Cleaning

```
#Keep the useful data columns
 reduced_data <-
  rawdata %>%
  select(excerpt,
         headline,
         lede,
         slug,
         eyecatcher_id,
         clicks,
         impressions)

#Remove the na values
reduced_data <- na.omit(reduced_data)
```

```
#Create 4 variables which count the length of ecerpt, headline, lede and slug
reduced_data$excerpt_ct <- nchar(reduced_data$excerpt)
reduced_data$headline_ct <- nchar(reduced_data$headline)
reduced_data$lede_ct <- nchar(reduced_data$lede)
reduced_data$slug_ct <- nchar(reduced_data$slug)
```

```
#Adding the 5th variable -- click_rate = clicks/impressions
reduced_data$click_rate <- reduced_data$clicks/reduced_data$impressions
```

```
#Select only the useful columns, excluding the original string columns, clicks and im
pressions
 cleaned_data <-
  reduced_data %>%
  select(excerpt_ct,
         headline_ct,
         lede_ct,
         slug_ct,
         click_rate)
 head(cleaned_data)
```

```
## # A tibble: 6 x 5
##   excerpt_ct headline_ct lede_ct slug_ct click_rate
##        <int>       <int>   <int>   <int>      <dbl>
## 1         32          84     130      83     0.0491
## 2         32          84     130      83     0.0402
## 3         32          84     130      83     0.0356
## 4         32          43     294      50     0.0255
## 5         32          43     294      50     0.0342
## 6         32          43     294      50     0.0290
```

##Methodology(Data and Model)

```
#Construct a logistic regression model for propensity_score
propensity_score <- glm(click_rate ~ excerpt_ct+ headline_ct + lede_ct + slug_ct,
                        family = binomial,
                        data = cleaned_data)
```

```
## Warning in eval(family$initialize): non-integer #successes in a binomial glm!
```

```
#Add our forecast to our dataset
data <-
  augment(propensity_score,
          data = cleaned_data,
          type.predict = "response") %>%
  dplyr::select(-.resid, -.std.resid, -.hat, -.sigma, -.cooksd)

#Use forecast to create matches
data <-
  data %>%
  arrange(.fitted, click_rate)
```

In order to set up a 'treated' group, there has to be a barrier for the qualified click_rate to be considered as relatively high click rate among the group. While belows is a summary of the 'click_rate' column. As we can see from the summary: median of click_rate is 0.012837, 3rd quantile of click_rate is 0.020686.

```
summary(data$click_rate)
```

```
##     Min.  1st Qu.   Median     Mean  3rd Qu.     Max.
## 0.000000 0.007648 0.012837 0.016097 0.020686 0.136063
```

```
# Use a matching function from the arm package to pair the closest of the ones that w
ere not treated, to the one that was treated.
data$treated <-
  if_else(data$click_rate >= 0.020686, 1, 0)

data$treated <-
  as.integer(data$treated)

matches <- arm::matching(z = data$treated,
                         score = data$.fitted)

data <- cbind(data, matches)
```

```
# Now we reduce the dataset to just those that are matched
data_matched <-
  data %>%
  filter(match.ind != 0) %>%
  dplyr::select(-match.ind, -pairs, -treated)

head(data_matched)
```

```
##   excerpt_ct headline_ct lede_ct slug_ct  click_rate     .fitted     .se.fit
## 1         96           40     287      95 0.022305654 0.003235511 0.002680595
## 2         32           59     191      90 0.007550336 0.005545153 0.002463804
## 3         32           66     294      87 0.030349014 0.008207906 0.002379438
## 4         32           74     267      94 0.021818182 0.008536003 0.002309403
## 5         32           83      40      97 0.003956624 0.009654912 0.001857801
## 6         32           64     337      80 0.015468608 0.009828235 0.002166451
##   cnts
## 1    1
## 2    1
## 3    1
## 4    1
## 5    1
## 6    1
```

```
# Examining the 'effect' of being treated on average
# spend in the 'usual' way.

propensity_score_regression <-
  lm(click_rate ~ excerpt_ct+ headline_ct + lede_ct + slug_ct,
                data = data_matched)

huxtable::huxreg(propensity_score_regression)
```

	(1)
(Intercept)	0.019 ***
	(0.001)
excerpt_ct	0.000 ***
	(0.000)
headline_ct	0.000 ***
	(0.000)
lede_ct	0.000
	(0.000)
slug_ct	-0.000 ***

	(0.000)
N	10104
R2	0.007
logLik	28242.152
AIC	-56472.304

*** p < 0.001; ** p < 0.01; * p < 0.05.

##Results ##Discussion weakness: eyecatcher column was deleted ##References Work Cited(to be continued) Matias, J. Nathan, et al. The Upworthy Research Archive. upworthy.natematias.com/index. ##Appendix(Optional)