Outline, Review and Ethics

Jameson Watts, Ph.D.
1/16/2020

Agenda

1. Course Overview
2. Review of Multiple Regression
3. Ethics of data in math
4. Ethics of data in policy

But first…

Rescheduling Saturday the 25th?

Setup

```
knitr::opts_chunk$set(echo = TRUE, message = FALSE, warning = FALSE)
library(tidyverse)
source('theme.R')
wine = read_rds("../resources/wine.rds")
```

Course Overview

Expectations and assignments

1. Data Camp Assignments
2. R vs. Python
3. Exams
4. Modeling Project

Review of Multiple Regression

Basic model

```
library(moderndive)
wine <- wine %>% mutate(bordeaux=(province=="Bordeaux"))
get_regression_table(lm(price ~ points, data = wine))
```

term	estimate	std_error	statistic	p_value	lower_ci	upper_ci
intercept	-489.251	3.969	-123.278	0	-497.029	-481.472
points	5.920	0.045	132.312	0	5.832	6.008

Multiple regression

```
get_regression_table(lm(price ~ points+bordeaux, data = wine))
```

term	estimate	std_error	statistic	p_value	lower_ci	upper_ci
intercept	-491.883	3.971	-123.863	0	-499.667	-484.100
points	5.946	0.045	132.842	0	5.858	6.034
bordeauxTRUE	8.703	0.661	13.170	0	7.408	9.999

Model diagnostics on full data set

```
get_regression_summaries(lm(price ~ points, data = wine))
```

r_squared	adj_r_squared	mse	rmse	sigma	statistic	p_value	df
0.164	0.164	1578.136	39.72576	39.726	17506.39	0	2

```
get_regression_summaries(lm(price ~ points+bordeaux, data = wine))
```

r_squared	adj_r_squared	mse	rmse	sigma	statistic	p_value	df
0.165	0.165	1575.905	39.69766	39.698	8852.104	0	3

Split sample using Caret

```
library(caret)
set.seed(5004) #for reproducibility
train_index <- createDataPartition(wine$price, times = 1, p = 0.8, list = FALSE)
train <- wine[train_index, ]
test <- wine[-train_index, ]

m1 <- lm(price~points, data = train)
m2 <- lm(price~points+bordeaux, data = train)
```

Comparing RMSE

```
get_regression_points(m1, newdata = test) %>%
  drop_na(residual) %>%
  mutate(sq_residuals = residual^2) %>%
  summarize(rmse = sqrt(mean(sq_residuals)))
```

rmse
47.44972

```
get_regression_points(m2, newdata = test) %>%
  drop_na(residual) %>%
  mutate(sq_residuals = residual^2) %>%
  summarize(rmse = sqrt(mean(sq_residuals)))
```

rmse
47.41283

What about an interaction?

```
m3 <- lm(price~points*bordeaux, data = train)
get_regression_table(m3)
```

term	estimate	std_error	statistic	p_value	lower_ci	upper_ci
intercept	-464.134	4.261	-108.924	0	-472.485	-455.782
points	5.633	0.048	117.288	0	5.539	5.727
bordeauxTRUE	-669.904	19.716	-33.977	0	-708.547	-631.260
points:bordeauxTRUE	7.698	0.224	34.411	0	7.259	8.136

```
get_regression_points(m3, newdata = test) %>%
  drop_na(residual) %>%
  mutate(sq_residuals = residual^2) %>%
  summarize(rmse = sqrt(mean(sq_residuals)))
```

rmse
47.1951

So what is machine learning?

Next steps…

Definition: using data to find a function that minimizes prediction error.

- Feature Engineering
- Variable Selection
- Cross validation
- Classification
    - Confusion matrix
    - ROC curves

Ethics of data

The math of it…

Suppose I'm trying to predict gender based on height. We start by defining the outcome and predictors and creating training and test data.

```
library(dslabs)
data(heights)
y <- heights$sex
x <- heights$height
set.seed(5004)
test_index <- createDataPartition(y, times = 1, p = 0.5, list = FALSE)
test_set <- heights[test_index, ]
train_set <- heights[-test_index, ]
```

Note: this vignette is adapted from this book

Guessing.

Let's start by developing the simplest possible machine algorithm: guessing the outcome.

```
y_hat <- sample(c("Male", "Female"), length(test_index), replace = TRUE) %>%
  factor(levels = levels(test_set$sex))
```

The overall accuracy is simply defined as the overall proportion that is predicted correctly:

```
mean(y_hat == test_set$sex)
```

```
## [1] 0.4933333
```

Let’s do better…

```
heights %>% group_by(sex) %>% summarize(mean(height), sd(height))
```

sex	mean(height)	sd(height)
Female	64.93942	3.760656
Male	69.31475	3.611024

Predict male if within 2 standard deviations

```
y_hat <- ifelse(x > 62, "Male", "Female") %>%
  factor(levels = levels(test_set$sex))

mean(y == y_hat)
```

```
## [1] 0.7933333
```

The accuracy goes up from 0.50 to about 0.80!!

Let’s optimize

```
cutoff <- seq(61, 70)
accuracy <- map_dbl(cutoff, function(x){
  y_hat <- ifelse(train_set$height > x, "Male", "Female") %>%
    factor(levels = levels(test_set$sex))
  mean(y_hat == train_set$sex)
})
```

```
max(accuracy)
```

```
## [1] 0.847619
```

which is much higher than 0.5. The cutoff resulting in this accuracy is:

```
best_cutoff <- cutoff[which.max(accuracy)]
best_cutoff
```

```
## [1] 65
```

How does it do on the test data?

```
y_hat <- ifelse(test_set$height > best_cutoff, "Male", "Female") %>%
  factor(levels = levels(test_set$sex))
y_hat <- factor(y_hat)
mean(y_hat == test_set$sex)
```

```
## [1] 0.8057143
```

Not quite as good as the training set, but pretty good nonetheless.

…but does this make sense?

Confusion matrix

```
table(predicted = y_hat, actual = test_set$sex)
```

```
##           actual
## predicted Female Male
##    Female     63   46
##    Male       56  360
```

what do you see?

Accuracy by sex

```
test_set %>%
  mutate(y_hat = y_hat) %>%
  group_by(sex) %>%
  summarize(accuracy = mean(y_hat == sex))
```

sex	accuracy
Female	0.5294118
Male	0.8866995

There is an imbalance in the force! We are literally calling almost half of the females male!

So why is the overall accuracy so high then?

Moral of the story

…too many men.

Other ethical issues

- Demographic data
- Profit optimizing
- Autonomous cars
- Recommendation engines
- Criminal sentencing
- Choice of classification model
- Killer robots

Reasonable people will disagree over subtle matters of right and wrong… thus, the important part of data ethics is committing to *consider* the ethical consequences of your choices.

The difference between “regular” ethics and data ethics is that algorithms scale really easily. Thus, seemingly small decisions can have wide-ranging impact.

with my friend Jeff Gaus.

Ethics policy and technology