Feature Engineering and Review

K Arnold

Logistics

- Project: outline is posted
- Midterm Quiz posted

Plan:

- TODAY: review
- Wed: decision trees
- Fri: lab about **overfitting**

General Hints

- Visualization:
  - What *glyph* represents each *observation*?
  - What *attributes*/aesthetics does that glyph have? (x, y, width, color, ...)
  - What *controls* each aesthetic? ("each party has a *y* position", ...)
- Wrangling
  - What does the *input* look like? (Translate the first row into a data sentence in English.)
  - What does the *output* need to look like? (Again, write a sentence.)
  - What *sequence of steps* needs to happen? (e.g., `filter-group_by-summarize-arrange`)
- Modeling
  - What *quantity* are you trying to predict?
  - What *error measure* will tell you the prediction is good / bad?
  - What *features* can help you make that prediction?

Q&A

> Recipes vs Data Wrangling Pipelines?

- recipe (from `recipes` package) = data wrangling pipeline
  - ... that can be easily applied to new data (e.g., *test set*)
  - ... that can have learnable state (like ranges of data values)

> Why did the prediction on the example test set house come out the same when we re-scaled the data?

- We'd applied *exactly the same* transformation to the example house as we did to the training data.
- Linear regression is *linear*: it doesn't care what units the data is in.
- (So the specific *range* didn't matter.)

> Are we still going to work in cohorts/teams?

```
library(tidymodels)
data(ames, package = "modeldata")
ames <- ames %>%
  filter(Gr_Liv_Area < 4000, Sale_Condition == "Normal") %>%
  mutate(across(where(is.integer), as.double))
```

```
set.seed(10) # Seed the random number generator
ames_split <- initial_split(ames, prop = 2/3) # Split our data randomly
ames_train <- training(ames_split)
ames_test <- testing(ames_split)
```

We'll use one example home from the test set.

```
example_home <- ames_test %>% slice(1)
example_home %>% select(Gr_Liv_Area, Sale_Price)
```

```
## # A tibble: 1 x 2
##   Gr_Liv_Area Sale_Price
##         <dbl>      <dbl>
## 1        1656     215000
```

Recipes

```
ames_recipe <-
  recipe(
    Sale_Price ~ Latitude + Longitude + Neighborhood + Year_Sold + Gr_Liv_Area,
    data = ames_train
  ) %>%
  step_other(Neighborhood) %>%
  step_dummy(all_nominal()) %>%
  #step_interact(~ starts_with("Neighborhood_") : Year_Sold) %>%
  step_center(all_predictors()) %>%
  step_scale(all_predictors()) %>%
  step_log(Sale_Price, base = 10) %>%
  prep(trainig = ames_train, retain = TRUE)
ames_recipe %>% summary()
```

```
## # A tibble: 11 x 4
##   variable                  type    role      source
##   <chr>                     <chr>   <chr>     <chr>
## 1 Latitude                  numeric predictor original
## 2 Longitude                 numeric predictor original
## 3 Year_Sold                 numeric predictor original
## 4 Gr_Liv_Area               numeric predictor original
## 5 Sale_Price                numeric outcome   original
## 6 Neighborhood_College_Creek numeric predictor derived
## # … with 5 more rows
```

```
ames_recipe %>% bake(new_data = ames_train)
```

```
## # A tibble: 1,608 x 11
##   Latitude Longitude Year_Sold Gr_Liv_Area Sale_Price Neighborhood_Co… Neighborhood_Ol…
##      <dbl>     <dbl>     <dbl>       <dbl>      <dbl>            <dbl>            <dbl>
## 1     1.07     0.876      1.61      -1.20        5.02           -0.329           -0.295
## 2     1.05     0.890      1.61      -0.317       5.24           -0.329           -0.295
## 3     1.51     0.145      1.61       0.298       5.28           -0.329           -0.295
## 4     1.51     0.146      1.61       0.247       5.29           -0.329           -0.295
## 5     1.42     0.140      1.61       0.657       5.28           -0.329           -0.295
## 6     1.38     0.221      1.61       0.351       5.25           -0.329           -0.295
## # … with 1,602 more rows, and 4 more variables: Neighborhood_Edwards <dbl>,
## #   Neighborhood_Gilbert <dbl>, Neighborhood_Sawyer <dbl>, Neighborhood_other <dbl>
```