Cross-Validation Review

K Arnold

Logistics

- Today: review, continue lab10
- Wednesday: Wrangling and Modeling in Python (*classification*).
  - *lab10 due*.
  - *Project milestone 1*: Data
- Friday: Classification *lab*
- Next Monday: (probably) brief notes about inference
- Next Tuesday: Discussion 11 due (fairness in classification)

Midterm notes

- Remember grammar of graphics: each aesthetic maps to one *variable*.
- Think about the shape of your data!
- Don't wait for the last minute.

(*Academic integrity note.*)

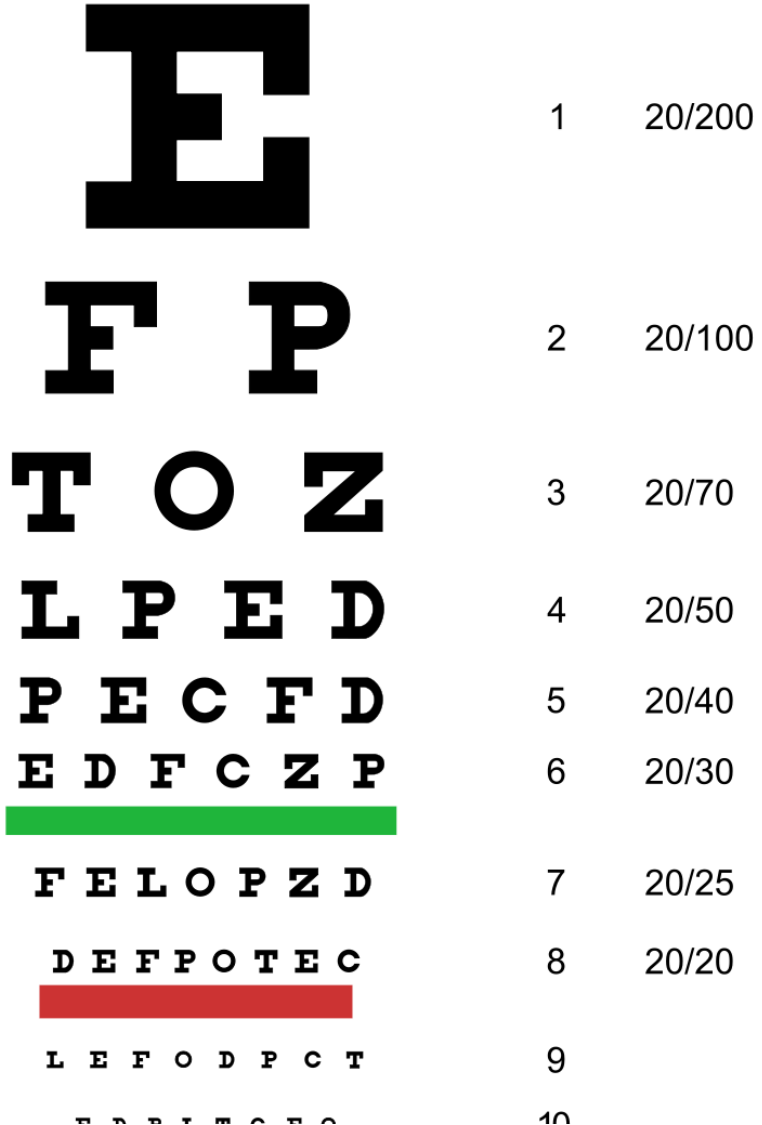
Feedback

Common themes in your comments:

- Modeling is fun
- Cross Validation is cool... but still confusing
- All that code is *really* confusing

Indeed. Let's review.

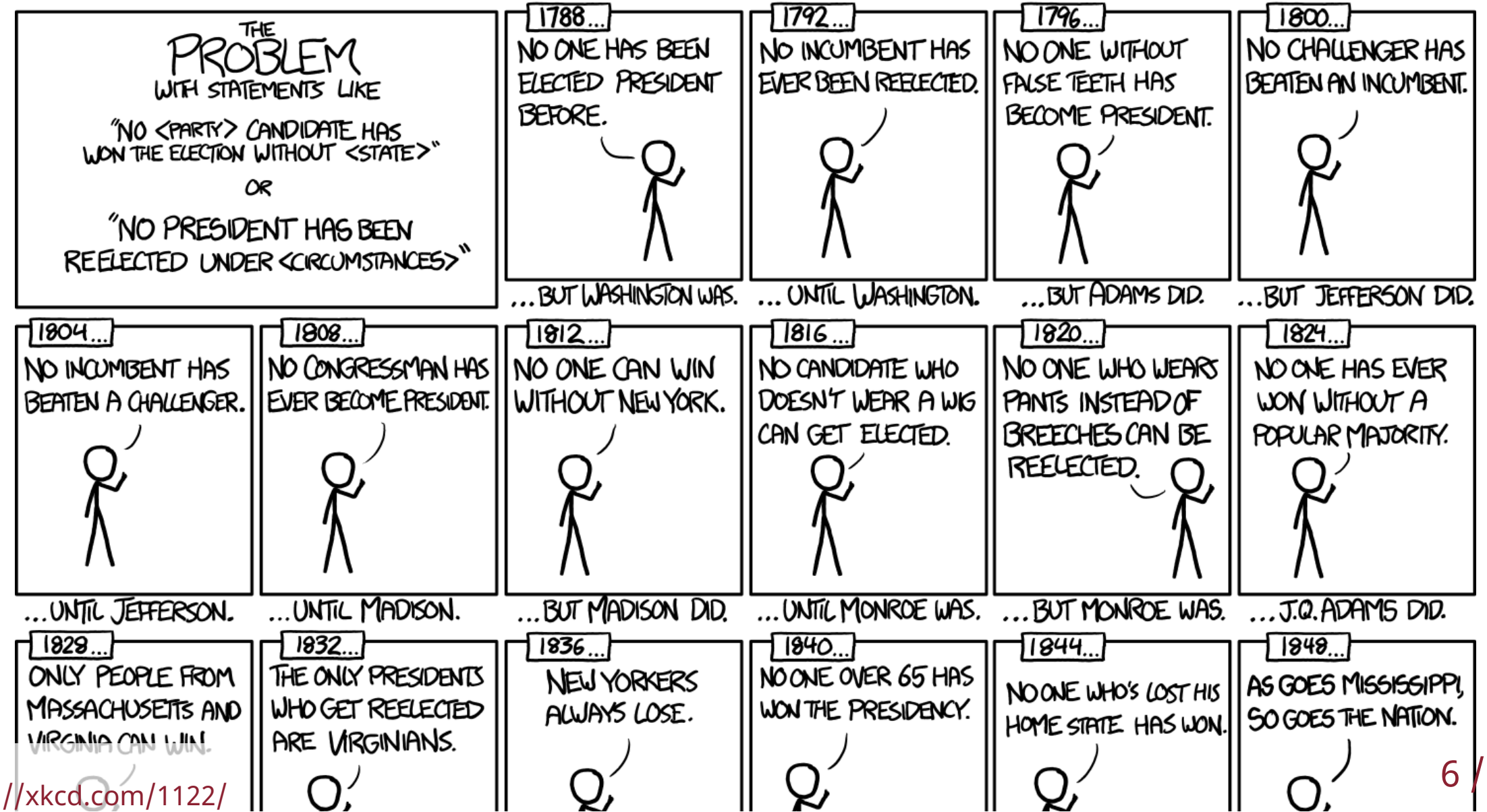
Why train-test split? Memorizing the eye chart



Snellen chart on Wikimedia, CC-BY-SA

Analogy by Clem Wang

Overfitting



https://xkcd.com/1122/

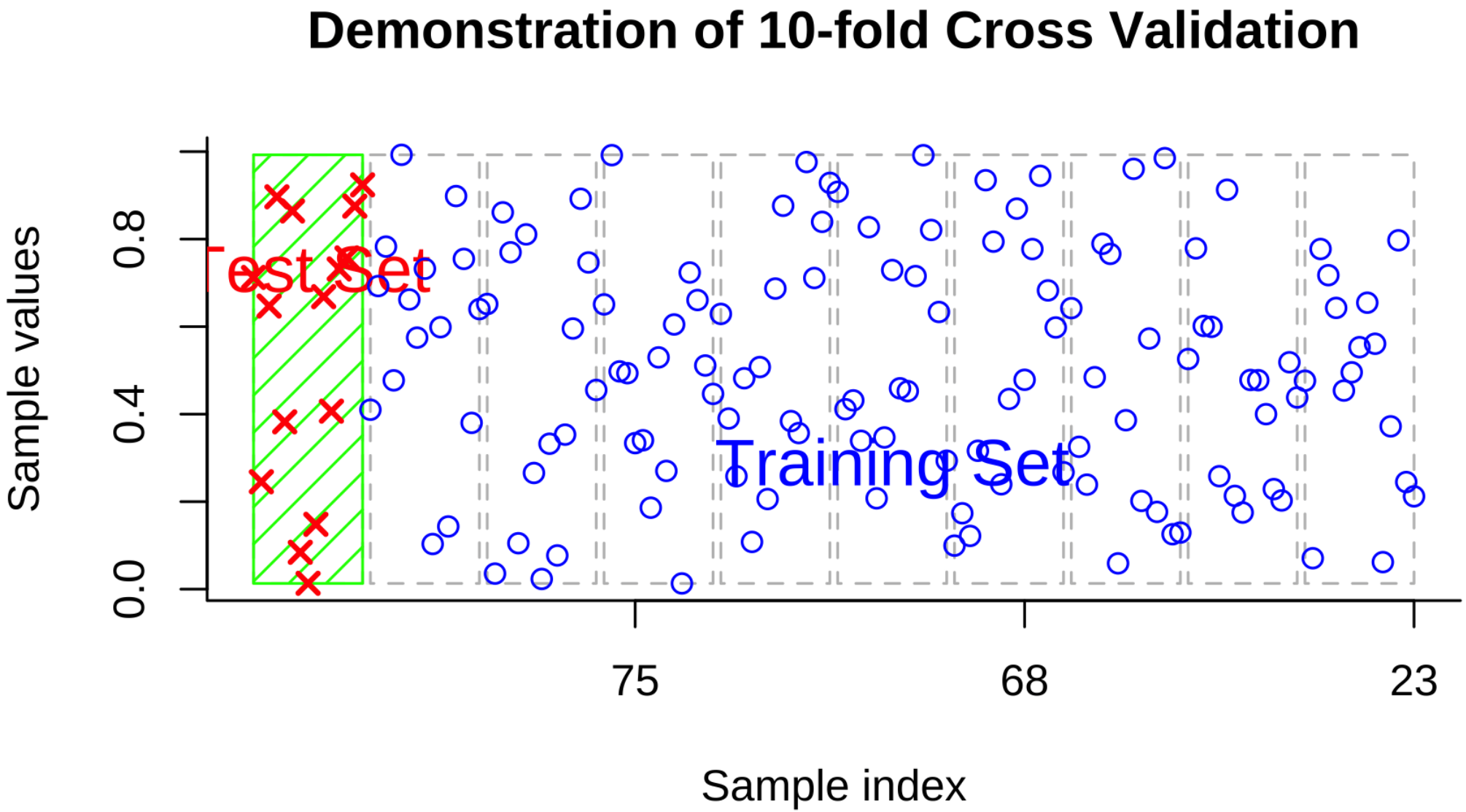
Why Cross-Validation?

Puzzle

- We want to pick the model that works best on *unseen* data
- ... but as soon as we try one model, **we've peeked at the data!**

Solution

- Divide training data into *V* piles (e.g., 10)
- Hide one pile from yourself.
  - train on ("analyze") the rest,
  - evaluate ("assess") on the one you held out.
- Repeat for each of the *V* piles.


Demonstration of 10-fold Cross Validation
Sample values
0.0
0.4
0.8
Test Set
Training Set
75
68
23
Sample index

In code...

```
cross_val_scores <- function(complete_model_spec, training_data, v, metrics = metric_set(mae)) {
  # Split the data into V folds.
  set.seed(0)
  resamples <- vfold_cv(training_data, v = v)

  ...
}
```

In code...

```
cross_val_scores <- function(complete_model_spec, training_data, v, metrics = metric_set(mae)) {
  # Split the data into V folds.
  set.seed(0)
  resamples <- vfold_cv(training_data, v = v)

  # For each of the V folds, assess the result of analyzing on the rest.
  raw_cv_results <- complete_model_spec %>%
    fit_resamples(resamples = resamples, metrics = metrics)

  # Return the collected metrics.
  collect_metrics(raw_cv_results, summarize = FALSE)
}
```

What's a complete model spec?

Workflow = recipe + model_spec.

```
spec <- workflow() %>%
  add_recipe(recipe) %>%
  add_model(model)
```

e.g.,

```
spec <- workflow() %>%
  add_recipe(
    recipe(Sale_Price ~ Latitude + Longitude, data = ames_train)
  ) %>%
  add_model(
    linear_reg()
  )
```

Continuing with Lab 10

Instructions