Actuarial Experience Studies and Assumption Setting in R

New York R Conference 2022

Matt Heaphy, FSA, MAAA

June 9, 2022

Agenda

- Primer on actuaries and actuarial modeling
- Experience studies with `dplyr`
- Assumption setting with `tidymodels`
- Measuring performance using actuarial models
- Wrap-up

The views herein are based on the speaker's experience and opinions only and do not represent the views of the Society of Actuaries or the American Academy of Actuaries. The

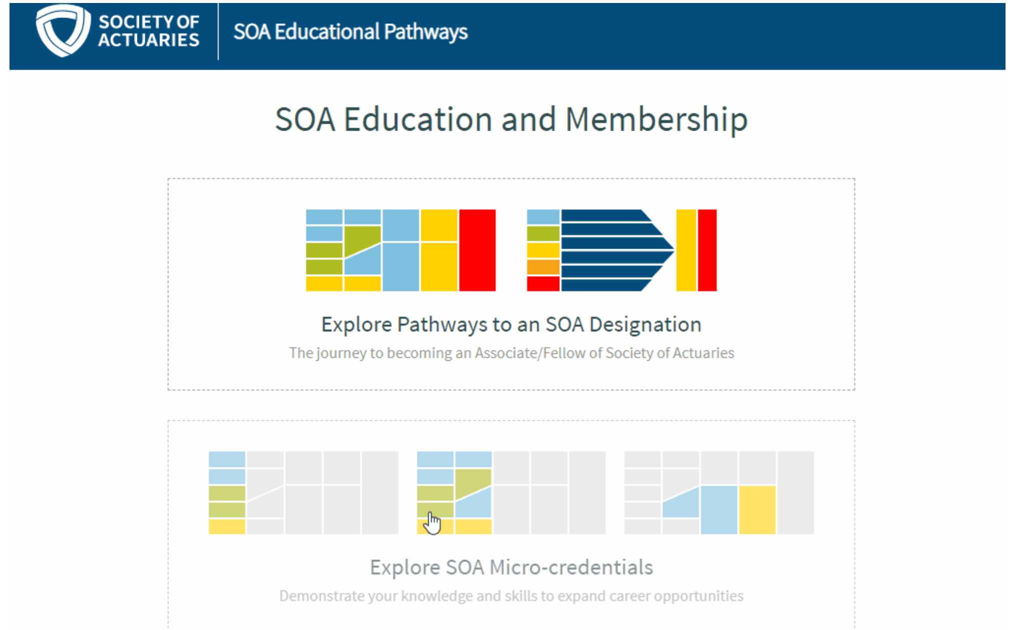
data and analysis included in this presentation are theoretical only and contain simplifying assumptions that may not be true in the real world.

Actuaries and Actuarial Models

What is an Actuary?

> *An actuary is someone who uses statistics, financial mathematics, and deep domain expertise to quantify, price, and manage risks.*

- One of the Original data professions
- Typically practicing in insurance (life, health, P&C) and pensions
- Society of Actuaries: 32K members worldwide[1]



1. https://www.soa.org/about/total-membership/

Actuarial Models

> Actuarial Models are long term projection engines of cashflows, assets, and liabilities

Uses

- Projections / planning
- Valuation
- Pricing
- Risk management

Actuarial Present Value (APV)

$$ APV=\sum_{t=1}^{\Omega}v^t{}_{t-1}p_x^{\tau}(q_{x+t}^dDB_t+q_{x+t}^sSV_t+WD_t) $$

- v^t = discount factor, the value of \$1 `t` years in the future, assuming a constant discount rate
- ${}_{t}p_x^{\tau}$ = the probability that a policy issued at age `x` survives `t` years
- q_{x+t}^h = the probability that a policy age `x+t` expires due to hazard `h`
- DB_t, SV_t, WD_t = death, surrender, and withdrawal claim payments

Experience Studies

Simulated Deferred Annuity Data

Topic: predicting surrender rates on a deferred annuity product with an optional lifetime income benefit.

- **Training data**: 20,000 policies (3,585 surrendered)
- **Test data**: 5,000 policies (861 surrendered)

```r
library(tidymodels)
glimpse(census)
```

```
Rows: 20,000
Columns: 9
$ pol_num  <int> 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17,
18~
$ term     <int> 0, 0, 0, 1, 0, 0, 0, 0, 0, 1, 0, 0, 1, 1, 0, 1, 0, 0, 0, 0,
0~
$ pol_yr   <int> 6, 13, 8, 12, 1, 2, 9, 15, 10, 4, 1, 8, 11, 10, 3, 11, 5, 2,
~
$ inc_guar <fct> FALSE, TRUE, TRUE, TRUE, FALSE, TRUE, TRUE, TRUE, TRUE,
TRUE,~
$ qual     <fct> TRUE, FALSE, TRUE, TRUE, TRUE, TRUE, TRUE, TRUE, TRUE, TRUE,
```

Cross-Tab Example

- `pol_yr` = policy year
- `claims` = # contract surrenders
- `exposures` = # policy years exposed to the hazard (surrenders)
- `q_obs` = Observed probability of surrender, `claims / exposures`
- `q_exp` = Expected probability of surrender
- `ae_q_exp` = Actual-to-expected ratio, `q_obs` / `q_exp`

pol_yr	claims	exposures	q_obs	q_exp	ae_q_exp
1	89	20,000	0.4%	0.4%	88.0%
2	144	18,635	0.8%	0.8%	99.5%
3	101	17,167	0.6%	0.8%	128.0%
4	168	15,773	1.1%	1.0%	96.3%
5	189	14,265	1.3%	1.3%	96.3%
6	173	12,804	1.4%	1.3%	99.7%
7	180	11,365	1.6%	1.6%	99.5%
8	207	9,977	2.1%	2.0%	98.2%
9	228	8,476	2.7%	2.5%	92.2%
10	197	7,010	2.8%	3.0%	106.0%
11	1,090	5,676	19.2%	19.1%	99.6%

Creating Exposure Records

Census data → Exposed data

Census data

pol_yr	age	term
Policy 1		
3	73	0
Policy 2		
4	66	1

Exposed data

pol_yr	age	term
Policy 1		
1	73	0
2	74	0
3	75	0
Policy 2		
1	66	0
2	67	0
3	68	0
4	69	1

```r
expose <- function(dat) {
  dat |>
    slice(rep(row_number(), pol_yr)) |>
    group_by(pol_num) |>
    mutate(
      term = ifelse(row_number() == pol_yr, term, 0),
      pol_yr = row_number(),
      age = age + pol_yr - 1) |>
    ungroup()
}

(study_data <- expose(census))
```

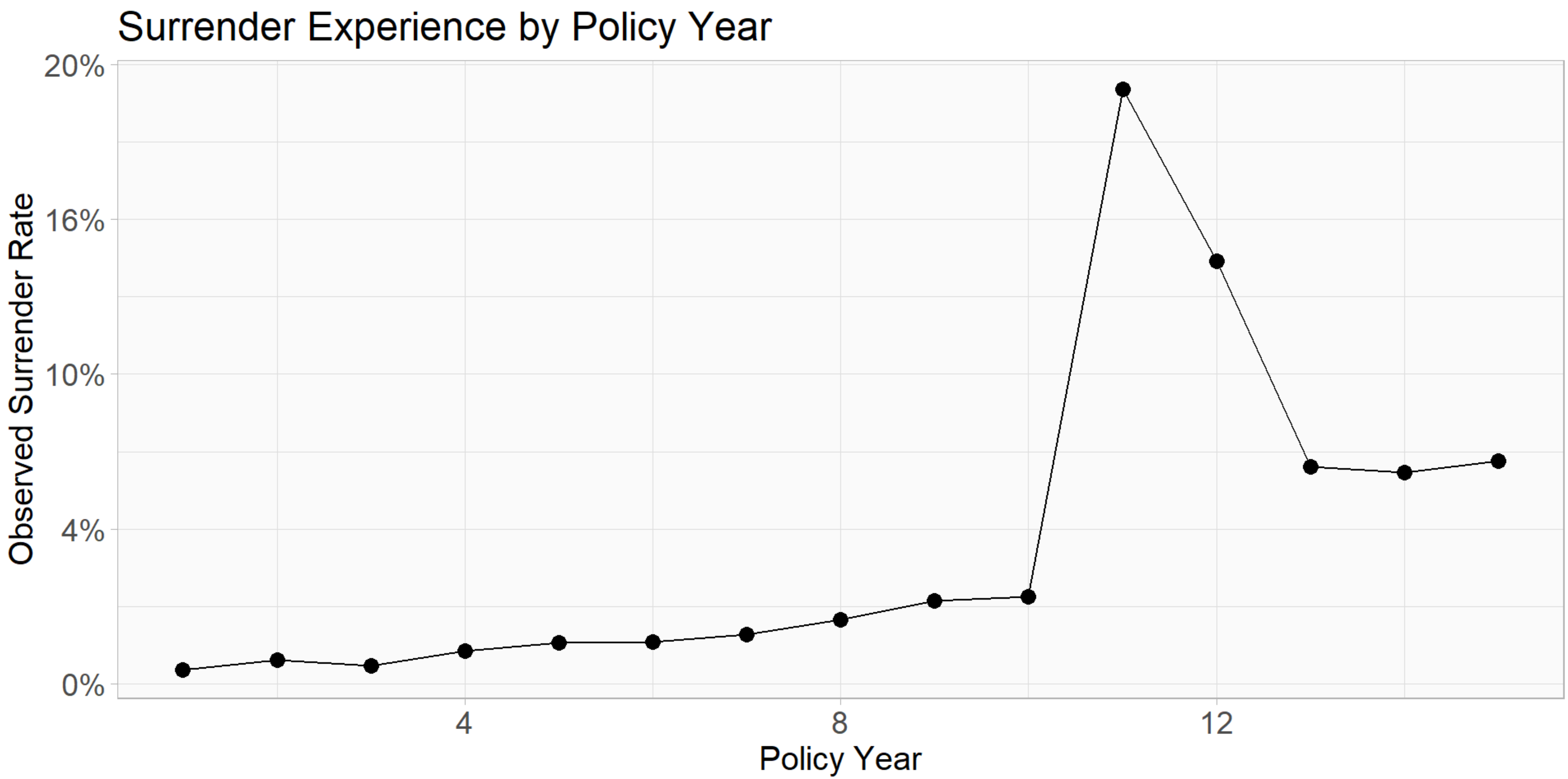
```
# A tibble: 149,367 x 9
   pol_num  term pol_yr inc_guar qual     age product gender wd_age
     <int> <dbl>  <int> <fct>    <fct> <dbl> <chr>   <chr>   <int>
 1       1     0      1 FALSE    TRUE     60 b       F          64
 2       1     0      2 FALSE    TRUE     61 b       F          64
 3       1     0      3 FALSE    TRUE     62 b       F          64
 4       1     0      4 FALSE    TRUE     63 b       F          64
 5       1     0      5 FALSE    TRUE     64 b       F          64
 6       1     0      6 FALSE    TRUE     65 b       F          64
 7       2     0      1 TRUE     FALSE    79 b       M          79
 8       2     0      2 TRUE     FALSE    80 b       M          79
 9       2     0      3 TRUE     FALSE    81 b       M          79
10       2     0      4 TRUE     FALSE    82 b       M          79
# ... with 149,357 more rows
```

Experience Summary Function

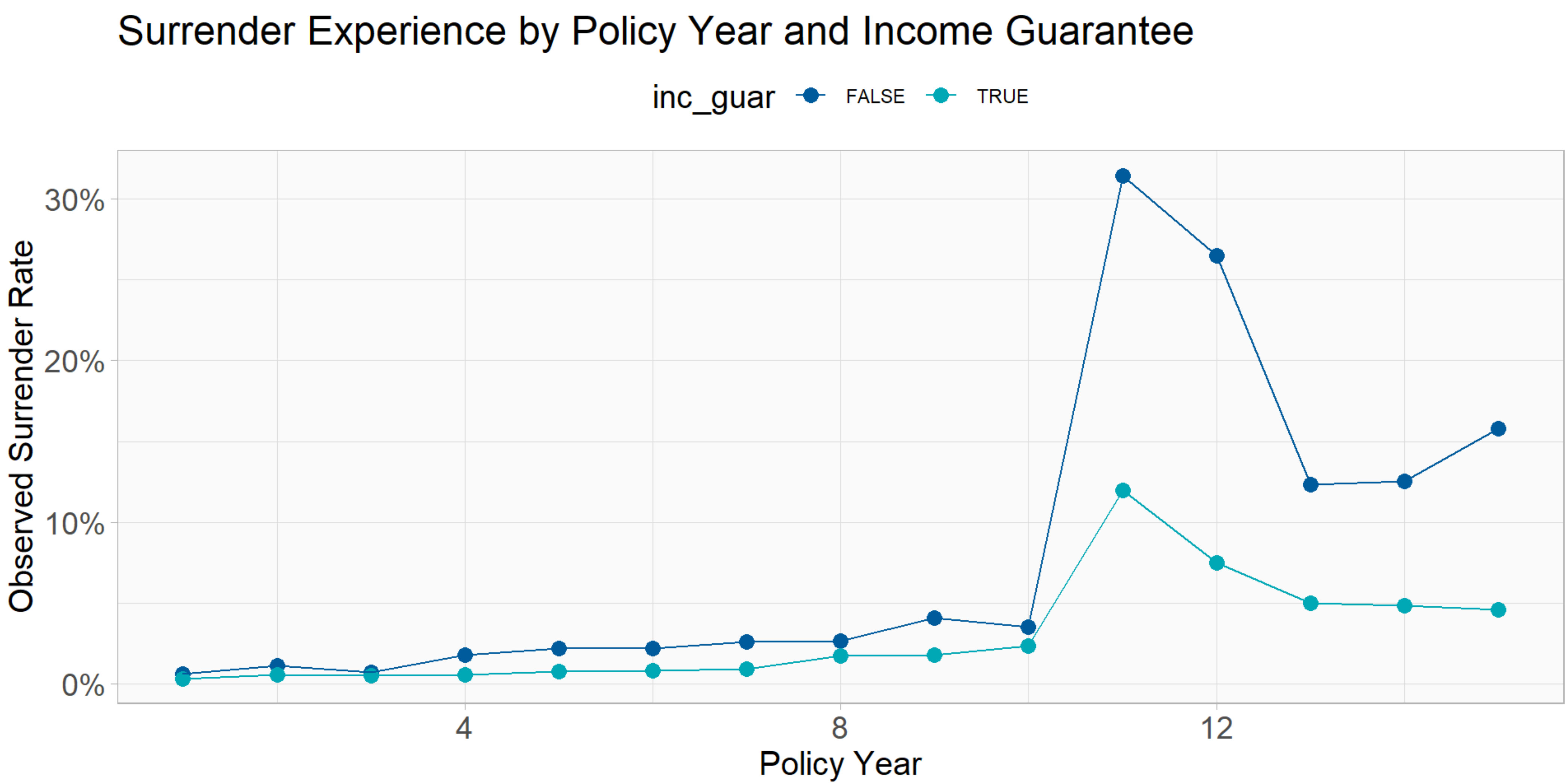
```r
exp_stats <- function(dat, expected = FALSE) {

  dat %>%
    summarize(claims = sum(term),
              exposures = n(),
              q_obs = mean(term),
              q_exp = if(expected) mean(q_exp) else NULL,
              ae_q_exp = if(expected) q_exp / q_obs else NULL,
              .groups = "drop")

}

exp_stats(study_data2, expected = TRUE)
```

claims	exposures	q_obs	q_exp	ae_q_exp
3,585	149,367	2.4%	2.4%	99.7%

```r
study_data2 |> group_by(pol_yr) |>
  exp_stats(expected = TRUE)
```



```
study_data2 |> group_by(pol_yr, inc_guar) |>
  exp_stats(expected = TRUE)
```

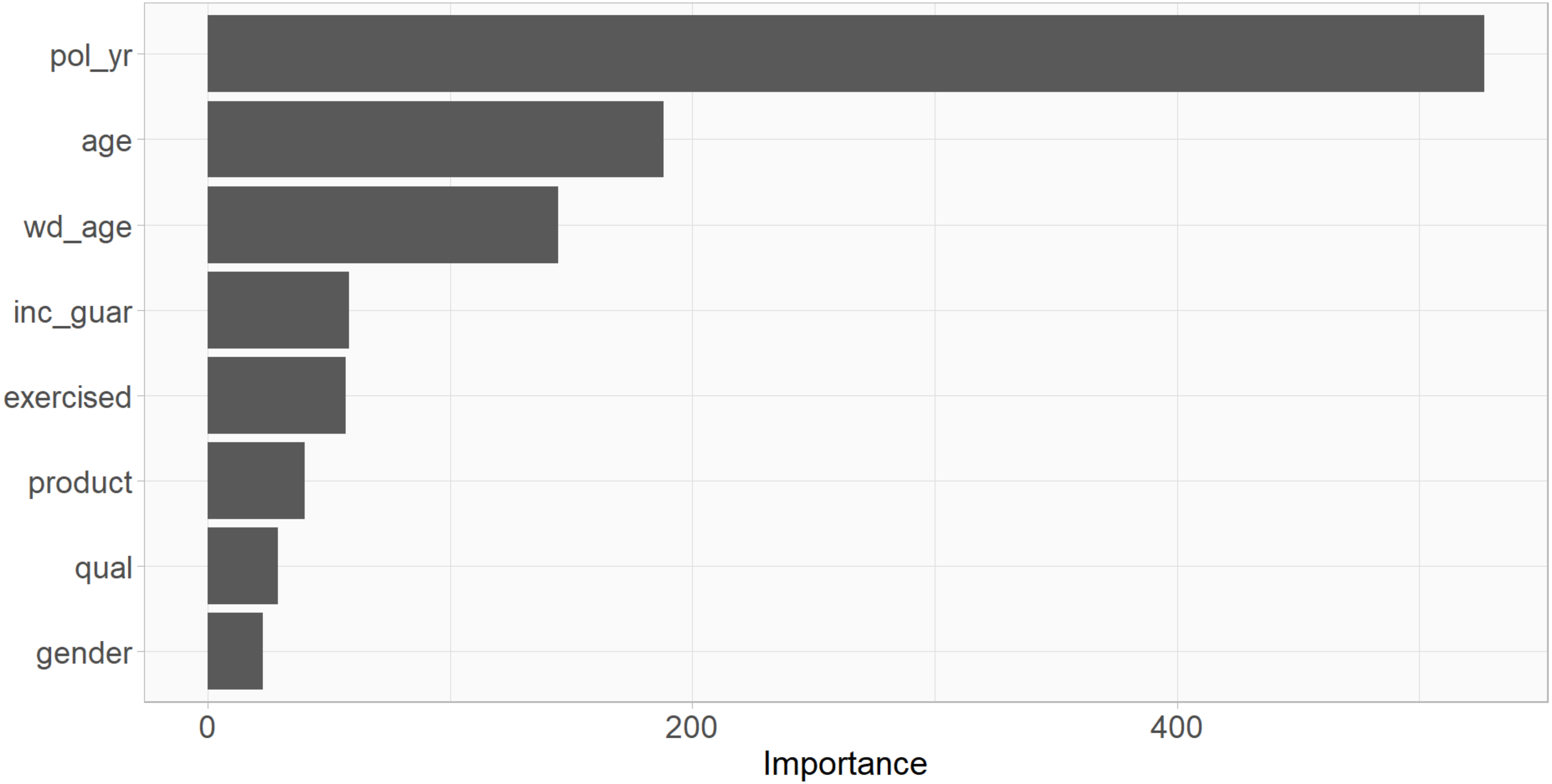


Variable Importance

Variable importance plots can quickly highlight notable features that might otherwise have been missed.

```r
model_dat <- study_data2 |>
  mutate(term = as.factor(term))

rf_rec <- recipe(term ~ ., data = model_dat) |>
  update_role(pol_num, q_exp, new_role = "ignore")

rf_spec <- rand_forest() |>
  set_mode("classification") |>
  set_engine("ranger", importance = "impurity")

rf_vip <- workflow(rf_rec, rf_spec) |> fit(model_dat)
```

```r
library(vip)
rf_vip |> extract_fit_parsnip() |> vip()
```



Assumption Setting

Assumption Setting Methods

> Goal: fit each model below and compare against observed experience and "correct" experience.

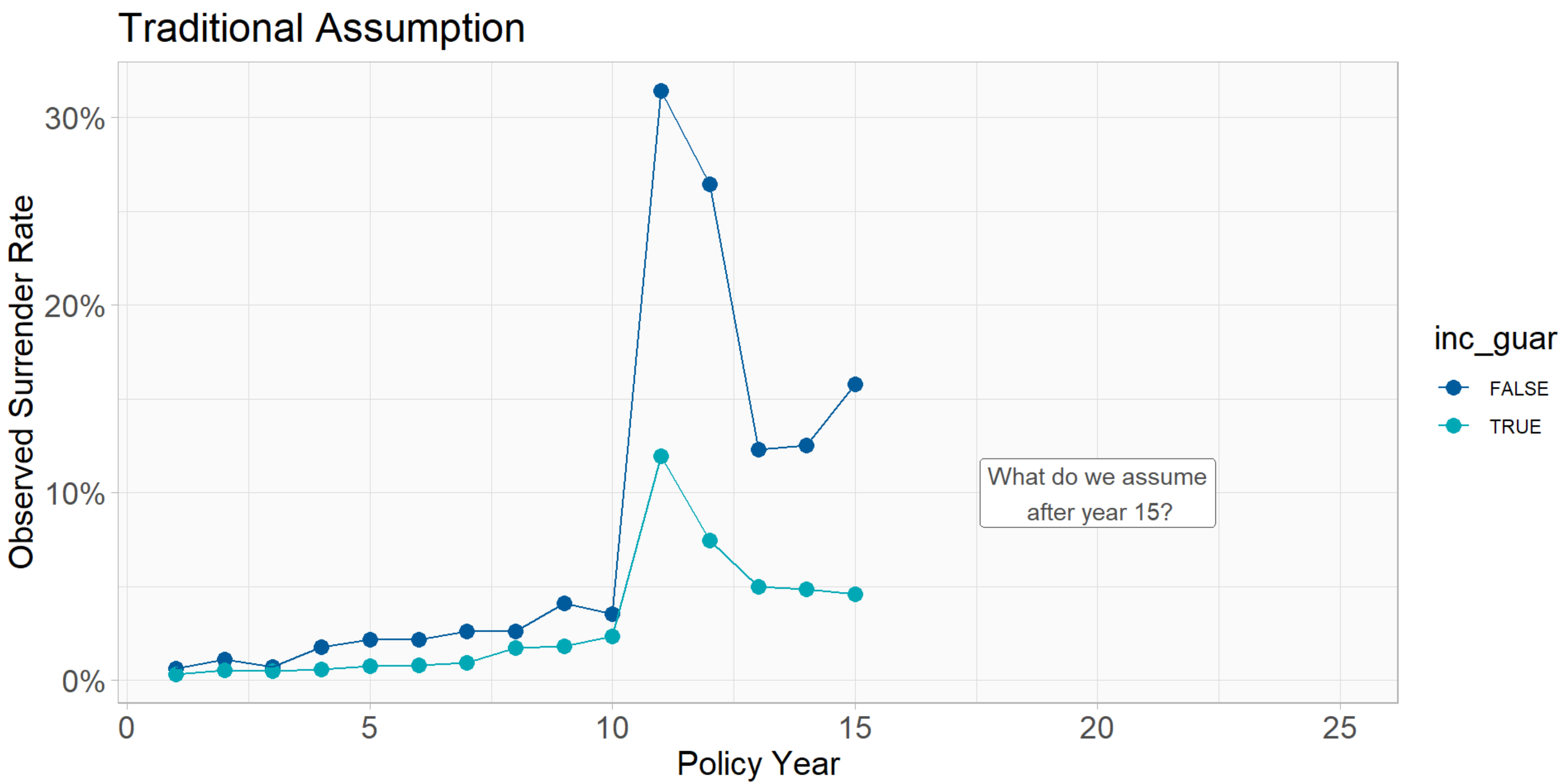
1. Traditional Tabular Assumption
2. Logistic Regression
3. Random Forest

Traditional Tabular Assumptions

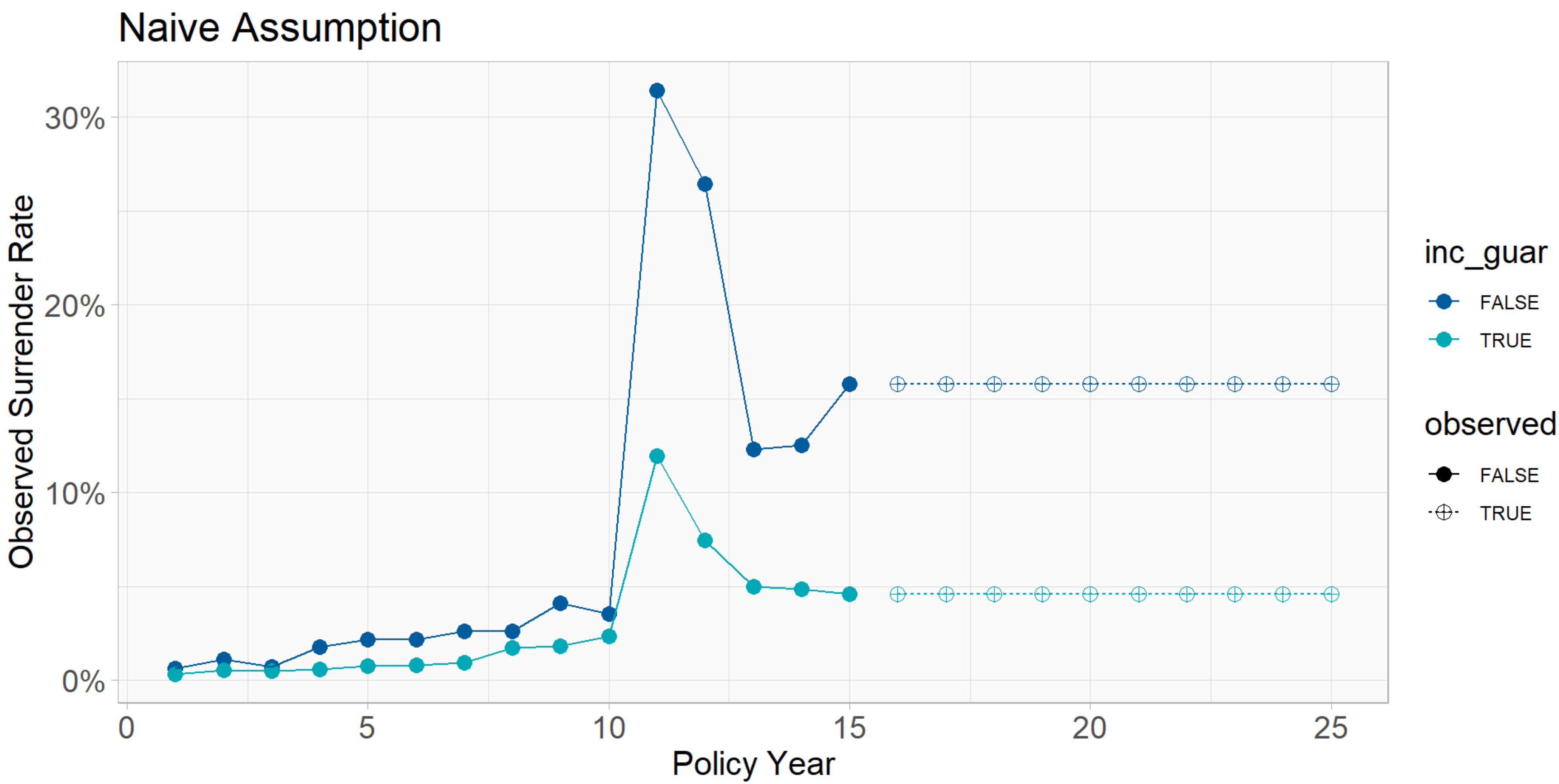
- Start with experience studies
- Apply judgment, smoothing, and topsides as needed

```r
trad_assump <- study_data |> group_by(pol_yr, inc_guar) |> exp_stats()
```

Initial Assumption



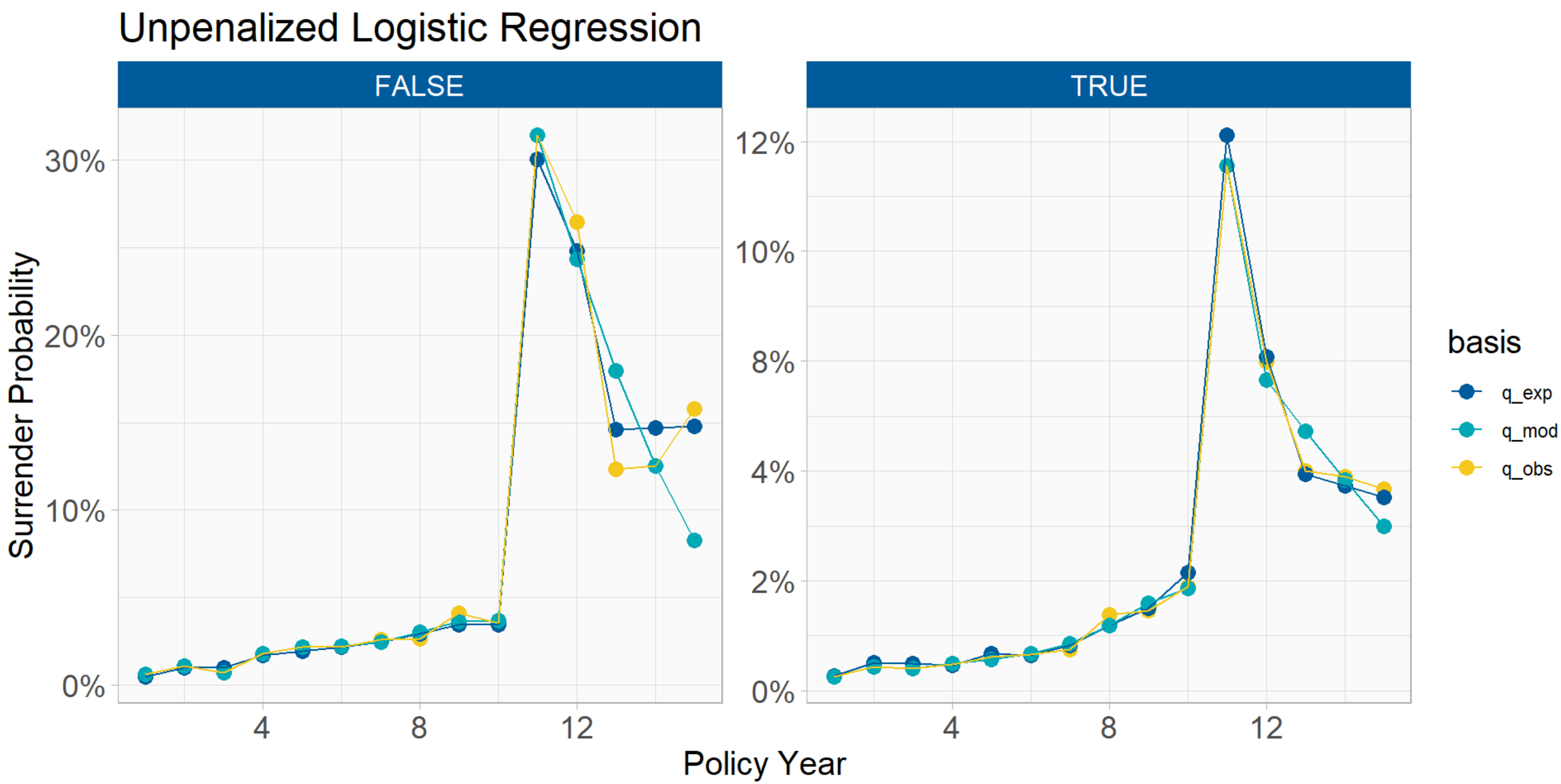
One Approach



Logistic Regression

```
log_spec <- logistic_reg() |> set_engine("glm")

log_rec <- recipe(term ~ ., data = model_dat) |>
  update_role(pol_num, q_exp, new_role = "ignore") |>
  step_mutate(sc_group = case_when(
    pol_yr <= 10 ~ "SC Period",
    pol_yr == 11 ~ "Shock",
    TRUE ~ "PostShock"
  ) |> factor()) |>
  step_dummy(all_nominal_predictors()) |>
  step_ns(pol_yr, age, deg_free = 7) |>
  step_interact(terms = ~starts_with("sc_group"):starts_with("inc_guar")) |
  step_interact(terms = ~starts_with("pol_yr"):starts_with("inc_guar")) |>
  step_interact(terms = ~starts_with("age"):starts_with("inc_guar"))

log_wf <- workflow(log_rec, log_spec)

log_model <- fit(log_wf, model_dat)
```

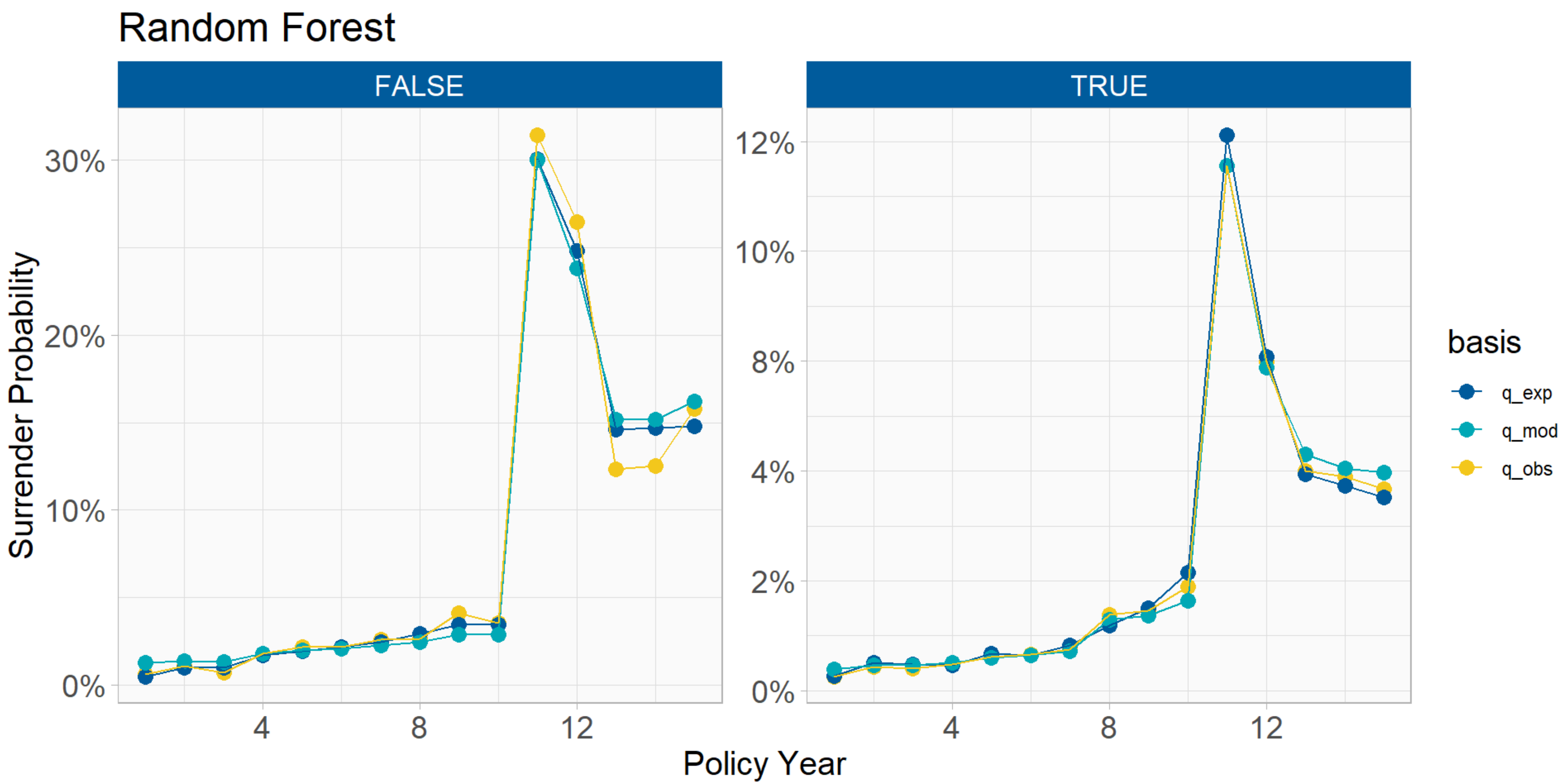
Performance



Random Forest

```r
rf_spec <- rand_forest() |>
  set_mode("classification") |>
  set_engine("ranger")

rf_rec <- recipe(term ~ ., data = model_dat) |>
  update_role(pol_num, q_exp, new_role = "ignore") |>
  step_mutate(sc_group = case_when(
    pol_yr <= 10 ~ "SC Period",
    pol_yr == 11 ~ "Shock",
    TRUE ~ "PostShock"
  ) |> factor()) |>
  step_dummy(all_nominal_predictors()) |>
  step_interact(terms = ~starts_with("sc_group"):starts_with("inc_guar")) |
  step_interact(terms = ~starts_with("pol_yr"):starts_with("inc_guar"))

rf_wf <- workflow(rf_rec, rf_spec)

rf_model <- fit(rf_wf, model_dat)
```

Performance



Actuarial Model Performance

Process

- Use the test data for active policies (4,139 records)
- For each model:
  - Generate predictions for future surrender probabilities
  - Calculate actuarial present values

$$ APV=\sum_{t=1}^{\Omega}v^t{}_{t-1}p_x^{\tau}(q_{x+t}^dDB_t+q_{x+t}^sSV_t+WD_t) $$

- Compare performance against the “correct” assumption

Other Assumptions

- Initial account value = $2,000
- Annual withdrawals with income benefit = $100 for life
- Annual withdrawals without income benefit = 5% of account value
- Interest credited rate = 3%
- Mortality = 2012 IAM Basic[1]

1. https://mort.soa.org/ViewTable.aspx?&TableIdentity=2581, https://mort.soa.org/ViewTable.aspx?&TableIdentity=2582

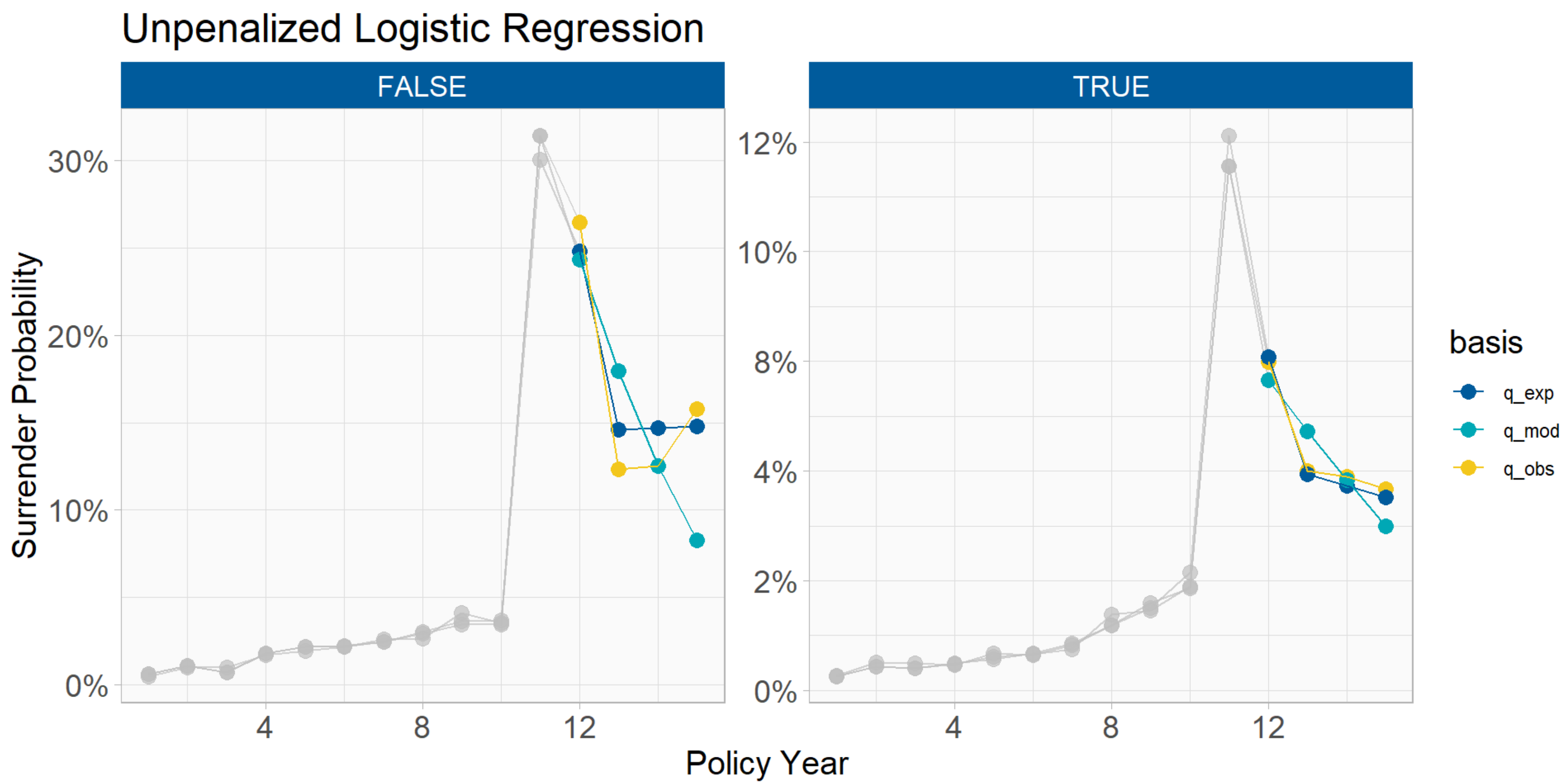
Results

- All Actual-to-expected ratios near 100%
- Higher RMSE on the logistic model

Assumption	APV	A/E	Abs Diff	RMSE
Random Forest	$7,376,821	100.0%	$3,689	5.76
Logistic	$7,402,574	100.3%	$22,064	11.97
Tabular	$7,375,744	99.9%	$4,766	5.73

Why is the Logistic Model an Outlier?

- Poor fit to surrender rates after year 10
- Extrapolation beyond year 15 is not accurate

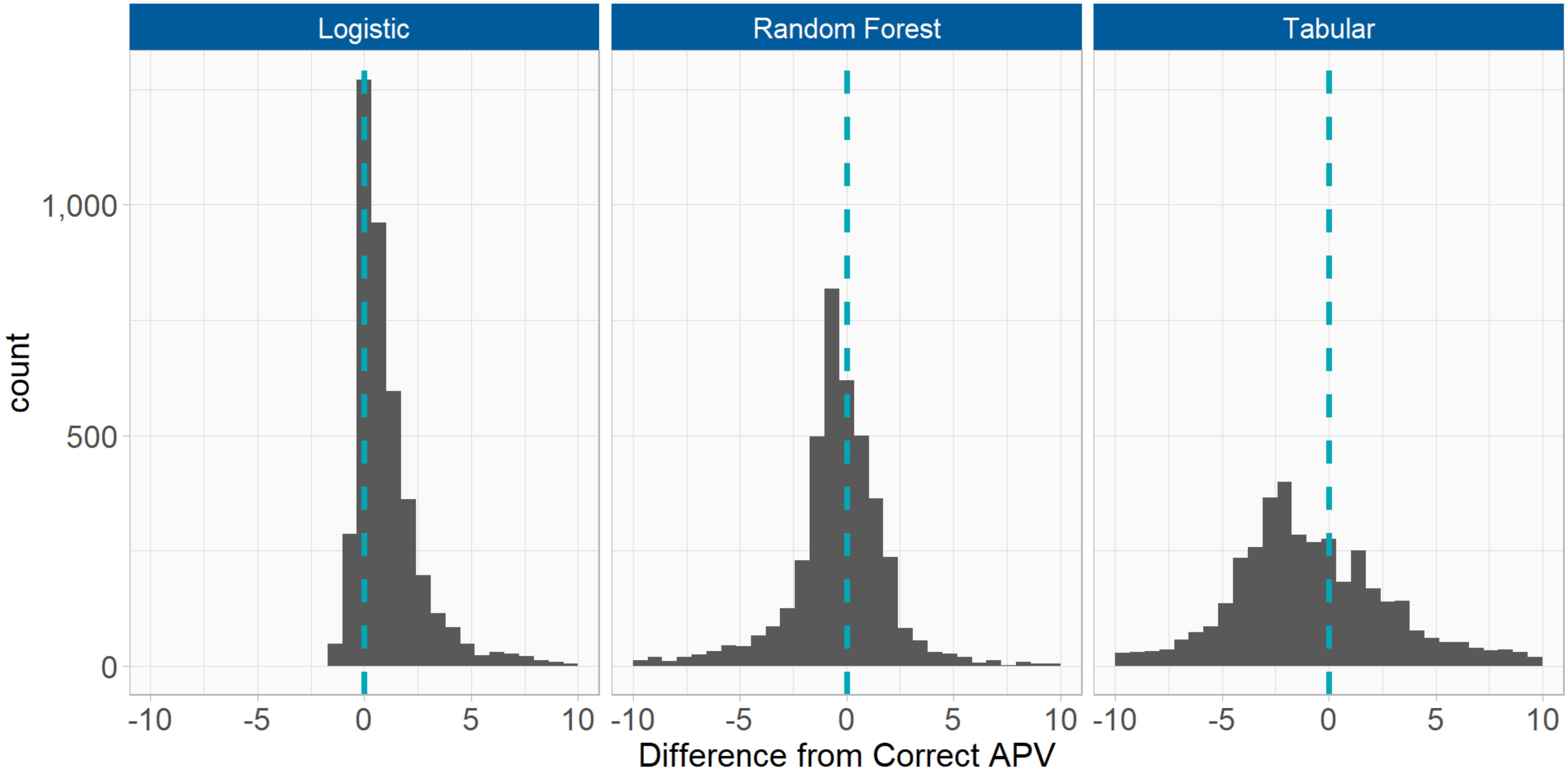


Results v2

- Applying a bit of post-processing judgment, we cap the `pol_yr` variable at 15
- The logistic model now performs much better

Assumption	APV	A/E	Abs Diff	RMSE
Random Forest	$7,376,821	100.0%	$3,689	5.76
Logistic	$7,385,579	100.1%	$5,069	2.69
Tabular	$7,375,744	99.9%	$4,766	5.73

Distribution of APV Residuals

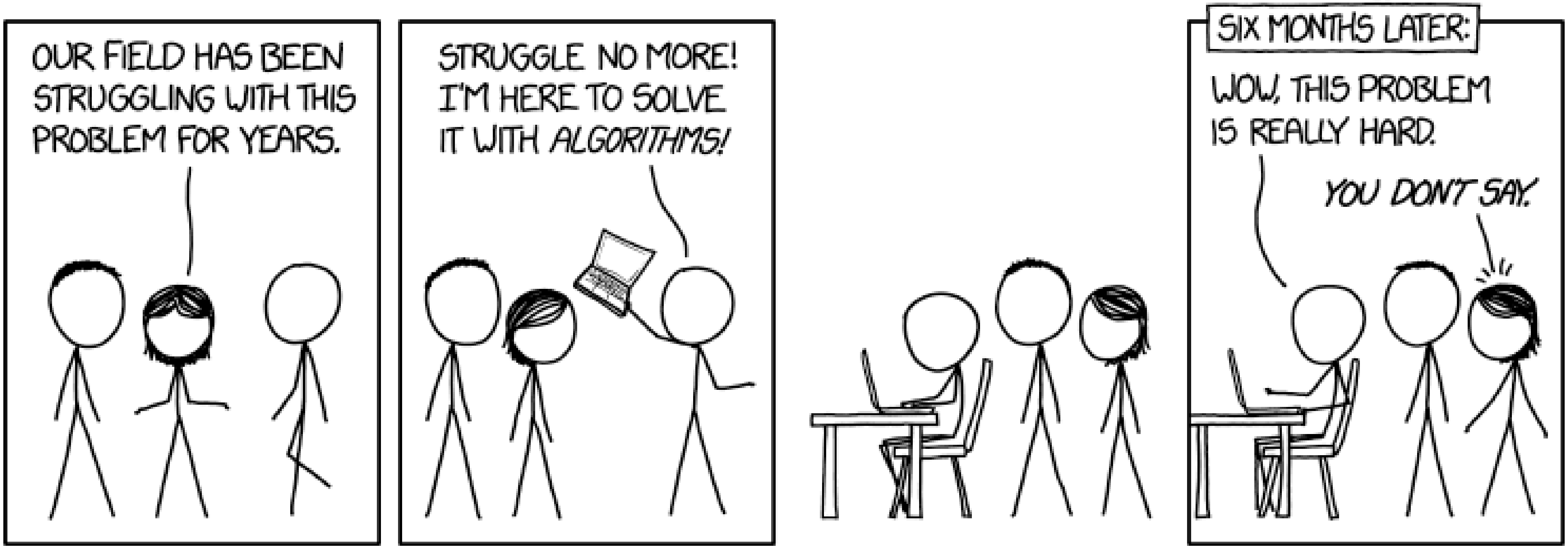


Wrap-Up

Lessons

- Modern tools like R and `tidymodels` save time and unlock deeper insights, resulting in more precise models.
- Assumption setting should always consider downstream usage in actuarial models.

Expertise and professional judgment are still required!



https://xkcd.com/1831

Thank you!