The tidymodels ecosystem

MODELING WITH TIDYMODELS IN R

David Svancer
Data Scientist

datacamp

Collection of machine learning packages

Collection of machine learning packages

Data resampling

Collection of machine learning packages

Data resampling

Feature engineering

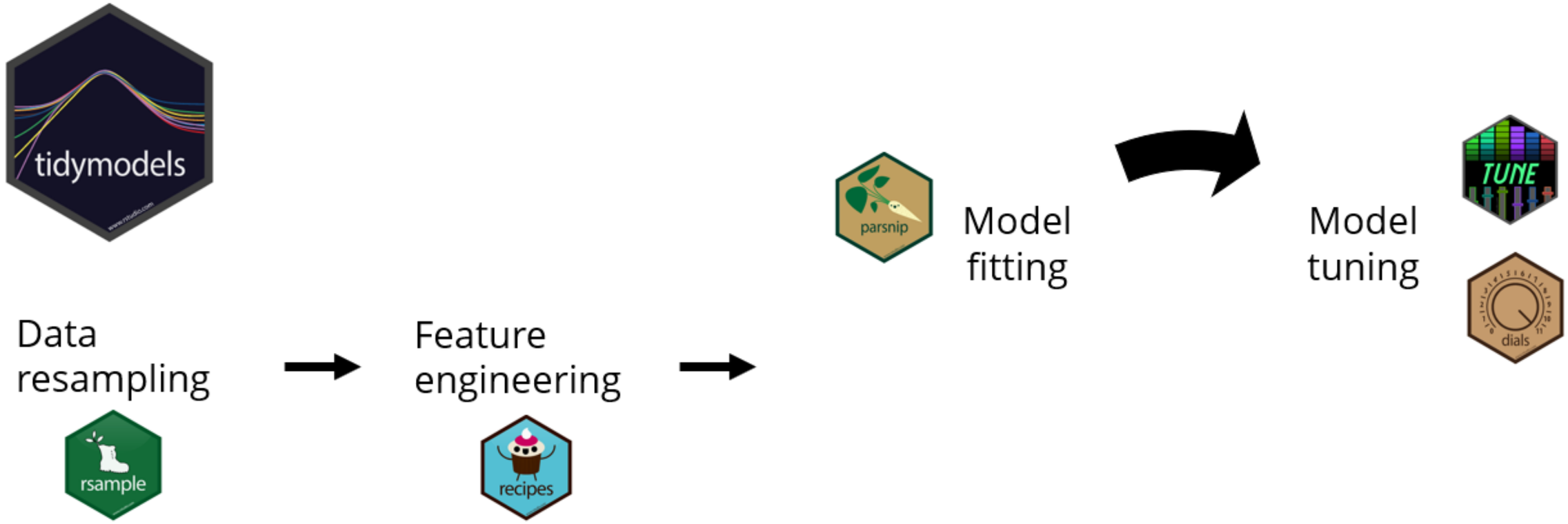
Collection of machine learning packages

Model fitting

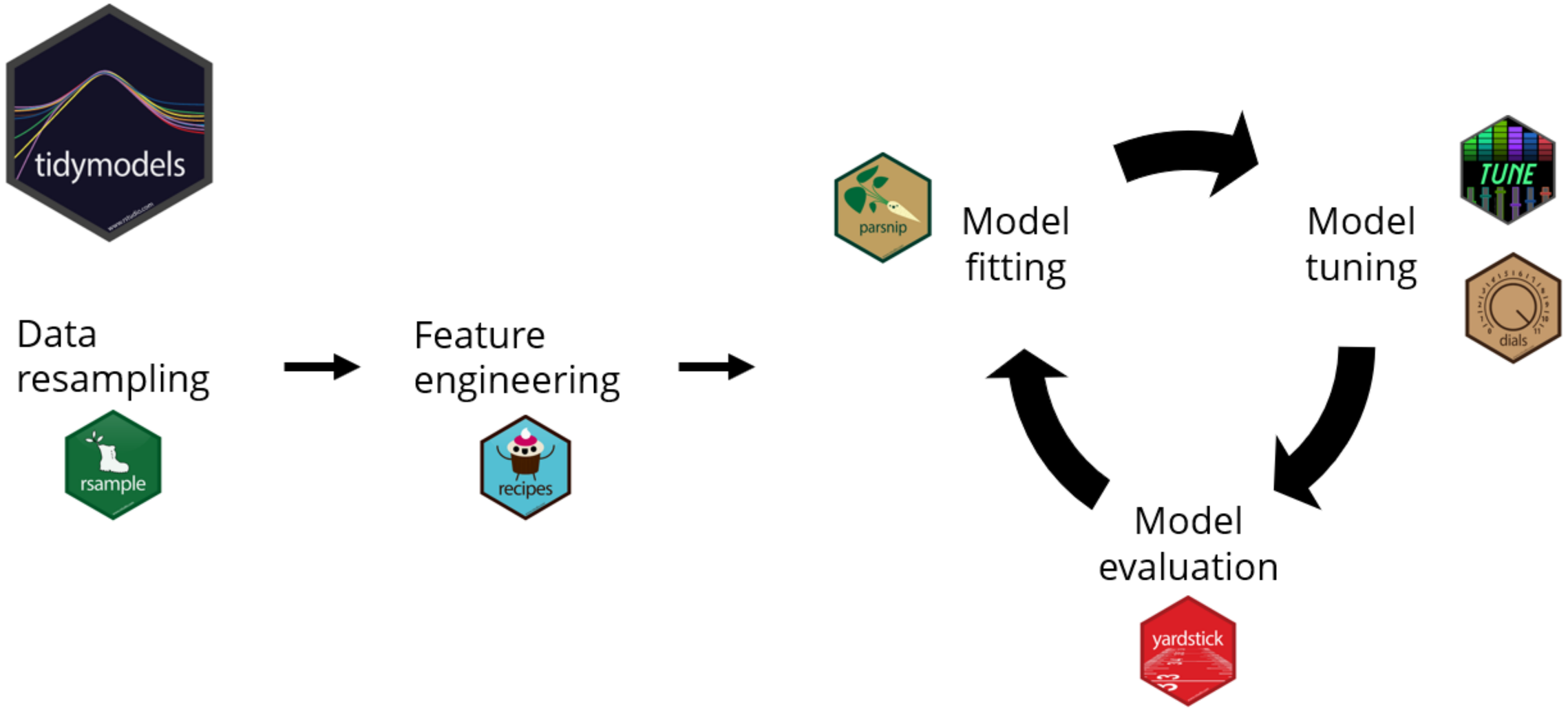
Data resampling

Feature engineering

Collection of machine learning packages



Collection of machine learning packages



Supervised machine learning

Branch of machine learning that uses labeled data for model fitting

Regression

- Predicting **quantitative** outcomes
  - Selling price of a home

Classification

- Predicting **categorical** outcomes
  - Whether an employee will leave a company

left_company	miles_from_home	salary
no	1	84500
yes	10	64820
no	5	76490
yes	19	68540

`tidymodels` **variable roles**

- *left_company* is an outcome variable
- *miles_from_home* and *salary* are predictor variables

Data resampling

Create training and test sets

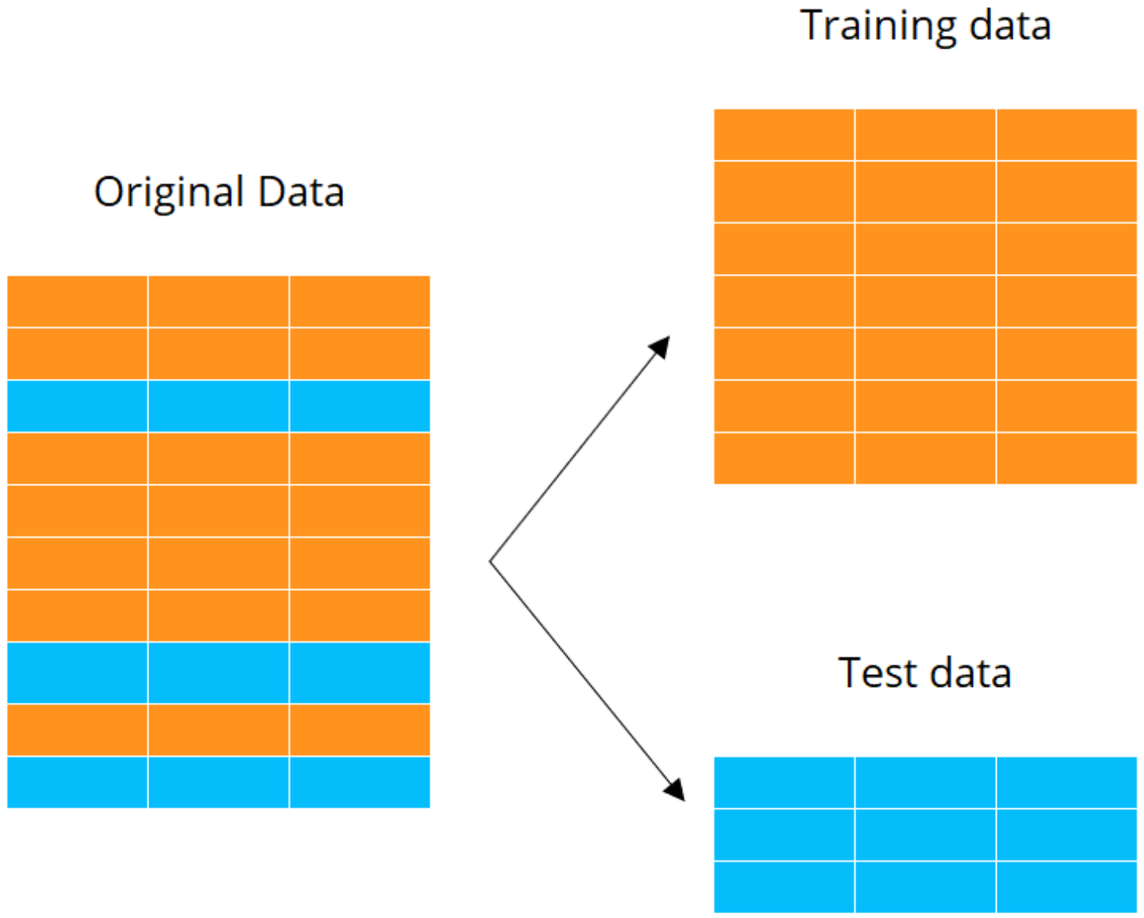
- Guards against **overfitting**
- Common ratio is 75% training, 25% test

Training data

- Feature engineering
- Model fitting and tuning

Test data

- Estimate model performance on new data



Fuel efficiency data

Vehicle fuel efficiency data from the U.S. Environmental Protection Agency

- Outcome variable is `hwy` - highway fuel efficiency in miles per gallon (mpg)

```
mpg
```

```
# A tibble: 234 x 11
     hwy   cty displ   cyl manufacturer model       year trans      drv   fl    class
   <int> <int> <dbl> <int> <chr>        <chr>      <int> <chr>      <chr> <chr> <chr>
 1    29    18   1.8     4 audi         a4          1999 auto(l5)   f     p     compact
 2    29    21   1.8     4 audi         a4          1999 manual(m5) f     p     compact
 3    31    20   2       4 audi         a4          2008 manual(m6) f     p     compact
 4    30    21   2       4 audi         a4          2008 auto(av)   f     p     compact
 5    26    16   2.8     6 audi         a4          1999 auto(l5)   f     p     compact
# ... with 224 more rows
```

Data resampling with tidymodels

- `initial_split()`
  - Specifies instructions for creating training and test datasets
  - `prop` specifies the proportion to place into training
  - `strata` provides stratification by the outcome variable
- Pass split object to `training()` function
- Pass split object to `testing()` function

```
library(tidymodels)
```

```
mpg_split <- initial_split(mpg,
                           prop = 0.75,
                           strata = hwy)
```

```
mpg_training <- mpg_split %>%
  training()
```

```
mpg_test <- mpg_split %>%
  testing()
```

Home sales data

Home sales from the Seattle, Washington area between 2015 and 2016

```
home_sales
```

```
# A tibble: 1,492 x 8
   selling_price home_age bedrooms bathrooms sqft_living sqft_lot sqft_basement floors
           <dbl>    <dbl>    <dbl>     <dbl>       <dbl>    <dbl>          <dbl>  <dbl>
 1        487000       10        4       2.5        2540     5001              0      2
 2        465000       10        3      2.25        1530     1245            480      2
 3        411000       18        2         2        1130     1148            330      2
 4        635000        4        3       2.5        3350     4007            800      2
 5        380000       24        5       2.5        2130     8428              0      2
# ... with 1,482 more rows
```

Let's practice!

MODELING WITH TIDYMODELS IN R

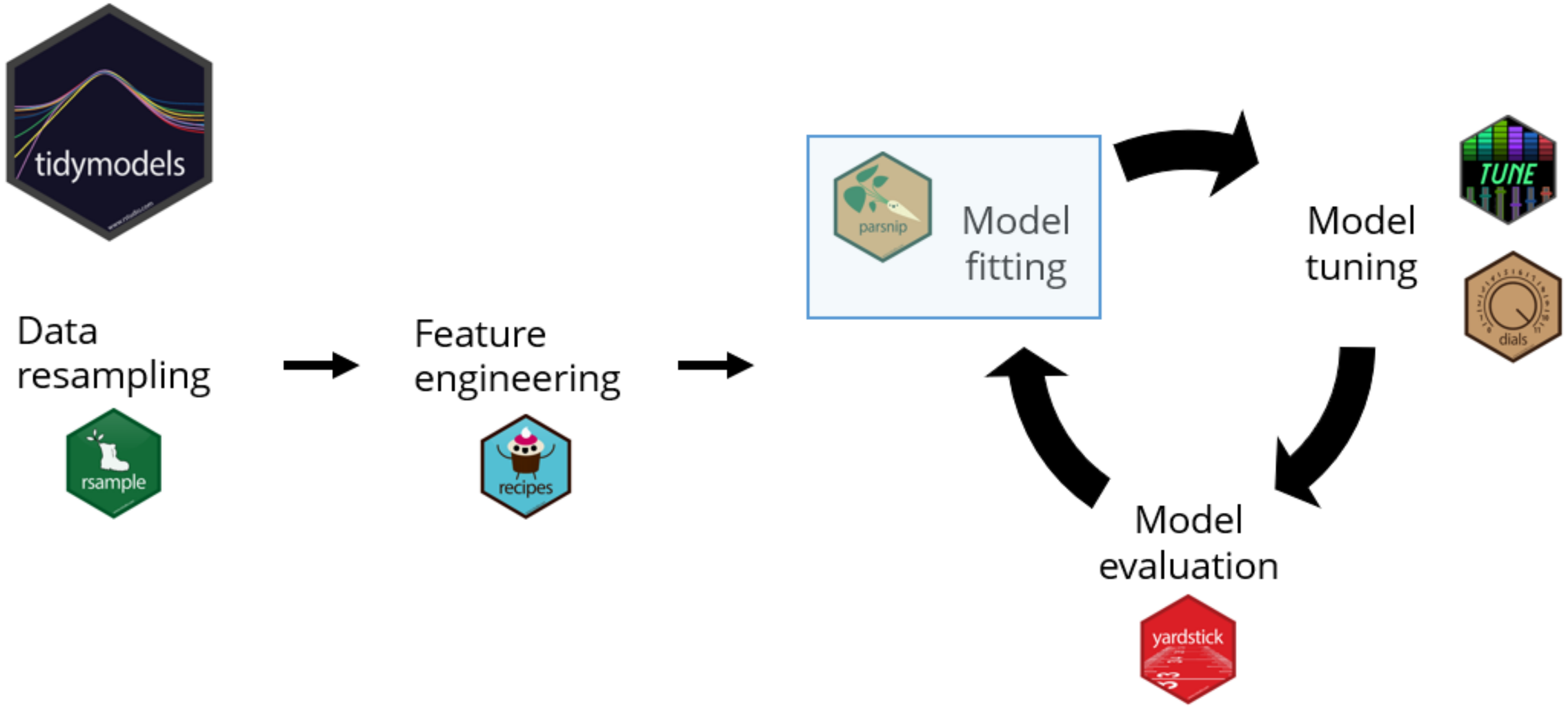
Linear regression with tidymodels

MODELING WITH TIDYMODELS IN R

David Svancer
Data Scientist

datacamp

Model fitting with parsnip



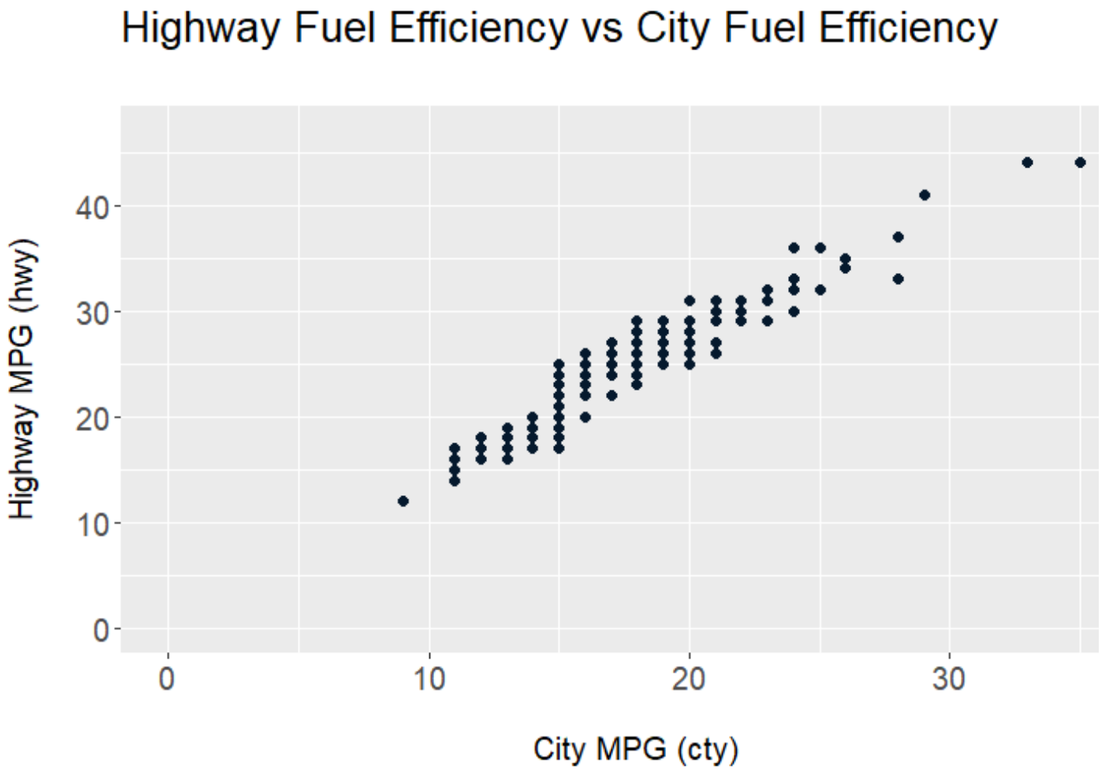
Linear regression model

Predicting `hwy` using `cty` as a predictor

$$hwy = \beta_0 + \beta_1 cty$$

Model parameters

- β_0 is the intercept
- β_1 is the slope



Linear regression model

Predicting `hwy` using `cty` as a predictor

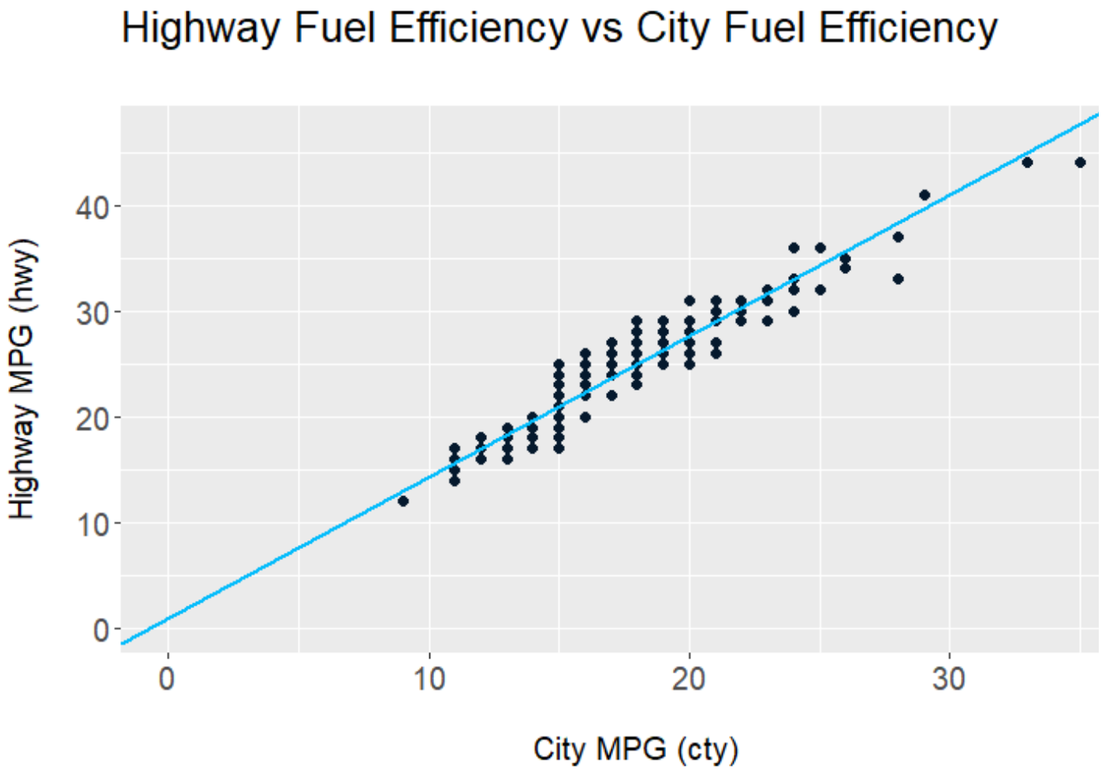
$$hwy = \beta_0 + \beta_1 cty$$

Model parameters

- β_0 is the intercept
- β_1 is the slope

Estimated paramters from training data

$$hwy = 0.77 + 1.35(cty)$$



Model formulas

Model formulas in `parsnip`

- Used to assign column roles
  - Outcome variable
  - Predictor variables

General form

```
outcome ~ predictor_1 + predictor_2 + ...
```

Shorthand notation

```
outcome ~ .
```

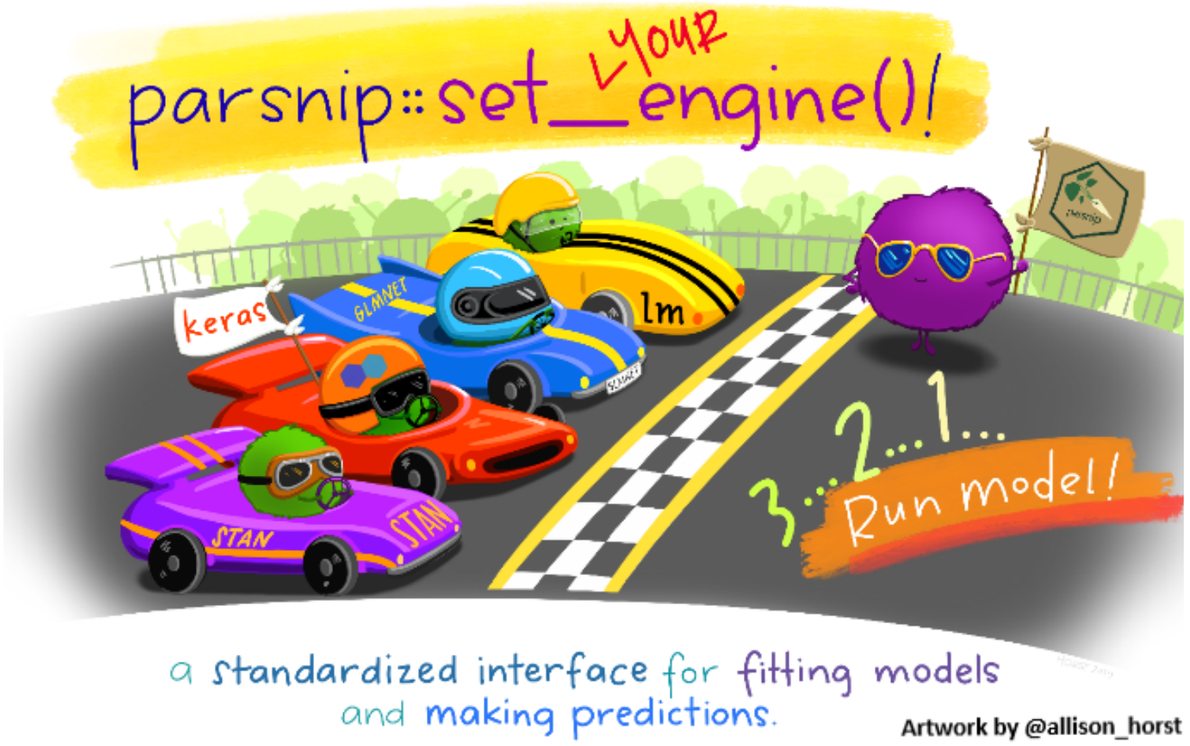
Predicting `hwy` using `cty` as a predictor variable

```
hwy ~ cty
```

The parsnip package

Unified syntax for model specification in R

1. Specify the **model type**
   - Linear regression or other model type
2. Specify the **engine**
   - Different engines correspond to different underlying R packages
3. Specify the **mode**
   - Either regression or classification



Fitting a linear regression model

Define model specification with `parsnip`

- `linear_reg()`

```
lm_model <- linear_reg() %>%
  set_engine('lm') %>%
  set_mode('regression')
```

Pass `lm_model` to the `fit()` function

- Specify model formula
- `data` to use for model fitting

```
lm_fit <- lm_model %>%
  fit(hwy ~ cty, data = mpg_training)
```

Obtaining the estimated parameters

The `tidy()` function

- Takes a trained `parsnip` model object
- Creates a model summary tibble
- `term` and `estimate` column provide estimated parameters

```
tidy(lm_fit)
```

```
# A tibble: 2 x 5
  term        estimate std.error statistic  p.value
  <chr>          <dbl>     <dbl>     <dbl>    <dbl>
1 (Intercept)    0.769    0.528       1.46 1.47e- 1
2 cty            1.35     0.0305     44.2  6.32e-97
```

Making predictions

Pass trained `parsnip` model to the `predict()` function

- `new_data` specifies dataset on which to predict new values

```
hwy_predictions <- lm_fit %>%
  predict(new_data = mpg_test)

hwy_predictions
```

```
# A tibble: 57 x 1
   .pred
   <dbl>
 1  25.0
 2  27.7
 3  25.0
 4  25.0
 5  22.3
# ... with 47 more rows
```

Standardized output from `predict()`

1. Returns a tibble
2. Keep rows in the same order as `new_data` input
3. Names prediction column `.pred`

Adding predictions to the test data

The `bind_cols()` function

- Combines two or more tibbles along the column axis
- Useful for creating a model results tibble

Steps

- Select `hwy` and `cty` from `mpg_test`
- Pass to `bind_cols()` and add predictions column

```
mpg_test_results <- mpg_test %>%
  select(hwy, cty) %>%
  bind_cols(hwy_predictions)

mpg_test_results
```

```
# A tibble: 57 x 3
     hwy   cty .pred
   <int> <int> <dbl>
 1    29    18  25.0
 2    31    20  27.7
 3    27    18  25.0
 4    26    18  25.0
 5    25    16  22.3
# ... with 47 more rows
```

Let's model!

MODELING WITH TIDYMODELS IN R

Evaluating model performance

MODELING WITH TIDYMODELS IN R

David Svancer
Data Scientist

datacamp

Input to yardstick functions

All `yardstick` functions require a tibble with model results

- Column with the true outcome variable values
  - `hwy` for mpg data
- Column with model predictions
  - `.pred`

```
mpg_test_results
```

```
# A tibble: 57 x 3
     hwy   cty .pred
   <int> <int> <dbl>
 1    29    18  25.0
 2    31    20  27.7
 3    27    18  25.0
 4    26    18  25.0
 5    25    16  22.3
# ... with 47 more rows
```

Root mean squared error (RMSE)

RMSE estimates the average prediction error

- Calculated with the `rmse()` function from `yardstick`
  - Takes a tibble with model results
  - `truth` is the column with true outcome values
  - `estimate` is the column with predicted outcome values

```
mpg_test_results %>%
  rmse(truth = hwy, estimate = .pred)
```

```
# A tibble: 1 x 3
  .metric .estimator .estimate
  <chr>   <chr>          <dbl>
1 rmse    standard        1.93
```

R squared metric

Measures the squared correlation between actual and predicted values

- Also called the **coefficient of determination**
- Ranges from 0 to 1
  - When all predictions equal the true outcome values, R squared is 1
- Calculated with the `rsq()` function from `yardstick`

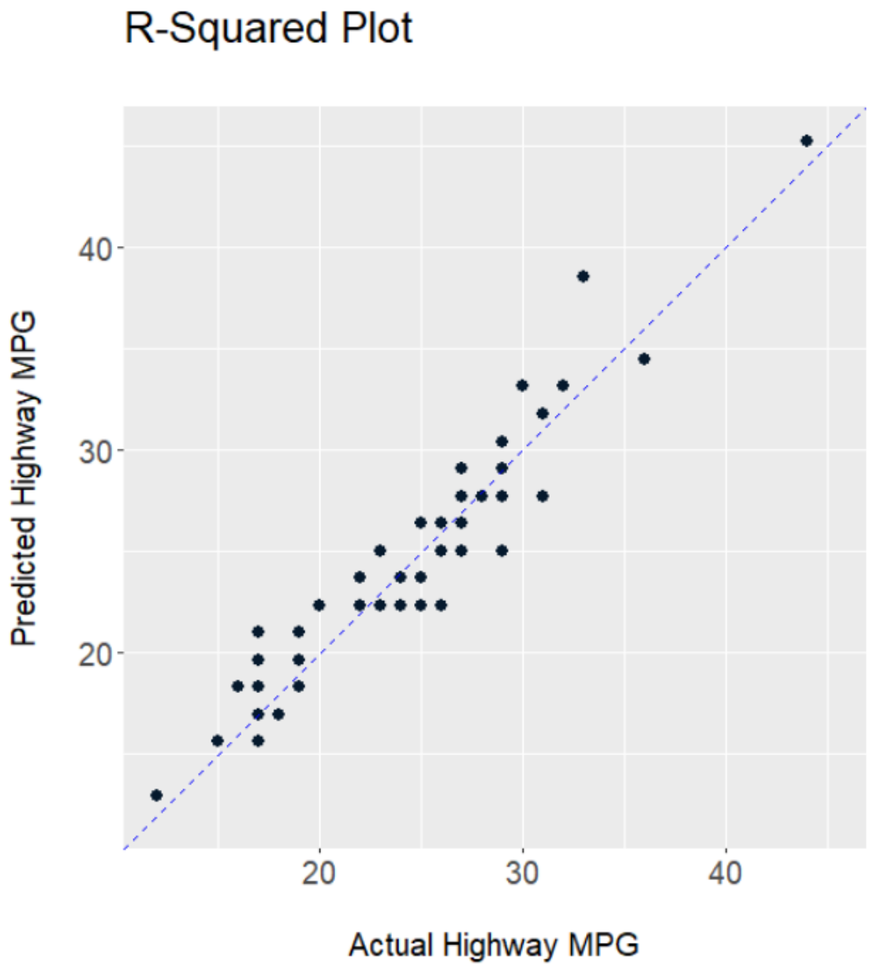
```
mpg_test_results %>%
  rsq(truth = hwy, estimate = .pred)
```

```
# A tibble: 1 x 3
  .metric .estimator .estimate
  <chr>   <chr>          <dbl>
1 rsq     standard       0.904
```

R squared plots

Visualization of the R squared metric

- Model predictions versus the true outcome
- The line y = x
  - Represents R squared of 1
- Used to find potential problems with model performance
  - Non-linear patterns
  - Regions where model is predicting poorly

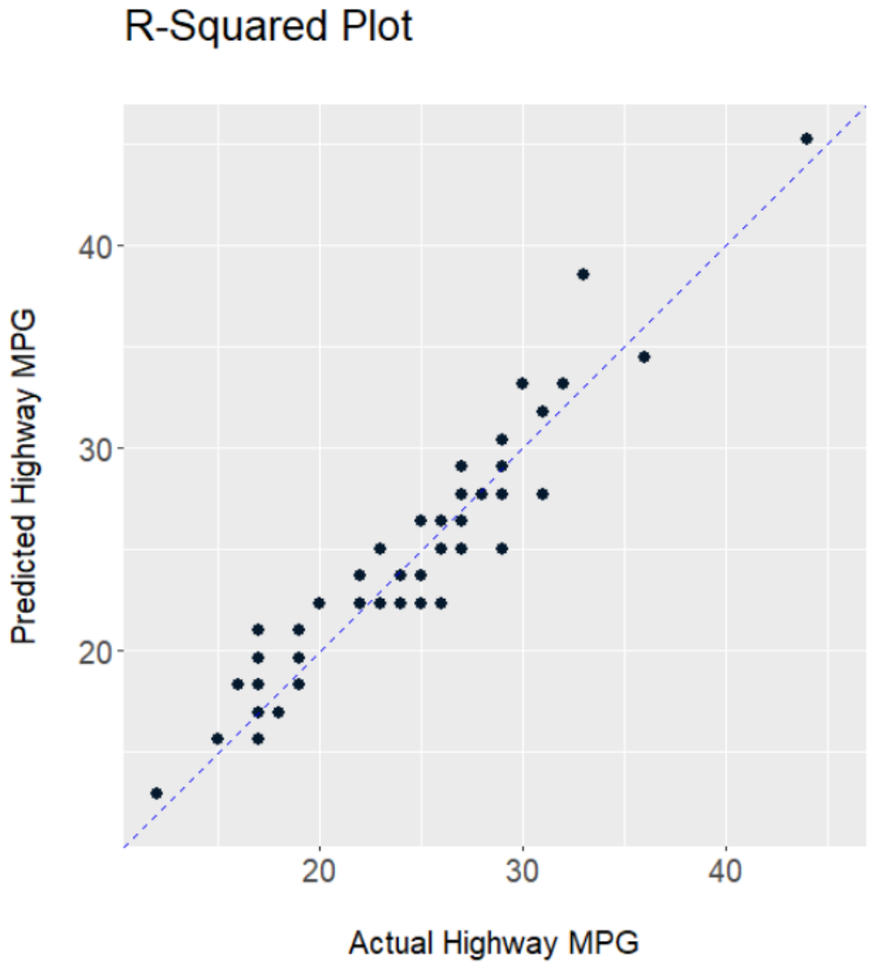


Plotting R squared plots

Making R squared plots with `ggplot2`

- Tibble of model results
- `geom_point()`
- `geom_abline()`
- `coord_obs_pred()`

```
ggplot(mpg_test_results, aes(x = hwy, y = .pred)) +
  geom_point() +
  geom_abline(color = 'blue', linetype = 2) +
  coord_obs_pred() +
  labs(title = 'R-Squared Plot',
       y = 'Predicted Highway MPG',
       x = 'Actual Highway MPG')
```



Streamlining model fitting

The `last_fit()` function

- Takes a model specification, model formula, and data split object
- Performs the following:
  1. Creates training and test datasets
  2. Fits the model to the training data
  3. Calculates metrics and predictions on the test data
  4. Returns an object with all results

```
lm_last_fit <- lm_model %>%
  last_fit(hwy ~ cty,
           split = mpg_split)
```

Collecting metrics

The `collect_metrics()` function

- Takes the results of `last_fit()`
  - Returns a tibble with performance metrics obtained on the test dataset
- Default regression model metrics
  - RMSE
  - R squared

```
lm_last_fit %>%
  collect_metrics()
```

```
# A tibble: 2 x 3
  .metric .estimator .estimate
  <chr>   <chr>          <dbl>
1 rmse    standard        1.93
2 rsq     standard        0.904
```

Collecting predictions

The `collect_predictions()` function

- Takes the results of `last_fit()`
  - Returns a tibble with test dataset predictions
  - Predictions column is named `.pred`
  - Outcome variable and other row identifier columns included

```
lm_last_fit %>%
  collect_predictions()
```

```
# A tibble: 57 x 4
   id                .pred  .row   hwy
   <chr>             <dbl> <int> <int>
 1 train/test split  25.0      1    29
 2 train/test split  27.7      3    31
 3 train/test split  25.0      7    27
 4 train/test split  25.0      8    26
 5 train/test split  22.3      9    25
# ... with 47 more rows
```

Let's evaluate some models!

MODELING WITH TIDYMODELS IN R