Introdução ao tidyverse

5 Resultados de modelos e tidymodels

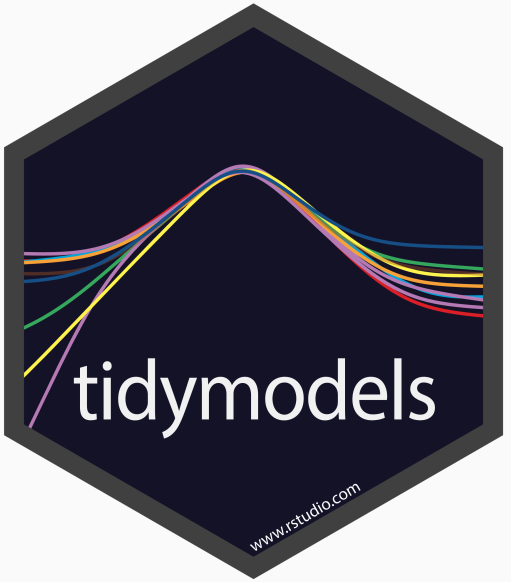
xaringan [presentation ninja]

Maurício Vancine
26/04/2019


broom
www.tidyverse.org

5 Modelos, tidyverse e tidymodels

Conteúdo



5 Modelos, tidyverse e tidymodels

Script

`script_aula_05.R`

5.1 Pacote broom

Resume informações importantes sobre modelos em tidy tibbles

Organiza mais de 100 modelos de pacotes de modelagem populares e quase todos os objetos de saída de modelos que acompanha o Base R

Atualmente, os seguintes métodos estão disponíveis no **broom**: https://broom.tidyverse.org/articles/available-methods.html

5.2 Funções tidying do broom

O pacote **broom** fornece três funções para transformar modelos em data frames *tidy*:

broom::tidy(): resume informações sobre os componentes do modelo

broom::glance(): apresenta informações sobre o modelo inteiro

broom::augment(): informações sobre observações do modelo

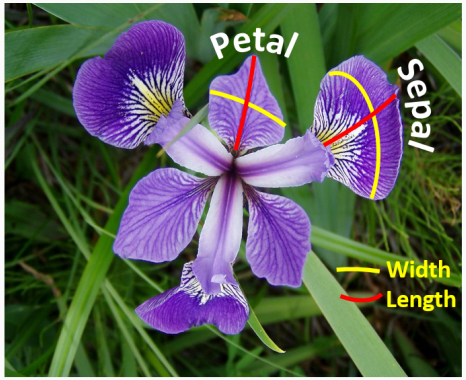
5.3 Aplicações

iris data

```
# data
dplyr::glimpse(iris)
```

```
## Rows: 150
## Columns: 5
## $ Sepal.Length <dbl> 5.1, 4.9, 4.7, 4.6, 5.0, 5.4, 4.6, 5.0, 4.4, 4.9, 5.4, 4
## $ Sepal.Width  <dbl> 3.5, 3.0, 3.2, 3.1, 3.6, 3.9, 3.4, 3.4, 2.9, 3.1, 3.7, 3
## $ Petal.Length <dbl> 1.4, 1.4, 1.3, 1.5, 1.4, 1.7, 1.4, 1.5, 1.4, 1.5, 1.5, 1
## $ Petal.Width  <dbl> 0.2, 0.2, 0.2, 0.2, 0.2, 0.4, 0.3, 0.2, 0.2, 0.1, 0.2, 0
## $ Species      <fct> setosa, setosa, setosa, setosa, setosa, setosa, setosa, s
```

Iris Virginica



Iris Versicolor

Iris Setosa

5.3 Aplicações

Modelo Linear (LM)

```
# linear model
lmfit <- lm(Sepal.Length ~ Petal.Length, iris)
lmfit
```

```
##
## Call:
## lm(formula = Sepal.Length ~ Petal.Length, data = iris)
##
## Coefficients:
##  (Intercept)  Petal.Length
##       4.3066        0.4089
```

5.3 Aplicações

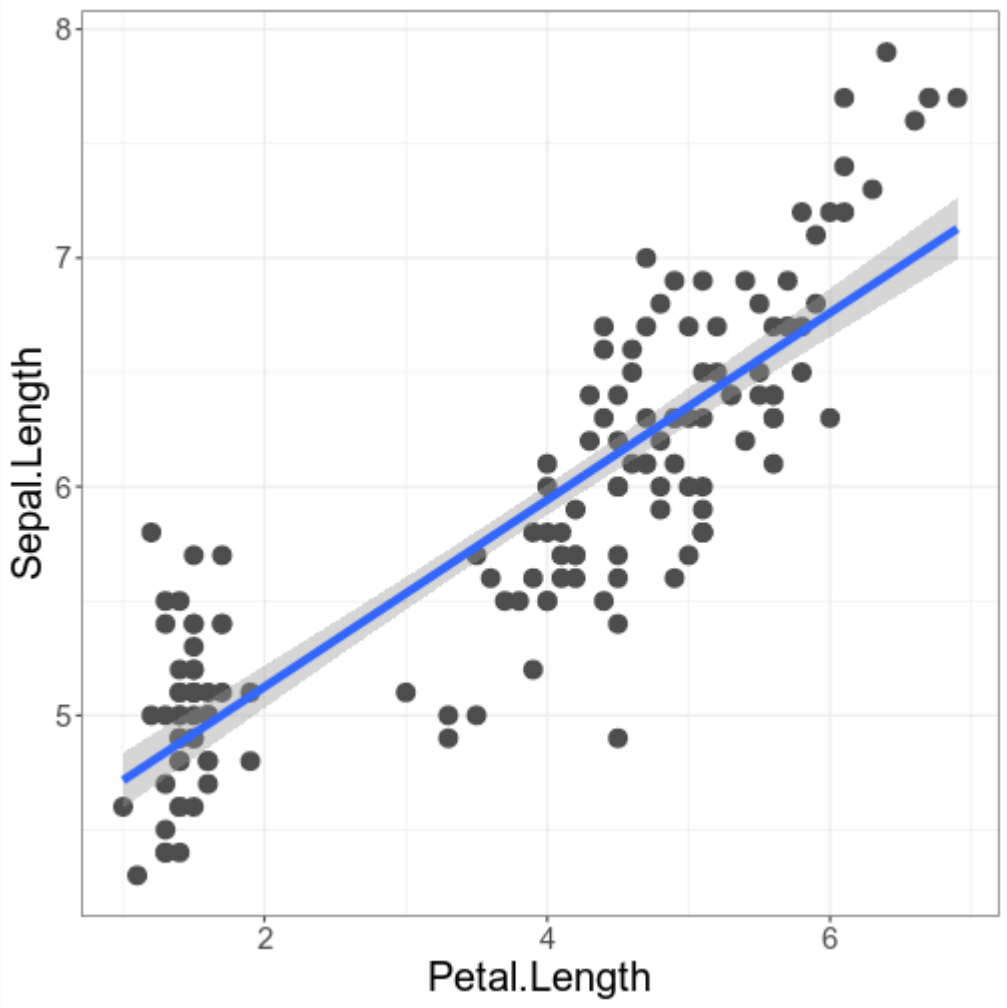
Modelo Linear (LM)

```
summary(lmfit)
```

```
##
## Call:
## lm(formula = Sepal.Length ~ Petal.Length, data = iris)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1.24675 -0.29657 -0.01515  0.27676  1.00269
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)   4.30660    0.07839   54.94   <2e-16 ***
## Petal.Length  0.40892    0.01889   21.65   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.4071 on 148 degrees of freedom
```

5.3 Aplicações

Modelo Linear (LM)



5.4 Função tidy

Informações sobre os componentes do modelo

```
# package
library(broom)

# componentes do modelo
broom::tidy(lmfit)
```

```
## # A tibble: 2 x 5
##   term         estimate std.error statistic   p.value
##   <chr>           <dbl>     <dbl>     <dbl>     <dbl>
## 1 (Intercept)     4.31     0.0784      54.9 2.43e-100
## 2 Petal.Length    0.409    0.0189      21.6 1.04e- 47
```

5.5 Função glance

Informações sobre o modelo inteiro

```
# package
library(broom)

# informacoes sobre o modelo inteiro
broom::glance(lmfit)
```

```
## # A tibble: 1 x 11
##   r.squared adj.r.squared sigma statistic  p.value    df logLik   AIC   BIC de
##       <dbl>         <dbl> <dbl>     <dbl>    <dbl> <int>  <dbl> <dbl> <dbl>
## 1     0.760         0.758 0.407      469. 1.04e-47     2  -77.0  160.  169.
```

5.6 Função augment

Informações sobre observações do modelo

```
# package
library(broom)

# observacoes do modelo
broom::augment(lmfit)
```

```
## # A tibble: 150 x 9
##    Sepal.Length Petal.Length .fitted .se.fit  .resid   .hat .sigma   .cooksd
##           <dbl>        <dbl>   <dbl>   <dbl>   <dbl>  <dbl>  <dbl>     <dbl>
##  1          5.1          1.4    4.88  0.0556  0.221  0.0186  0.408 0.00285
##  2          4.9          1.4    4.88  0.0556  0.0209 0.0186  0.408 0.0000255
##  3          4.7          1.3    4.84  0.0571 -0.138  0.0197  0.408 0.00118
##  4          4.6          1.5    4.92  0.0541 -0.320  0.0176  0.408 0.00565
##  5          5            1.4    4.88  0.0556  0.121  0.0186  0.408 0.000854
##  6          5.4          1.7    5.00  0.0511  0.398  0.0158  0.407 0.00780
##  7          4.6          1.4    4.88  0.0556 -0.279  0.0186  0.408 0.00455
##  8          5            1.5    4.92  0.0541  0.0800 0.0176  0.408 0.000353
##  9          4.4          1.4    4.88  0.0556 -0.479  0.0186  0.407 0.0134
## 10          4.9          1.5    4.92  0.0541 -0.0200 0.0176  0.408 0.0000220
```

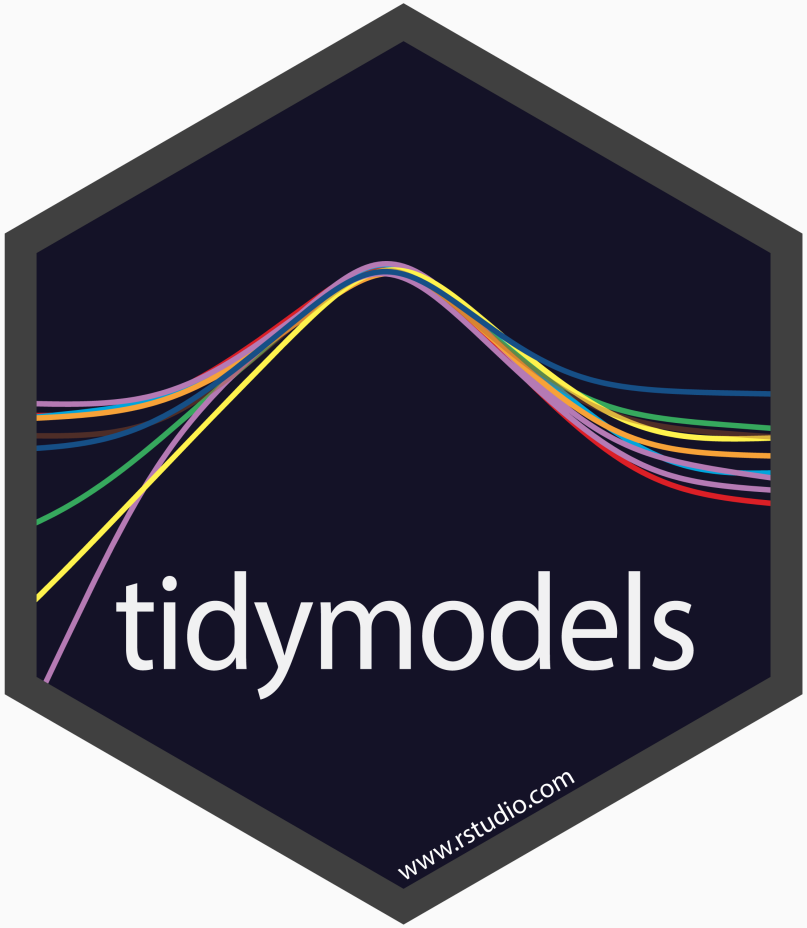
5.6 Função augment

Add informações sobre observações do modelo aos dados

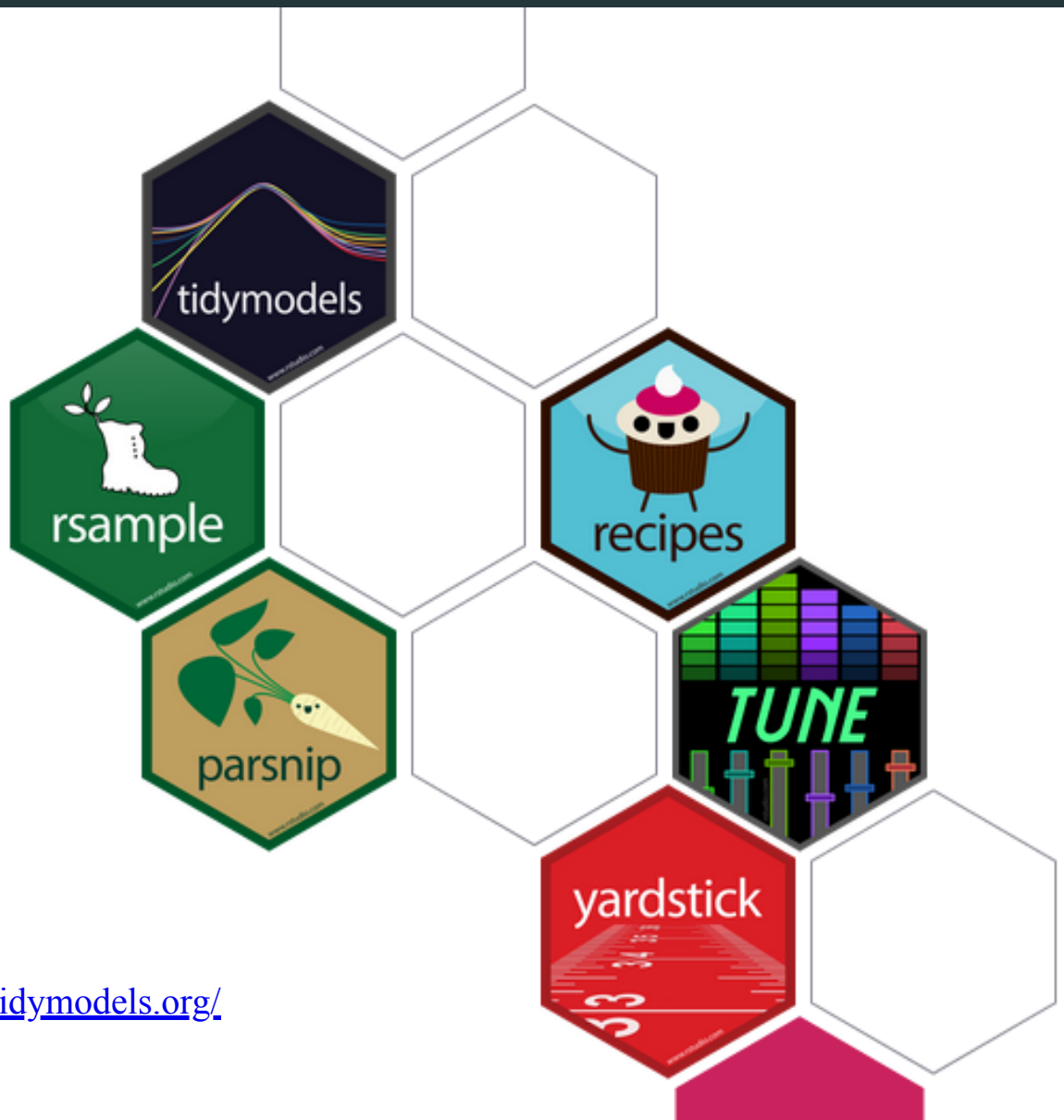
```
# package
library(broom)

# observacoes do modelo
broom::augment(lmfit, data = iris)
```

```
## # A tibble: 150 x 12
##    Sepal.Length Sepal.Width Petal.Length Petal.Width Species .fitted .se.fit
##           <dbl>       <dbl>        <dbl>       <dbl> <fct>     <dbl>   <dbl>
##  1          5.1         3.5          1.4         0.2 setosa     4.88  0.0556
##  2          4.9         3            1.4         0.2 setosa     4.88  0.0556
##  3          4.7         3.2          1.3         0.2 setosa     4.84  0.0571
##  4          4.6         3.1          1.5         0.2 setosa     4.92  0.0541
##  5          5           3.6          1.4         0.2 setosa     4.88  0.0556
##  6          5.4         3.9          1.7         0.4 setosa     5.00  0.0511
##  7          4.6         3.4          1.4         0.3 setosa     4.88  0.0556
##  8          5           3.4          1.5         0.2 setosa     4.92  0.0541
##  9          4.4         2.9          1.4         0.2 setosa     4.88  0.0556
## 10          4.9         3.1          1.5         0.1 setosa     4.92  0.0541
```


tidymodels
www.rstudio.com

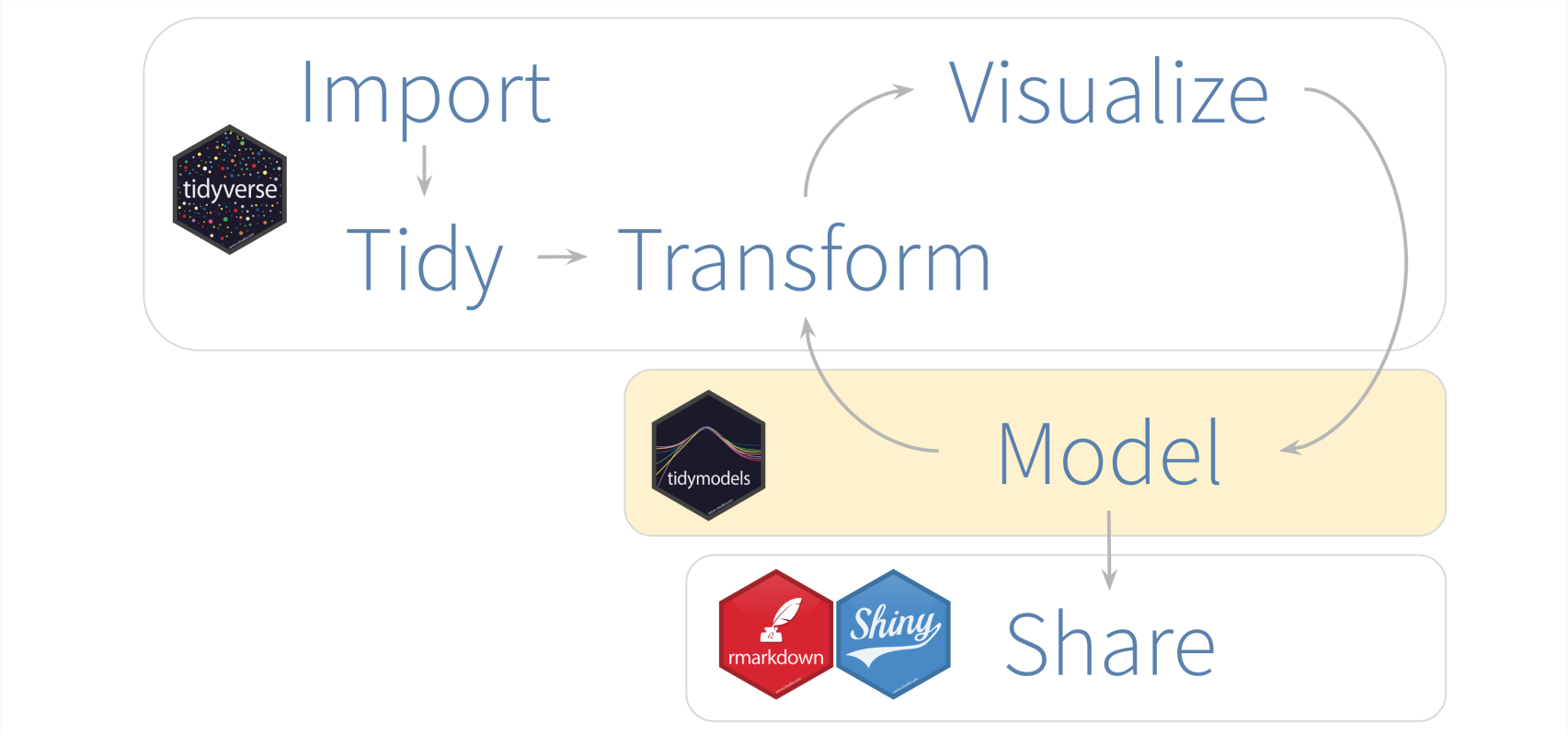
5.7 Pacote tidymodels



[*] https://www.tidymodels.org/

5.7 Pacote tidymodels

Coleção de pacotes para **modelagem e aprendizado de máquina** usando princípios do **tidyverse**



[*] https://rviews.rstudio.com/2019/06/19/a-gentle-intro-to-tidymodels/

5.7 Pacote tidymodels

Principais Pacotes

- rsample - Diferentes tipos de reamostragens
- recipes - Transformações para pré-processamento de dados do modelo
- parnip - Uma interface comum para criação de modelo
- yardstick - Medir o desempenho do modelo

[*] https://rviews.rstudio.com/2019/06/19/a-gentle-intro-to-tidymodels/

5.7 Pacote tidymodels

```
library(tidymodels)
```

```
## Registered S3 method overwritten by 'parsnip':
##   method           from
##   print.nullmodel vegan

## ── Attaching packages ──────────────────────────────────────────────────────

## ✓ dials     0.0.6      ✓ rsample   0.0.6
## ✓ infer     0.5.1      ✓ tune      0.1.0
## ✓ parsnip   0.1.0      ✓ workflows 0.1.1
## ✓ recipes   0.1.10     ✓ yardstick 0.0.6

## ── Conflicts ─────────────────────────────────────────────────────────────
## x psych::%+%()           masks ggplot2::%+%()
## x scales::alpha()        masks psych::alpha(), ggplot2::alpha()
## x recipes::check()       masks permute::check()
## x scales::discard()      masks purrr::discard()
## x magrittr::extract()    masks tidyr::extract()
## x dplyr::filter()        masks stats::filter()
## x recipes::fixed()       masks stringr::fixed()
```

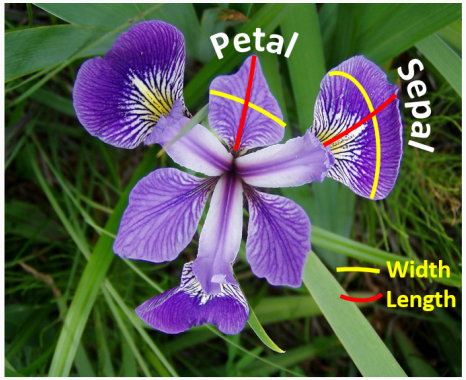
5.7 Pacote tidymodels

iris data

```
# data
dplyr::glimpse(iris)
```

```
## Rows: 150
## Columns: 5
## $ Sepal.Length <dbl> 5.1, 4.9, 4.7, 4.6, 5.0, 5.4, 4.6, 5.0, 4.4, 4.9, 5.4, 4
## $ Sepal.Width  <dbl> 3.5, 3.0, 3.2, 3.1, 3.6, 3.9, 3.4, 3.4, 2.9, 3.1, 3.7, 3
## $ Petal.Length <dbl> 1.4, 1.4, 1.3, 1.5, 1.4, 1.7, 1.4, 1.5, 1.4, 1.5, 1.5, 1
## $ Petal.Width  <dbl> 0.2, 0.2, 0.2, 0.2, 0.2, 0.4, 0.3, 0.2, 0.2, 0.1, 0.2, 0
## $ Species      <fct> setosa, setosa, setosa, setosa, setosa, setosa, setosa, s
```

Iris Virginica



Iris Versicolor

Iris Setosa

5.8 Pacote rsample

Data Sampling

```
# Data Sampling (rsample)
iris_split <- rsample::initial_split(iris, prop = 0.6)
iris_split
```

```
## <Training/Validation/Total>
## <90/60/150>
```

5.8 Pacote rsample

Data Sampling

```
# train
iris_split %>%
  rsample::training() %>%
  dplyr::glimpse()
```

```
## Rows: 90
## Columns: 5
## $ Sepal.Length <dbl> 5.1, 4.9, 4.7, 5.0, 4.4, 4.9, 5.4, 4.8, 5.8, 5.7, 5.1, 5
## $ Sepal.Width  <dbl> 3.5, 3.0, 3.2, 3.6, 2.9, 3.1, 3.7, 3.4, 4.0, 4.4, 3.5, 3
## $ Petal.Length <dbl> 1.4, 1.4, 1.3, 1.4, 1.4, 1.5, 1.5, 1.6, 1.2, 1.5, 1.4, 1
## $ Petal.Width  <dbl> 0.2, 0.2, 0.2, 0.2, 0.2, 0.1, 0.2, 0.2, 0.2, 0.4, 0.3, 0
## $ Species      <fct> setosa, setosa, setosa, setosa, setosa, setosa, setosa, s
```

5.8 Pacote rsample

Data Sampling

```
# test
iris_split %>%
  rsample::testing() %>%
  dplyr::glimpse()
```

```
## Rows: 60
## Columns: 5
## $ Sepal.Length <dbl> 4.6, 5.4, 4.6, 5.0, 4.8, 4.3, 5.4, 5.1, 4.6, 5.1, 4.8, 4
## $ Sepal.Width  <dbl> 3.1, 3.9, 3.4, 3.4, 3.0, 3.0, 3.9, 3.7, 3.6, 3.3, 3.4, 3
## $ Petal.Length <dbl> 1.5, 1.7, 1.4, 1.5, 1.4, 1.1, 1.3, 1.5, 1.0, 1.7, 1.9, 1
## $ Petal.Width  <dbl> 0.2, 0.4, 0.3, 0.2, 0.1, 0.1, 0.4, 0.4, 0.2, 0.5, 0.2, 0
## $ Species      <fct> setosa, setosa, setosa, setosa, setosa, setosa, setosa, s
```

5.9 Pacote recipes

```
# Pre-process interface (recipes)
iris_recipe <- rsample::training(iris_split) %>%
  recipes::recipe(Species ~.) %>%
  recipes::step_corr(all_predictors()) %>%
  recipes::step_center(all_predictors(), -all_outcomes()) %>%
  recipes::step_scale(all_predictors(), -all_outcomes()) %>%
  recipes::prep()
iris_recipe
```

```
## Data Recipe
##
## Inputs:
##
##       role #variables
##    outcome           1
##  predictor           4
##
## Training data contained 90 data points and no missing data.
##
## Operations:
##
```

5.9 Pacote recipes

```
# test data
iris_testing <- iris_recipe %>%
  recipes::bake(rsample::testing(iris_split))
dplyr::glimpse(iris_testing)
```

```
## Rows: 60
## Columns: 4
## $ Sepal.Length <dbl> -1.51055730, -0.51367255, -1.51055730, -1.01211492, -1.26
## $ Sepal.Width  <dbl> 0.08563602, 1.84729135, 0.74625677, 0.74625677, -0.13457
## $ Petal.Width  <dbl> -1.2404295, -0.9795916, -1.1100105, -1.2404295, -1.37084
## $ Species      <fct> setosa, setosa, setosa, setosa, setosa, setosa, setosa, s
```

5.9 Pacote recipes

```
# train data
iris_training <- recipes::juice(iris_recipe)
dplyr::glimpse(iris_training)
```

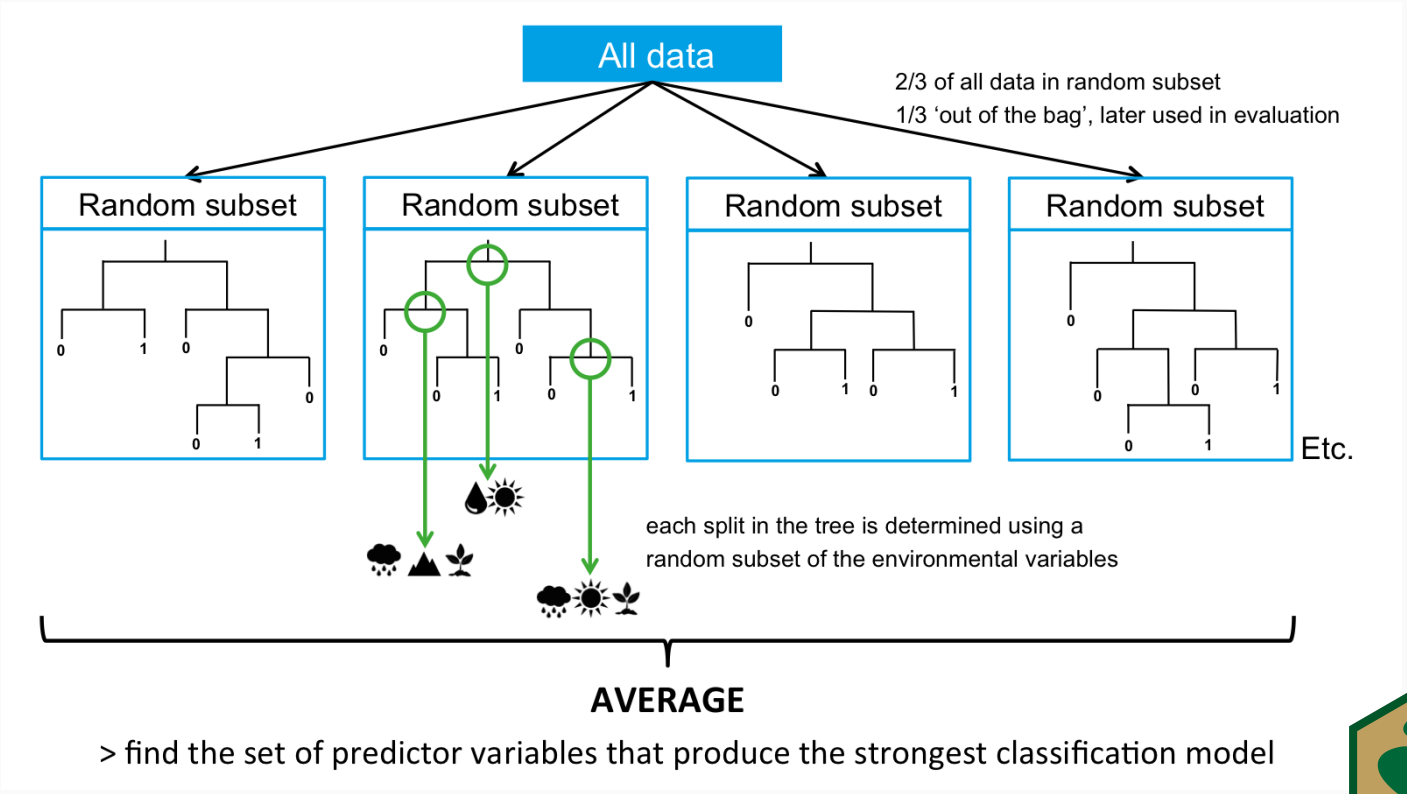
```
## Rows: 90
## Columns: 4
## $ Sepal.Length <dbl> -0.88750433, -1.13672552, -1.38594670, -1.01211492, -1.75
## $ Sepal.Width  <dbl> 0.96646369, -0.13457089, 0.30584294, 1.18667060, -0.3547
## $ Petal.Width  <dbl> -1.2404295, -1.2404295, -1.2404295, -1.2404295, -1.24042
## $ Species      <fct> setosa, setosa, setosa, setosa, setosa, setosa, setosa, s
```

5.10 Pacote parsnip

```
# Model Training (parsnip)
iris_rf <- parsnip::rand_forest(trees = 100, mode = "classification") %>%
  parsnip::set_engine("randomForest") %>%
  parsnip::fit(Species ~ ., data = iris_training)
```

5.10 Pacote parsnip

Random Forest



> find the set of predictor variables that produce the strongest classification model

5.10 Pacote parsnip

```
# Predictions
stats::predict(iris_rf, iris_testing)
```

```
## # A tibble: 60 x 1
##    .pred_class
##    <fct>
##  1 setosa
##  2 setosa
##  3 setosa
##  4 setosa
##  5 setosa
##  6 setosa
##  7 setosa
##  8 setosa
##  9 setosa
## 10 setosa
## # … with 50 more rows
```

5.10 Pacote parsnip

```
iris_rf %>%
  stats::predict(iris_testing) %>%
  dplyr::bind_cols(iris_testing) %>%
  dplyr::glimpse()
```

```
## Rows: 60
## Columns: 5
## $ .pred_class  <fct> setosa, setosa, setosa, setosa, setosa, setosa, setosa, s
## $ Sepal.Length <dbl> -1.51055730, -0.51367255, -1.51055730, -1.01211492, -1.26
## $ Sepal.Width  <dbl> 0.08563602, 1.84729135, 0.74625677, 0.74625677, -0.13457
## $ Petal.Width  <dbl> -1.2404295, -0.9795916, -1.1100105, -1.2404295, -1.37084
## $ Species      <fct> setosa, setosa, setosa, setosa, setosa, setosa, setosa, s
```

5.11 Pacote tyardstick

```
# Model Validation (yardstick)
iris_rf %>%
  stats::predict(iris_testing) %>%
  dplyr::bind_cols(iris_testing) %>%
  yardstick::metrics(truth = Species, estimate = .pred_class)
```

```
## # A tibble: 2 x 3
##   .metric  .estimator .estimate
##   <chr>    <chr>          <dbl>
## 1 accuracy multiclass     0.967
## 2 kap      multiclass     0.950
```

5.11 Pacote tyardstick

```
# Per classifier metrics
iris_probs <- iris_rf %>%
  stats::predict(iris_testing, type = "prob") %>%
  dplyr::bind_cols(iris_testing)
dplyr::glimpse(iris_probs)
```

```
## Rows: 60
## Columns: 7
## $ .pred_setosa     <dbl> 1.00, 0.99, 1.00, 1.00, 0.98, 0.99, 0.99, 1.00, 1.00,
## $ .pred_versicolor <dbl> 0.00, 0.01, 0.00, 0.00, 0.02, 0.01, 0.01, 0.00, 0.00,
## $ .pred_virginica  <dbl> 0.00, 0.00, 0.00, 0.00, 0.00, 0.00, 0.00, 0.00, 0.00,
## $ Sepal.Length     <dbl> -1.51055730, -0.51367255, -1.51055730, -1.01211492, -
## $ Sepal.Width      <dbl> 0.08563602, 1.84729135, 0.74625677, 0.74625677, -0.13
## $ Petal.Width      <dbl> -1.2404295, -0.9795916, -1.1100105, -1.2404295, -1.37
## $ Species          <fct> setosa, setosa, setosa, setosa, setosa, setosa, setos
```

5.11 Pacote tyardstick

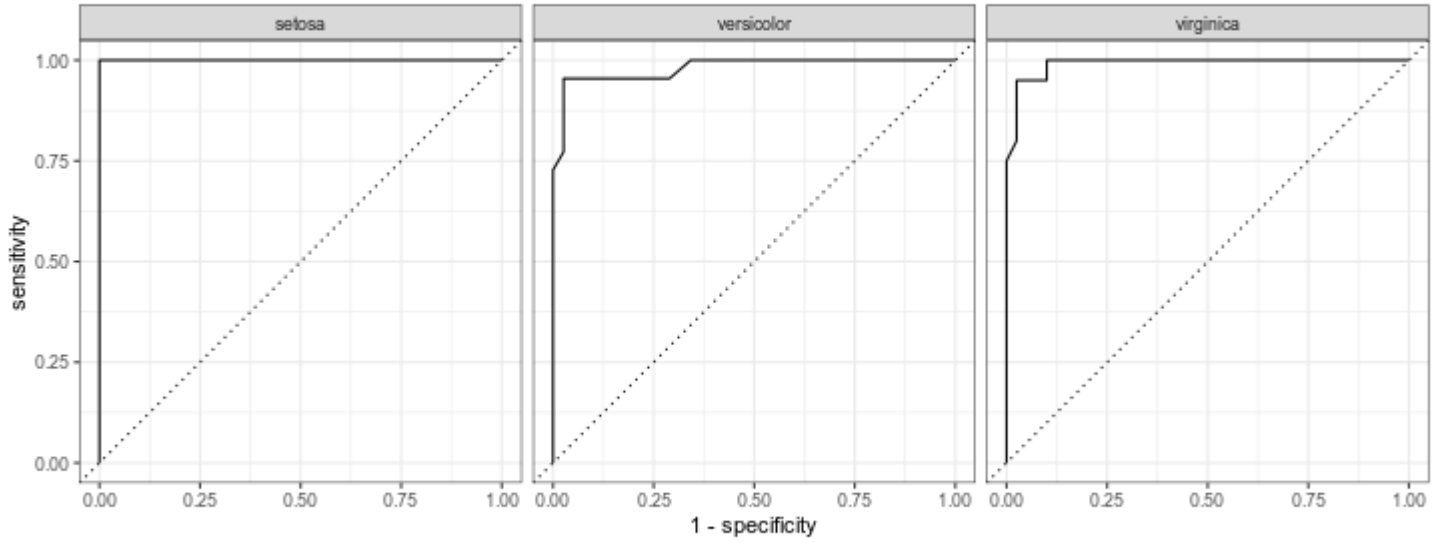
```
iris_probs %>%
  yardstick::gain_curve(Species, .pred_setosa:.pred_virginica) %>%
  dplyr::glimpse()
```

```
## Rows: 75
## Columns: 5
## $ .level          <chr> "setosa", "setosa", "setosa", "setosa", "setosa", "set
## $ .n              <dbl> 0, 7, 12, 14, 16, 17, 18, 19, 20, 21, 22, 23, 25, 29,
## $ .n_events       <dbl> 0, 7, 12, 14, 16, 17, 18, 18, 18, 18, 18, 18, 18, 18,
## $ .percent_tested <dbl> 0.000000, 11.666667, 20.000000, 23.333333, 26.666667,
## $ .percent_found  <dbl> 0.000000, 38.888889, 66.666667, 77.777778, 88.888889,
```

5.11 Pacote tyardstick

ROC e AUC

```
# roc e auc
iris_probs %>%
  yardstick::roc_curve(Species, .pred_setosa:.pred_virginica) %>%
  ggplot2::autoplot()
```



5.11 Pacote tyardstick

```
# predicao
predict(iris_rf, iris_testing, type = "prob") %>%
  dplyr::bind_cols(predict(iris_rf, iris_testing)) %>%
  dplyr::bind_cols(select(iris_testing, Species)) %>%
  dplyr::glimpse()
```

```
## Rows: 60
## Columns: 5
## $ .pred_setosa     <dbl> 1.00, 0.99, 1.00, 1.00, 0.98, 0.99, 0.99, 1.00, 1.00,
## $ .pred_versicolor <dbl> 0.00, 0.01, 0.00, 0.00, 0.02, 0.01, 0.01, 0.00, 0.00,
## $ .pred_virginica  <dbl> 0.00, 0.00, 0.00, 0.00, 0.00, 0.00, 0.00, 0.00, 0.00,
## $ .pred_class      <fct> setosa, setosa, setosa, setosa, setosa, setosa, setos
## $ Species          <fct> setosa, setosa, setosa, setosa, setosa, setosa, setos
```

5.11 Pacote tyardstick

```
predict(iris_rf, iris_testing, type = "prob") %>%
  dplyr::bind_cols(predict(iris_rf, iris_testing)) %>%
  dplyr::bind_cols(select(iris_testing, Species)) %>%
  metrics(Species, .pred_setosa:.pred_virginica, estimate = .pred_class)
```

```
## # A tibble: 4 x 3
##   .metric     .estimator .estimate
##   <chr>       <chr>          <dbl>
## 1 accuracy    multiclass     0.967
## 2 kap         multiclass     0.950
## 3 mn_log_loss multiclass     0.178
## 4 roc_auc     hand_till      0.991
```

Maurício Vancine

Contatos:

mauricio.vancine@gmail.com
mauriciovancine
mauriciovancine.netlify.com

Slides criados via pacote xaringan e tema Metropolis