

Supplementary Material for ‘Evaluating the Generalizability of Machine Learning Pipelines When Using Lexicase or Tournament Selection’

Anonymous

2025-01-29

Contents

1	Introduction	7
1.1	About our supplemental material	7
2	Helper functions	9
2.1	Setup	9
2.2	Permutaiton test	10
2.3	Test accuracy plot	13
2.4	Test accuracy results summary	14
2.5	Validation accuracy plot	14
2.6	Validation accuracy results summary	15
2.7	Complexity plot	15
2.8	Complexity results summary	16
3	Task 146818	17
3.1	5%	17
3.2	10%	22
3.3	50%	28
3.4	90%	34
3.5	95%	40
4	Task 359954	47
4.1	5%	47
4.2	10%	52
4.3	50%	58
4.4	90%	64
4.5	95%	70
5	Task 359955	77
5.1	5%	77
5.2	10%	82
5.3	50%	88
5.4	90%	94
5.5	95%	100

6 Task 190146	107
6.1 5%	107
6.2 10%	112
6.3 50%	118
6.4 90%	124
6.5 95%	130
7 Task 168757	137
7.1 5%	137
7.2 10%	142
7.3 50%	148
7.4 90%	154
7.5 95%	160
8 Task 359956	167
8.1 5%	167
8.2 10%	172
8.3 50%	178
8.4 90%	184
8.5 95%	190
9 Task 359958	197
9.1 5%	197
9.2 10%	202
9.3 50%	208
9.4 90%	214
9.5 95%	220
10 Task 359959	227
10.1 5%	227
10.2 10%	232
10.3 50%	238
10.4 90%	244
10.5 95%	250
11 Task 2073	257
11.1 5%	257
11.2 10%	262
11.3 50%	268
11.4 90%	274
11.5 95%	280

CONTENTS	5
12 Task 359960	287
12.1 5%	287
12.2 10%	291
12.3 50%	296
12.4 90%	302
12.5 95%	308
13 Task 168784	315
13.1 5%	315
13.2 10%	320
13.3 50%	326
13.4 90%	332
13.5 95%	338
14 Task 359962	345
14.1 5%	345
14.2 10%	350
14.3 50%	356
14.4 90%	362
14.5 95%	368

Chapter 1

Introduction

This is not intended as a stand-alone document, but as a companion to our manuscript.

1.1 About our supplemental material

As you may have noticed (unless you're reading a pdf version of this), our supplemental material is hosted using GitHub pages. We compiled our data analyses and supplemental documentation into this nifty web-accessible book using bookdown.

Our supplemental material includes the following:

- Helper functions (Section 2)
- Task 146818 (Section 3)
- Task 359954 (Section 4)
- Task 359955 (Section 5)
- Task 190146 (Section 6)
- Task 168757 (Section 7)
- Task 359956 (Section 8)
- Task 359958 (Section 9)
- Task 359959 (Section 10)
- Task 2073 (Section 11)
- Task 359960 (Section 12)
- Task 168784 (Section 13)

Chapter 2

Helper functions

Here we show the functions being used to generate the supplementary material. All of the following functions are composed of R code and are used to generate the figures, tables, and statistics.

2.1 Setup

```
library(ggplot2)
library(cowplot)
library(dplyr)
library(PupillometryR)
library(scales)

NAMES = c('tournament', 'lexicase')
SHAPE <- c(21, 21)
cb_palette <- c('#DC1E34', '#004D40')
data_dir <- './'
c_task_id_lists <- c(146818,359954,359955,190146,168757,359956,
359958,359959,2073,359960,168784,359962)

p_theme <- theme(
  plot.title = element_text(face = "bold", size = 17, hjust=0.5),
  panel.border = element_blank(),
  panel.grid.minor = element_blank(),
  legend.title=element_text(size=17),
  legend.text=element_text(size=17),
  axis.title = element_text(size=17),
  axis.text = element_text(size=11),
  axis.text.y = element_text(angle = 90, hjust = 0.5),
  legend.position="bottom",
  panel.background = element_rect(fill = "#f1f2f5",
                                    colour = "white",
                                    size = 0.5, linetype = "solid"))

results <- read.csv("./data.csv")
results$selection <- factor(results$selection, levels = NAMES)
```

2.2 Permutation test

```

# permutation test with t-test statistic
# assuming we are using an alpha of 0.05
permutation_test <- function(x, y, seed, alternative) {
  # Set the random seed for reproducibility
  set.seed(seed)

  # Number of permutations
  n_permutations = 100000

  # Calculate the observed difference in means
  observed_diff <- t.test(x, y, var.equal = FALSE)$statistic
  print(paste('observed_diff:', observed_diff))

  # Combine both samples
  combined <- c(x, y)
  n_x <- length(x)

  # Generate permutation differences
  permutation_diffs <- numeric(n_permutations)

  # Use a reproducible random sequence for each permutation
  # Generate unique seeds for each permutation
  seeds <- sample.int(1e9, n_permutations)

  for (i in 1:n_permutations) {
    # Set seed for this permutation
    set.seed(seeds[i])
    # Shuffle the combined data
    permuted <- sample(combined)
    # First n_x elements to group 1
    perm_x <- permuted[1:n_x]
    # Remaining elements to group 2
    perm_y <- permuted[(n_x + 1):length(combined)]
    # Calculate the difference in t-test statistics
    permutation_diffs[i] <- t.test(perm_x, perm_y,
                                    var.equal = FALSE)$statistic
  }

  # sort permutation_diffs
  permutation_diffs <- sort(permutation_diffs)

  if (alternative == "1") {
    # is the observed difference < than the 5th percentile
    print(paste('permutation_diffs[0.05 * n_permutations]:',
               permutation_diffs[0.05 * n_permutations]))

    if (observed_diff < permutation_diffs[0.05 * n_permutations]) {
      print('reject null hypothesis')
    }
    else {
      print('fail to reject null hypothesis')
    }
  }
}

```

```

# if p_value is 0
p_value <- mean(permuation_diffs < observed_diff)
if (p_value == 0.0) {
  print(paste('p-value:', 1/n_permutations))
} else {
  print(paste('p-value:', p_value))
}

# make histogram plot
df <- data.frame(difference = permuation_diffs)
df$category <- ifelse(df$difference < observed_diff, 'not', 'extreme')

plot <- ggplot(df, aes(x = difference, fill = category)) +
  geom_histogram(bins = 100,
                 color = "black",
                 alpha = 0.7) +
  geom_vline(xintercept = observed_diff,
             color = "red",
             linetype = "dotted",
             size = 1
            ) +
  labs(title = "Permutation Test: Frequency of T-test Statistic Differences",
       x = "Difference in t-test statistics",
       y = "Frequency"
      ) +
  theme_minimal() +
  scale_colour_manual(values = c('black', 'green')) +
  scale_fill_manual(values = c('black', 'green'))

print(plot)

} else if (alternative == "g") {
  # is the observed difference > than the 95th percentile
  print(paste('permuation_diffs[0.95 * n_permutations]:',
             permuation_diffs[0.95 * n_permutations]))

  if (permuation_diffs[0.95 * n_permutations] < observed_diff) {
    print('reject null hypothesis')
  }
  else{
    print('fail to reject null hypothesis')
  }

  # if p_value is 0
  p_value <- mean(permuation_diffs > observed_diff)
  if (p_value == 0.0) {
    print(paste('p-value:', 1/n_permutations))
  } else {
    print(paste('p-value:', p_value))
  }

  # make histogram plot
  df <- data.frame(difference = permuation_diffs)
  df$category <- ifelse(df$difference < observed_diff, 'not', 'extreme')

  plot <- ggplot(df, aes(x = difference, fill = category)) +

```

```

geom_histogram(bins = 100,
               color = "black",
               alpha = 0.7) +
geom_vline(xintercept = observed_diff,
            color = "red",
            linetype = "dotted",
            size = 1
            ) +
labs(title = "Permutation Test: Frequency of T-test Statistic Differences",
     x = "Difference in t-test statistics",
     y = "Frequency"
     ) +
theme_minimal() +
scale_shape_manual(values = SHAPE,) +
scale_colour_manual(values = c('green', 'black')) +
scale_fill_manual(values = c('green', 'black'))

print(plot)
} else if (alternative == "t") {
  # is the observed difference within 2.5th and 97.5th percentile
  lower <- observed_diff < permutation_diffs[0.025 * n_permutations]
  print(paste('lower:', permutation_diffs[0.025 * n_permutations]))
  upper <- observed_diff > permutation_diffs[0.975 * n_permutations]
  print(paste('upper:', permutation_diffs[0.975 * n_permutations]))

  if (lower | upper) {
    print('reject null hypothesis')
  }
  else{
    print('fail to reject null hypothesis')
  }

  # if p_value is 0
  p_value <- mean(abs(permutation_diffs) > abs(observed_diff))
  if (p_value == 0.0) {
    print(paste('p-value:', 1/n_permutations))
  } else {
    print(paste('p-value:', p_value))
  }

  # make histogram plot
  df <- data.frame(difference = abs(permutation_diffs))
  df$category <- ifelse(df$difference > abs(observed_diff), 'extreme', 'not')

  plot <- ggplot(df, aes(x = difference, fill = category)) +
  geom_histogram(bins = 100,
                color = "black",
                alpha = 0.7) +
  geom_vline(xintercept = abs(observed_diff),
            color = "red",
            linetype = "dotted",
            size = 1
            ) +
  labs(title = "Permutation Test: Frequency of T-test Statistic Differences",
       x = "Difference in t-test statistics",
       y = "Frequency"
       )
}

```

```
  ) +
  theme_minimal() +
  scale_shape_manual(values = SHAPE,) +
  scale_colour_manual(values = c('green', 'black')) +
  scale_fill_manual(values = c('green', 'black'))  
  
print(plot)  
  
} else {  
  stop("Invalid alternative (less, greater, or two-sided.)")  
}  
}
```

2.3 Test accuracy plot

```
test_plot <- function(data) { return(
  ggplot(data,
  aes(x = selection, y = testing_performance, color = selection,
      fill = selection, shape = selection)) +
  geom_flat_violin(position = position_nudge(x = 0.1, y = 0),
                     scale = "width", alpha = 0.2, width = 1.5) +
  geom_boxplot(color = "black", width = .08, outlier.shape = NA,
               alpha = 0.0, linewidth = 0.8,
               position = position_nudge(x = .15, y = 0)) +
  geom_point(position = position_jitter(width = 0.03,
                                         height = 0.0), size = 1.5, alpha = 1.0) +
  scale_y_continuous(
    name = "Accuracy %",
    labels = scales::percent,
  ) +
  scale_x_discrete(
    name = "Selection scheme"
  ) +
  scale_shape_manual(values = SHAPE,) +
  scale_colour_manual(values = cb_palette,) +
  scale_fill_manual(values = cb_palette,) +
  ggtitle('Accuracy on test set') +
  p_theme +
  guides(
    shape=guide_legend(nrow = 1, title.position = "left",
                       title = "Selection scheme"),
    color=guide_legend(nrow = 1, title.position = "left",
                       title = "Selection scheme"),
    fill=guide_legend(nrow = 1, title.position = "left",
                      title = "Selection scheme"))
  theme(axis.ticks.x = element_blank(),
        axis.text.x = element_blank(),
        axis.title.x = element_blank(),
        axis.text.y = element_text(angle = 90, hjust = 0.5))
)}
```

2.4 Test accuracy results summary

```
test_results_summary <- function(data) {  
  return(  
    data %>%  
      group_by(selection) %>%  
      dplyr::summarise(  
        count = n(),  
        na_cnt = sum(is.na(testing_performance)),  
        min = min(testing_performance, na.rm = TRUE),  
        median = median(testing_performance, na.rm = TRUE),  
        mean = mean(testing_performance, na.rm = TRUE),  
        max = max(testing_performance, na.rm = TRUE),  
        IQR = IQR(testing_performance, na.rm = TRUE))  
  )  
}
```

2.5 Validation accuracy plot

```
validation_plot <- function(data) { return(  
  ggplot(data,  
    aes(x = selection, y = training_performance, color = selection,  
        fill = selection, shape = selection)) +  
  geom_flat_violin(position = position_nudge(x = 0.1, y = 0),  
    scale = "width", alpha = 0.2, width = 1.5) +  
  geom_boxplot(color = "black", width = .08, outlier.shape = NA,  
    alpha = 0.0, linewidth = 0.8,  
    position = position_nudge(x = .15, y = 0)) +  
  geom_point(position = position_jitter(width = 0.03,  
    height = 0.0), size = 1.5, alpha = 1.0) +  
  scale_y_continuous(  
    name = "Accuracy %",  
    labels = scales::percent,  
  
) +  
  scale_x_discrete(  
    name = "Selection Scheme"  
) +  
  scale_shape_manual(values = SHAPE) +  
  scale_colour_manual(values = cb_palette,) +  
  scale_fill_manual(values = cb_palette,) +  
  ggtitle('Accuracy on validation set') +  
  p_theme +  
  guides(  
    shape=guide_legend(nrow = 1, title.position = "left",  
      title = "Selection Scheme"),  
    color=guide_legend(nrow = 1, title.position = "left",  
      title = "Selection Scheme"),  
    fill=guide_legend(nrow = 1, title.position = "left",  
      title = "Selection Scheme")) +  
  theme(axis.ticks.x = element_blank(),  
    axis.text.x = element_blank(),
```

```

    axis.title.x = element_blank(),
    axis.text.y = element_text(angle = 90, hjust = 0.5))
)}
```

2.6 Validation accuracy results summary

```

validation_accuracy_summary <- function(data) {
  return(
    data %>%
      group_by(selection) %>%
      dplyr::summarise(
        count = n(),
        na_cnt = sum(is.na(training_performance)),
        min = min(training_performance, na.rm = TRUE),
        median = median(training_performance, na.rm = TRUE),
        mean = mean(training_performance, na.rm = TRUE),
        max = max(training_performance, na.rm = TRUE),
        IQR = IQR(training_performance, na.rm = TRUE)
      )
}
```

2.7 Complexity plot

```

complexity_plot <- function(data) { return(
  ggplot(data,
    aes(x = selection, y = testing_complexity, color = selection,
        fill = selection, shape = selection)) +
    geom_flat_violin(position = position_nudge(x = 0.1, y = 0),
                      scale = "width", alpha = 0.2, width = 1.5) +
    geom_boxplot(color = "black", width = .08, outlier.shape = NA,
                  alpha = 0.0, linewidth = 0.8,
                  position = position_nudge(x = .15, y = 0)) +
    geom_point(position = position_jitter(width = 0.03,
                                           height = 0.0), size = 1.5, alpha = 1.0) +
    scale_y_log10(
      name="Complexity",
      breaks = trans_breaks("log10", function(x) 10^x),
      labels = trans_format("log10", math_format(10^.x)))
  ) +
    scale_x_discrete(
      name = "Selection Scheme"
    ) +
    scale_shape_manual(values = SHAPE,) +
    scale_colour_manual(values = cb_palette,) +
    scale_fill_manual(values = cb_palette,) +
    ggtitle('Pipeline Complexity') +
    p_theme +
    guides(
      shape=guide_legend(nrow = 1, title.position = "left",
                         title = "Selection Scheme"),
```

```
color=guide_legend(nrow = 1, title.position = "left",
                   title = "Selection Scheme"),
fill=guide_legend(nrow = 1, title.position = "left",
                   title = "Selection Scheme")) +
theme(axis.ticks.x = element_blank(),
      axis.text.x = element_blank(),
      axis.title.x = element_blank(),
      axis.text.y = element_text(angle = 90, hjust = 0.5))
)}
```

2.8 Complexity results summary

```
complexity_summary <- function(data) {
  return(
    data %>%
      group_by(selection) %>%
      dplyr::summarise(
        count = n(),
        na_cnt = sum(is.na(testing_complexity)),
        min = min(testing_complexity, na.rm = TRUE),
        median = median(testing_complexity, na.rm = TRUE),
        mean = mean(testing_complexity, na.rm = TRUE),
        max = max(testing_complexity, na.rm = TRUE),
        IQR = IQR(testing_complexity, na.rm = TRUE))
  )
}
```

Chapter 3

Task 146818

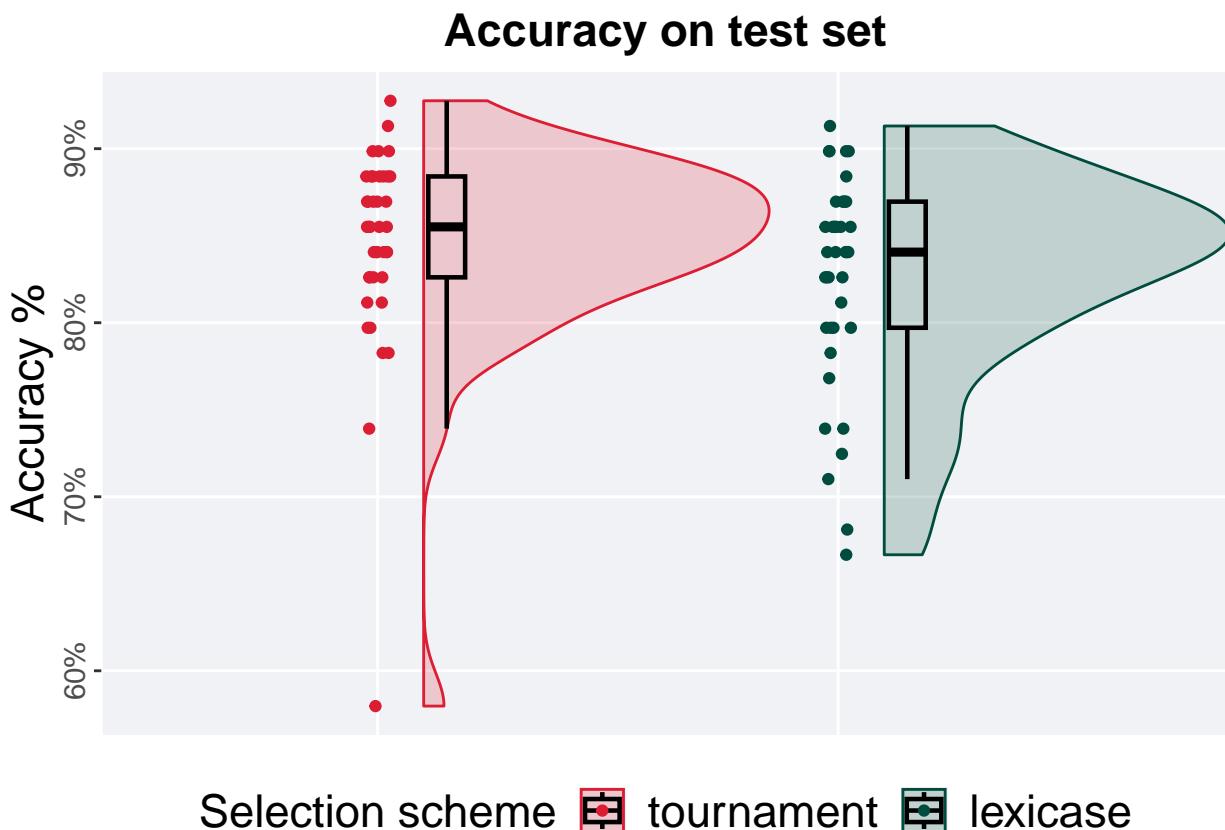
We present the results of our analysis of task 146818 with the different selection set splits used in our study.

```
task_data <- filter(results, task_id == 146818)
```

3.1 5%

3.1.1 Test accuracy

```
test_plot(filter(task_data, split == '5%'))
```



Summary statistics for the testing performance of the selection schemes at the 5% selection set split:

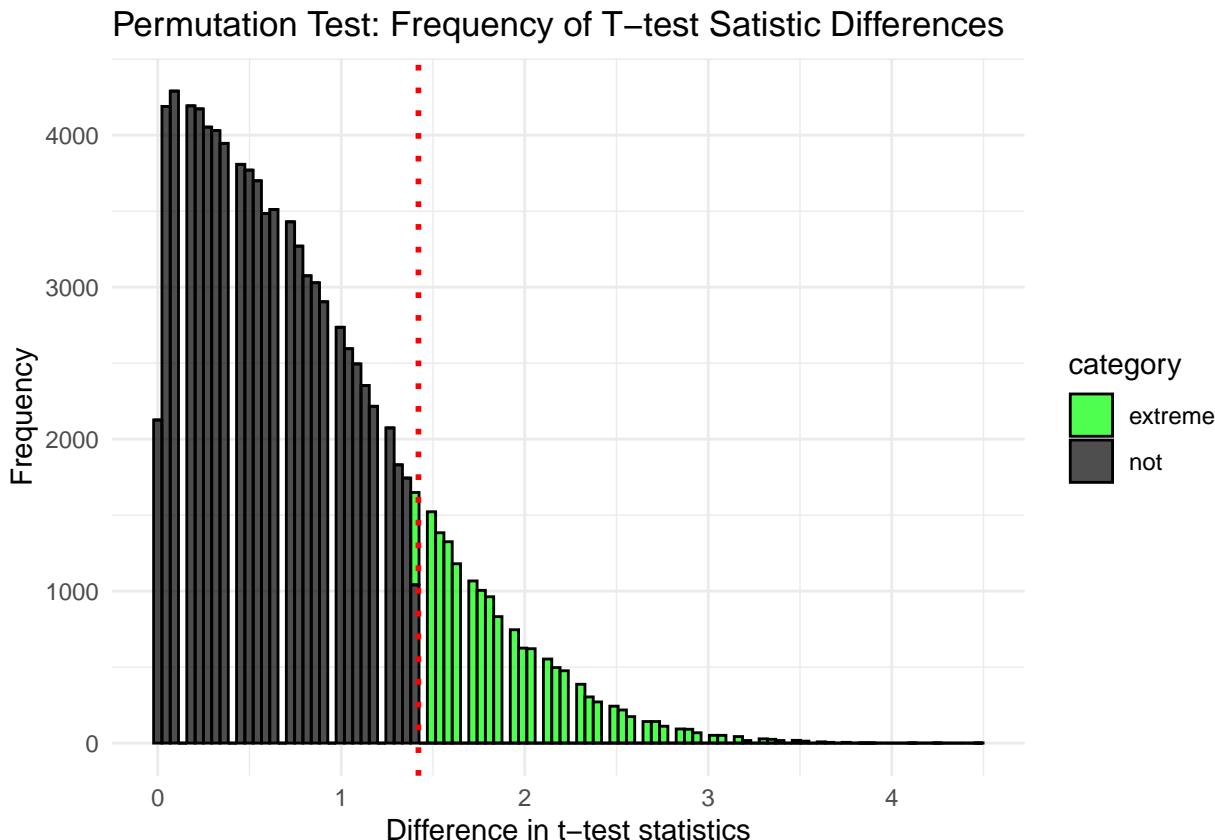
```
test_results_summary(filter(task_data, split == '5%'))
```

```
## # A tibble: 2 x 8
##   selection count na_cnt  min median  mean   max   IQR
##   <fct>     <int> <int> <dbl> <dbl> <dbl> <dbl> <dbl>
## 1 tournament     40     0 0.580 0.855 0.845 0.928 0.0580
## 2 lexicase       40     0 0.667 0.841 0.826 0.913 0.0725
```

The permutation test revealed that the results are:

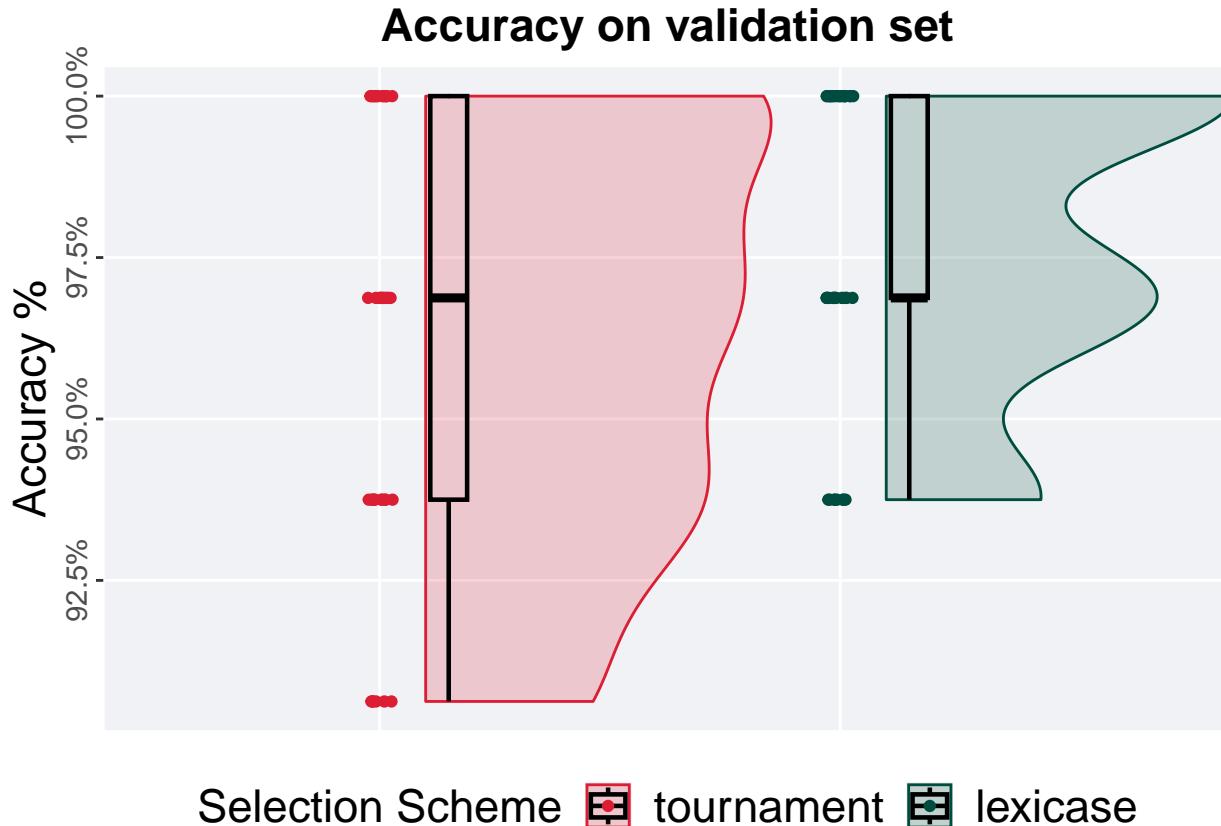
```
tournament_results <- filter(task_data, split == '5%' & selection == 'tournament')
lexicase_results <- filter(task_data, split == '5%' & selection == 'lexicase')
permutation_test(tournament_results$testing_performance,
                 lexicase_results$testing_performance,
                 seed = 1,
                 alternative = "t")
```

```
## [1] "observed_diff: 1.42058620905175"
## [1] "lower: -1.99072649993754"
## [1] "upper: 1.99072649993754"
## [1] "fail to reject null hypothesis"
## [1] "p-value: 0.15934"
```



3.1.2 Validation accuracy

```
validation_plot(filter(task_data, split == '5%'))
```



Summary statistics for the testing performance of the selection schemes at the 5% selection set split:

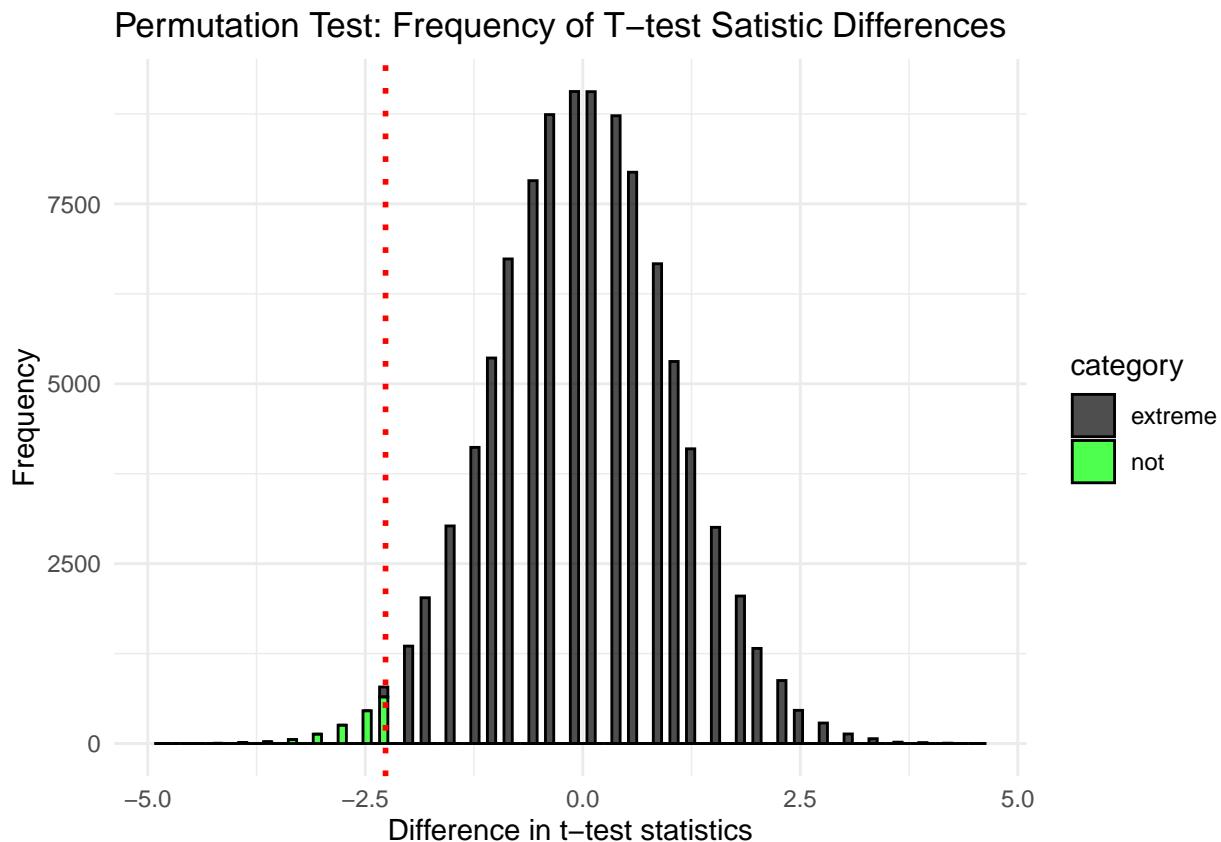
```
validation_accuracy_summary(filter(task_data, split == '5%'))
```

```
## # A tibble: 2 x 8
##   selection  count na_cnt   min median   mean   max     IQR
##   <fct>     <int> <int> <dbl> <dbl> <dbl> <dbl> <dbl>
## 1 tournament     40     0 0.906  0.969  0.962     1  0.0625
## 2 lexicase       40     0 0.938  0.969  0.977     1  0.0312
```

The permutation test revealed that the results are:

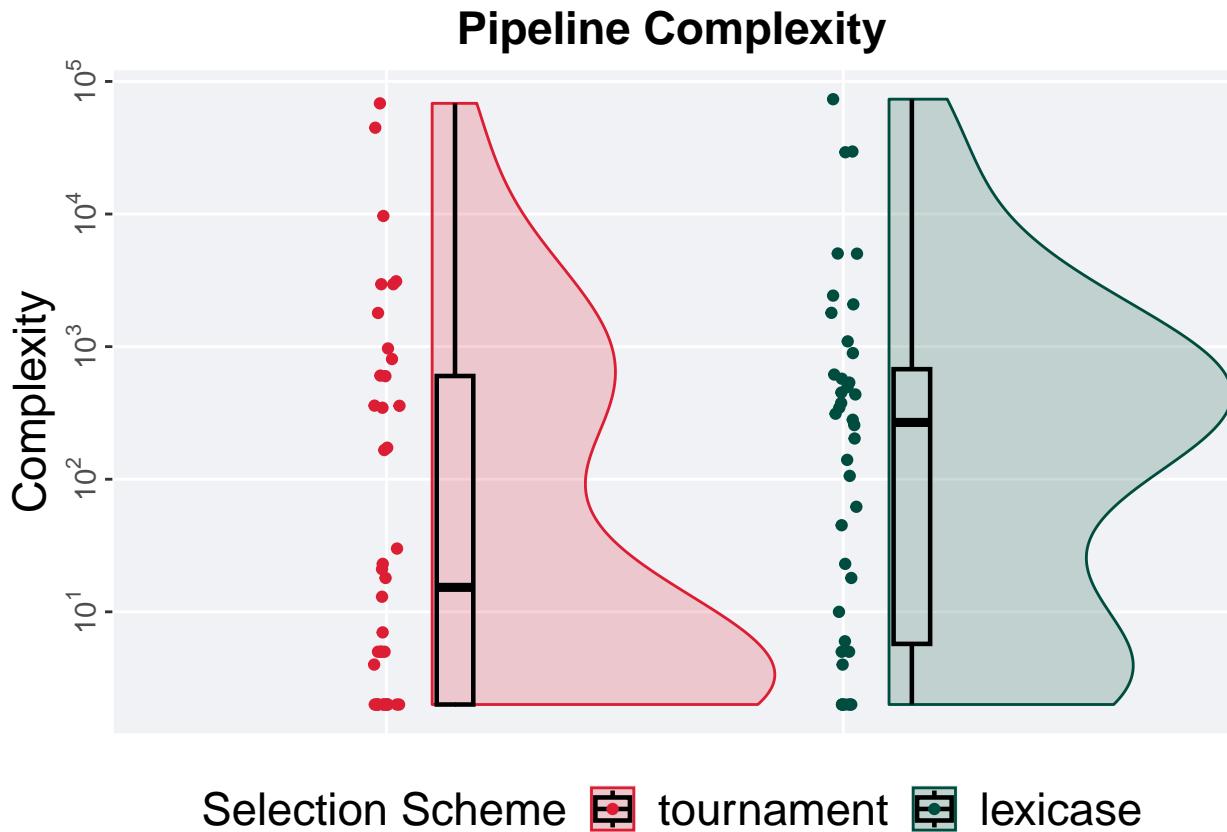
```
tournament_results <- filter(task_data, split == '5%' & selection == 'tournament')
lexicase_results <- filter(task_data, split == '5%' & selection == 'lexicase')
permutation_test(tournament_results$training_performance,
                 lexicase_results$training_performance,
                 seed = 2,
                 alternative = "1")
```

```
## [1] "observed_diff: -2.26720028939778"
## [1] "permutation_diffs[0.05 * n_permutations]: -1.76808181038323"
## [1] "reject null hypothesis"
## [1] "p-value: 0.01586"
```



3.1.3 Pipeline complexity

```
complexity_plot(filter(task_data, split == '5%'))
```



Summary statistics for the pipeline complexity of the selection schemes at the 5% selection set split:

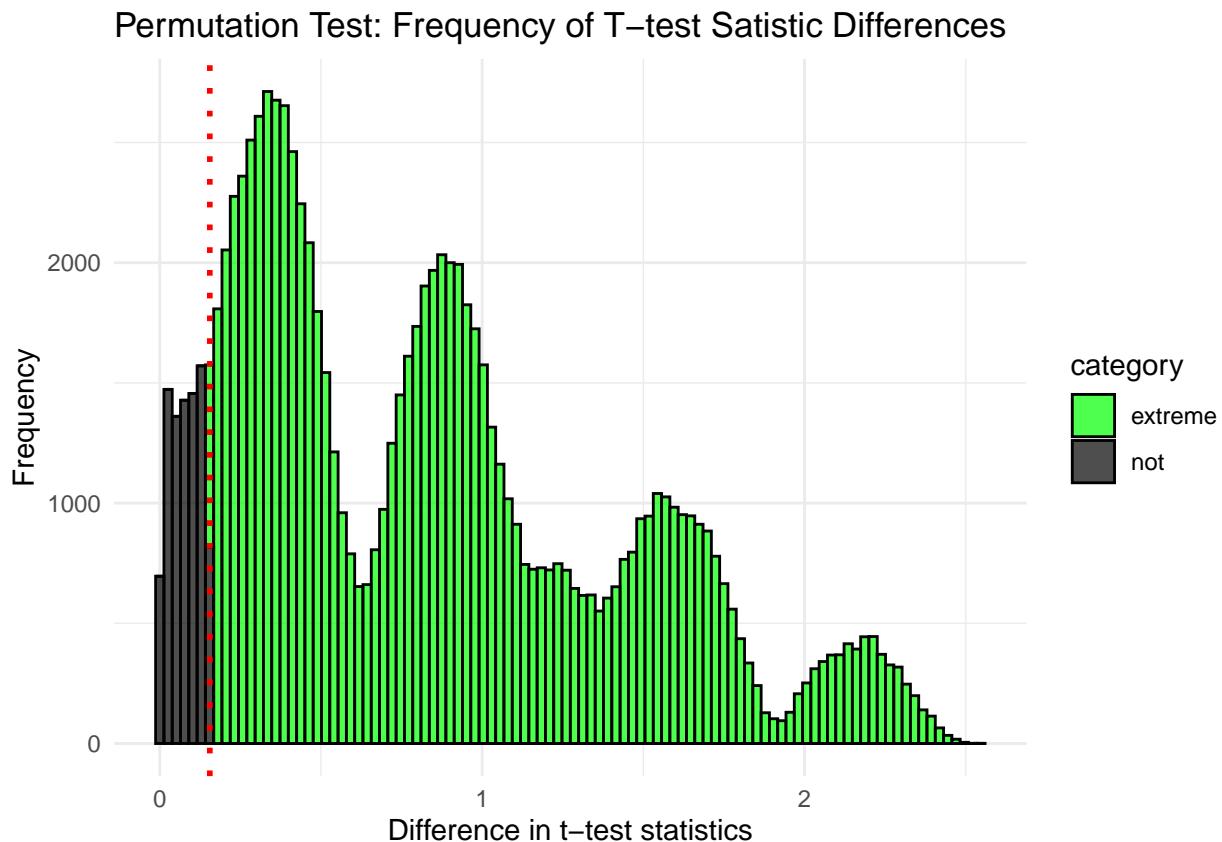
```
complexity_summary(filter(task_data, split == '5%'))
```

```
## # A tibble: 2 x 8
##   selection count na_cnt  min median  mean   max   IQR
##   <fct>     <int> <int> <dbl> <dbl> <dbl> <dbl>
## 1 tournament     40     0     2  15.5 3455. 68371  598.
## 2 lexicase       40     0     2  268.  3901. 73491  680.
```

The permutation test revealed that the results are:

```
tournament_results <- filter(task_data, split == '5%' & selection == 'tournament')
lexicase_results <- filter(task_data, split == '5%' & selection == 'lexicase')
permutation_test(tournament_results$testing_complexity,
                 lexicase_results$testing_complexity,
                 seed = 200,
                 alternative = "t")
```

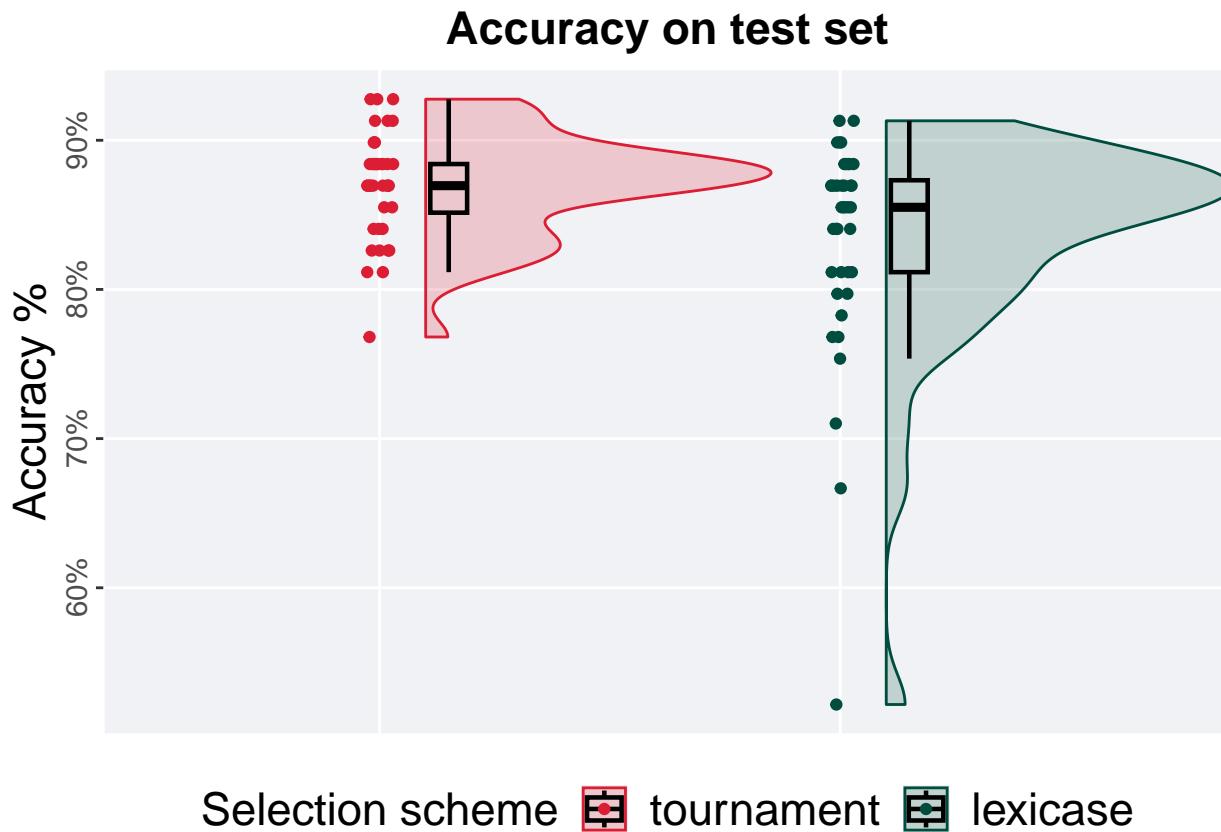
```
## [1] "observed_diff: -0.15520529269803"
## [1] "lower: -2.00626898206809"
## [1] "upper: 2.01796244439775"
## [1] "fail to reject null hypothesis"
## [1] "p-value: 0.91199"
```



3.2 10%

3.2.1 Test accuracy

```
test_plot(filter(task_data, split == '10%'))
```



Summary statistics for the testing performance of the selection schemes at the 5% selection set split:

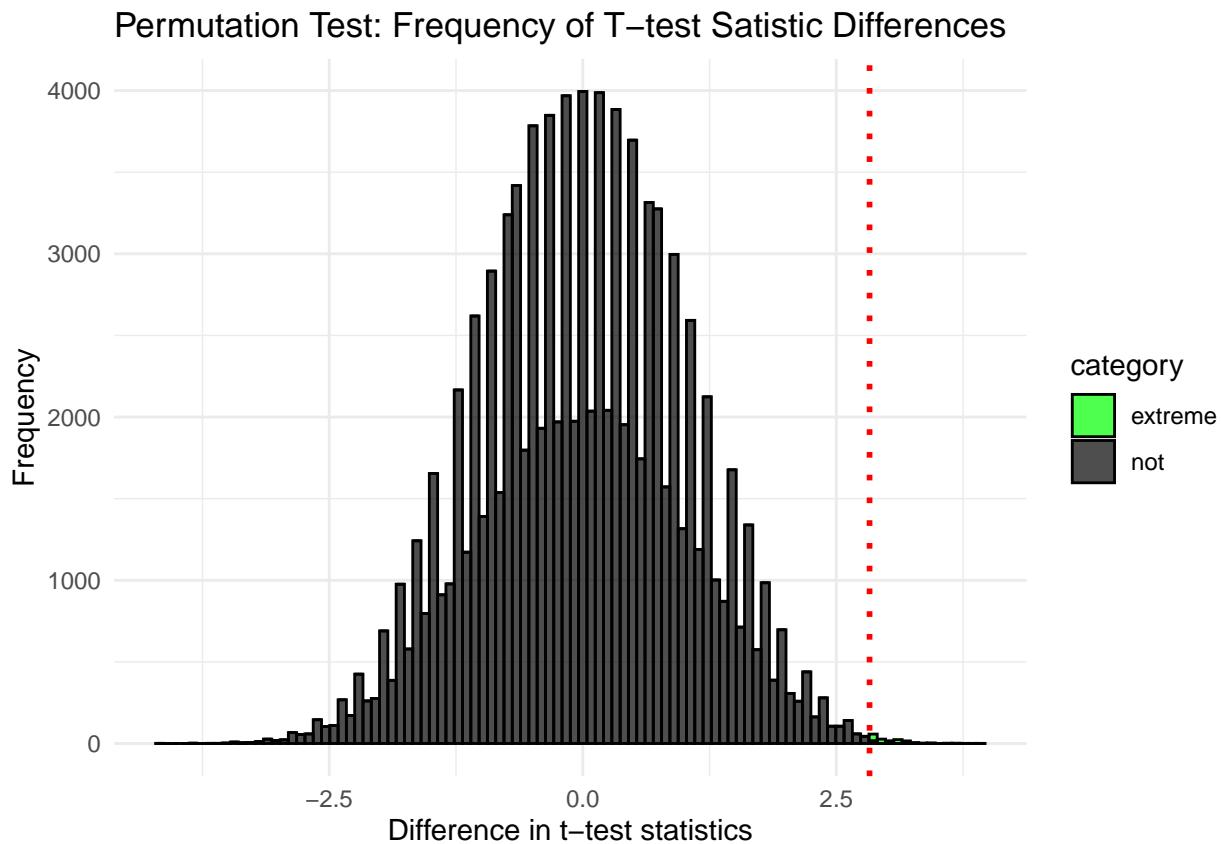
```
test_results_summary(filter(task_data, split == '10%'))
```

```
## # A tibble: 2 x 8
##   selection  count  na_cnt  min  median  mean  max  IQR
##   <fct>      <int>   <int> <dbl>  <dbl> <dbl> <dbl> <dbl>
## 1 tournament     40      0  0.768  0.870  0.870  0.928 0.0326
## 2 lexicase       40      0  0.522  0.855  0.834  0.913 0.0616
```

The permutation test revealed that the results are:

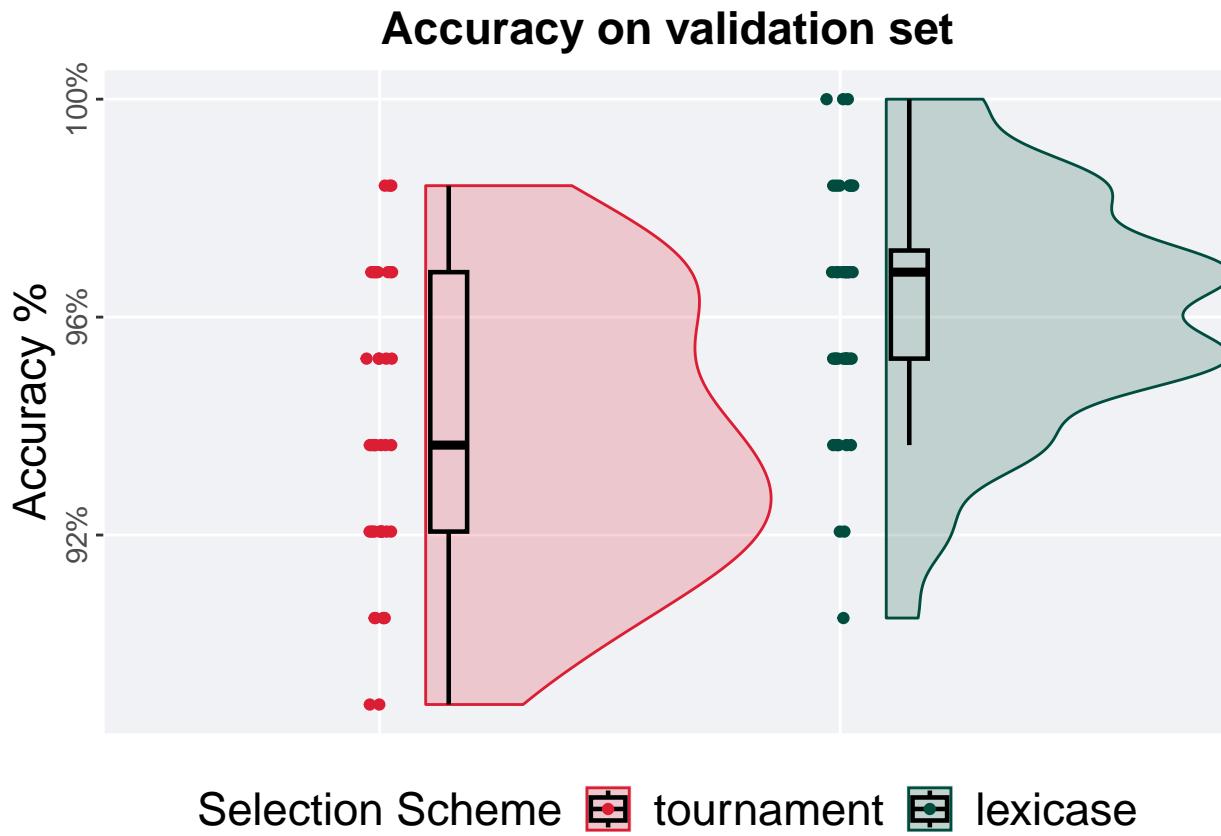
```
tournament_results <- filter(task_data, split == '10%' & selection == 'tournament')
lexicase_results <- filter(task_data, split == '10%' & selection == 'lexicase')
permutation_test(tournament_results$testing_performance,
                 lexicase_results$testing_performance,
                 seed = 3,
                 alternative = "g")
```

```
## [1] "observed_diff: 2.82771300817792"
## [1] "permutation_diffs[0.95 * n_permutations]: 1.65479687273561"
## [1] "reject null hypothesis"
## [1] "p-value: 0.00146"
```



3.2.2 Validation accuracy

```
validation_plot(filter(task_data, split == '10%'))
```



Summary statistics for the testing performance of the selection schemes at the 5% selection set split:

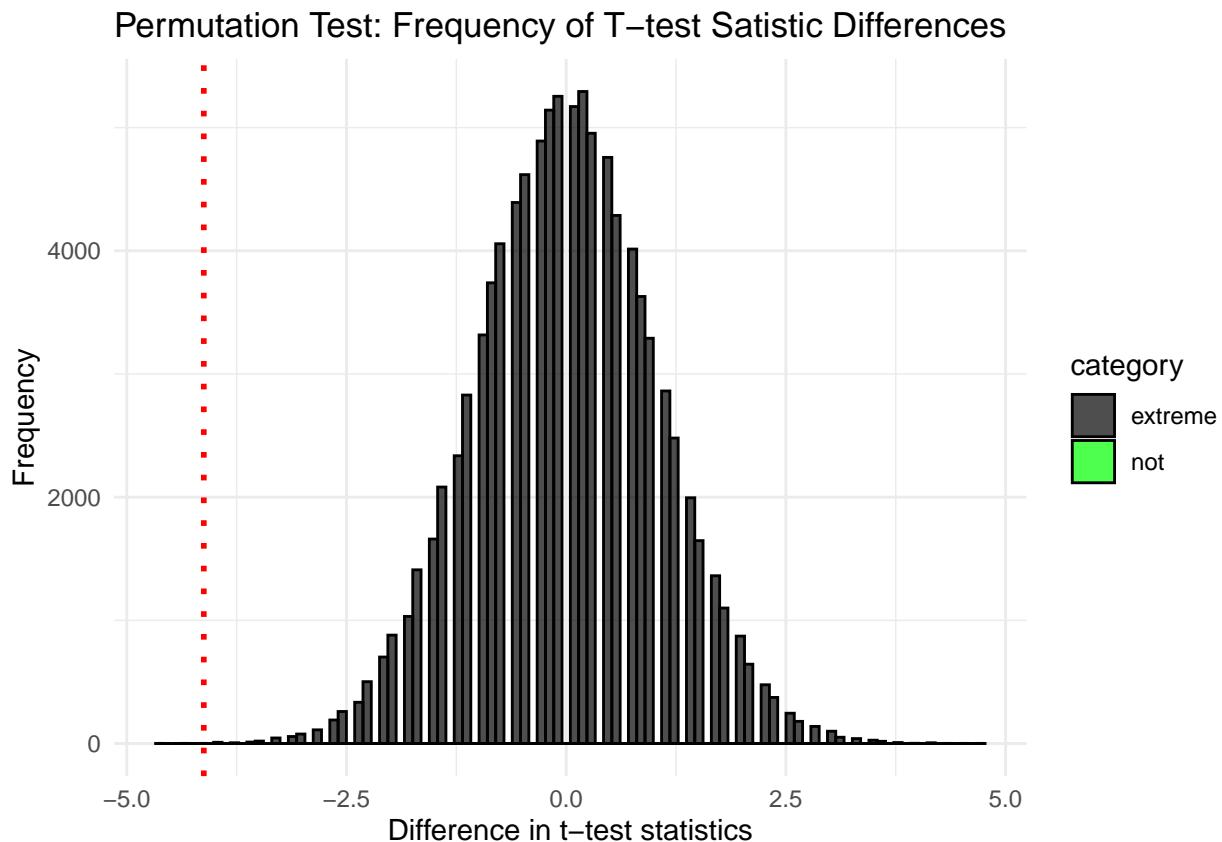
```
validation_accuracy_summary(filter(task_data, split == '10%'))
```

```
## # A tibble: 2 x 8
##   selection count na_cnt  min median  mean  max    IQR
##   <fct>     <int>  <int> <dbl>  <dbl> <dbl> <dbl> <dbl>
## 1 tournament     40      0  0.889  0.937  0.938  0.984  0.0476
## 2 lexicase       40      0  0.905  0.968  0.961  1       0.0198
```

The permutation test revealed that the results are:

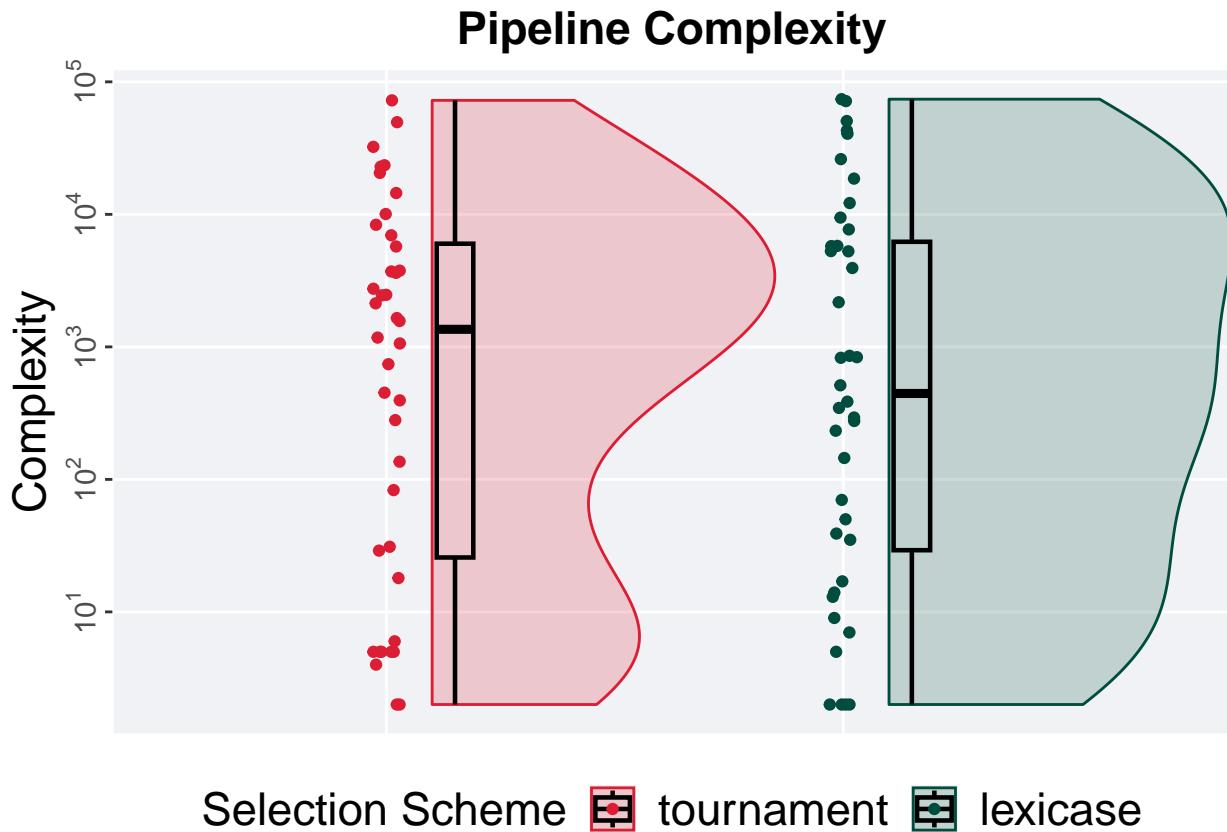
```
tournament_results <- filter(task_data, split == '10%' & selection == 'tournament')
lexicase_results <- filter(task_data, split == '10%' & selection == 'lexicase')
permutation_test(tournament_results$training_performance,
                 lexicase_results$training_performance,
                 seed = 4,
                 alternative = "1")
```

```
## [1] "observed_diff: -4.12382195839694"
## [1] "permutation_diffs[0.05 * n_permutations]: -1.66780738405742"
## [1] "reject null hypothesis"
## [1] "p-value: 2e-05"
```



3.2.3 Pipeline complexity

```
complexity_plot(filter(task_data, split == '10%'))
```



Summary statistics for the pipeline complexity of the selection schemes at the 5% selection set split:

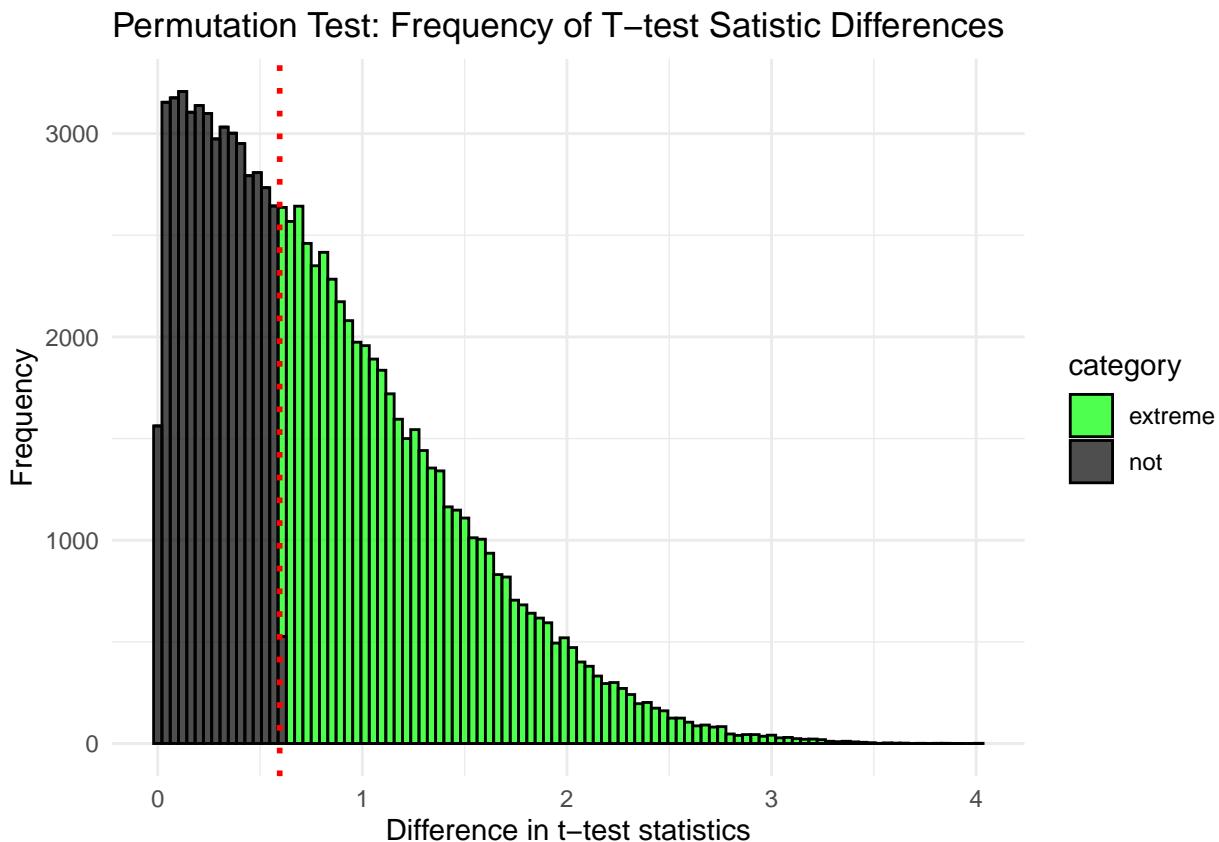
```
complexity_summary(filter(task_data, split == '10%'))
```

```
## # A tibble: 2 x 8
##   selection count na_cnt  min median  mean   max   IQR
##   <fct>     <int> <int> <dbl> <dbl> <dbl> <dbl>
## 1 tournament     40     0     2  1371  7386. 72486 6002.
## 2 lexicase       40     0     2   450.  9674. 73981 6229.
```

The permutation test revealed that the results are:

```
tournament_results <- filter(task_data, split == '10%' & selection == 'tournament')
lexicase_results <- filter(task_data, split == '10%' & selection == 'lexicase')
permutation_test(tournament_results$testing_complexity,
                 lexicase_results$testing_complexity,
                 seed = 201,
                 alternative = "t")

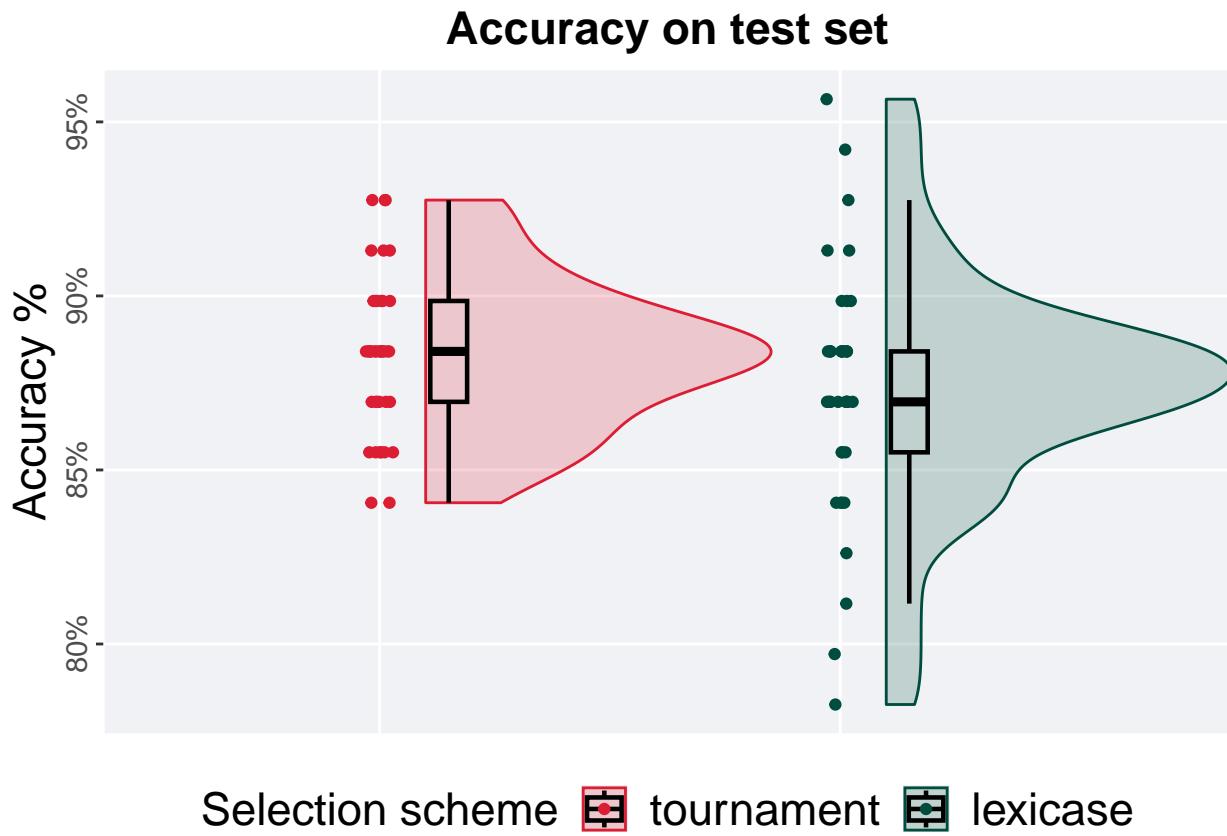
## [1] "observed_diff: -0.595867454849943"
## [1] "lower: -1.98097933747457"
## [1] "upper: 1.96657925417945"
## [1] "fail to reject null hypothesis"
## [1] "p-value: 0.56095"
```



3.3 50%

3.3.1 Test accuracy

```
test_plot(filter(task_data, split == '50%'))
```



Summary statistics for the testing performance of the selection schemes at the 5% selection set split:

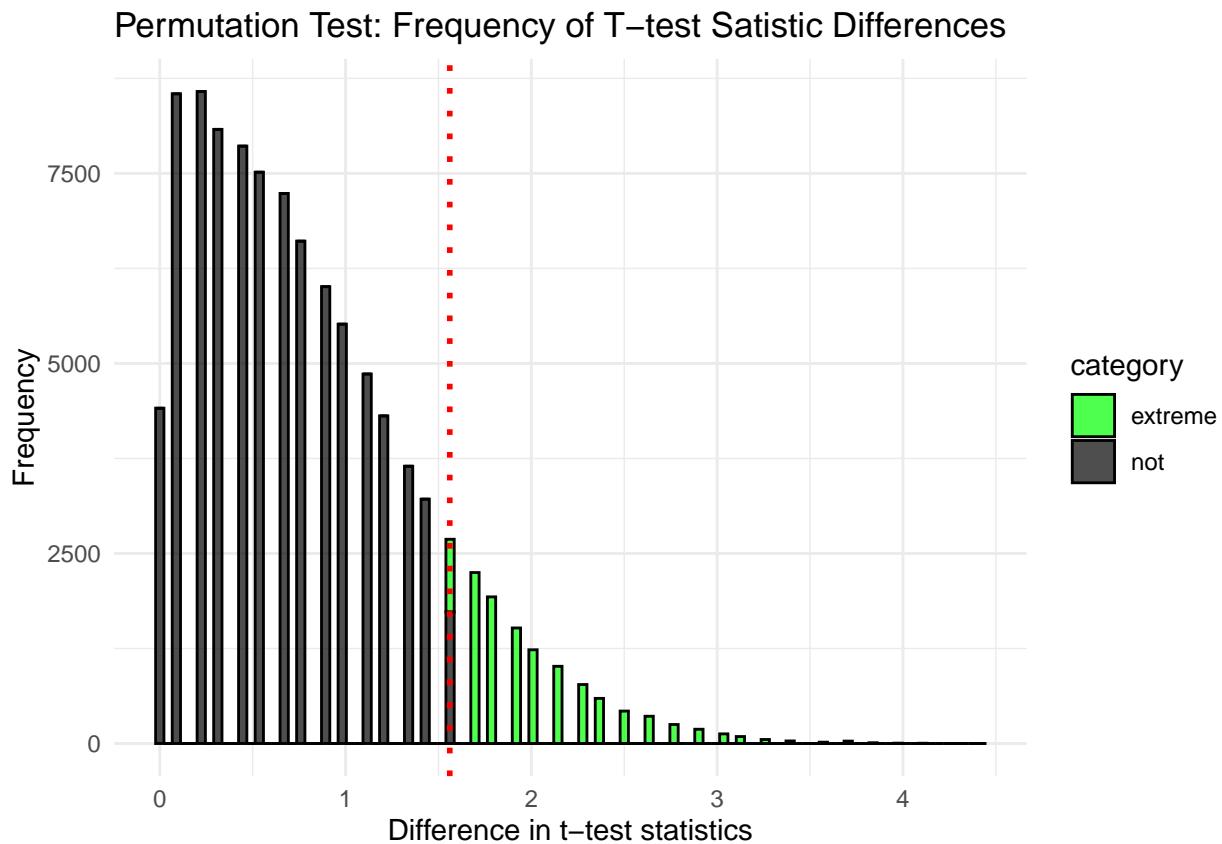
```
test_results_summary(filter(task_data, split == '50%'))
```

```
## # A tibble: 2 x 8
##   selection count na_cnt  min median  mean   max     IQR
##   <fct>     <int> <int> <dbl> <dbl> <dbl> <dbl> <dbl>
## 1 tournament     40     0 0.841 0.884 0.883 0.928 0.0290
## 2 lexicase       40     0 0.783 0.870 0.873 0.957 0.0290
```

The permutation test revealed that the results are:

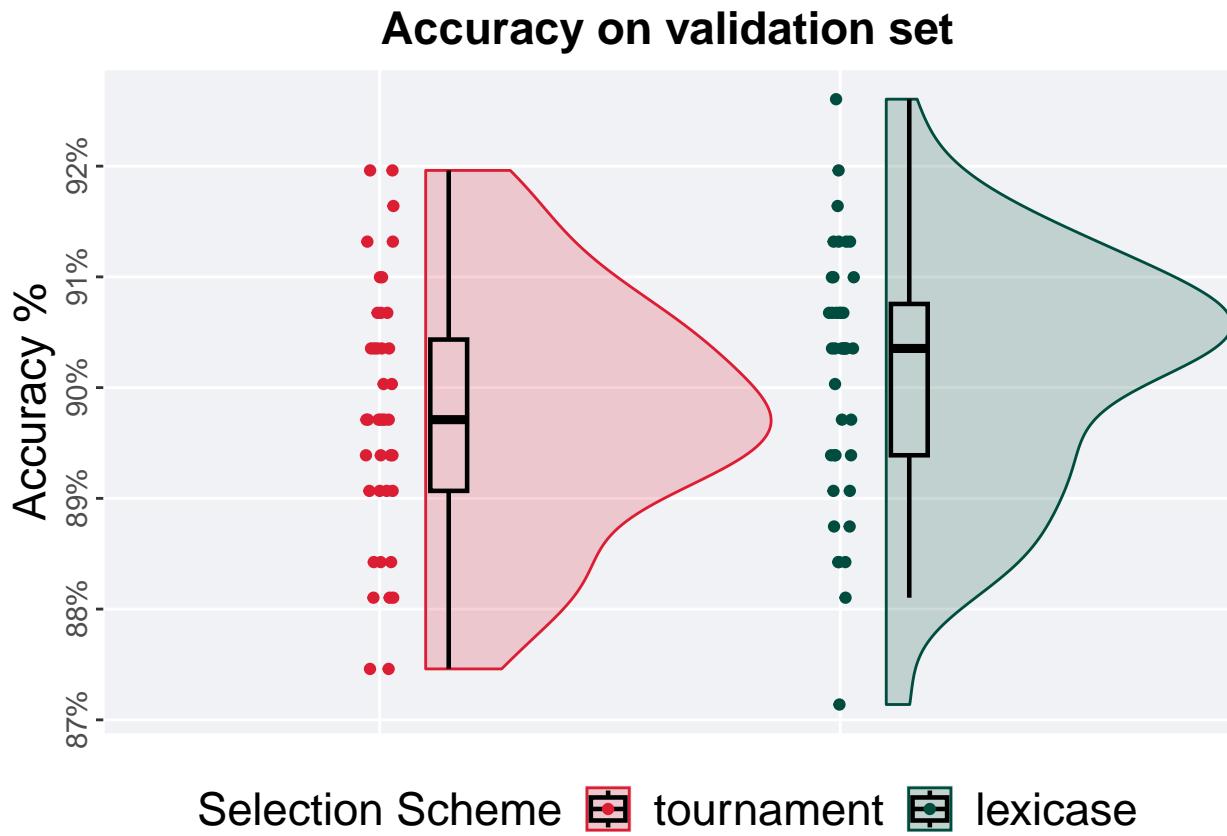
```
tournament_results <- filter(task_data, split == '50%' & selection == 'tournament')
lexicase_results <- filter(task_data, split == '50%' & selection == 'lexicase')
permutation_test(tournament_results$testing_performance,
                 lexicase_results$testing_performance,
                 seed = 5,
                 alternative = "t")
```

```
## [1] "observed_diff: 1.5608765017161"
## [1] "lower: -2.02762682929956"
## [1] "upper: 2.02762732023459"
## [1] "fail to reject null hypothesis"
## [1] "p-value: 0.11875"
```



3.3.2 Validation accuracy

```
validation_plot(filter(task_data, split == '50%'))
```



Summary statistics for the testing performance of the selection schemes at the 5% selection set split:

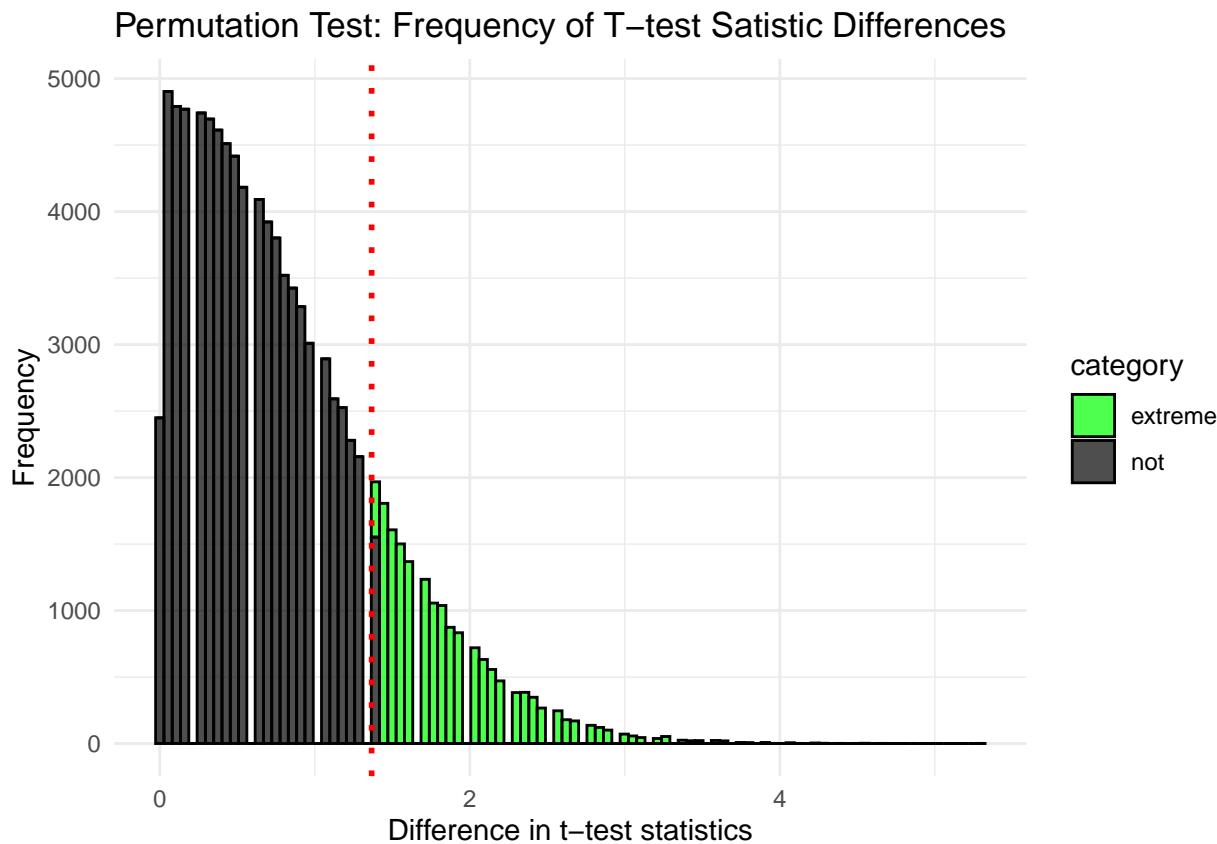
```
validation_accuracy_summary(filter(task_data, split == '50%'))
```

```
## # A tibble: 2 x 8
##   selection count na_cnt  min median  mean   max     IQR
##   <fct>     <int> <int> <dbl> <dbl> <dbl> <dbl> <dbl>
## 1 tournament     40     0 0.875 0.897 0.898 0.920 0.0137
## 2 lexicase       40     0 0.871 0.904 0.901 0.926 0.0137
```

The permutation test revealed that the results are:

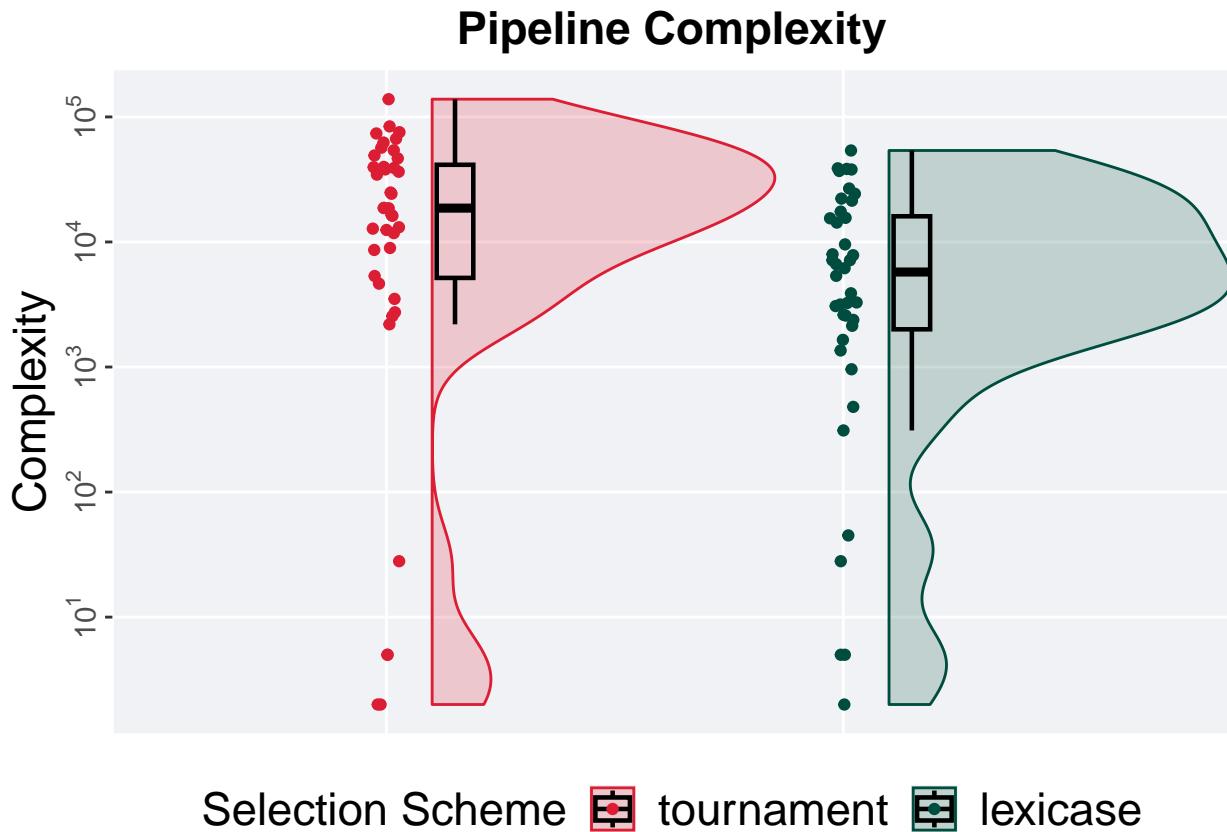
```
tournament_results <- filter(task_data, split == '50%' & selection == 'tournament')
lexicase_results <- filter(task_data, split == '50%' & selection == 'lexicase')
permutation_test(tournament_results$training_performance,
                 lexicase_results$training_performance,
                 seed = 6,
                 alternative = "t")
```

```
## [1] "observed_diff: -1.3666761390605"
## [1] "lower: -2.01498791705763"
## [1] "upper: 2.01499018334096"
## [1] "fail to reject null hypothesis"
## [1] "p-value: 0.16869"
```



3.3.3 Pipeline complexity

```
complexity_plot(filter(task_data, split == '50%'))
```



Summary statistics for the pipeline complexity of the selection schemes at the 5% selection set split:

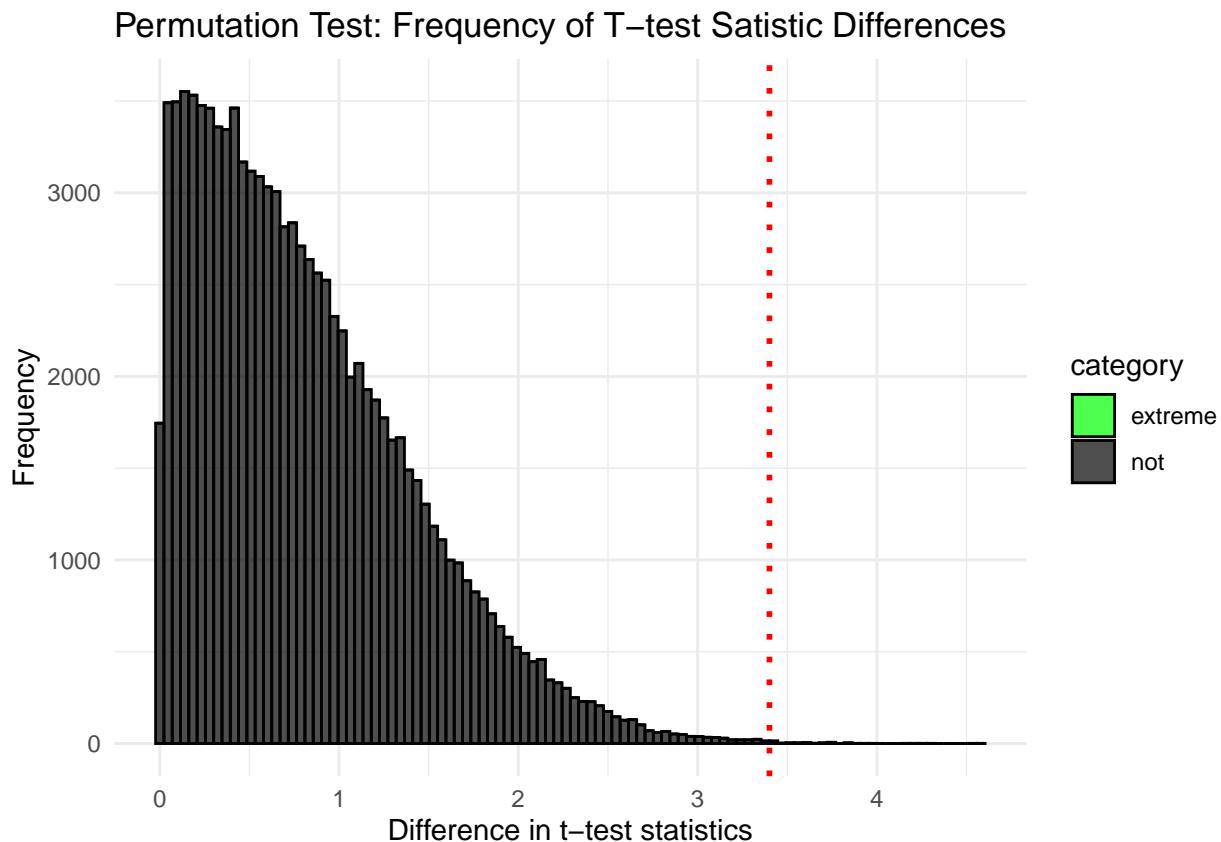
```
complexity_summary(filter(task_data, split == '50%'))
```

```
## # A tibble: 2 x 8
##   selection count na_cnt  min median   mean   max   IQR
##   <fct>     <int> <int> <dbl> <dbl> <dbl> <int> <dbl>
## 1 tournament     40     0     2 18642. 29034. 138561 36351
## 2 lexicase       40     0     2  5764  11312.  53886 14049.
```

The permutation test revealed that the results are:

```
tournament_results <- filter(task_data, split == '50%' & selection == 'tournament')
lexicase_results <- filter(task_data, split == '50%' & selection == 'lexicase')
permutation_test(tournament_results$testing_complexity,
                 lexicase_results$testing_complexity,
                 seed = 202,
                 alternative = "t")
```

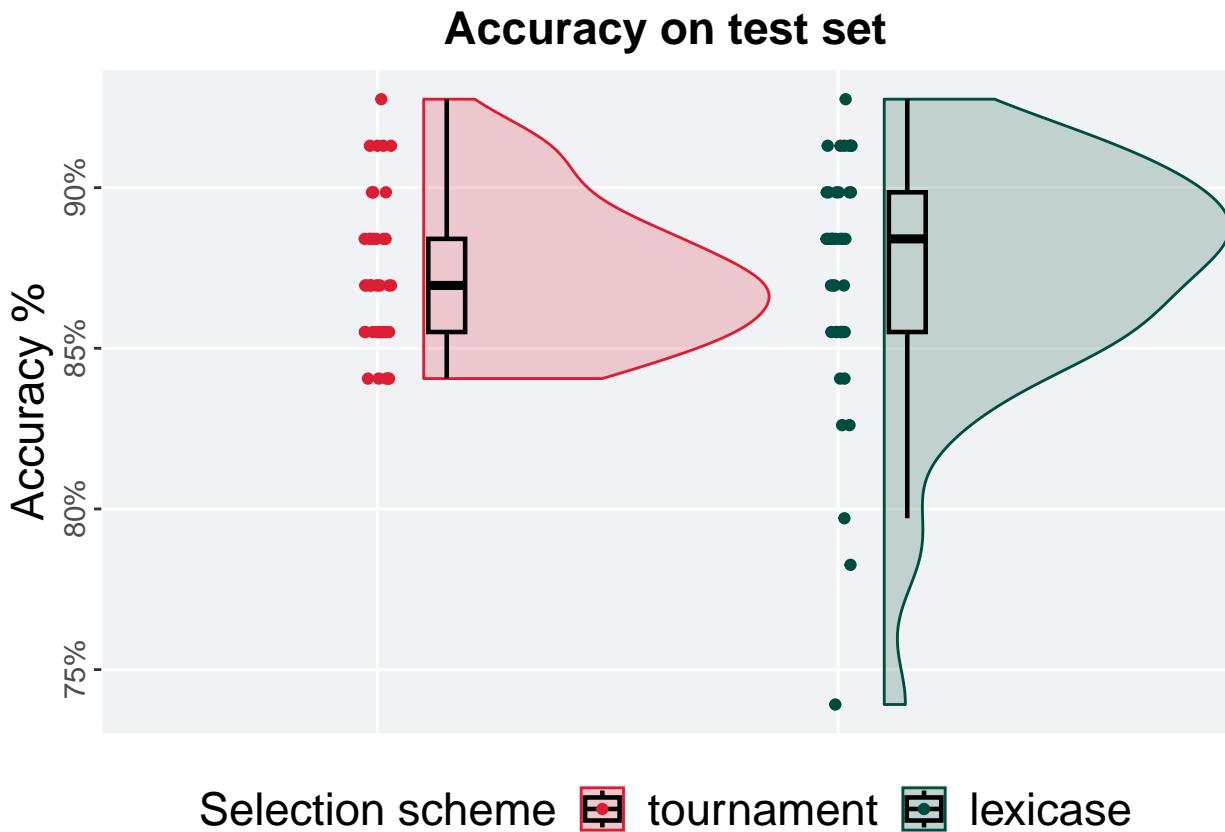
```
## [1] "observed_diff: 3.40063710370762"
## [1] "lower: -1.9770030894576"
## [1] "upper: 1.97740775447499"
## [1] "reject null hypothesis"
## [1] "p-value: 0.00062"
```



3.4 90%

3.4.1 Test accuracy

```
test_plot(filter(task_data, split == '90%'))
```



Summary statistics for the testing performance of the selection schemes at the 5% selection set split:

```
test_results_summary(filter(task_data, split == '90%'))
```

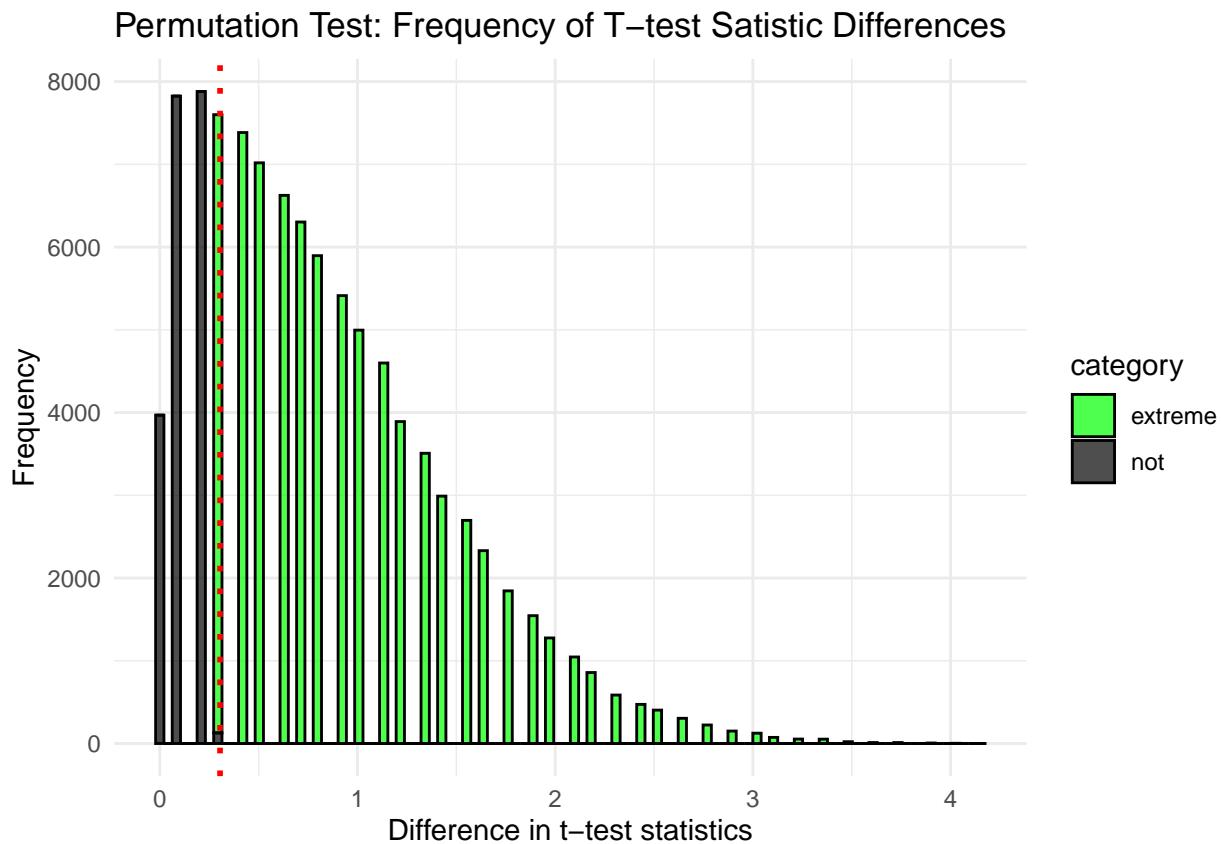
```
## # A tibble: 2 x 8
##   selection count na_cnt  min median  mean   max     IQR
##   <fct>     <int> <int> <dbl> <dbl> <dbl> <dbl> <dbl>
## 1 tournament     40     0 0.841  0.870  0.874  0.928  0.0290
## 2 lexicase      40     0 0.739  0.884  0.872  0.928  0.0435
```

The permutation test revealed that the results are:

```
tournament_results <- filter(task_data, split == '90%' & selection == 'tournament')
lexicase_results <- filter(task_data, split == '90%' & selection == 'lexicase')
```

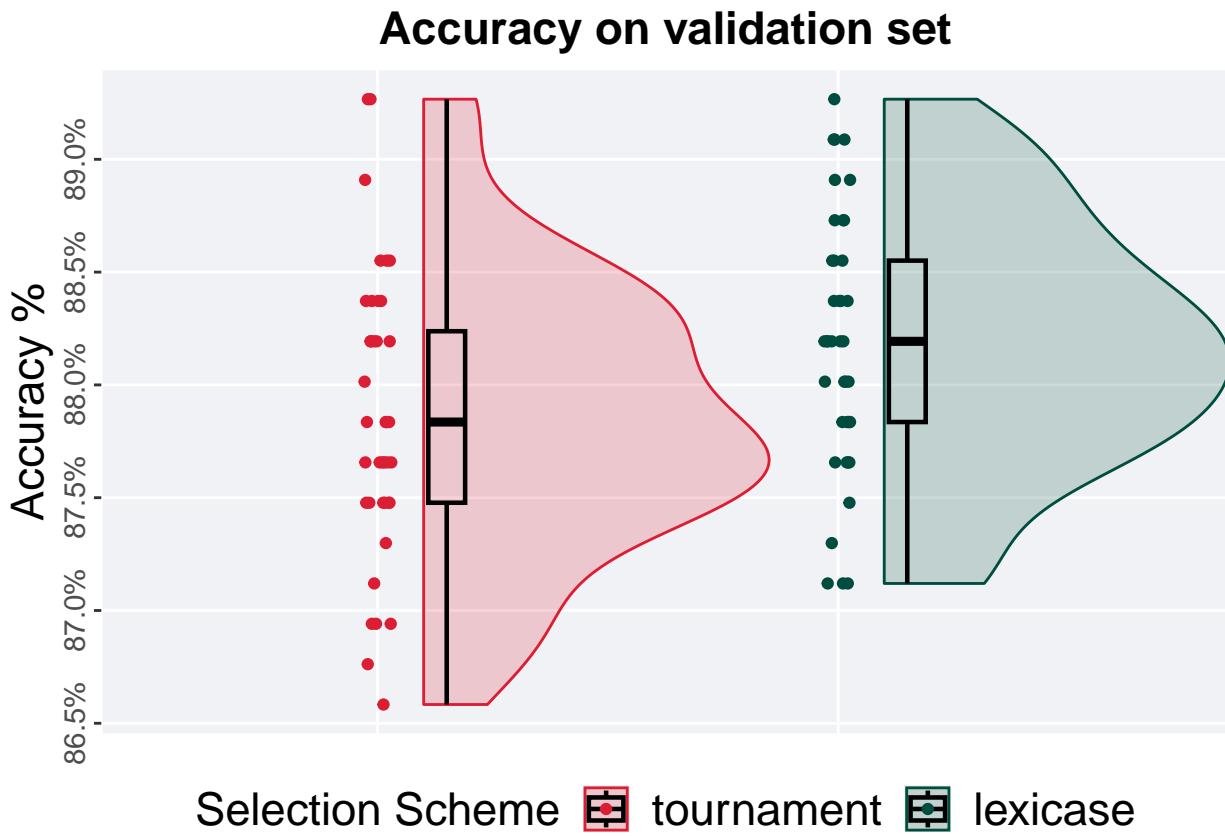
```
permutation_test(tournament_results$testing_performance,
                  lexicase_results$testing_performance,
                  seed = 7,
                  alternative = "t")
```

```
## [1] "observed_diff: 0.304925249770556"
## [1] "lower: -1.97785745966913"
## [1] "upper: 1.97785745966913"
## [1] "fail to reject null hypothesis"
## [1] "p-value: 0.80197"
```



3.4.2 Validation accuracy

```
validation_plot(filter(task_data, split == '90%'))
```



Summary statistics for the testing performance of the selection schemes at the 5% selection set split:

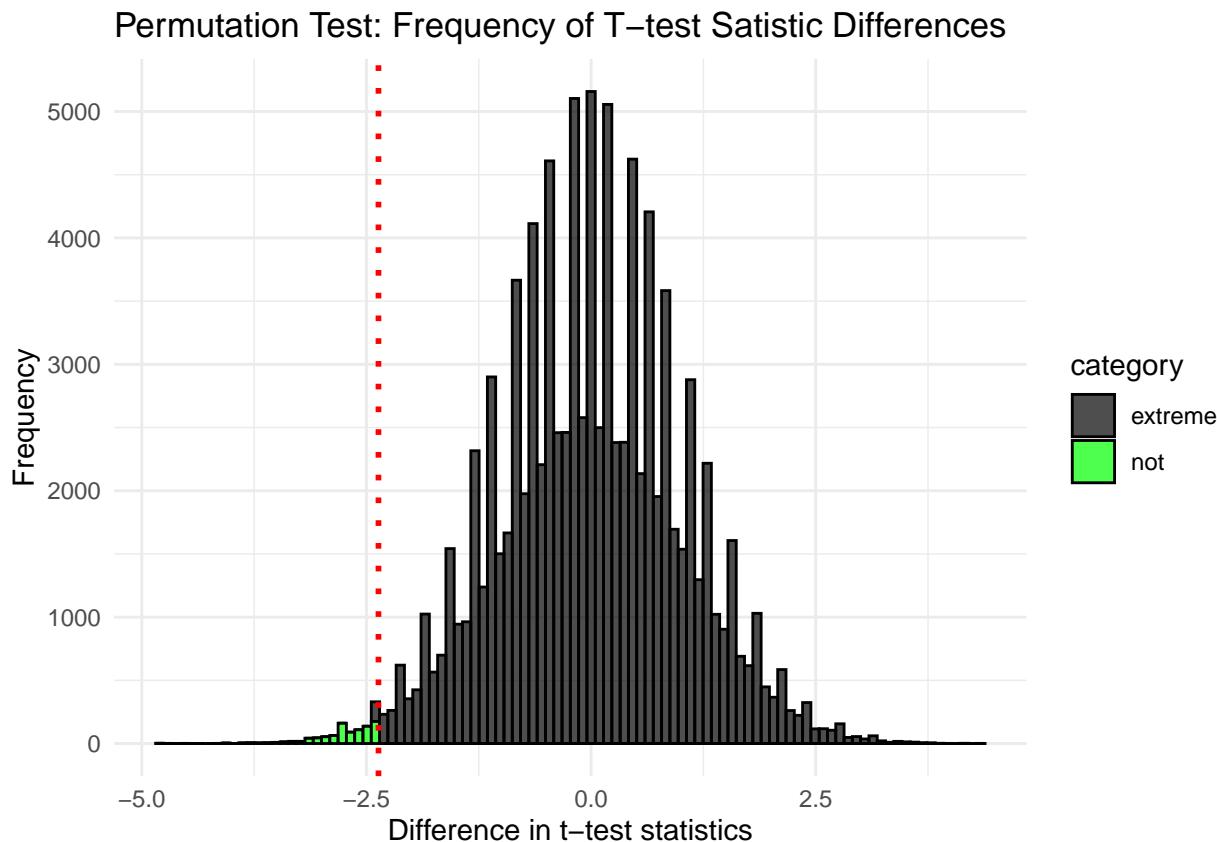
```
validation_accuracy_summary(filter(task_data, split == '90%'))
```

```
## # A tibble: 2 x 8
##   selection  count  na_cnt  min  median  mean  max      IQR
##   <fct>      <int>  <int>  <dbl>  <dbl>  <dbl>  <dbl>  <dbl>
## 1 tournament    40      0  0.866  0.878  0.879  0.893  0.00760
## 2 lexicase      40      0  0.871  0.882  0.882  0.893  0.00716
```

The permutation test revealed that the results are:

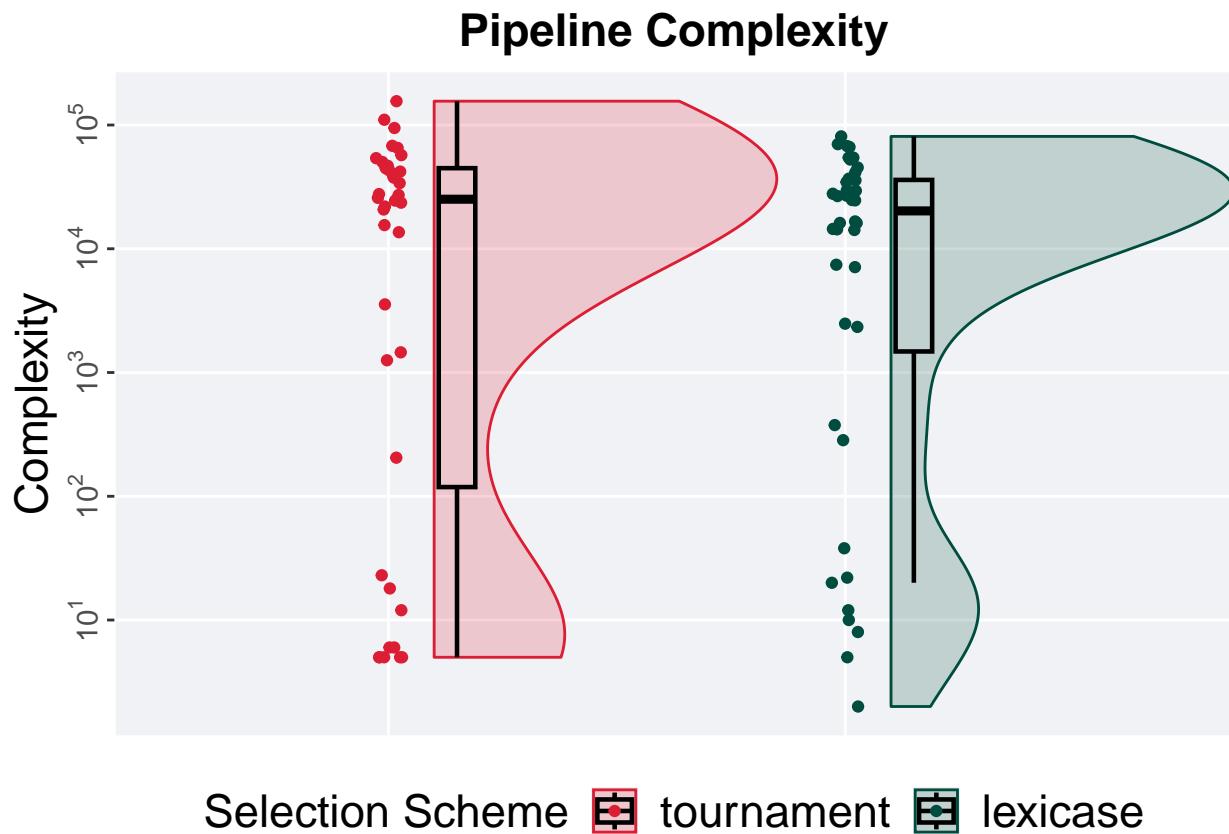
```
tournament_results <- filter(task_data, split == '90%' & selection == 'tournament')
lexicase_results <- filter(task_data, split == '90%' & selection == 'lexicase')
permutation_test(tournament_results$training_performance,
                 lexicase_results$training_performance,
                 seed = 8,
                 alternative = "1")
```

```
## [1] "observed_diff: -2.3660300252374"
## [1] "permutation_diffs[0.05 * n_permutations]: -1.67076701407627"
## [1] "reject null hypothesis"
## [1] "p-value: 0.00964"
```



3.4.3 Pipeline complexity

```
complexity_plot(filter(task_data, split == '90%'))
```



Summary statistics for the pipeline complexity of the selection schemes at the 5% selection set split:

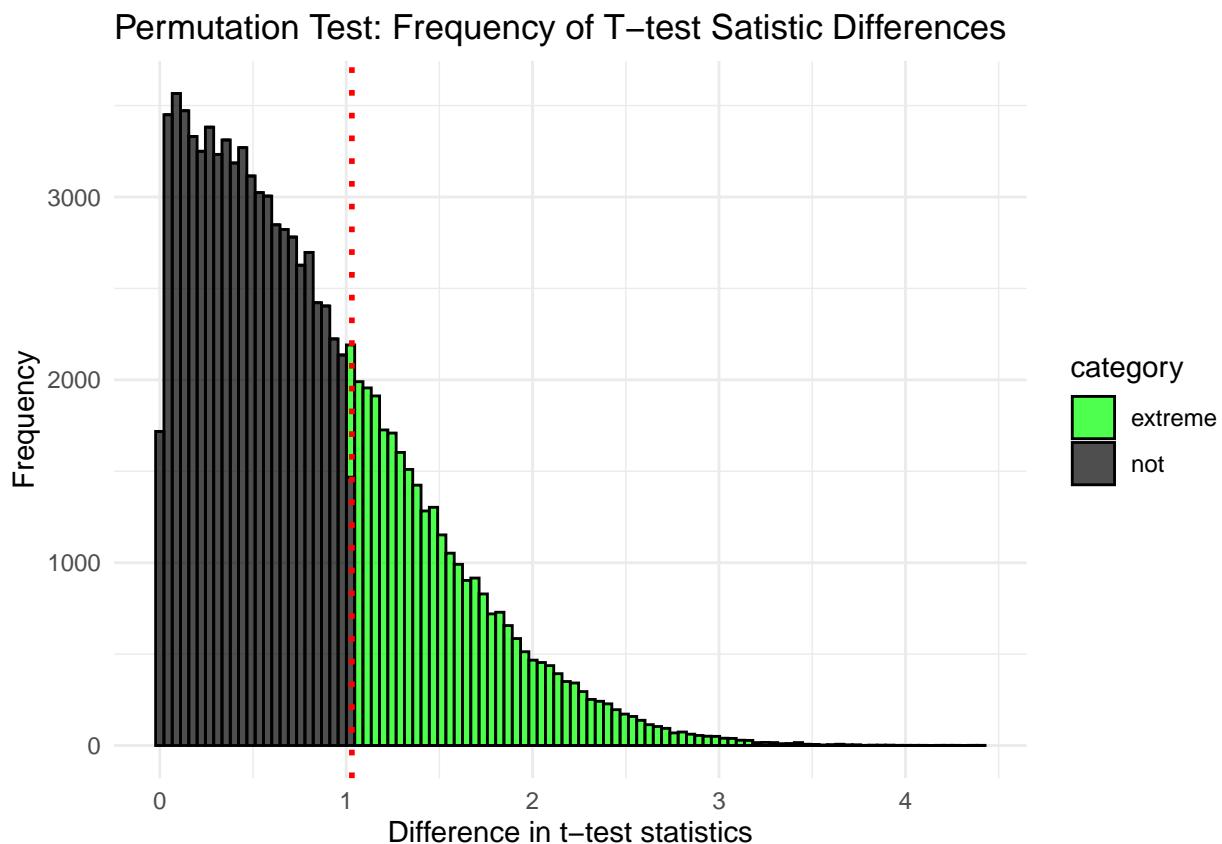
```
complexity_summary(filter(task_data, split == '90%'))
```

```
## # A tibble: 2 x 8
##   selection count na_cnt  min median   mean   max   IQR
##   <fct>     <int> <int> <dbl> <dbl> <dbl> <int> <dbl>
## 1 tournament     40     0     5 25148. 30979. 155771 44629.
## 2 lexicase       40     0     2 20624  24282.  80892 34170
```

The permutation test revealed that the results are:

```
tournament_results <- filter(task_data, split == '90%' & selection == 'tournament')
lexicase_results <- filter(task_data, split == '90%' & selection == 'lexicase')
permutation_test(tournament_results$testing_complexity,
                 lexicase_results$testing_complexity,
                 seed = 203,
                 alternative = "t")
```

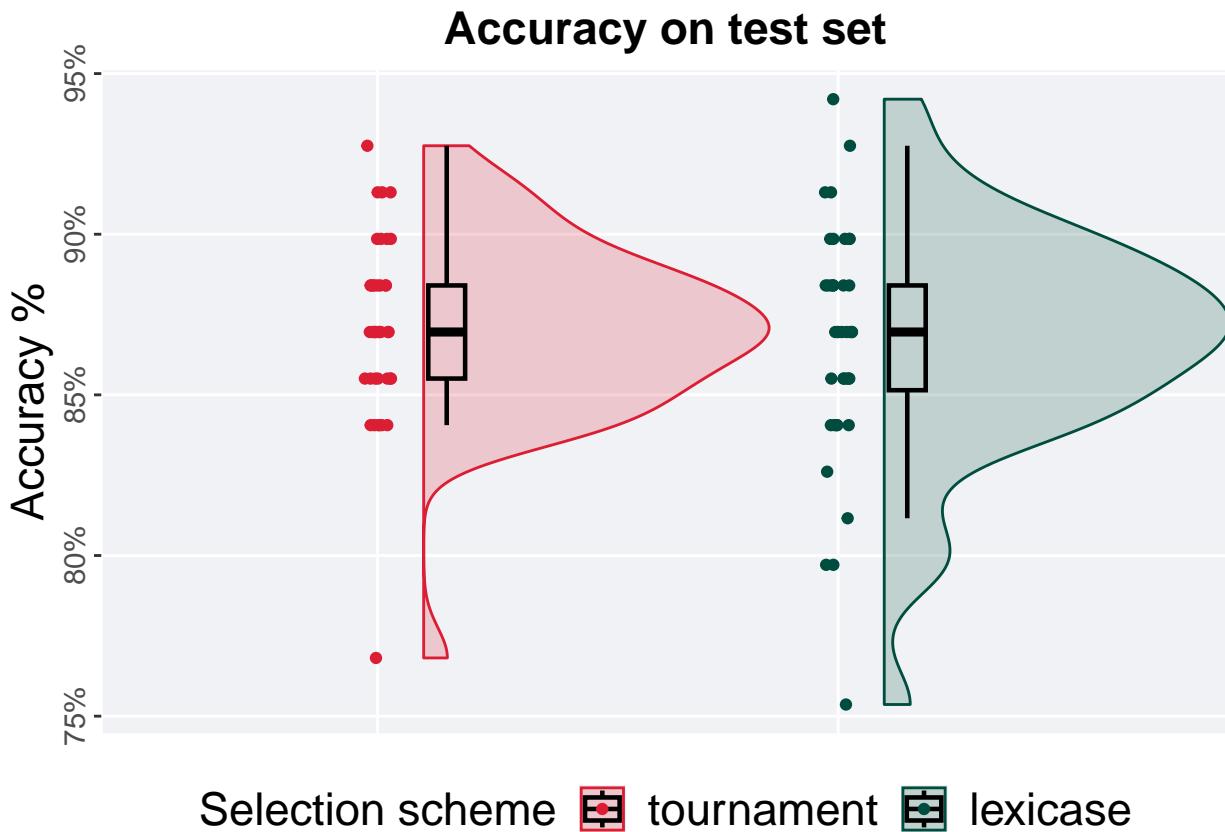
```
## [1] "observed_diff: 1.03068252751432"
## [1] "lower: -1.99740025729566"
## [1] "upper: 1.97729889993401"
## [1] "fail to reject null hypothesis"
## [1] "p-value: 0.31258"
```



3.5 95%

3.5.1 Test accuracy

```
test_plot(filter(task_data, split == '95%'))
```



Summary statistics for the testing performance of the selection schemes at the 5% selection set split:

```
test_results_summary(filter(task_data, split == '95%'))
```

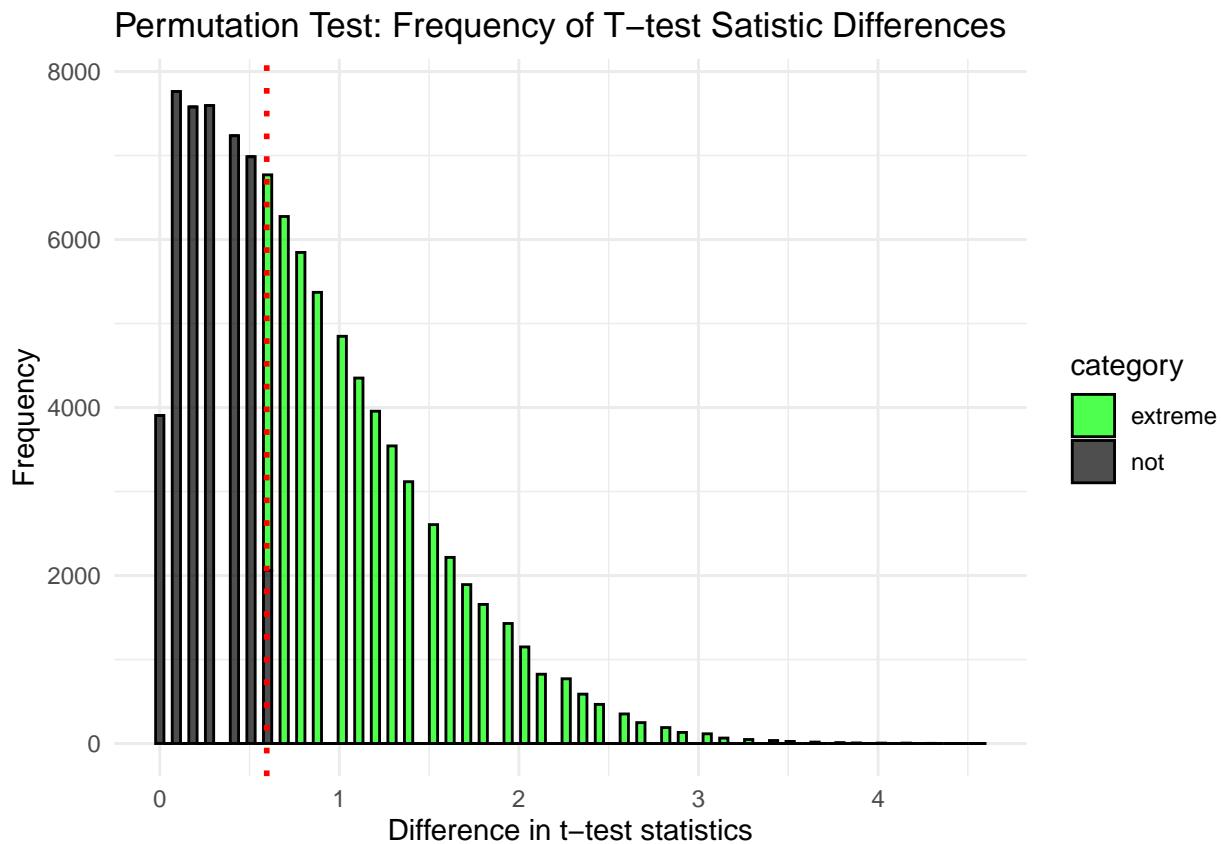
```
## # A tibble: 2 x 8
##   selection count na_cnt  min median  mean   max     IQR
##   <fct>     <int> <int> <dbl> <dbl> <dbl> <dbl> <dbl>
## 1 tournament     40     0 0.768 0.870 0.871 0.928 0.0290
## 2 lexicase       40     0 0.754 0.870 0.866 0.942 0.0326
```

The permutation test revealed that the results are:

```
tournament_results <- filter(task_data, split == '95%' & selection == 'tournament')
lexicase_results <- filter(task_data, split == '95%' & selection == 'lexicase')
```

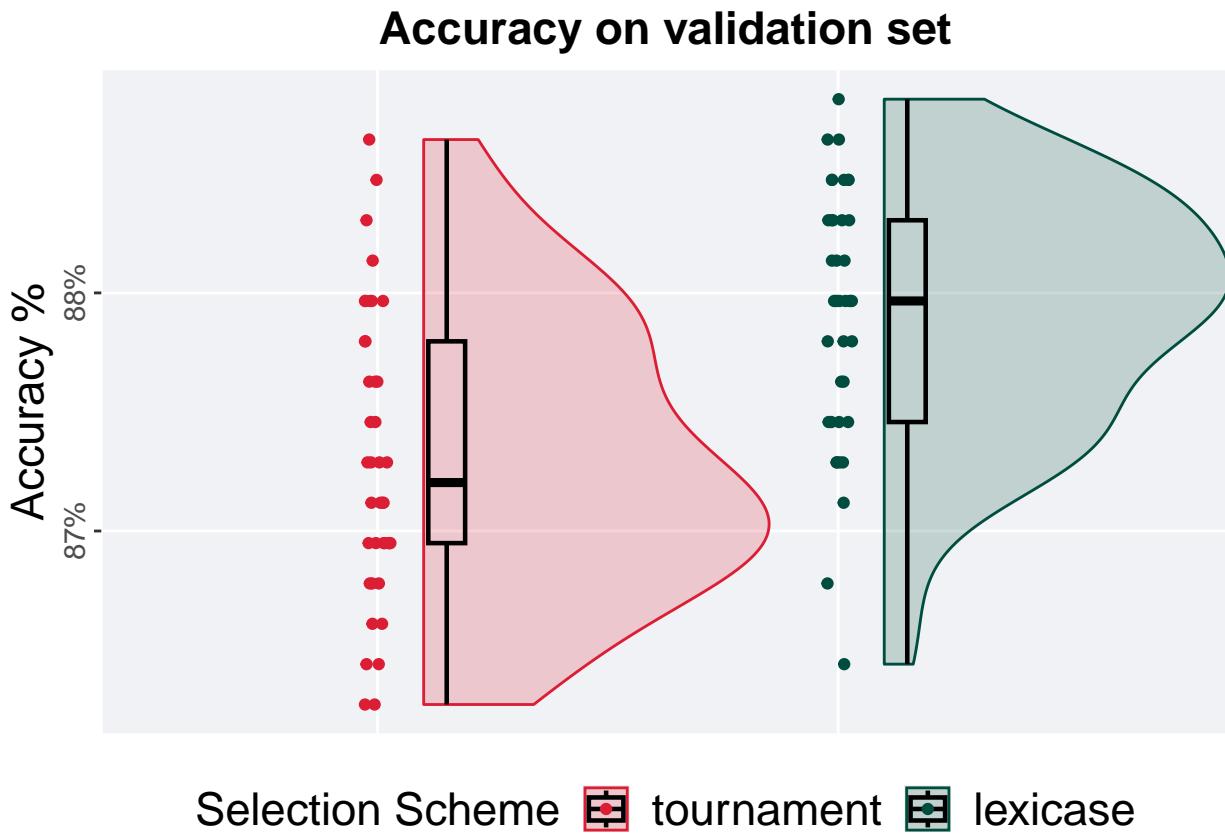
```
permutation_test(tournament_results$testing_performance,
                  lexicase_results$testing_performance,
                  seed = 9,
                  alternative = "t")
```

```
## [1] "observed_diff: 0.595624911096808"
## [1] "lower: -2.03270741712368"
## [1] "upper: 2.03270741712368"
## [1] "fail to reject null hypothesis"
## [1] "p-value: 0.56878"
```



3.5.2 Validation accuracy

```
validation_plot(filter(task_data, split == '95%'))
```



Summary statistics for the testing performance of the selection schemes at the 5% selection set split:

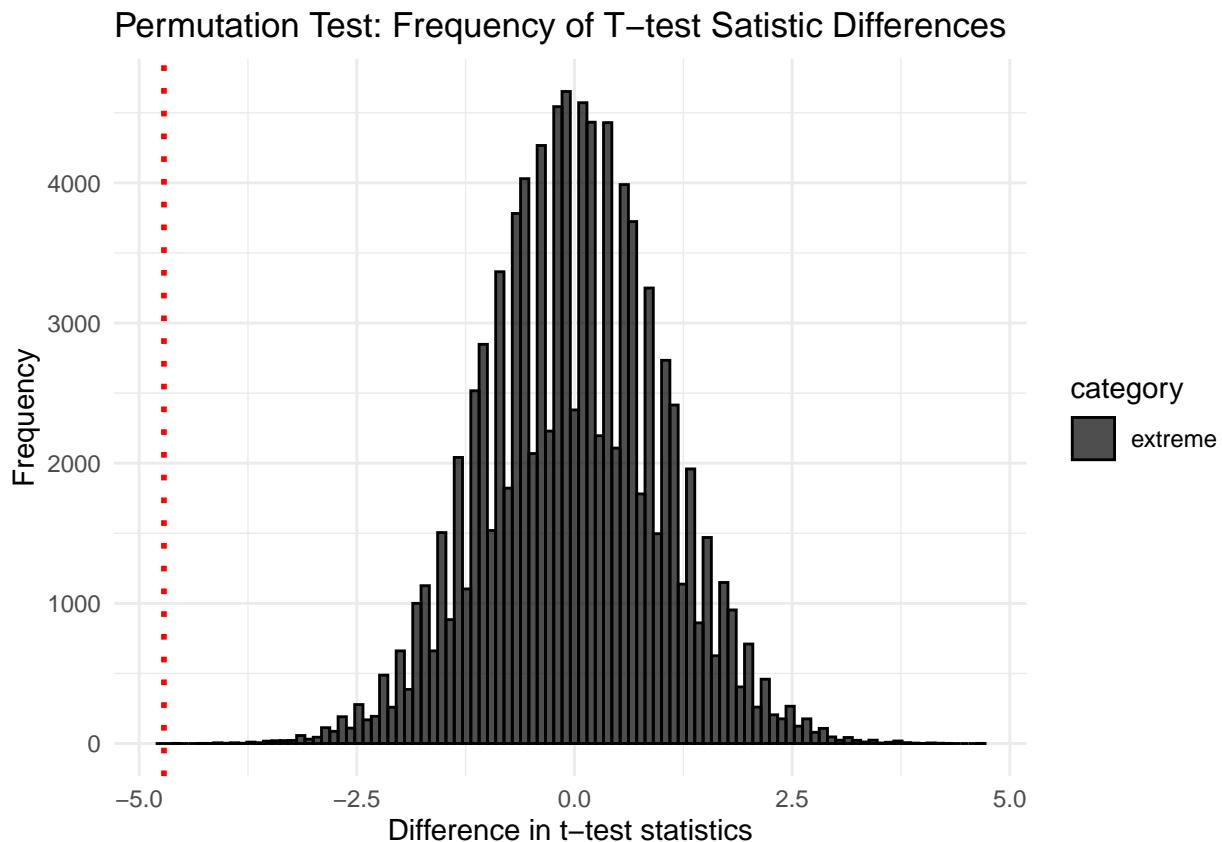
```
validation_accuracy_summary(filter(task_data, split == '95%'))
```

```
## # A tibble: 2 x 8
##   selection  count  na_cnt  min  median  mean  max      IQR
##   <fct>      <int>  <int>  <dbl>  <dbl>  <dbl>  <dbl>  <dbl>
## 1 tournament    40      0  0.863  0.872  0.873  0.886  0.00847
## 2 lexicase      40      0  0.864  0.880  0.879  0.888  0.00847
```

The permutation test revealed that the results are:

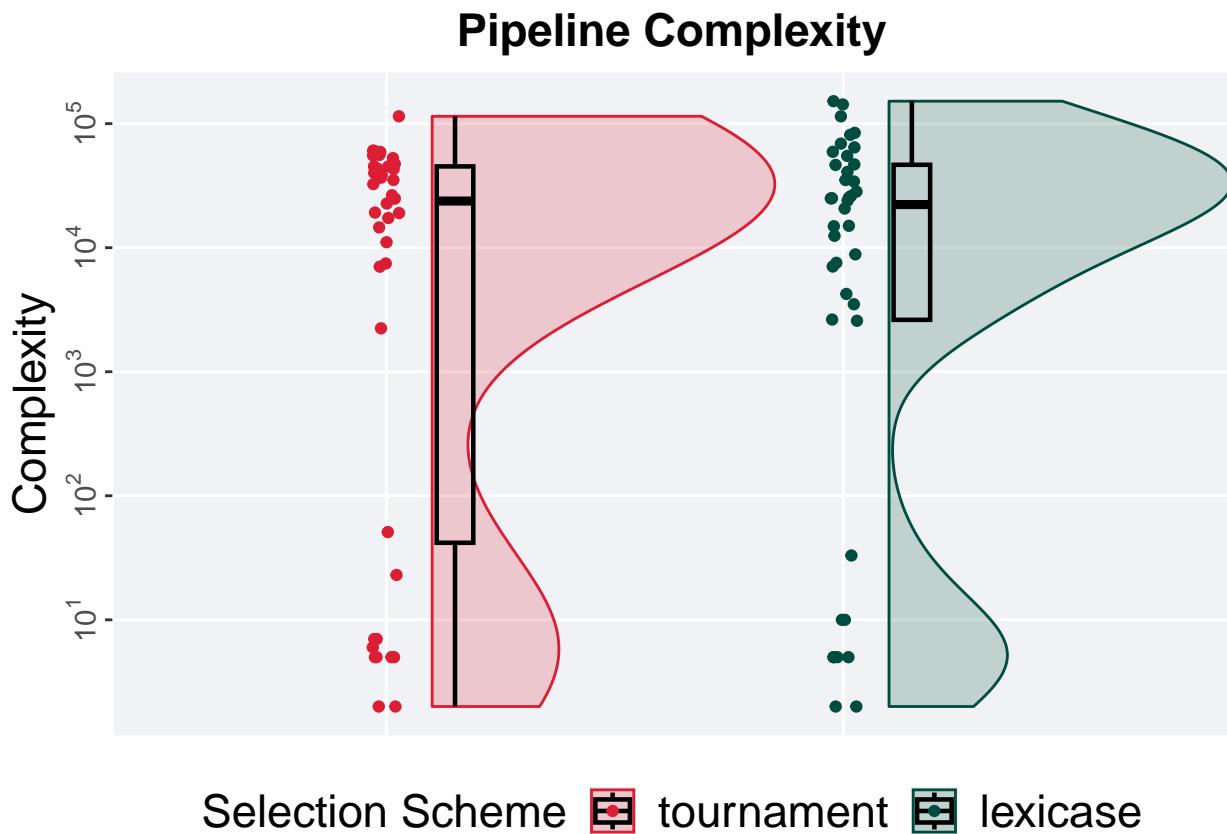
```
tournament_results <- filter(task_data, split == '95%' & selection == 'tournament')
lexicase_results <- filter(task_data, split == '95%' & selection == 'lexicase')
permutation_test(tournament_results$training_performance,
                 lexicase_results$training_performance,
                 seed = 10,
                 alternative = "1")
```

```
## [1] "observed_diff: -4.71550186673307"
## [1] "permutation_diffs[0.05 * n_permutations]: -1.6695026702645"
## [1] "reject null hypothesis"
## [1] "p-value: 1e-05"
```



3.5.3 Pipeline complexity

```
complexity_plot(filter(task_data, split == '95%'))
```



Summary statistics for the pipeline complexity of the selection schemes at the 5% selection set split:

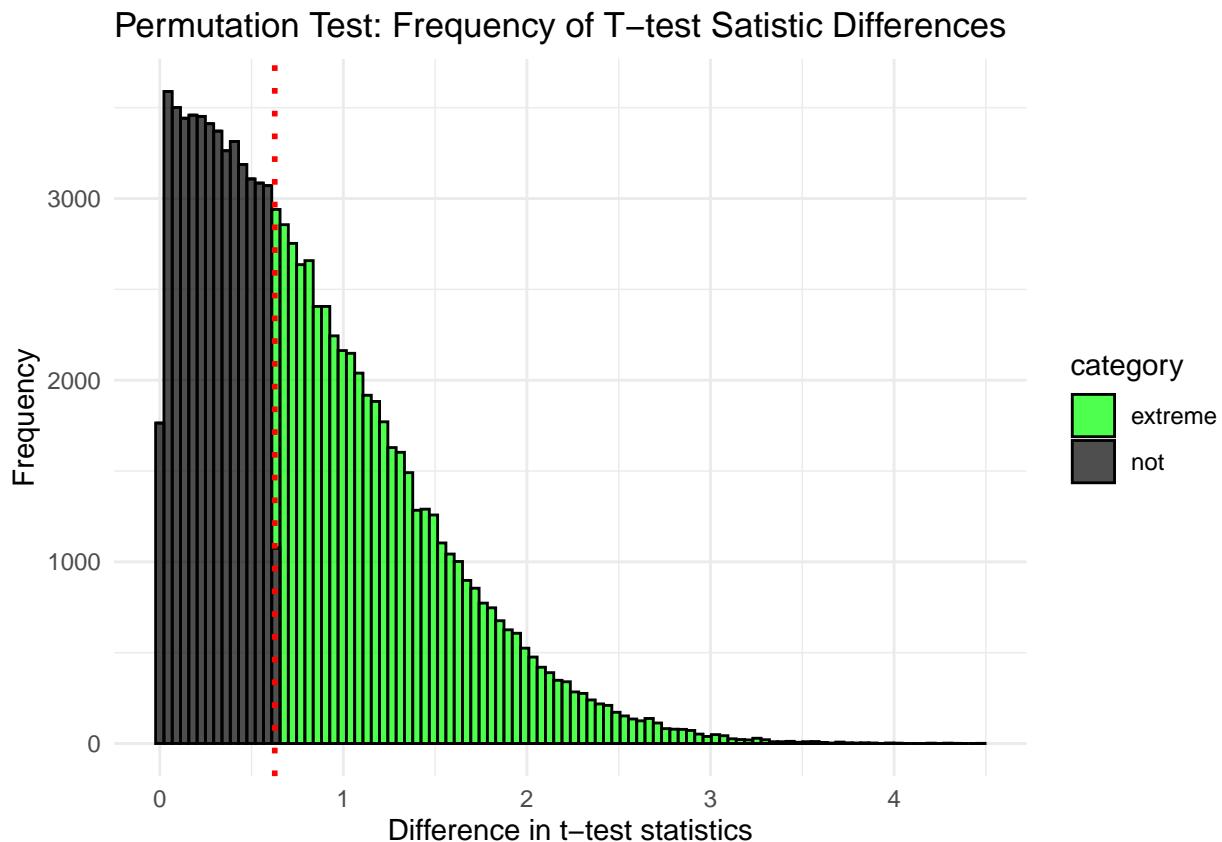
```
complexity_summary(filter(task_data, split == '95%'))
```

```
## # A tibble: 2 x 8
##   selection count na_cnt  min median   mean   max   IQR
##   <fct>     <int> <int> <dbl> <dbl> <dbl> <int> <dbl>
## 1 tournament     40     0     2 23799 27374. 114701 45177
## 2 lexicase       40     0     2 22349 31993. 151792 43991.
```

The permutation test revealed that the results are:

```
tournament_results <- filter(task_data, split == '95%' & selection == 'tournament')
lexicase_results <- filter(task_data, split == '95%' & selection == 'lexicase')
permutation_test(tournament_results$testing_complexity,
                 lexicase_results$testing_complexity,
                 seed = 204,
                 alternative = "t")
```

```
## [1] "observed_diff: -0.626438146998671"
## [1] "lower: -1.99293796073898"
## [1] "upper: 1.98291998227564"
## [1] "fail to reject null hypothesis"
## [1] "p-value: 0.53906"
```



Chapter 4

Task 359954

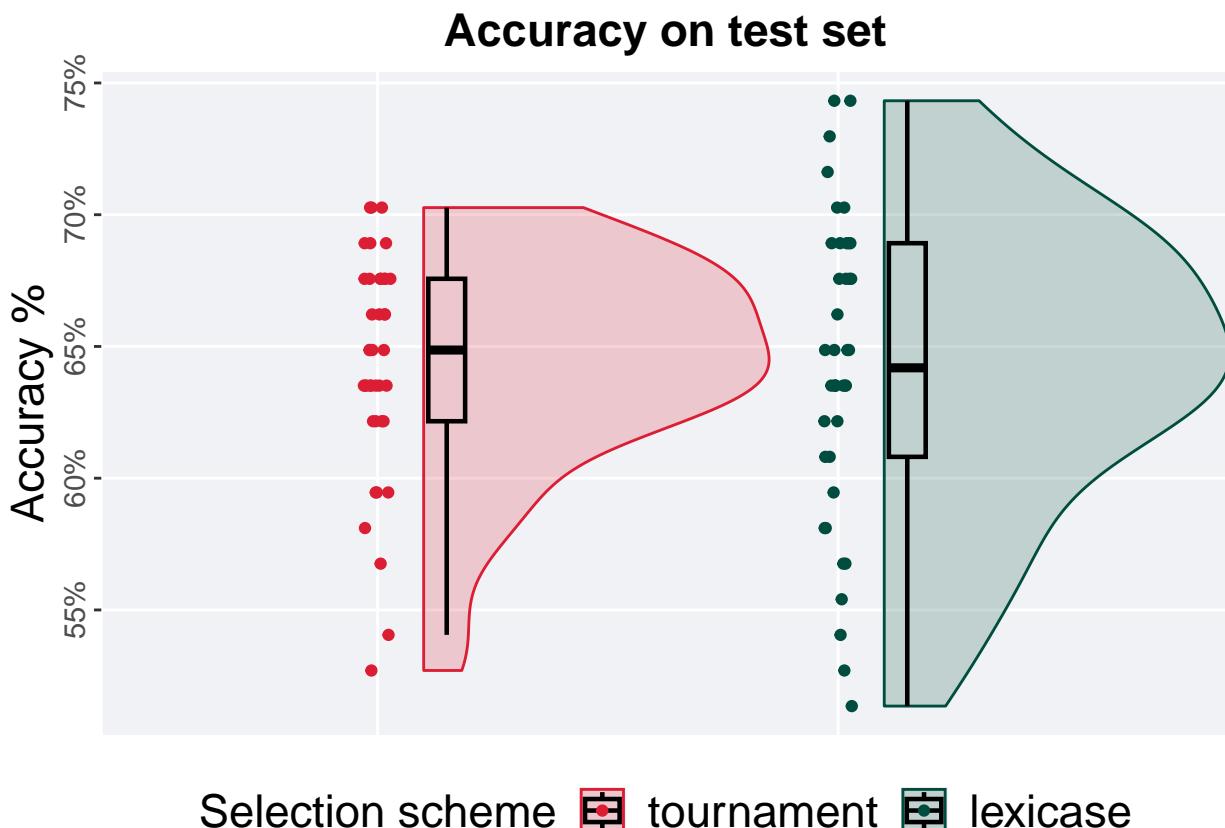
We present the results of our analysis of task 359954 with the different selection set splits used in our study.

```
task_data <- filter(results, task_id == 359954)
```

4.1 5%

4.1.1 Test accuracy

```
test_plot(filter(task_data, split == '5%'))
```



Summary statistics for the testing performance of the selection schemes at the 5% selection set split:

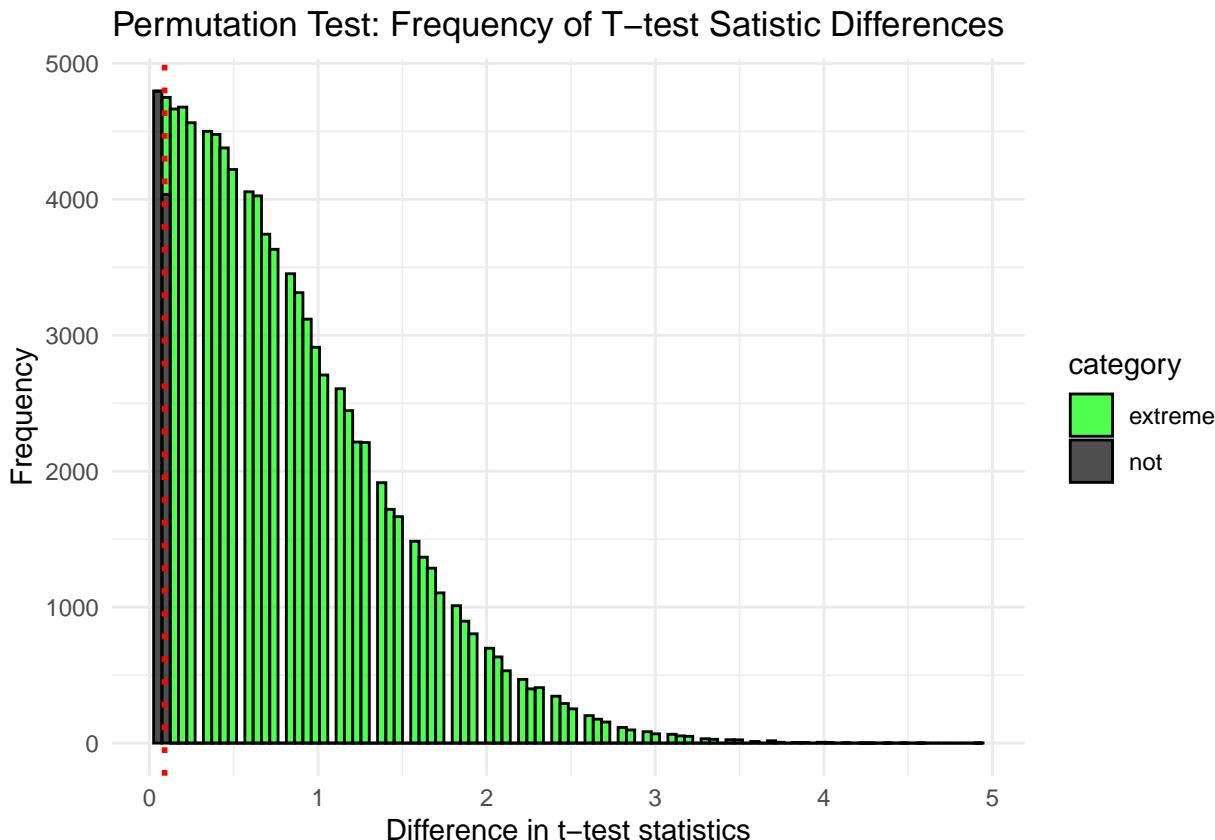
```
test_results_summary(filter(task_data, split == '5%'))
```

```
## # A tibble: 2 x 8
##   selection count na_cnt  min median  mean   max    IQR
##   <fct>     <int> <int> <dbl>  <dbl> <dbl> <dbl> <dbl>
## 1 tournament     40     0 0.527  0.649 0.643 0.703 0.0541
## 2 lexicase       40     0 0.514  0.642 0.642 0.743 0.0811
```

The permutation test revealed that the results are:

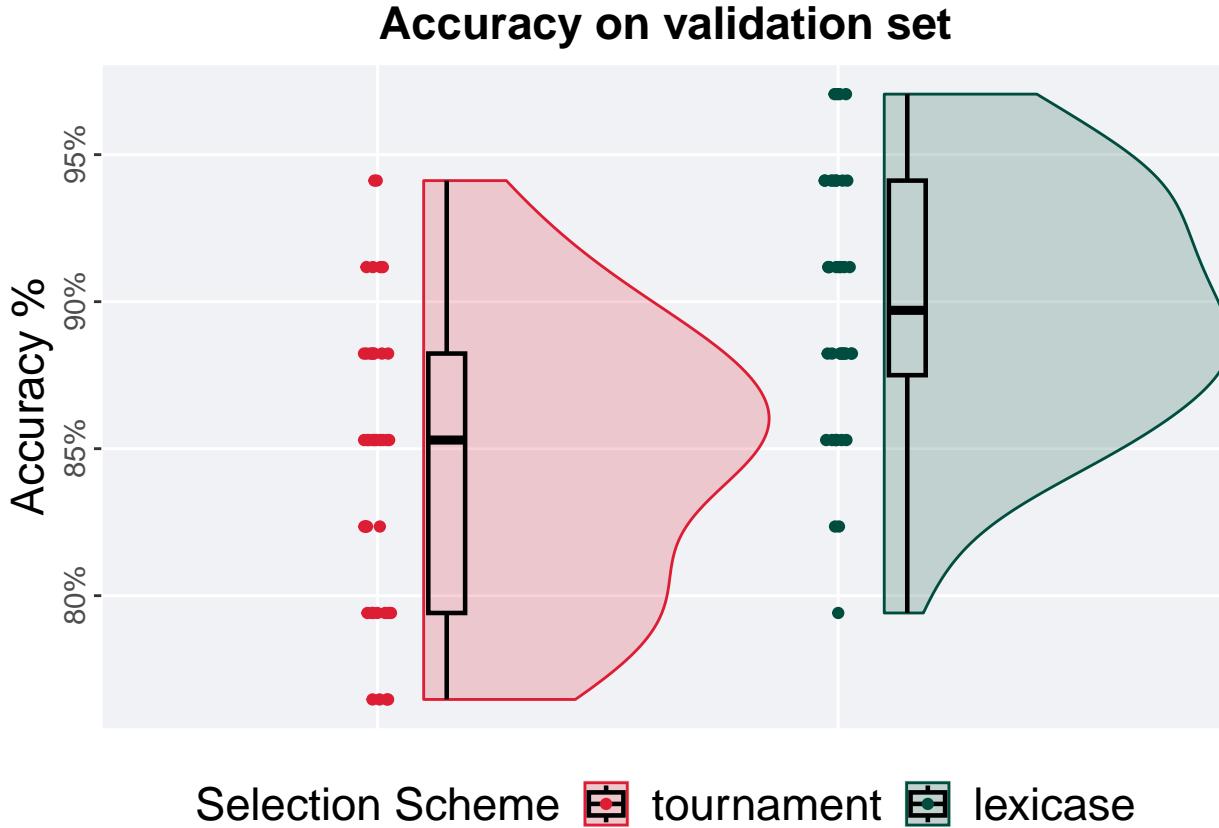
```
tournament_results <- filter(task_data, split == '5%' & selection == 'tournament')
lexicase_results <- filter(task_data, split == '5%' & selection == 'lexicase')
permutation_test(tournament_results$testing_performance,
                  lexicase_results$testing_performance,
                  seed = 11,
                  alternative = "t")
```

```
## [1] "observed_diff: 0.089899097880169"
## [1] "lower: -1.99686947621326"
## [1] "upper: 1.99686947621326"
## [1] "fail to reject null hypothesis"
## [1] "p-value: 0.91166"
```



4.1.2 Validation accuracy

```
validation_plot(filter(task_data, split == '5%'))
```



Summary statistics for the testing performance of the selection schemes at the 5% selection set split:

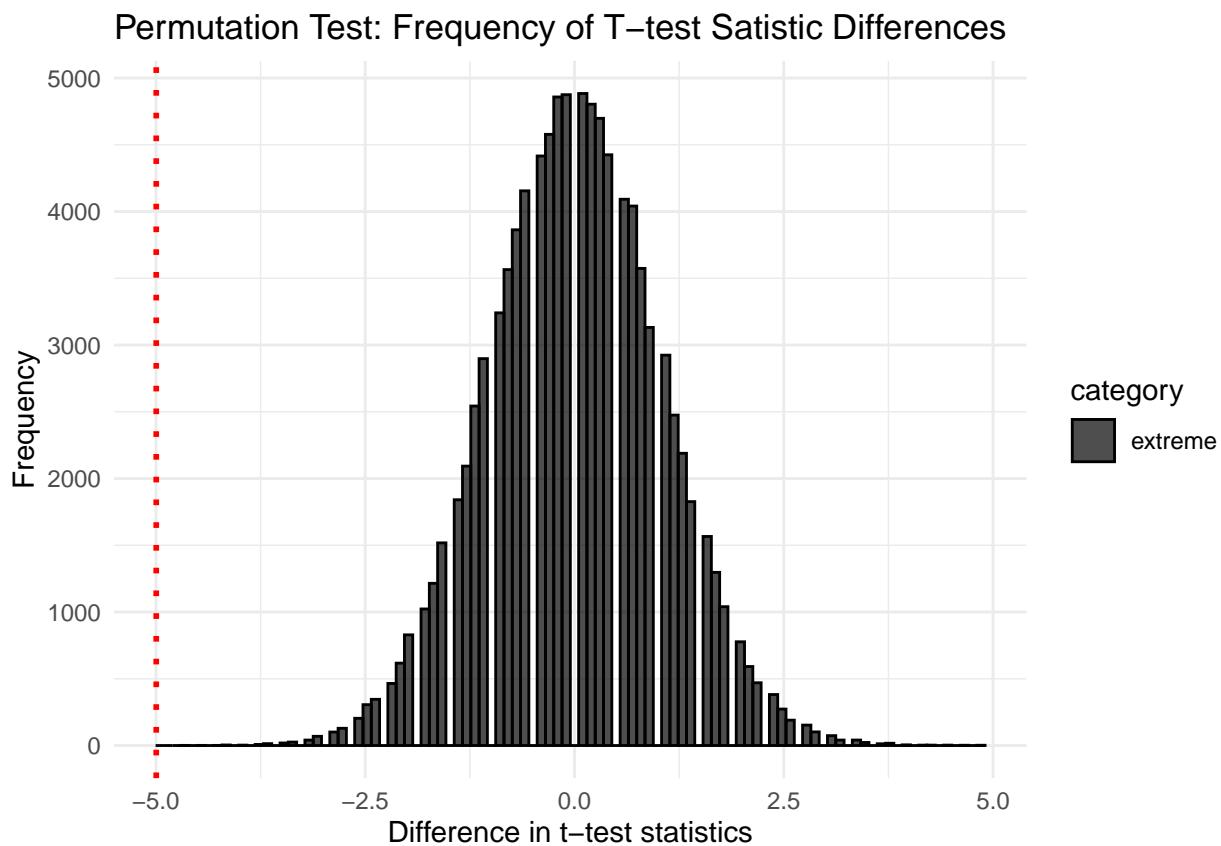
```
validation_accuracy_summary(filter(task_data, split == '5%'))
```

```
## # A tibble: 2 x 8
##   selection count na_cnt  min median  mean   max     IQR
##   <fct>     <int> <int> <dbl> <dbl> <dbl> <dbl> <dbl>
## 1 tournament     40     0 0.765  0.853  0.846  0.941  0.0882
## 2 lexicase       40     0 0.794  0.897  0.899  0.971  0.0662
```

The permutation test revealed that the results are:

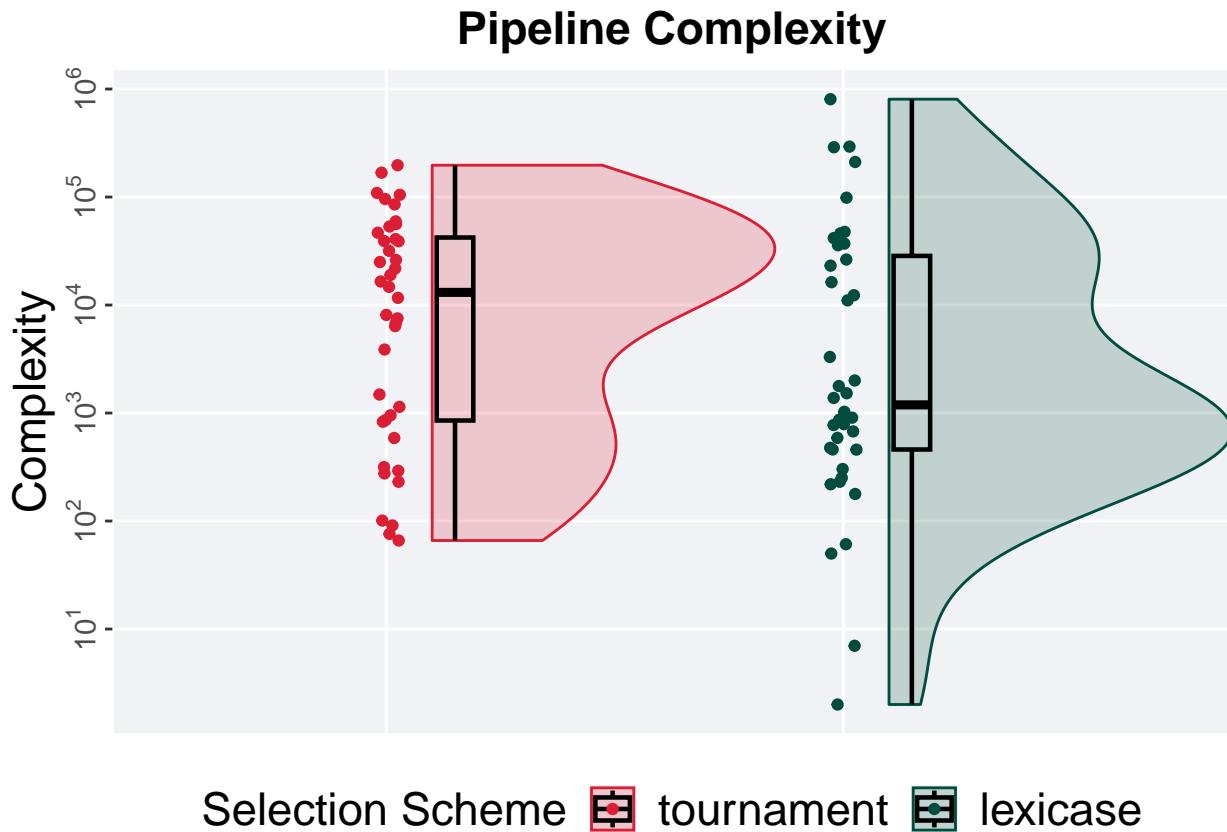
```
tournament_results <- filter(task_data, split == '5%' & selection == 'tournament')
lexicase_results <- filter(task_data, split == '5%' & selection == 'lexicase')
permutation_test(tournament_results$training_performance,
                 lexicase_results$training_performance,
                 seed = 12,
                 alternative = "1")
```

```
## [1] "observed_diff: -4.99649396670713"
## [1] "permutation_diffs[0.05 * n_permutations]: -1.68354186553216"
## [1] "reject null hypothesis"
## [1] "p-value: 1e-05"
```



4.1.3 Pipeline complexity

```
complexity_plot(filter(task_data, split == '5%'))
```



Summary statistics for the pipeline complexity of the selection schemes at the 5% selection set split:

```
complexity_summary(filter(task_data, split == '5%'))
```

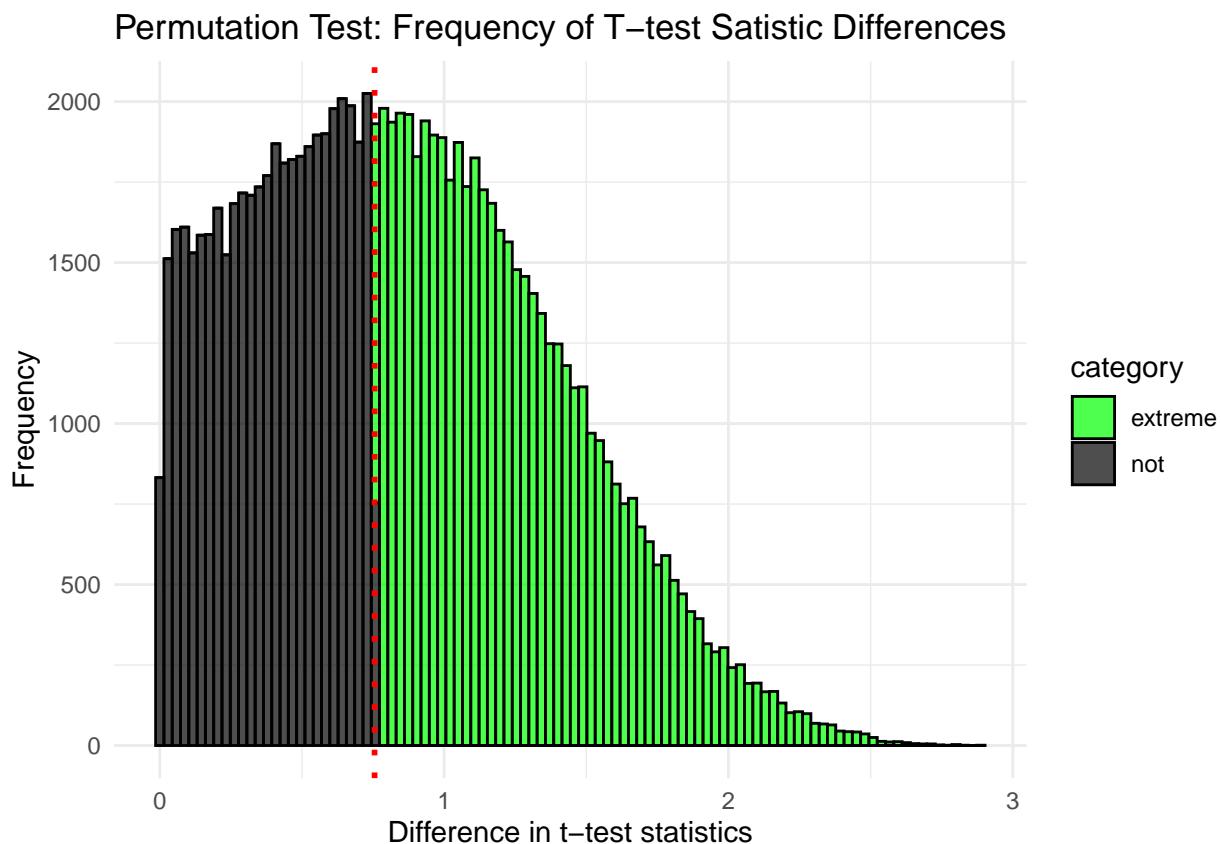
```
## # A tibble: 2 x 8
##   selection count na_cnt  min median   mean   max   IQR
##   <fct>     <int> <int> <dbl> <dbl> <dbl> <int> <dbl>
## 1 tournament     40     0    66  13192 32544. 197051 41406.
## 2 lexicase       40     0     2   1203 50344. 805192 28290.
```

The permutation test revealed that the results are:

```
tournament_results <- filter(task_data, split == '5%' & selection == 'tournament')
lexicase_results <- filter(task_data, split == '5%' & selection == 'lexicase')
```

```
permutation_test(tournament_results$testing_complexity,
                  lexicase_results$testing_complexity,
                  seed = 205,
                  alternative = "t")
```

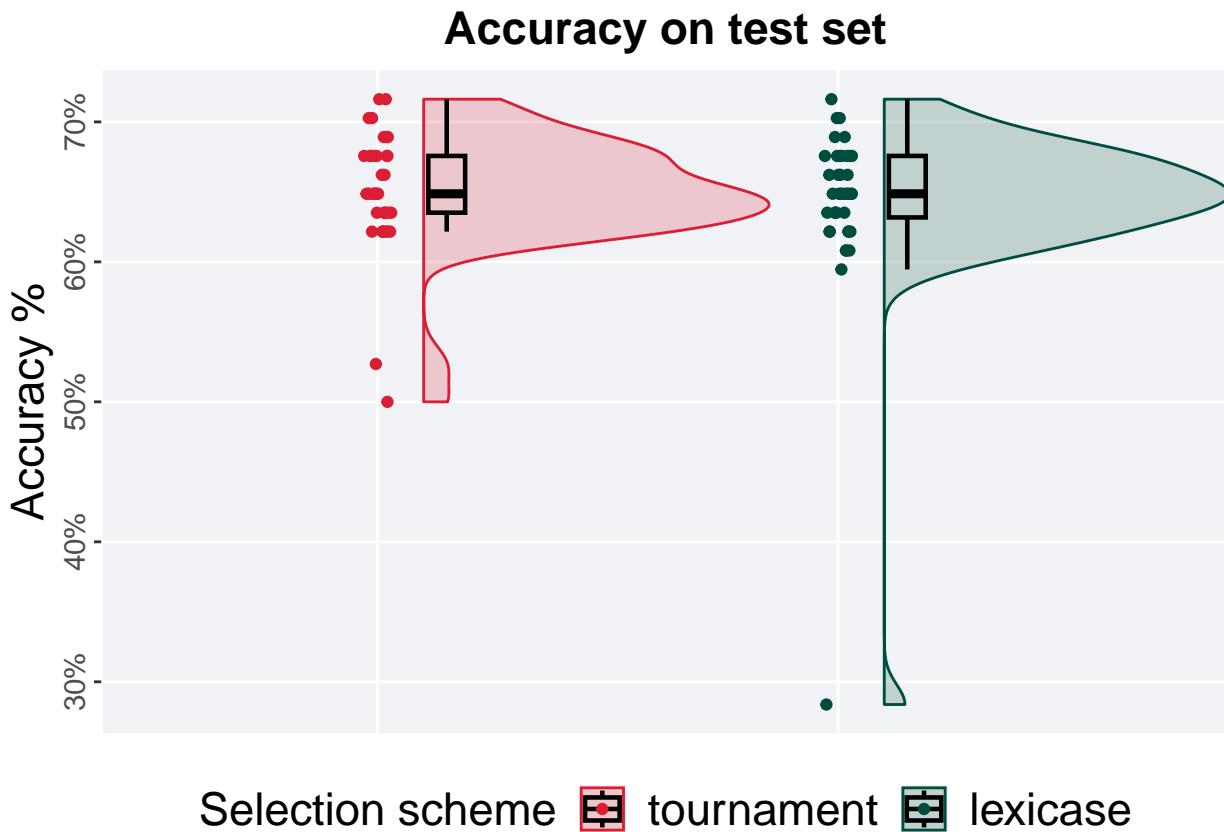
```
## [1] "observed_diff: -0.755505088010244"
## [1] "lower: -1.78370056240216"
## [1] "upper: 1.78619742112509"
## [1] "fail to reject null hypothesis"
## [1] "p-value: 0.54325"
```



4.2 10%

4.2.1 Test accuracy

```
test_plot(filter(task_data, split == '10%'))
```



Summary statistics for the testing performance of the selection schemes at the 5% selection set split:

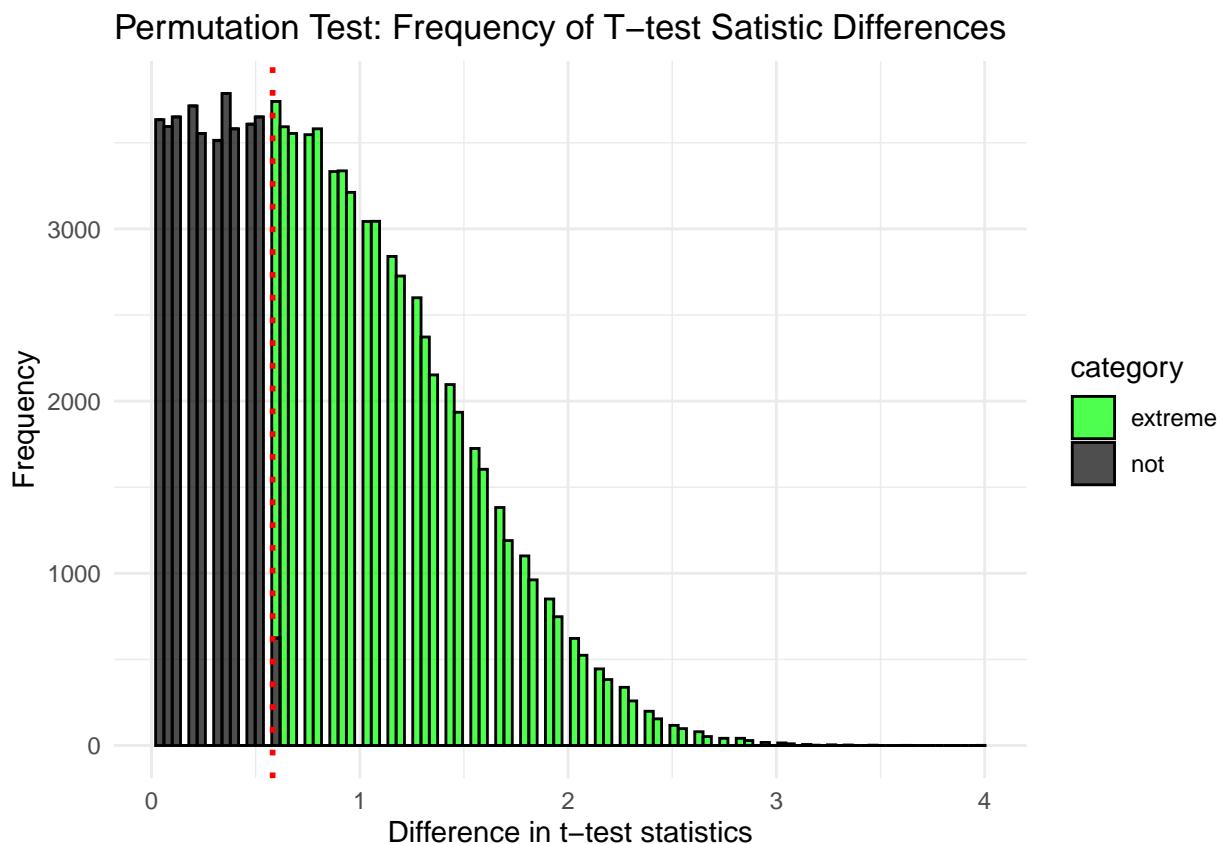
```
test_results_summary(filter(task_data, split == '10%'))
```

```
## # A tibble: 2 x 8
##   selection count na_cnt  min median  mean   max     IQR
##   <fct>     <int> <int> <dbl> <dbl> <dbl> <dbl> <dbl>
## 1 tournament     40     0 0.5    0.649 0.650 0.716 0.0405
## 2 lexicase       40     0 0.284  0.649 0.643 0.716 0.0439
```

The permutation test revealed that the results are:

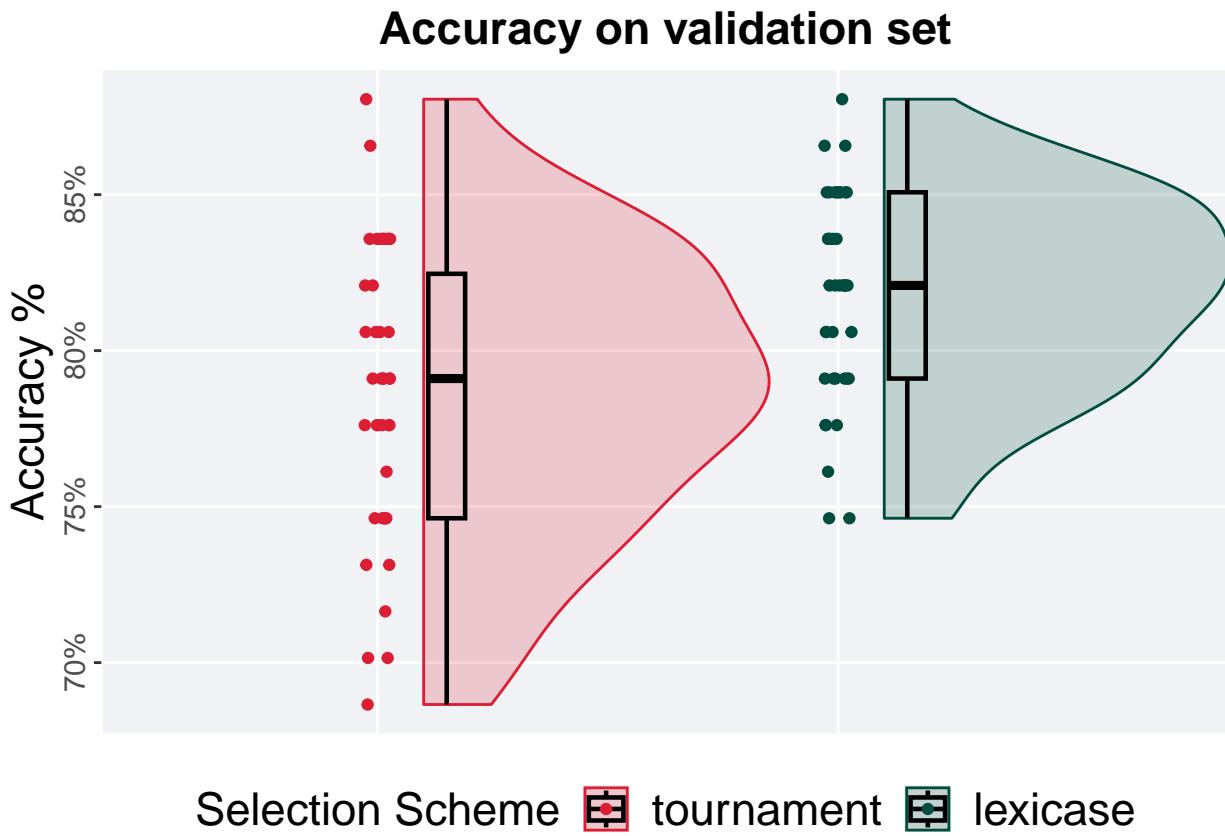
```
tournament_results <- filter(task_data, split == '10%' & selection == 'tournament')
lexicase_results <- filter(task_data, split == '10%' & selection == 'lexicase')
permutation_test(tournament_results$testing_performance,
                 lexicase_results$testing_performance,
                 seed = 13,
                 alternative = "t")
```

```
## [1] "observed_diff: 0.581706669215585"
## [1] "lower: -1.89401584174984"
## [1] "upper: 1.89401627937136"
## [1] "fail to reject null hypothesis"
## [1] "p-value: 0.63091"
```



4.2.2 Validation accuracy

```
validation_plot(filter(task_data, split == '10%'))
```



Summary statistics for the testing performance of the selection schemes at the 5% selection set split:

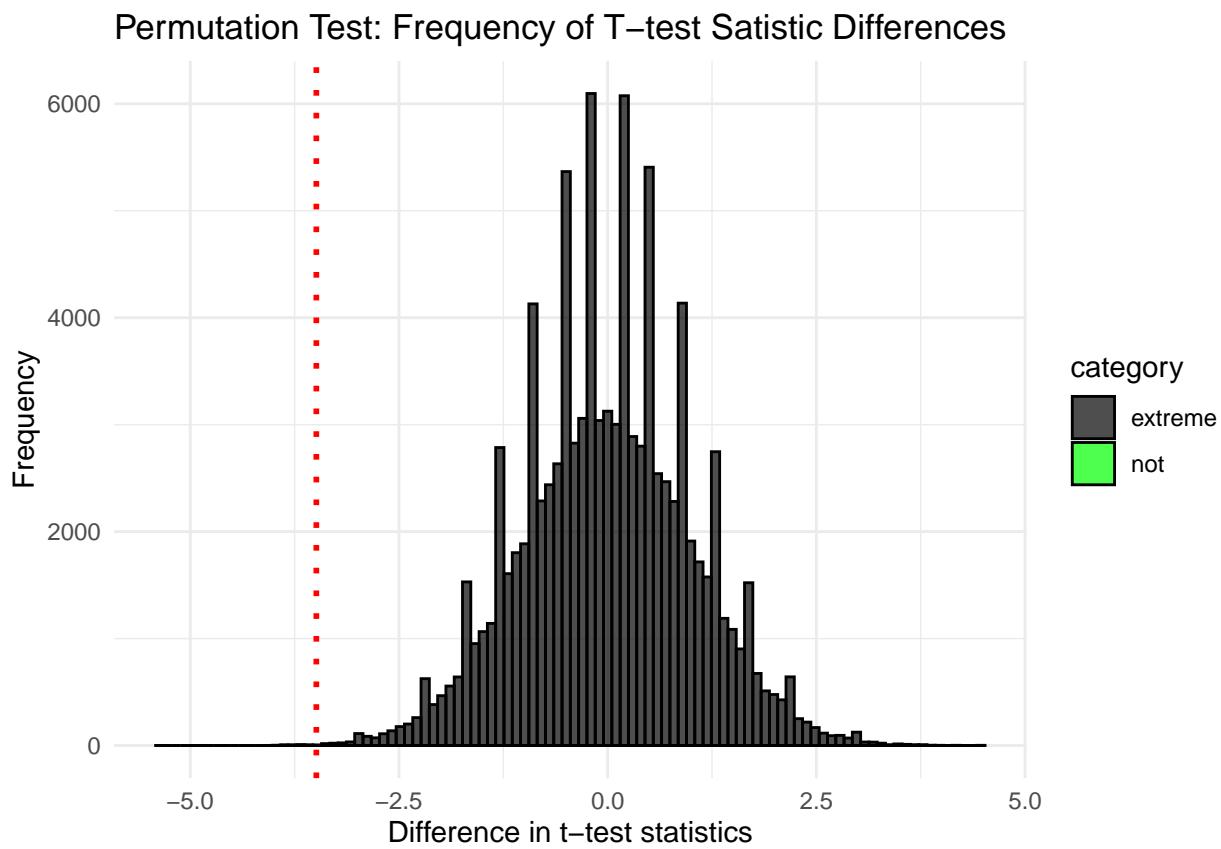
```
validation_accuracy_summary(filter(task_data, split == '10%'))
```

```
## # A tibble: 2 x 8
##   selection  count  na_cnt  min  median  mean  max    IQR
##   <fct>      <int>  <int>  <dbl>  <dbl>  <dbl> <dbl>  <dbl>
## 1 tournament    40      0  0.687  0.791  0.787  0.881  0.0784
## 2 lexicase      40      0  0.746  0.821  0.818  0.881  0.0597
```

The permutation test revealed that the results are:

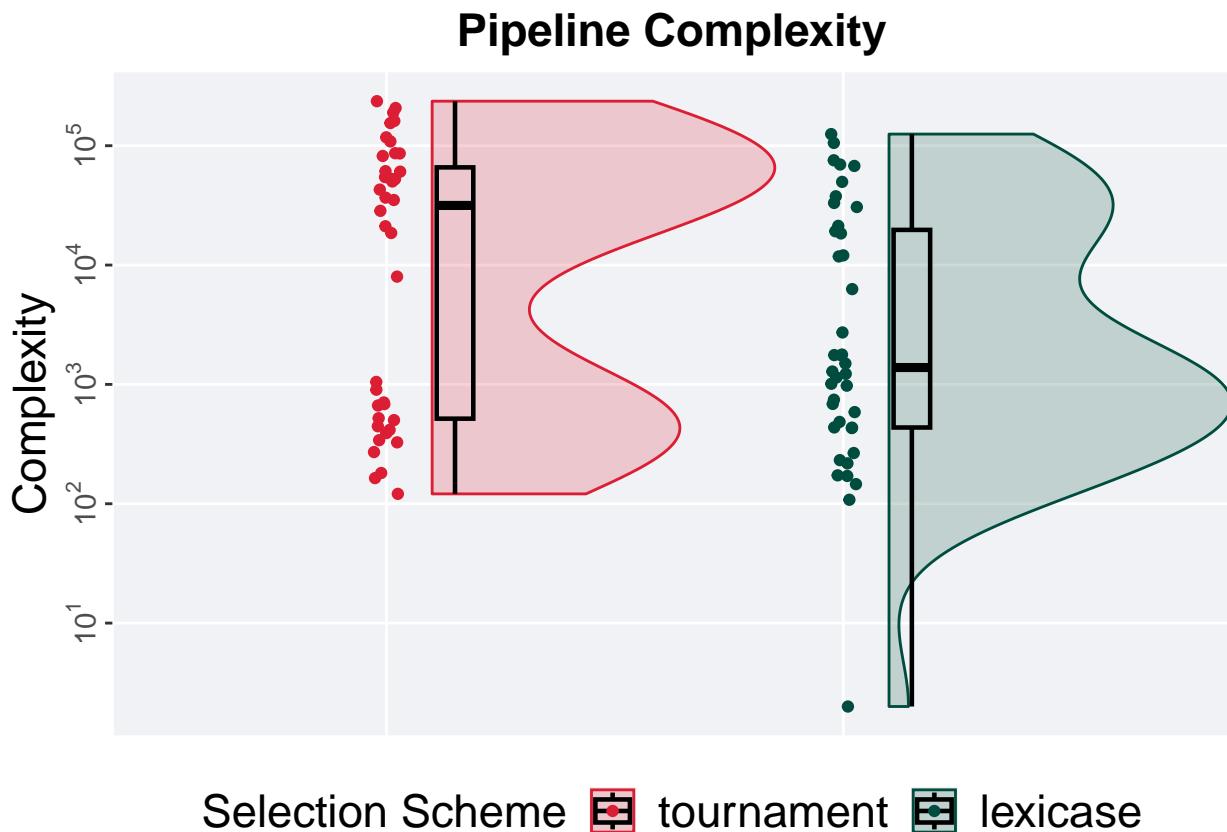
```
tournament_results <- filter(task_data, split == '10%' & selection == 'tournament')
lexicase_results <- filter(task_data, split == '10%' & selection == 'lexicase')
permutation_test(tournament_results$training_performance,
                 lexicase_results$training_performance,
                 seed = 14,
                 alternative = "1")
```

```
## [1] "observed_diff: -3.49007120627971"
## [1] "permutation_diffs[0.05 * n_permutations]: -1.65102702194372"
## [1] "reject null hypothesis"
## [1] "p-value: 0.00027"
```



4.2.3 Pipeline complexity

```
complexity_plot(filter(task_data, split == '10%'))
```



Summary statistics for the pipeline complexity of the selection schemes at the 5% selection set split:

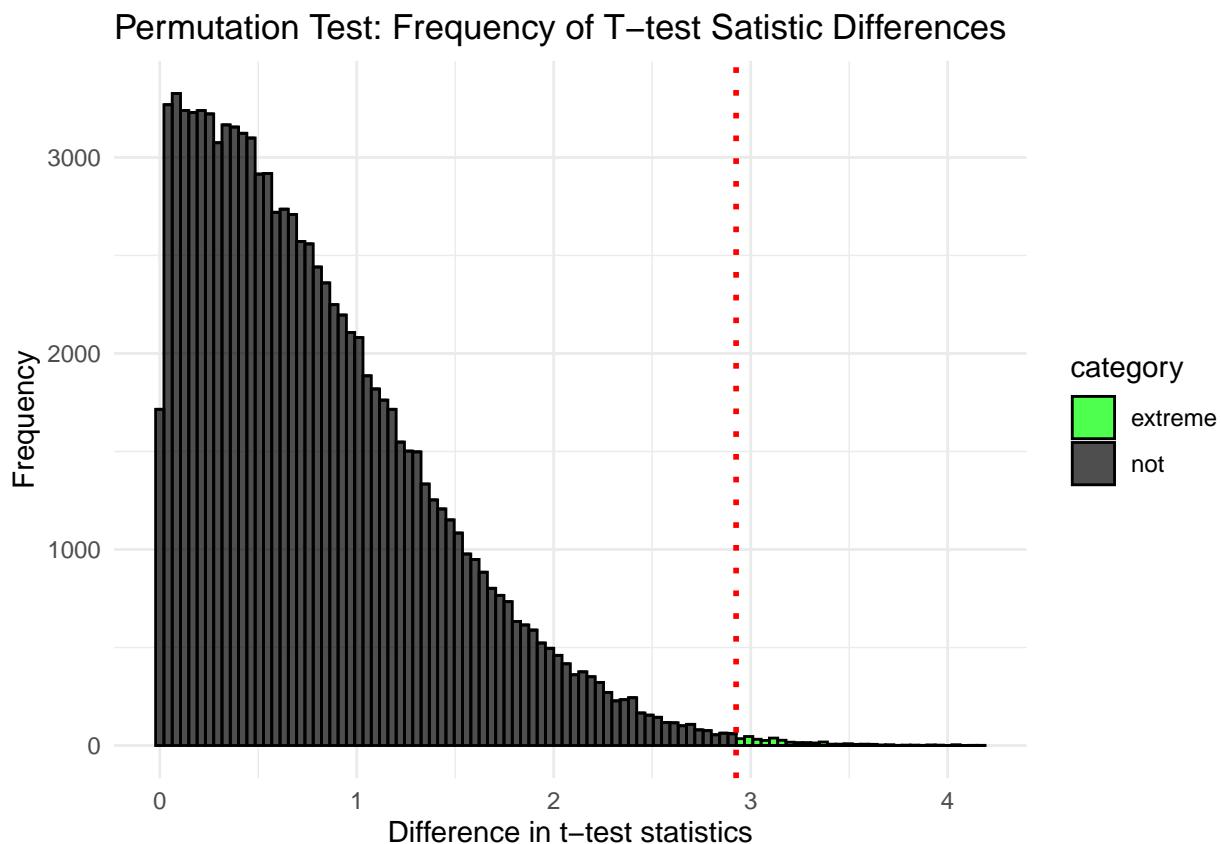
```
complexity_summary(filter(task_data, split == '10%'))
```

```
## # A tibble: 2 x 8
##   selection count na_cnt  min median   mean   max   IQR
##   <fct>     <int> <int> <dbl> <dbl> <dbl> <int> <dbl>
## 1 tournament     40     0  121 31778. 50292. 236081 65889.
## 2 lexicase       40     0     2  1389 17603. 125121 19298
```

The permutation test revealed that the results are:

```
tournament_results <- filter(task_data, split == '10%' & selection == 'tournament')
lexicase_results <- filter(task_data, split == '10%' & selection == 'lexicase')
permutation_test(tournament_results$testing_complexity,
                 lexicase_results$testing_complexity,
                 seed = 206,
                 alternative = "t")
```

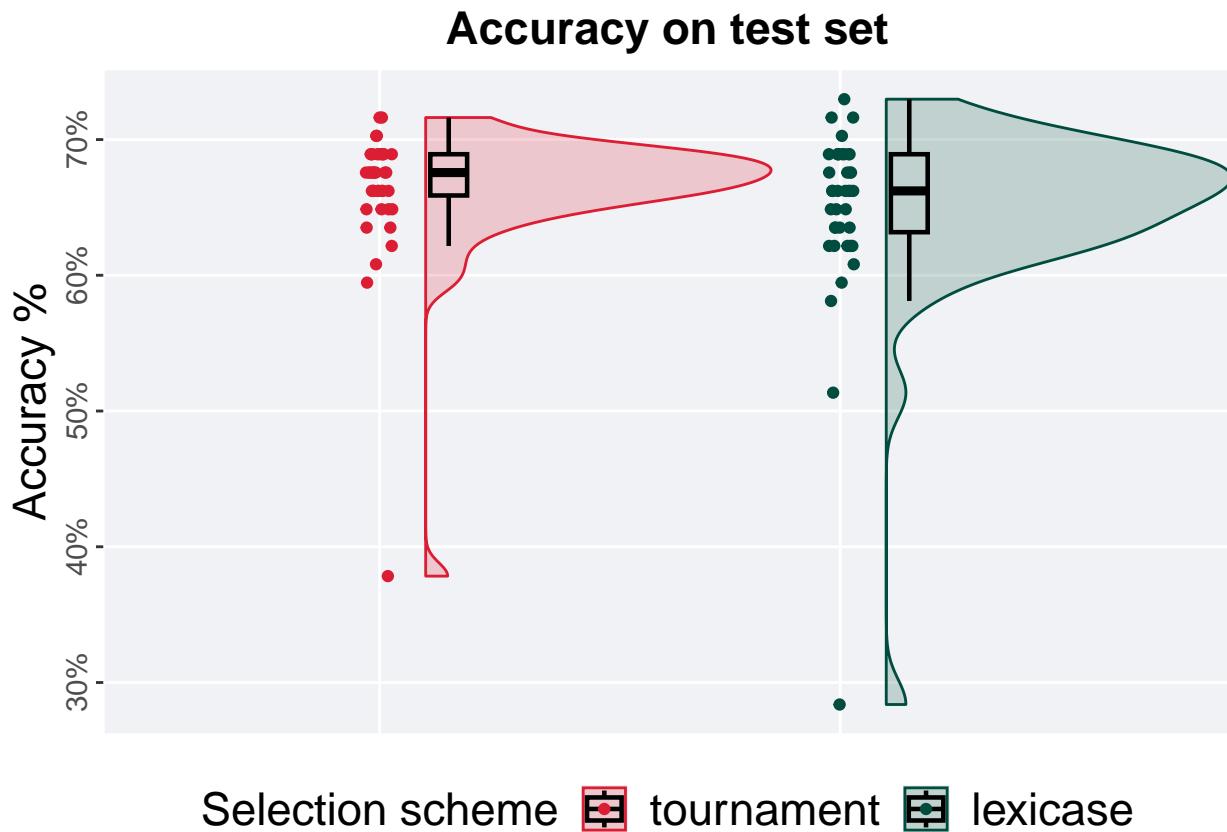
```
## [1] "observed_diff: 2.92603084073712"
## [1] "lower: -1.99246011044843"
## [1] "upper: 1.98335510453391"
## [1] "reject null hypothesis"
## [1] "p-value: 0.00349"
```



4.3 50%

4.3.1 Test accuracy

```
test_plot(filter(task_data, split == '50%'))
```



Summary statistics for the testing performance of the selection schemes at the 5% selection set split:

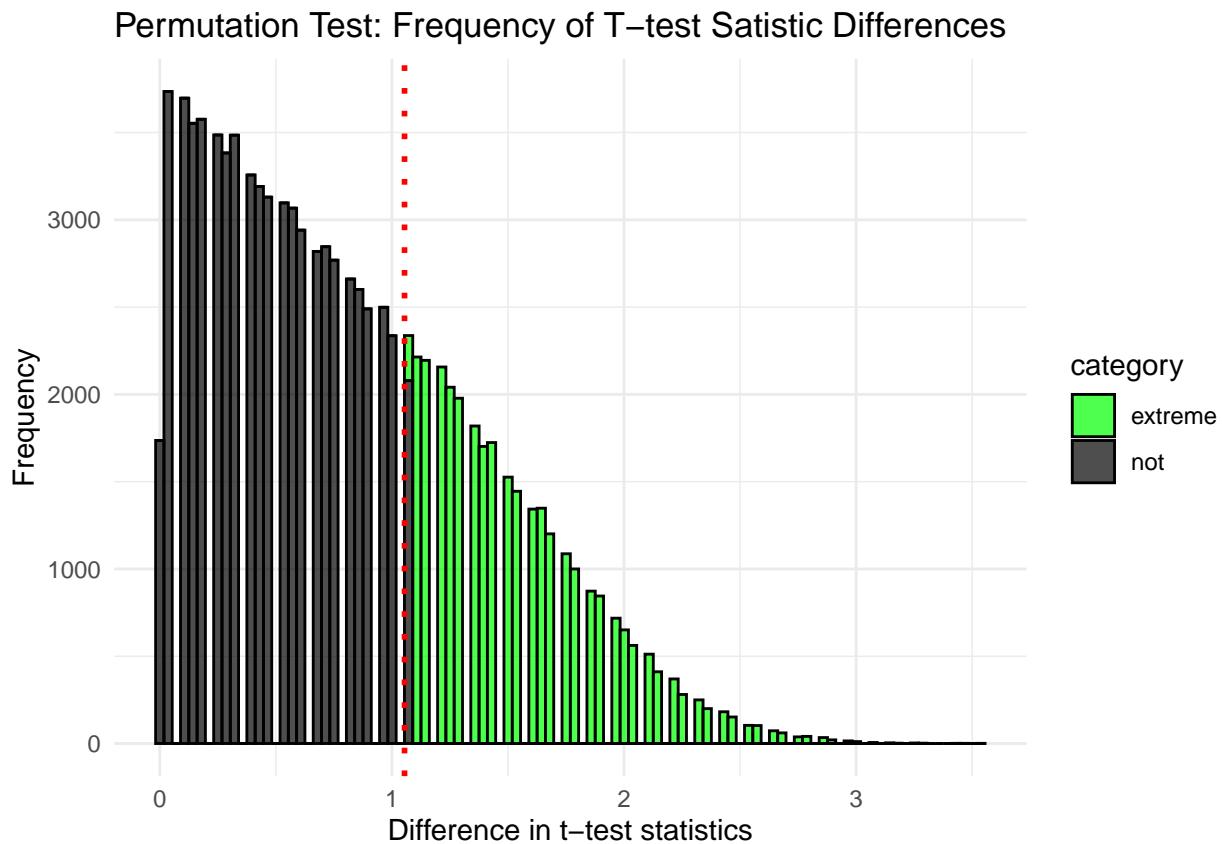
```
test_results_summary(filter(task_data, split == '50%'))
```

```
## # A tibble: 2 x 8
##   selection count na_cnt  min median  mean  max    IQR
##   <fct>     <int> <int> <dbl> <dbl> <dbl> <dbl> <dbl>
## 1 tournament     40     0 0.378 0.676 0.662 0.716 0.0304
## 2 lexicase       40     0 0.284 0.662 0.647 0.730 0.0574
```

The permutation test revealed that the results are:

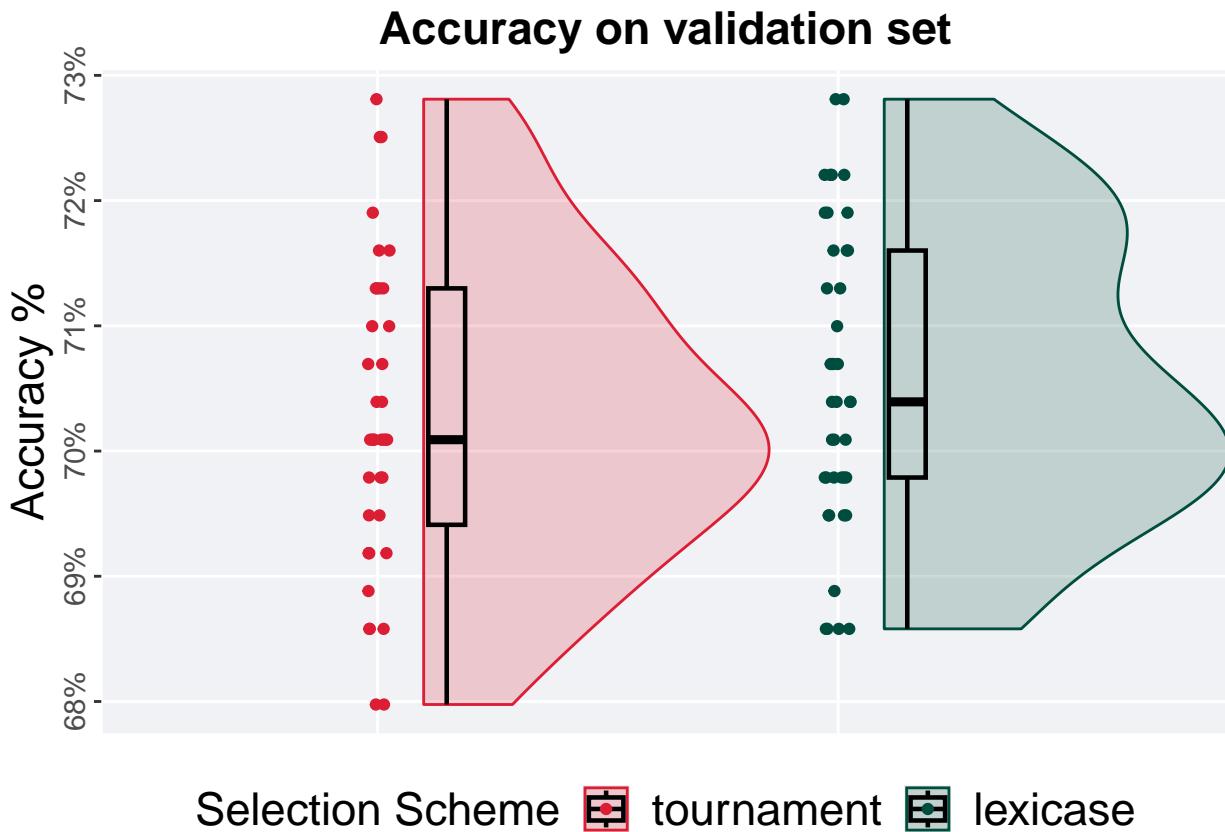
```
tournament_results <- filter(task_data, split == '50%' & selection == 'tournament')
lexicase_results <- filter(task_data, split == '50%' & selection == 'lexicase')
permutation_test(tournament_results$testing_performance,
                 lexicase_results$testing_performance,
                 seed = 15,
                 alternative = "t")
```

```
## [1] "observed_diff: 1.05447797672778"
## [1] "lower: -1.89851953697627"
## [1] "upper: 1.89851953697627"
## [1] "fail to reject null hypothesis"
## [1] "p-value: 0.31566"
```



4.3.2 Validation accuracy

```
validation_plot(filter(task_data, split == '50%'))
```



Summary statistics for the testing performance of the selection schemes at the 5% selection set split:

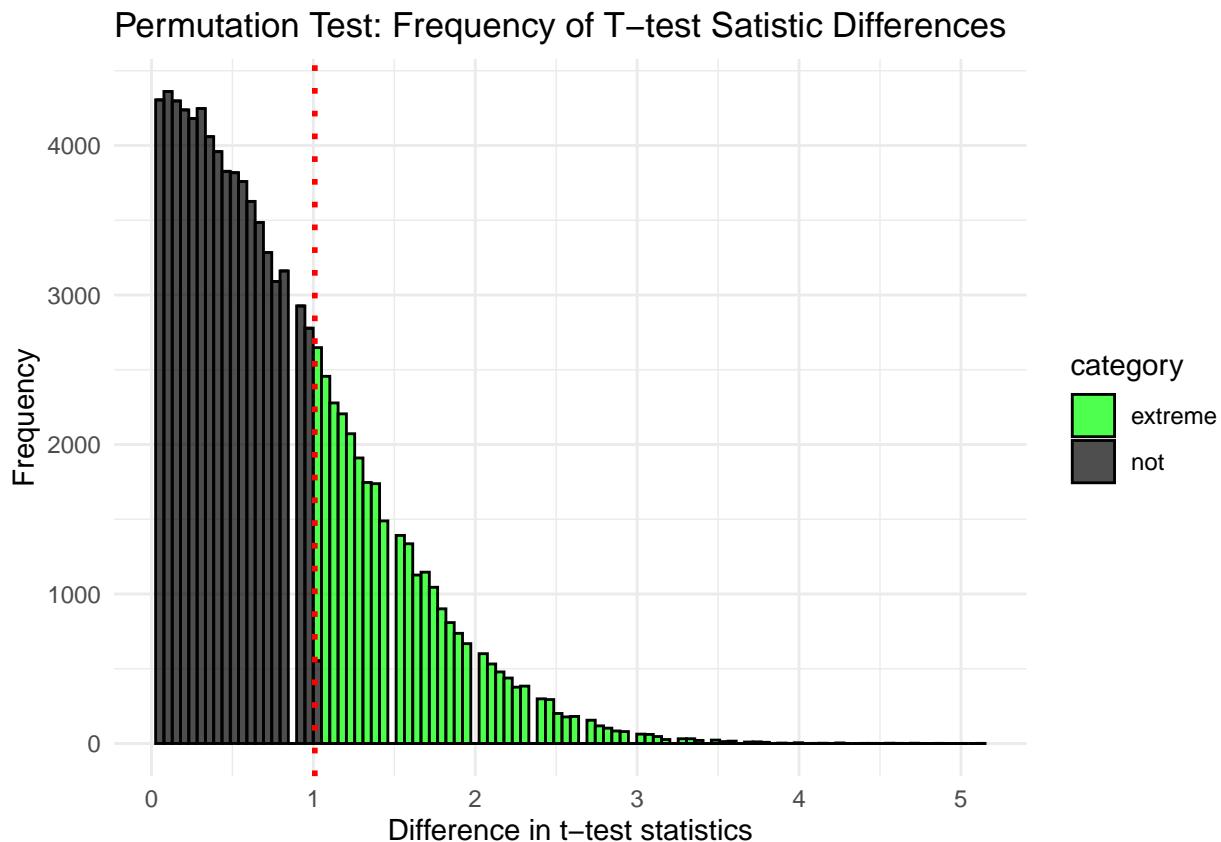
```
validation_accuracy_summary(filter(task_data, split == '50%'))
```

```
## # A tibble: 2 x 8
##   selection count na_cnt  min median  mean   max     IQR
##   <fct>     <int> <int> <dbl> <dbl> <dbl> <dbl> <dbl>
## 1 tournament     40     0 0.680 0.701 0.703 0.728 0.0189
## 2 lexicase       40     0 0.686 0.704 0.706 0.728 0.0181
```

The permutation test revealed that the results are:

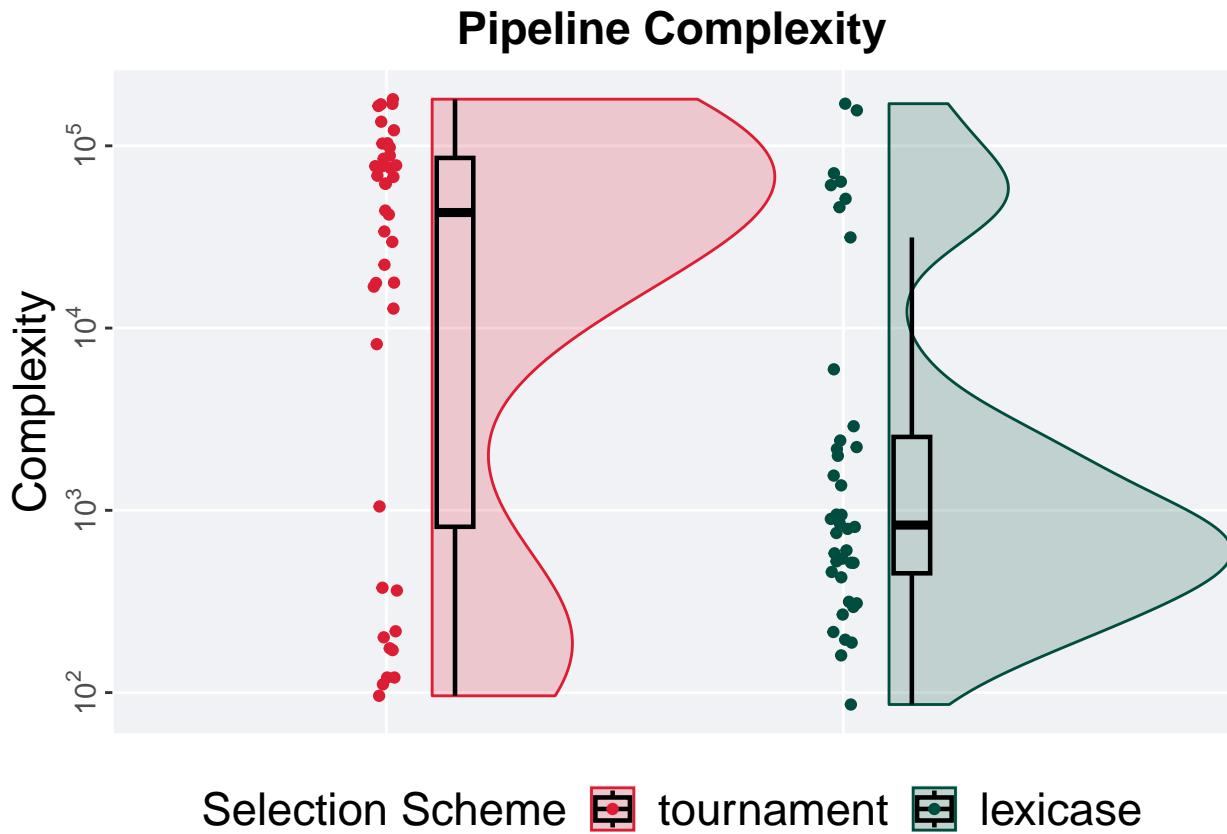
```
tournament_results <- filter(task_data, split == '50%' & selection == 'tournament')
lexicase_results <- filter(task_data, split == '50%' & selection == 'lexicase')
permutation_test(tournament_results$training_performance,
                 lexicase_results$training_performance,
                 seed = 16,
                 alternative = "t")
```

```
## [1] "observed_diff: -1.0089201210214"
## [1] "lower: -1.97083964694619"
## [1] "upper: 1.9708394540355"
## [1] "fail to reject null hypothesis"
## [1] "p-value: 0.32044"
```



4.3.3 Pipeline complexity

```
complexity_plot(filter(task_data, split == '50%'))
```



Summary statistics for the pipeline complexity of the selection schemes at the 5% selection set split:

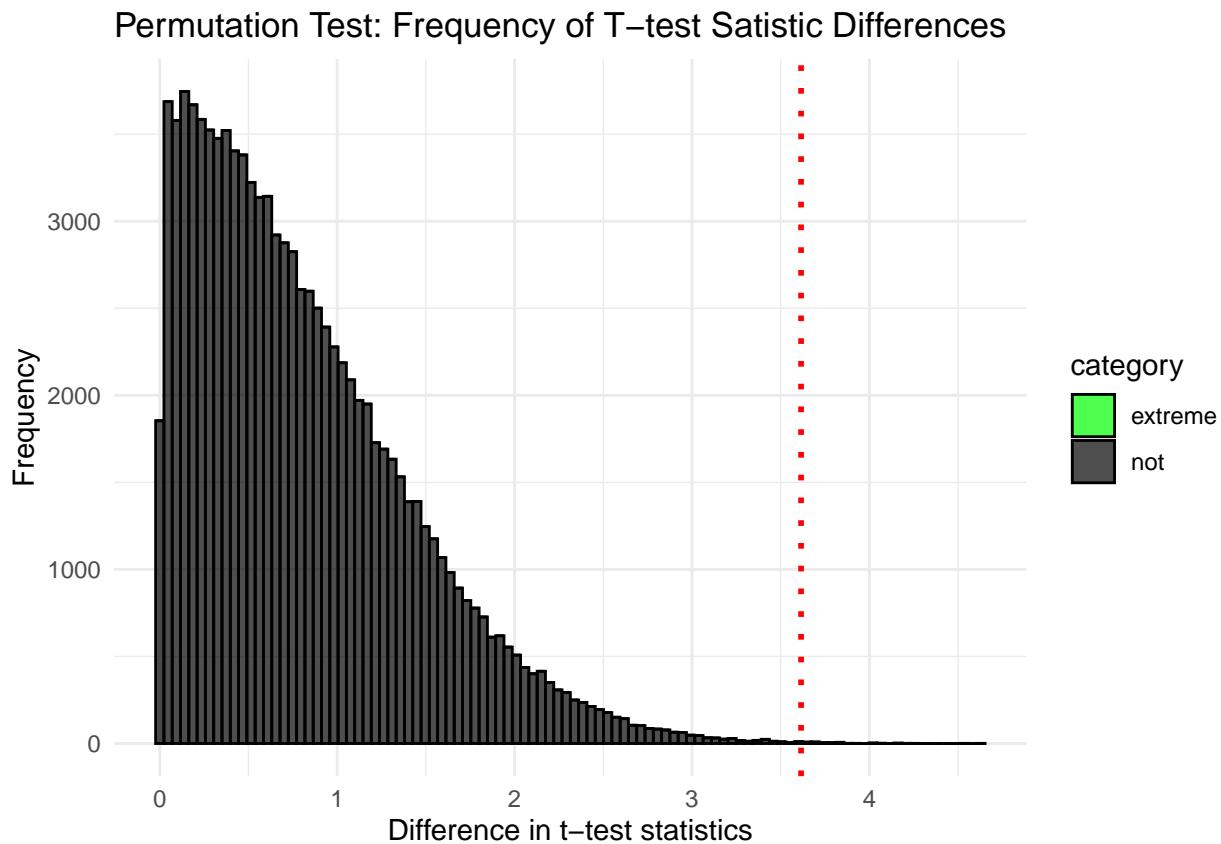
```
complexity_summary(filter(task_data, split == '50%'))
```

```
## # A tibble: 2 x 8
##   selection count na_cnt  min median   mean   max   IQR
##   <fct>     <int> <int> <dbl> <dbl> <dbl> <int> <dbl>
## 1 tournament     40     0    96 43061  55894. 180231 84942
## 2 lexicase       40     0    86   830. 17083. 170261 2084.
```

The permutation test revealed that the results are:

```
tournament_results <- filter(task_data, split == '50%' & selection == 'tournament')
lexicase_results <- filter(task_data, split == '50%' & selection == 'lexicase')
permutation_test(tournament_results$testing_complexity,
                 lexicase_results$testing_complexity,
                 seed = 207,
                 alternative = "t")
```

```
## [1] "observed_diff: 3.61488180927466"
## [1] "lower: -1.99801833753129"
## [1] "upper: 1.98261915374391"
## [1] "reject null hypothesis"
## [1] "p-value: 0.00055"
```

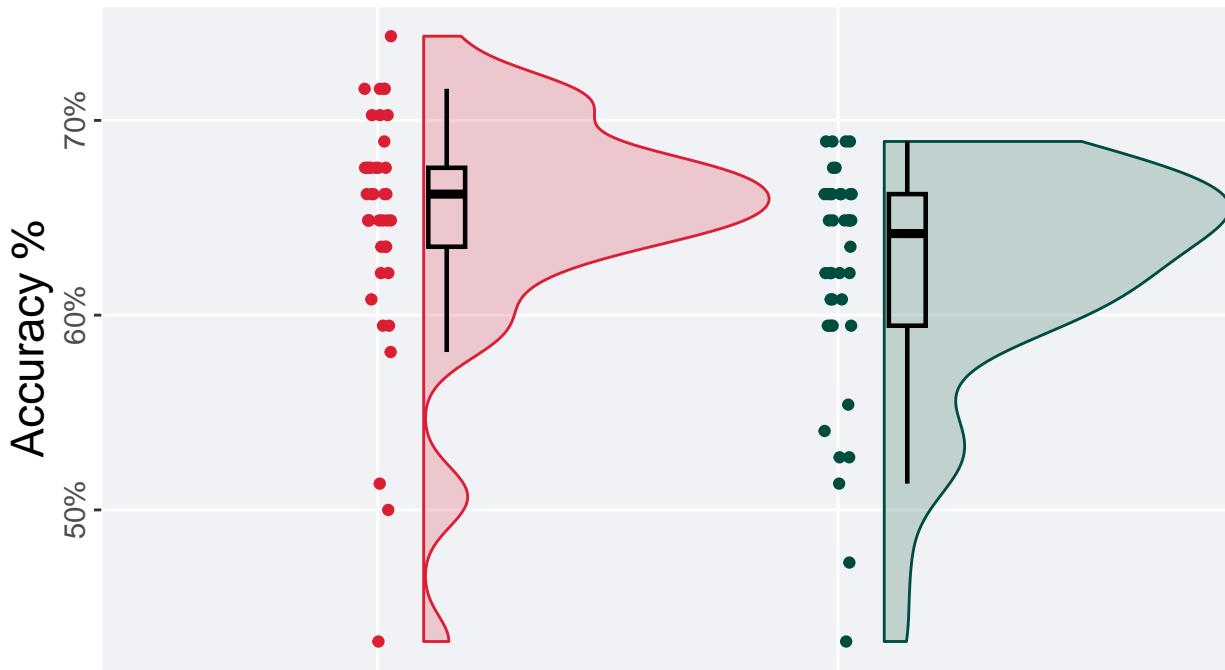


4.4 90%

4.4.1 Test accuracy

```
test_plot(filter(task_data, split == '90%'))
```

Accuracy on test set



Selection scheme tournament lexicase

Summary statistics for the testing performance of the selection schemes at the 5% selection set split:

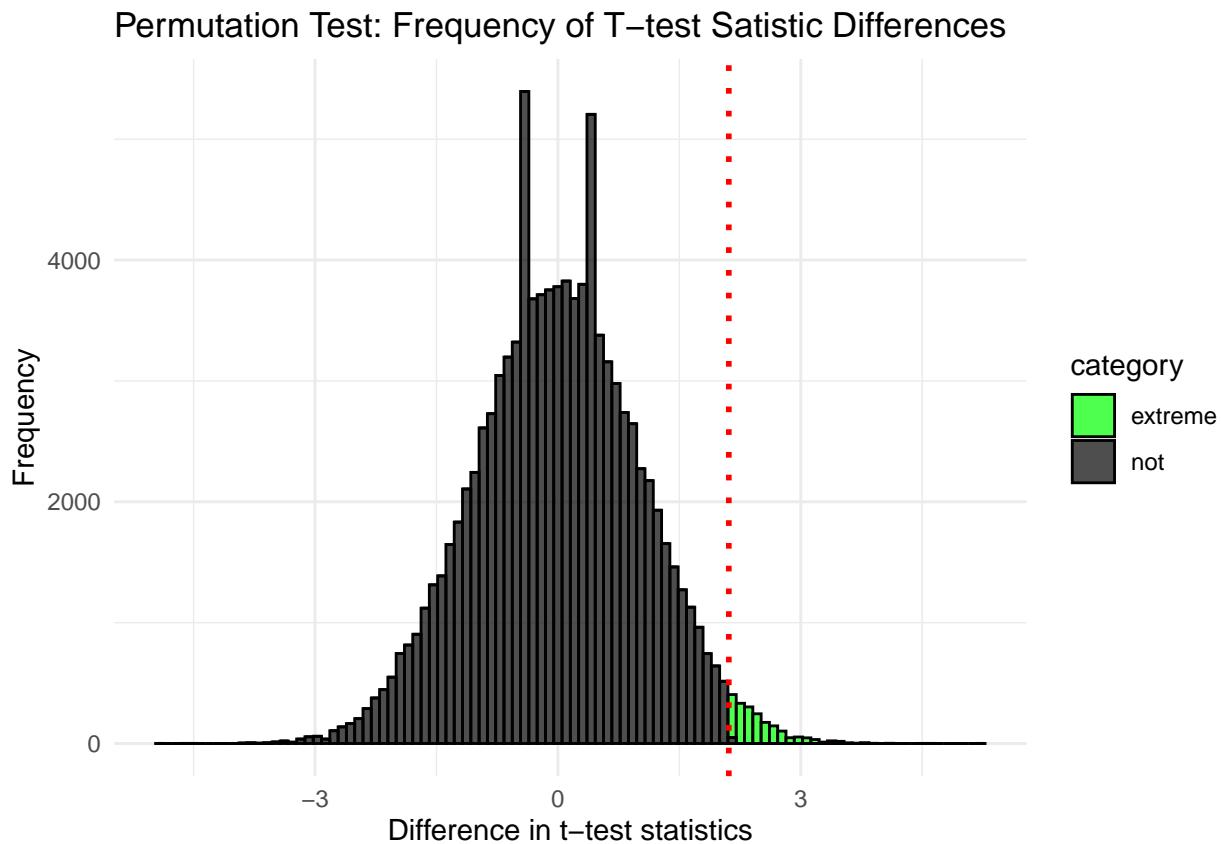
```
test_results_summary(filter(task_data, split == '90%'))
```

```
## # A tibble: 2 x 8
##   selection  count  na_cnt  min  median  mean  max  IQR
##   <fct>      <int>   <int> <dbl>  <dbl> <dbl> <dbl> <dbl>
## 1 tournament     40      0 0.432  0.662 0.649 0.743 0.0405
## 2 lexicase       40      0 0.432  0.642 0.620 0.689 0.0676
```

The permutation test revealed that the results are:

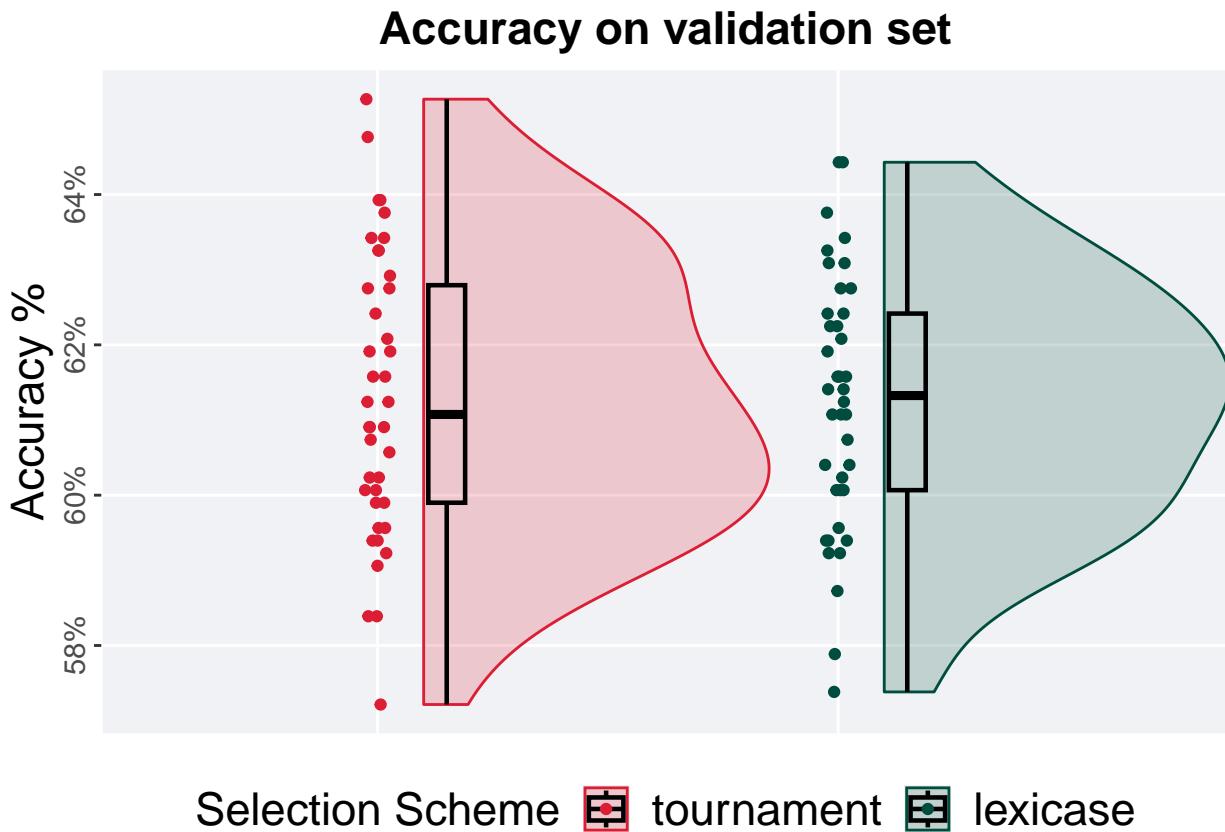
```
tournament_results <- filter(task_data, split == '90%' & selection == 'tournament')
lexicase_results <- filter(task_data, split == '90%' & selection == 'lexicase')
permutation_test(tournament_results$testing_performance,
                 lexicase_results$testing_performance,
                 seed = 17,
                 alternative = "g")
```

```
## [1] "observed_diff: 2.11112337167441"
## [1] "permutation_diffs[0.95 * n_permutations]: 1.64634620519505"
## [1] "reject null hypothesis"
## [1] "p-value: 0.01917"
```



4.4.2 Validation accuracy

```
validation_plot(filter(task_data, split == '90%'))
```



Summary statistics for the testing performance of the selection schemes at the 5% selection set split:

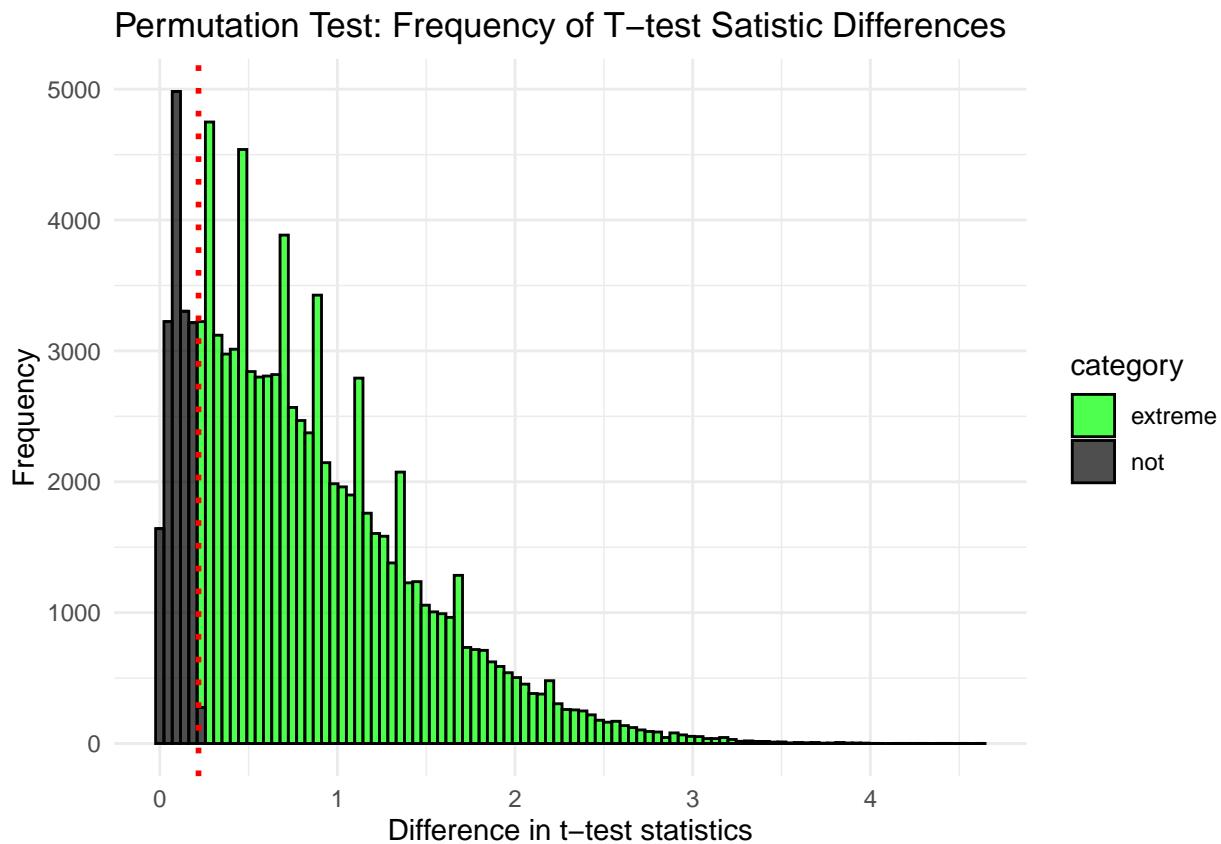
```
validation_accuracy_summary(filter(task_data, split == '90%'))
```

```
## # A tibble: 2 x 8
##   selection count na_cnt  min median  mean   max     IQR
##   <fct>     <int> <int> <dbl> <dbl> <dbl> <dbl> <dbl>
## 1 tournament     40     0 0.572 0.611 0.613 0.653 0.0289
## 2 lexicase       40     0 0.574 0.613 0.612 0.644 0.0235
```

The permutation test revealed that the results are:

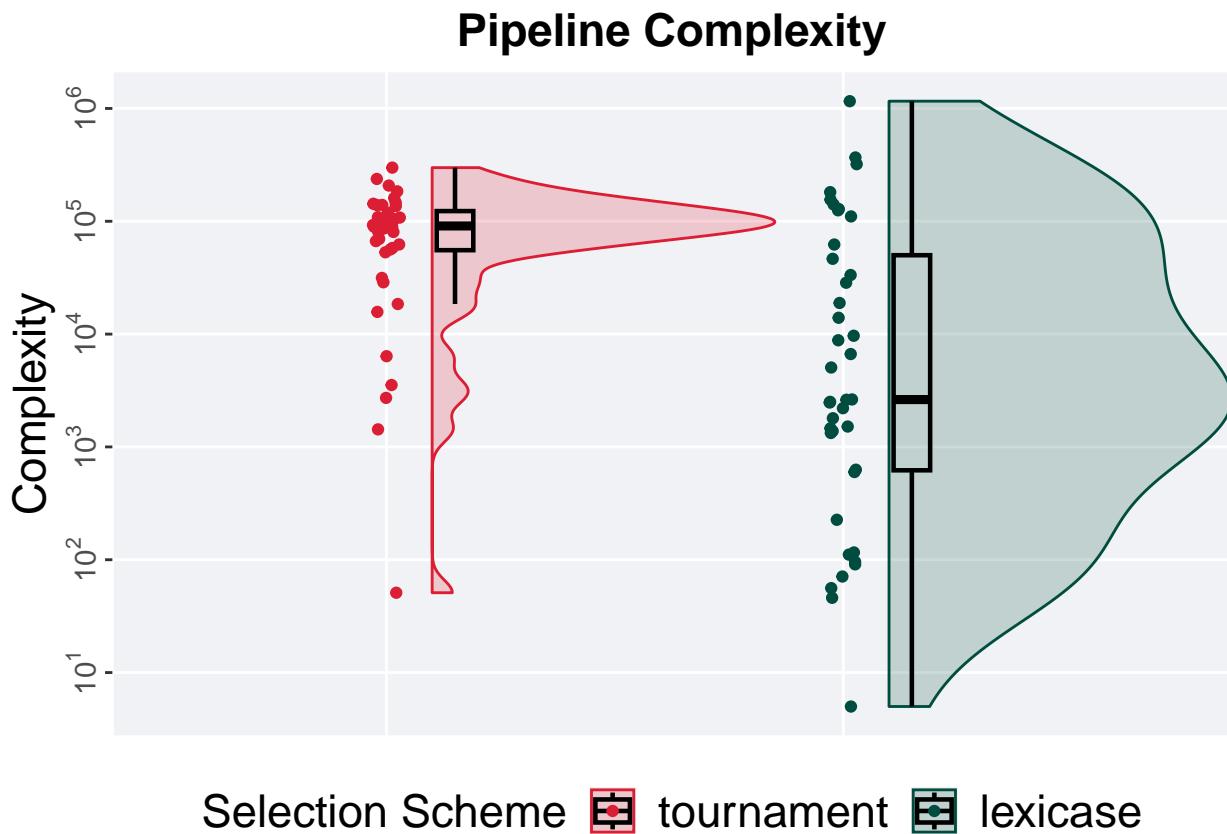
```
tournament_results <- filter(task_data, split == '90%' & selection == 'tournament')
lexicase_results <- filter(task_data, split == '90%' & selection == 'lexicase')
permutation_test(tournament_results$training_performance,
                 lexicase_results$training_performance,
                 seed = 18,
                 alternative = "t")

## [1] "observed_diff: 0.218175712374656"
## [1] "lower: -1.99095942630688"
## [1] "upper: 1.99096129334562"
## [1] "fail to reject null hypothesis"
## [1] "p-value: 0.83358"
```



4.4.3 Pipeline complexity

```
complexity_plot(filter(task_data, split == '90%'))
```



Summary statistics for the pipeline complexity of the selection schemes at the 5% selection set split:

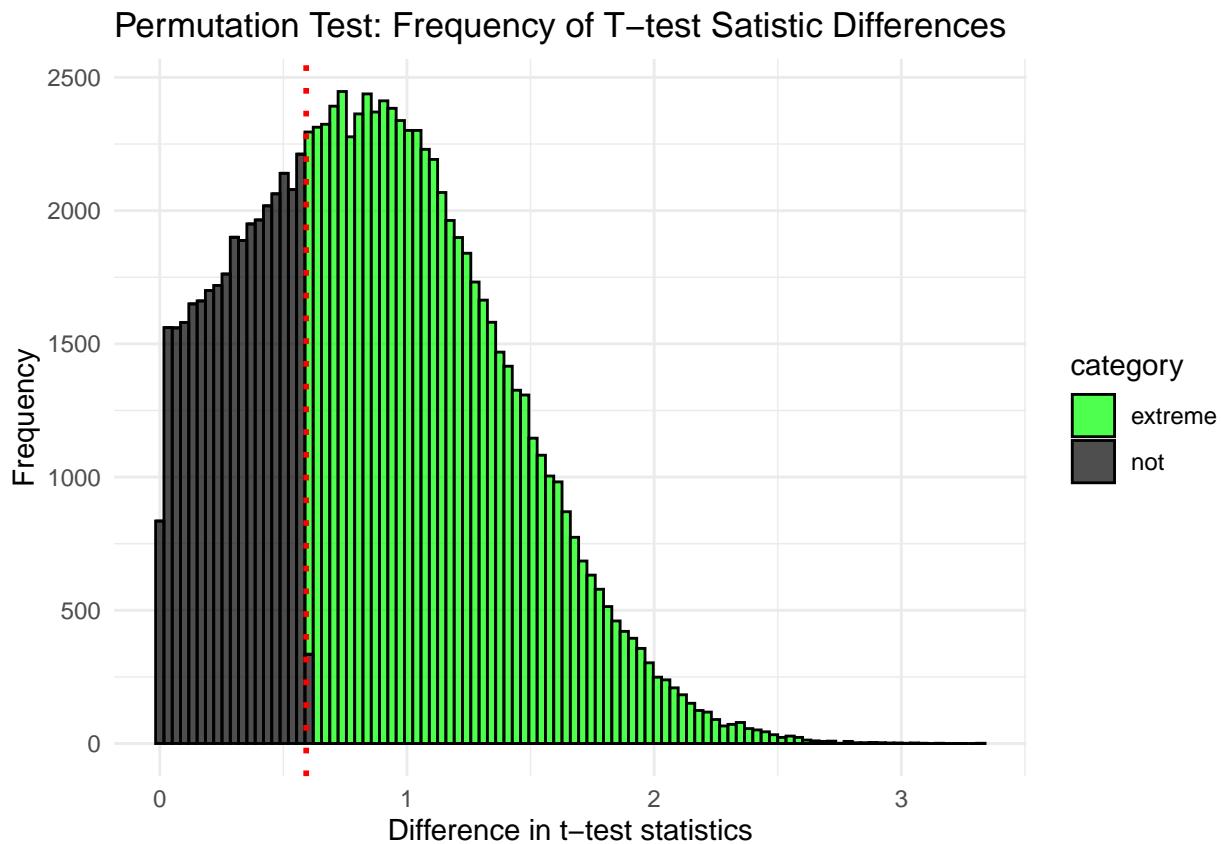
```
complexity_summary(filter(task_data, split == '90%'))
```

```
## # A tibble: 2 x 8
##   selection count na_cnt  min median  mean      max    IQR
##   <fct>     <int> <int> <dbl> <dbl> <dbl> <int> <dbl>
## 1 tournament     40     0    51 90881  92988. 298841 67888.
## 2 lexicase       40     0     5  2628.  73633. 1159841 49808.
```

The permutation test revealed that the results are:

```
tournament_results <- filter(task_data, split == '90%' & selection == 'tournament')
lexicase_results <- filter(task_data, split == '90%' & selection == 'lexicase')
permutation_test(tournament_results$testing_complexity,
                 lexicase_results$testing_complexity,
                 seed = 208,
                 alternative = "t")
```

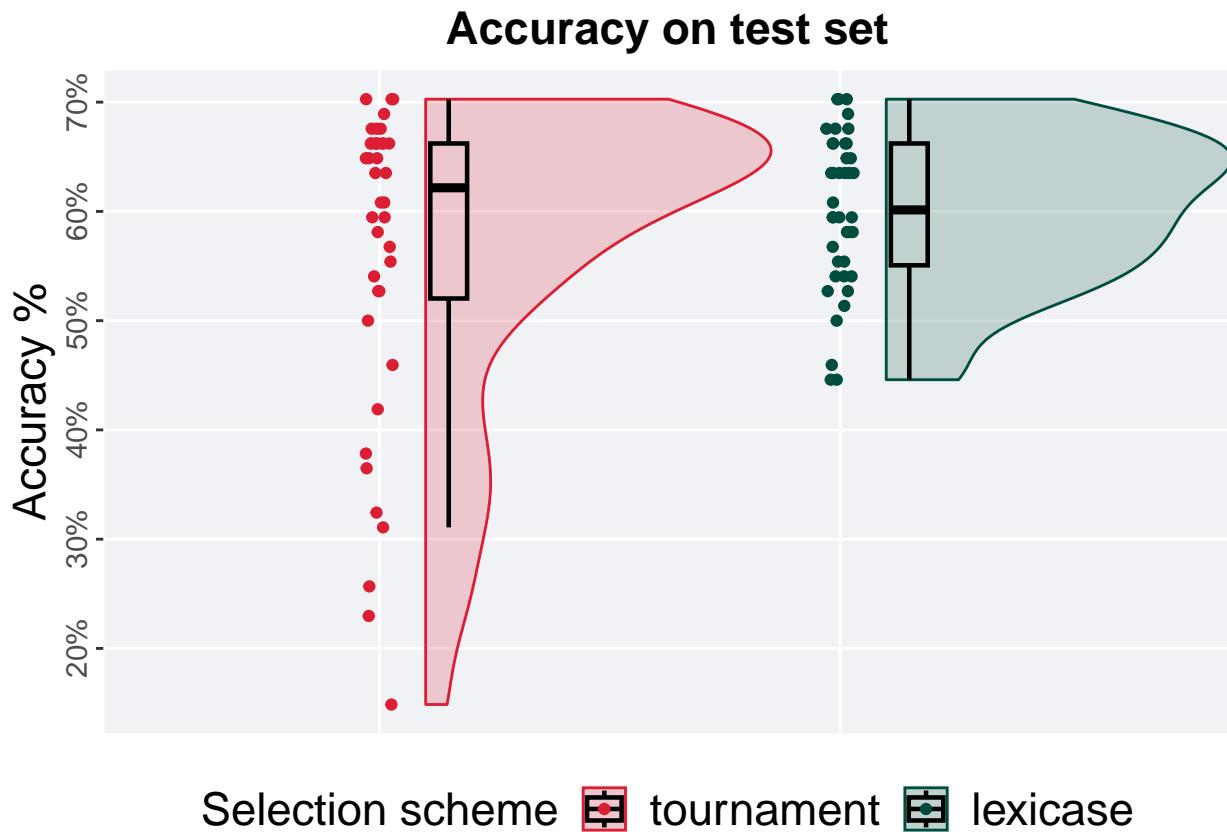
```
## [1] "observed_diff: 0.59233678961419"
## [1] "lower: -1.75930428651117"
## [1] "upper: 1.75817620360251"
## [1] "fail to reject null hypothesis"
## [1] "p-value: 0.67422"
```



4.5 95%

4.5.1 Test accuracy

```
test_plot(filter(task_data, split == '95%'))
```



Summary statistics for the testing performance of the selection schemes at the 5% selection set split:

```
test_results_summary(filter(task_data, split == '95%'))
```

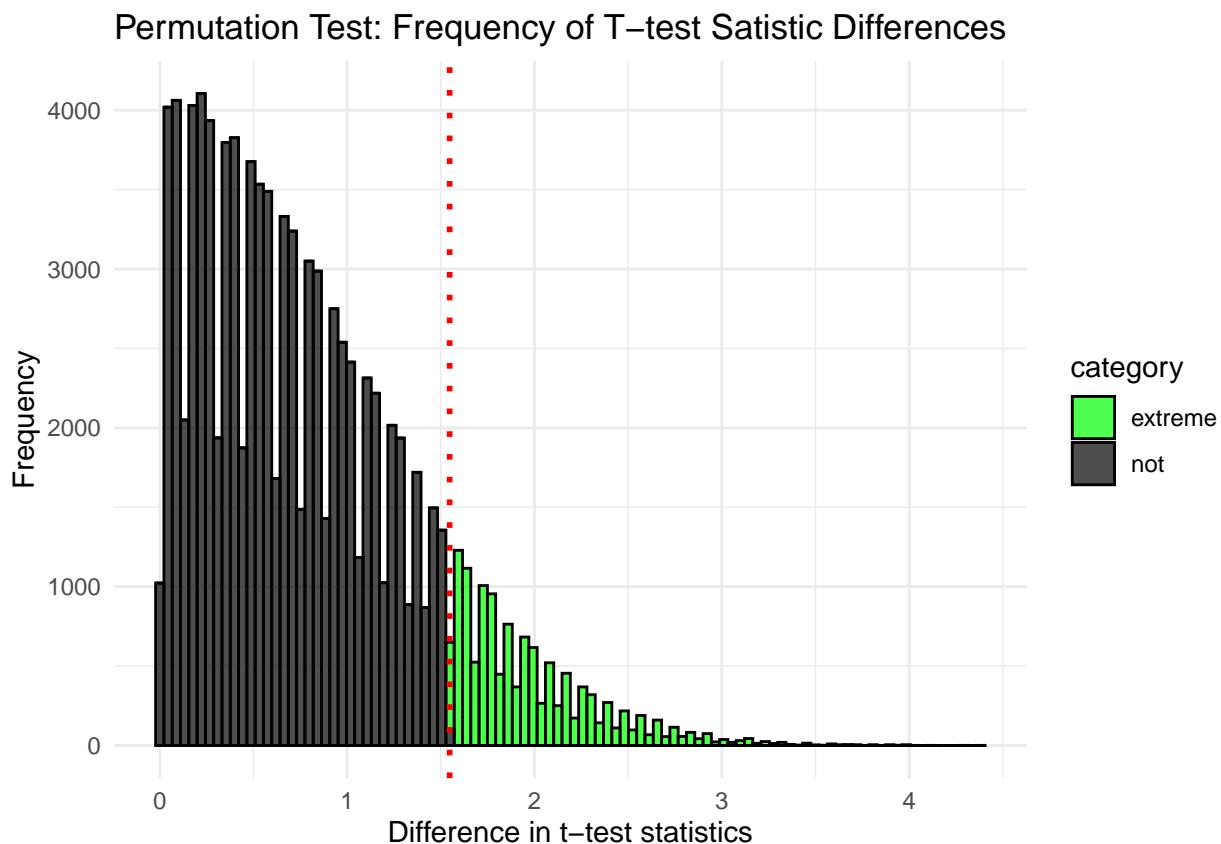
```
## # A tibble: 2 x 8
##   selection count na_cnt  min median  mean   max   IQR
##   <fct>     <int> <int> <dbl> <dbl> <dbl> <dbl> <dbl>
## 1 tournament     40     0 0.149  0.622  0.561  0.703  0.142
## 2 lexicase       40     0 0.446  0.601  0.601  0.703  0.111
```

The permutation test revealed that the results are:

```
tournament_results <- filter(task_data, split == '95%' & selection == 'tournament')
lexicase_results <- filter(task_data, split == '95%' & selection == 'lexicase')
```

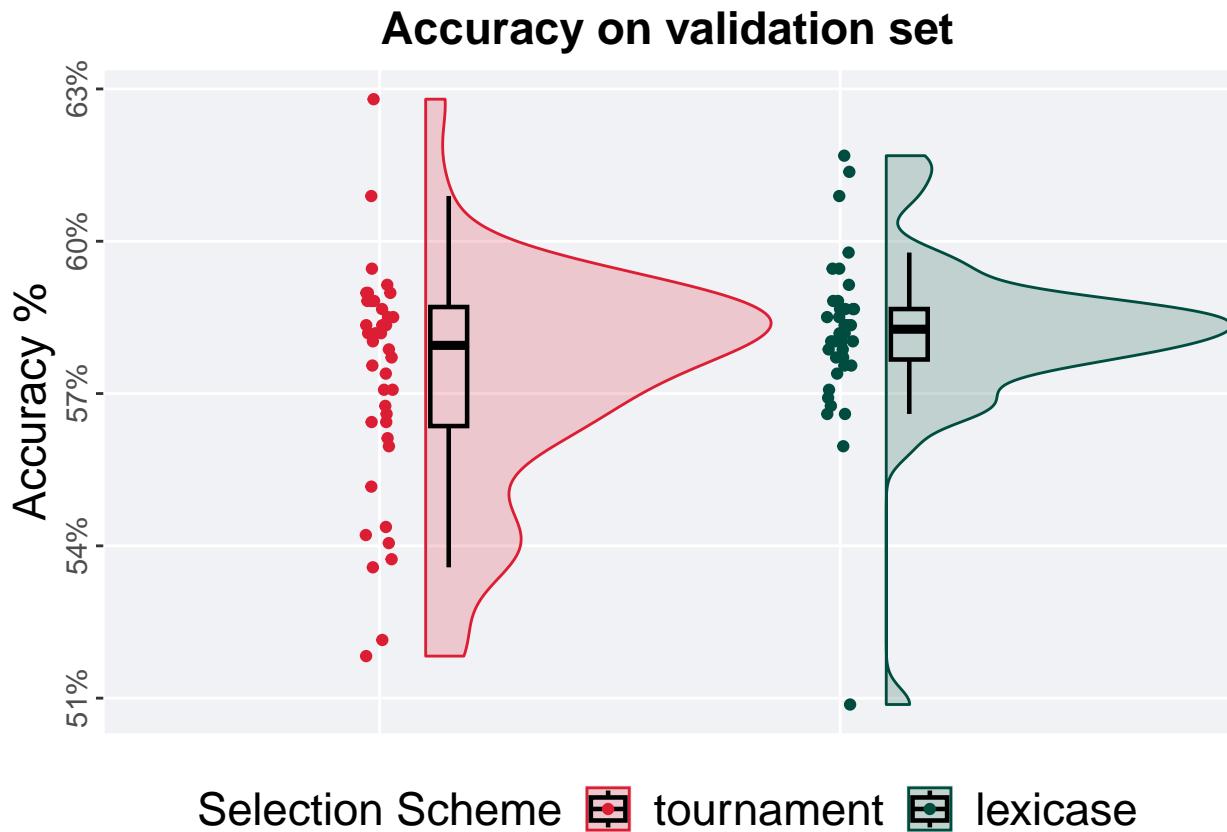
```
permutation_test(tournament_results$testing_performance,
                 lexicase_results$testing_performance,
                 seed = 19,
                 alternative = "t")
```

```
## [1] "observed_diff: -1.54710562137524"
## [1] "lower: -1.95772948677679"
## [1] "upper: 1.95772969216945"
## [1] "fail to reject null hypothesis"
## [1] "p-value: 0.12617"
```



4.5.2 Validation accuracy

```
validation_plot(filter(task_data, split == '95%'))
```



Summary statistics for the testing performance of the selection schemes at the 5% selection set split:

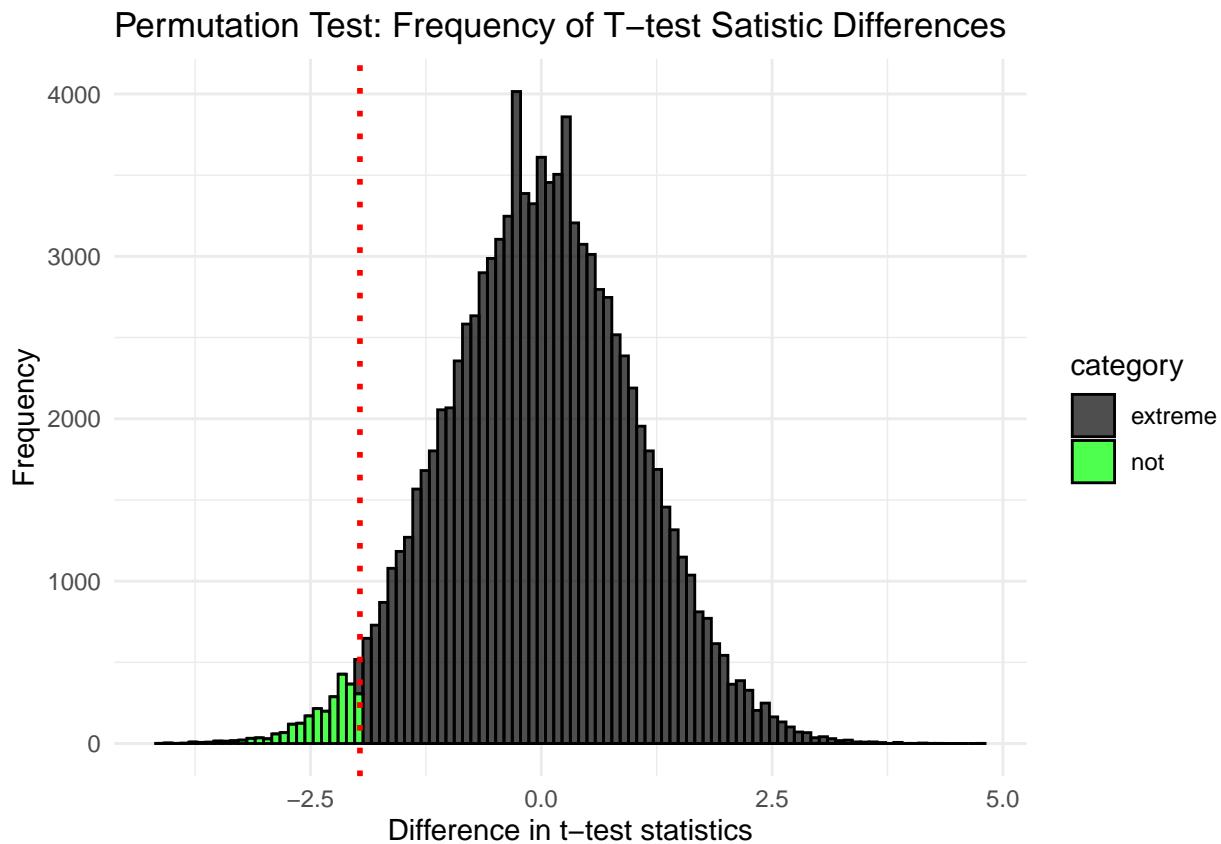
```
validation_accuracy_summary(filter(task_data, split == '95%'))
```

```
## # A tibble: 2 x 8
##   selection  count  na_cnt  min  median  mean  max      IQR
##   <fct>      <int>  <int>  <dbl>  <dbl>  <dbl>  <dbl>  <dbl>
## 1 tournament    40      0  0.518  0.579  0.573  0.628  0.0234
## 2 lexicase      40      0  0.509  0.583  0.582  0.617  0.00994
```

The permutation test revealed that the results are:

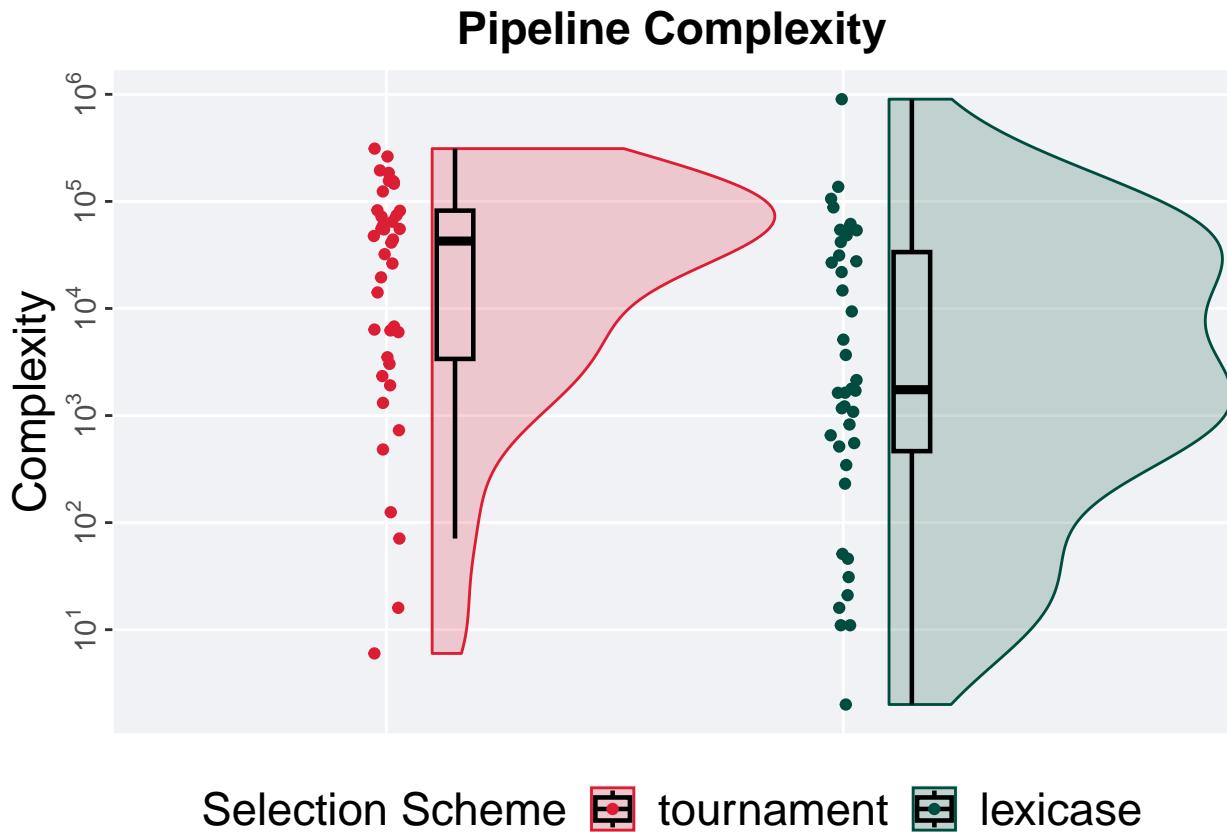
```
tournament_results <- filter(task_data, split == '95%' & selection == 'tournament')
lexicase_results <- filter(task_data, split == '95%' & selection == 'lexicase')
permutation_test(tournament_results$training_performance,
                 lexicase_results$training_performance,
                 seed = 20,
                 alternative = "1")
```

```
## [1] "observed_diff: -1.96456434004405"
## [1] "permutation_diffs[0.05 * n_permutations]: -1.64935315376094"
## [1] "reject null hypothesis"
## [1] "p-value: 0.02535"
```



4.5.3 Pipeline complexity

```
complexity_plot(filter(task_data, split == '95%'))
```



Summary statistics for the pipeline complexity of the selection schemes at the 5% selection set split:

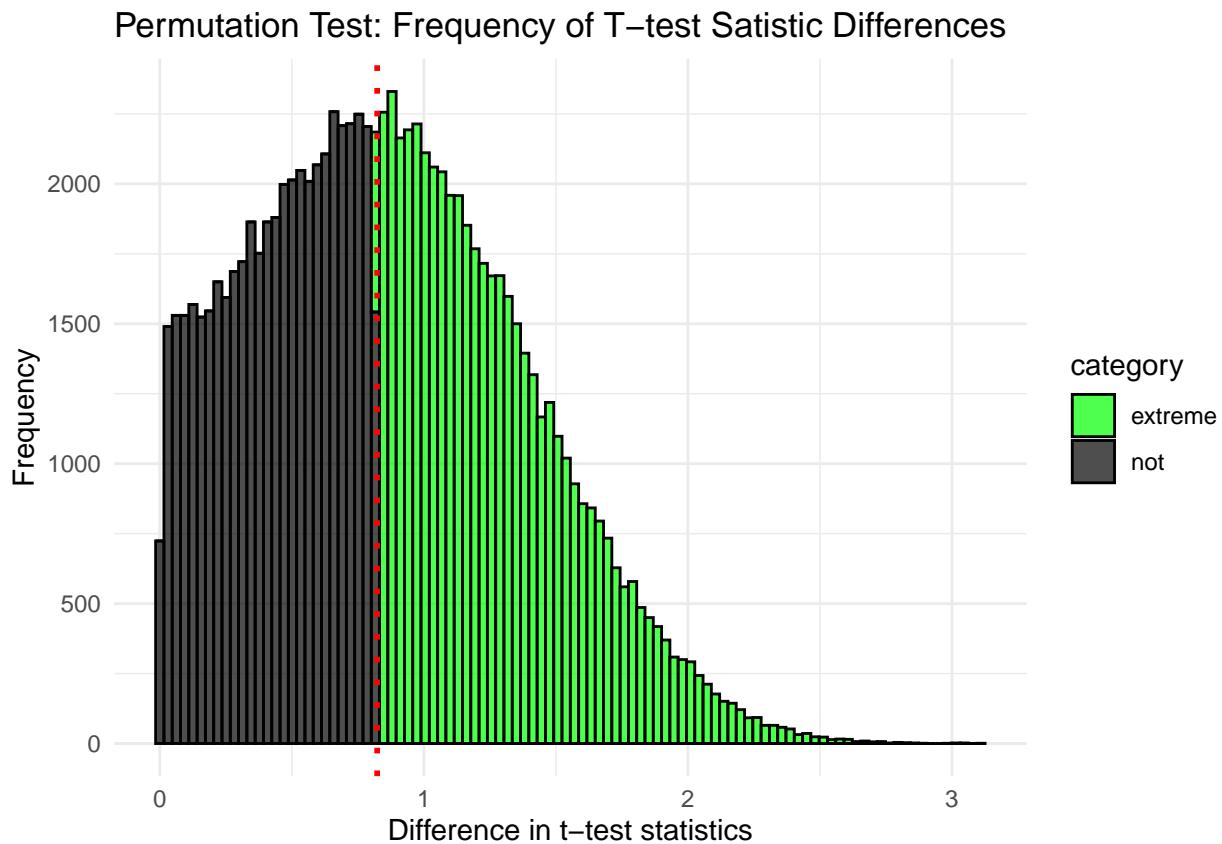
```
complexity_summary(filter(task_data, split == '95%'))
```

```
## # A tibble: 2 x 8
##   selection count na_cnt  min median   mean   max   IQR
##   <fct>     <int> <int> <dbl> <dbl> <dbl> <dbl>
## 1 tournament     40     0     6 42636 63777. 311351 78630.
## 2 lexicase       40     0     2 1744. 42617. 901261 33453
```

The permutation test revealed that the results are:

```
tournament_results <- filter(task_data, split == '95%' & selection == 'tournament')
lexicase_results <- filter(task_data, split == '95%' & selection == 'lexicase')
permutation_test(tournament_results$testing_complexity,
                 lexicase_results$testing_complexity,
                 seed = 209,
                 alternative = "t")
```

```
## [1] "observed_diff: 0.82326384377601"
## [1] "lower: -1.76792275980242"
## [1] "upper: 1.76844870536208"
## [1] "fail to reject null hypothesis"
## [1] "p-value: 0.51153"
```



Chapter 5

Task 359955

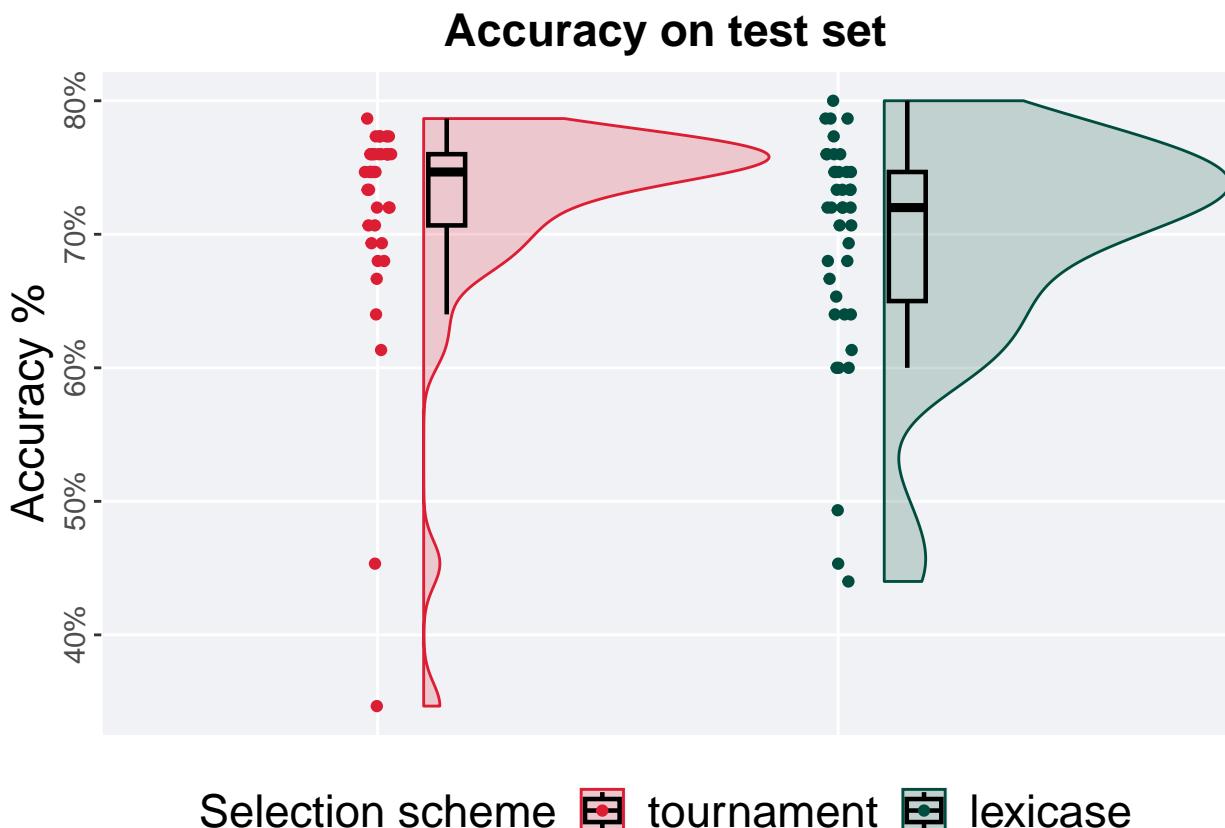
We present the results of our analysis of task 359955 with the different selection set splits used in our study.

```
task_data <- filter(results, task_id == 359955)
```

5.1 5%

5.1.1 Test accuracy

```
test_plot(filter(task_data, split == '5%'))
```



Summary statistics for the testing performance of the selection schemes at the 5% selection set split:

```
test_results_summary(filter(task_data, split == '5%'))
```

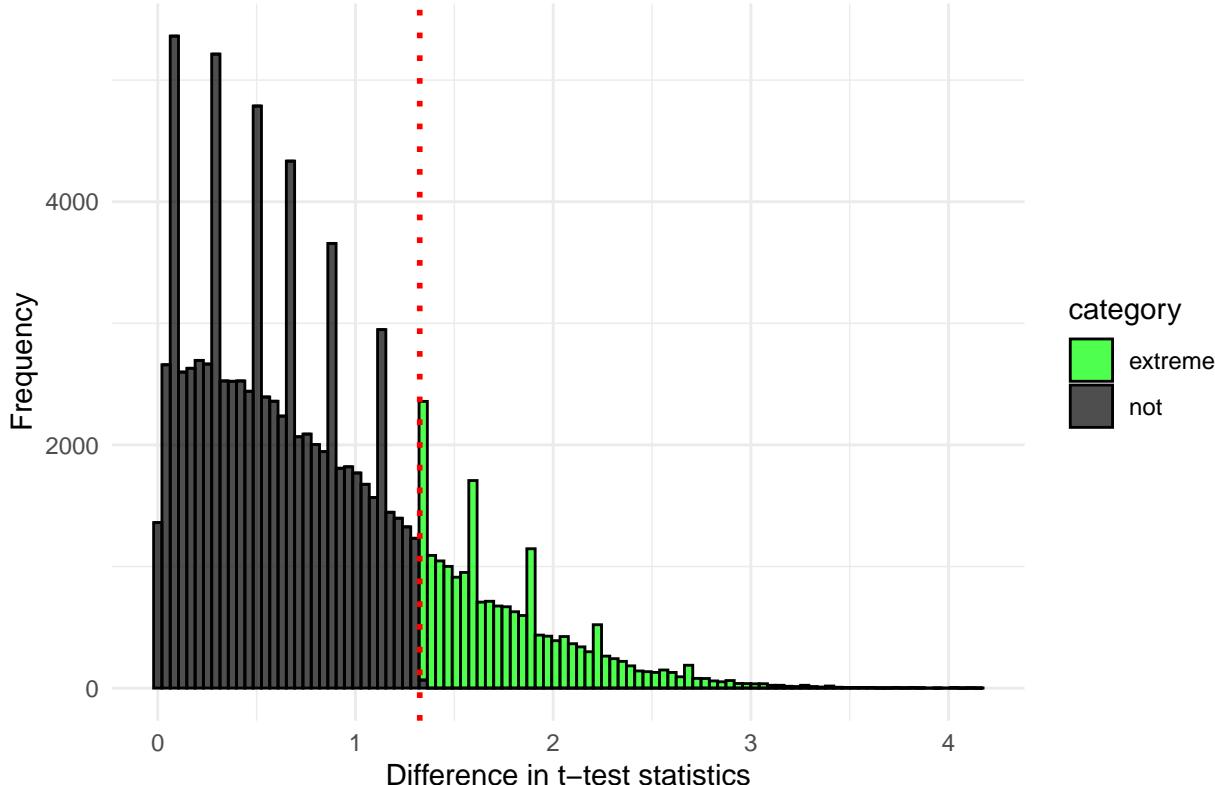
```
## # A tibble: 2 x 8
##   selection count na_cnt  min median  mean   max    IQR
##   <fct>     <int> <int> <dbl> <dbl> <dbl> <dbl> <dbl>
## 1 tournament     40     0 0.347  0.747  0.719  0.787 0.0533
## 2 lexicase       40     0 0.44    0.72   0.694  0.8    0.0967
```

The permutation test revealed that the results are:

```
tournament_results <- filter(task_data, split == '5%' & selection == 'tournament')
lexicase_results <- filter(task_data, split == '5%' & selection == 'lexicase')
permutation_test(tournament_results$testing_performance,
                 lexicase_results$testing_performance,
                 seed = 31,
                 alternative = "t")
```

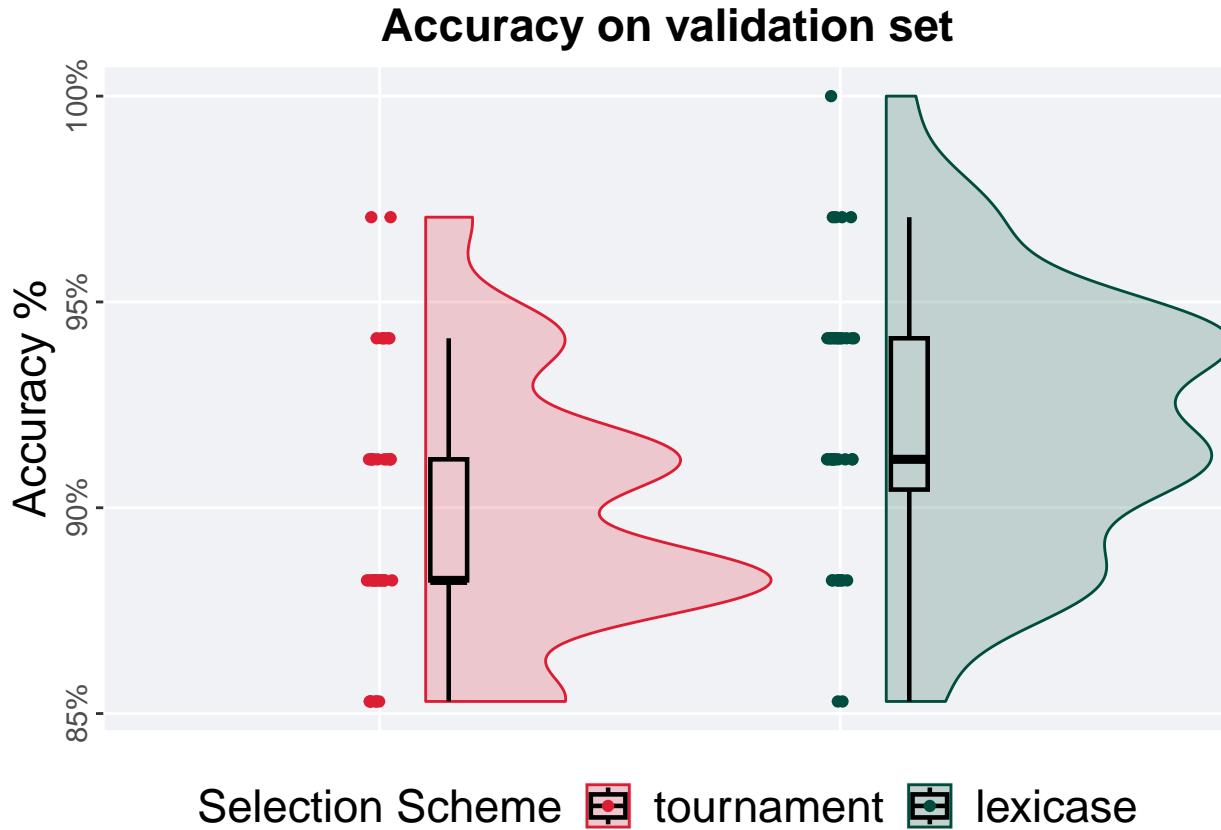
```
## [1] "observed_diff: 1.32499897361024"
## [1] "lower: -1.97890029487314"
## [1] "upper: 1.97890051754304"
## [1] "fail to reject null hypothesis"
## [1] "p-value: 0.19871"
```

Permutation Test: Frequency of T-test Statistic Differences



5.1.2 Validation accuracy

```
validation_plot(filter(task_data, split == '5%'))
```



Summary statistics for the testing performance of the selection schemes at the 5% selection set split:

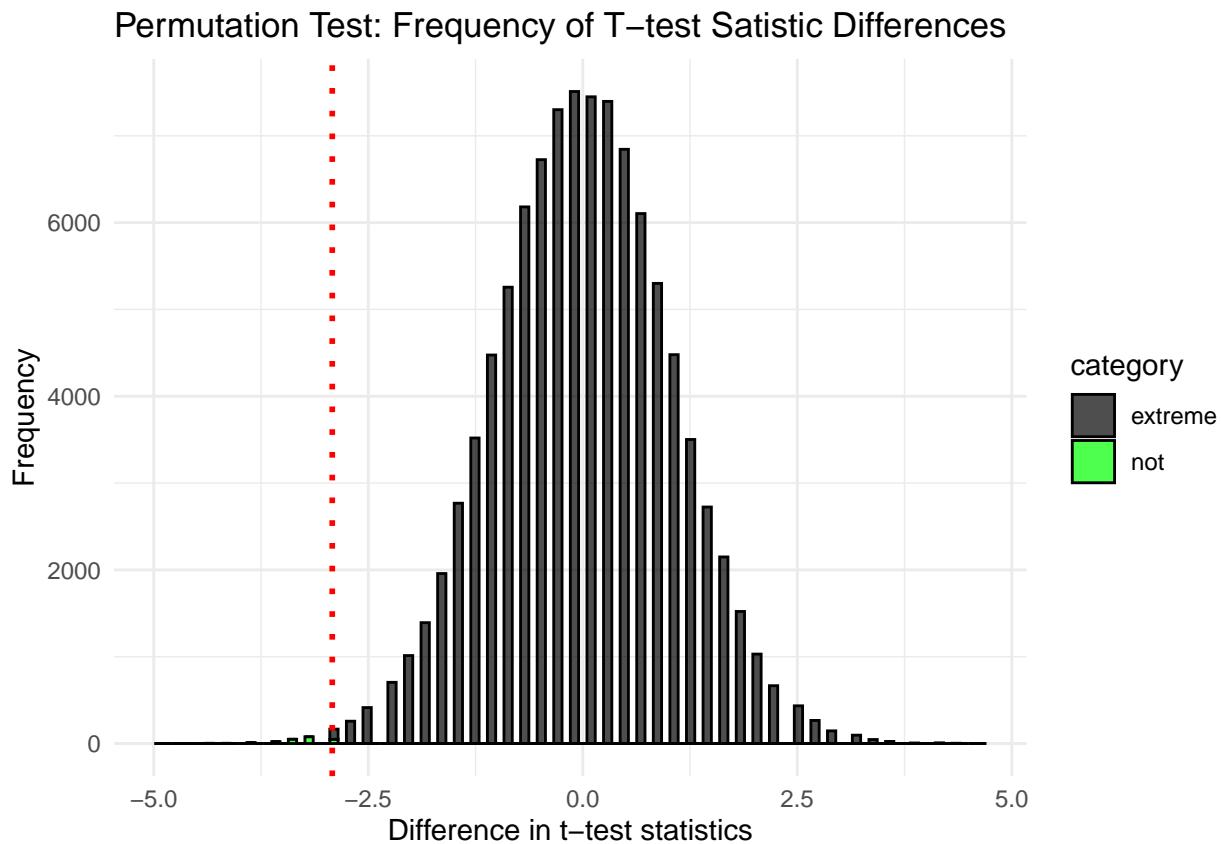
```
validation_accuracy_summary(filter(task_data, split == '5%'))
```

```
## # A tibble: 2 x 8
##   selection count na_cnt  min median  mean   max     IQR
##   <fct>     <int> <int> <dbl> <dbl> <dbl> <dbl> <dbl>
## 1 tournament     40     0 0.853 0.882 0.899 0.971 0.0294
## 2 lexicase       40     0 0.853 0.912 0.921 1       0.0368
```

The permutation test revealed that the results are:

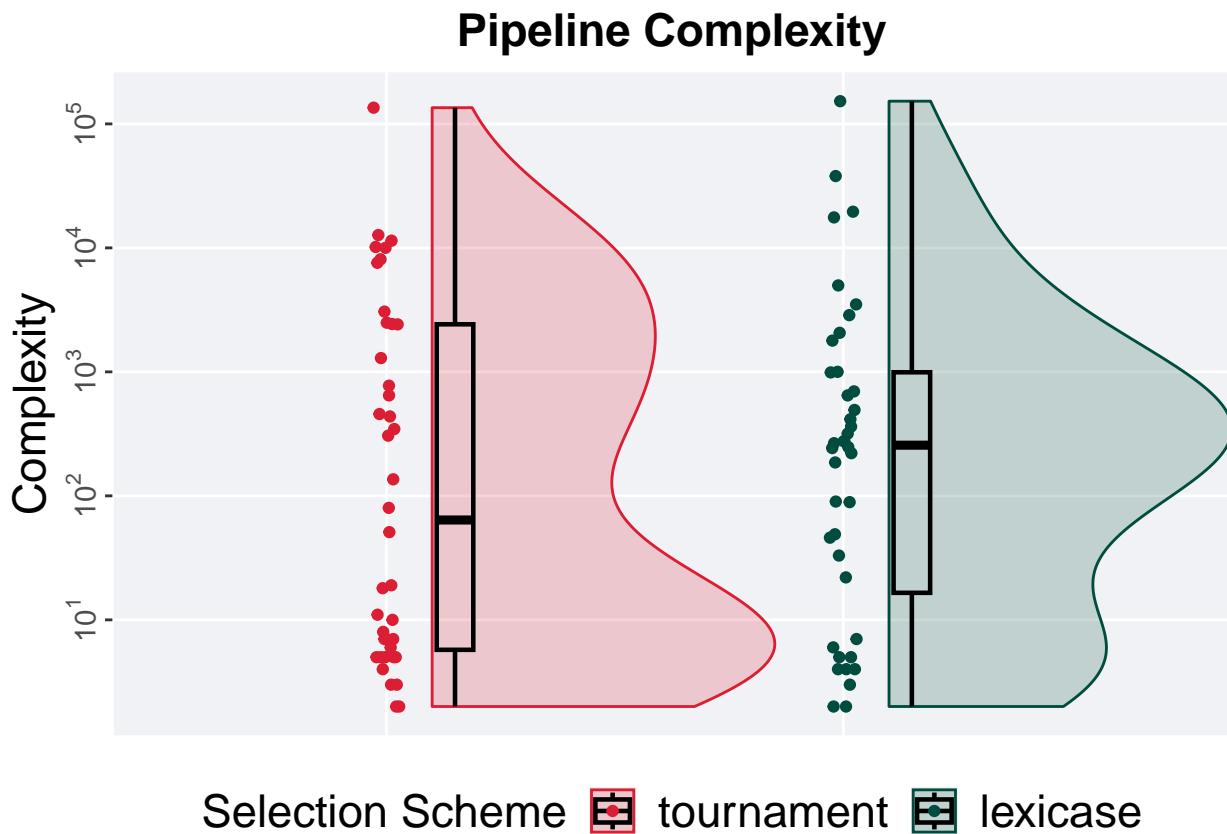
```
tournament_results <- filter(task_data, split == '5%' & selection == 'tournament')
lexicase_results <- filter(task_data, split == '5%' & selection == 'lexicase')
permutation_test(tournament_results$training_performance,
                 lexicase_results$training_performance,
                 seed = 32,
                 alternative = "1")
```

```
## [1] "observed_diff: -2.91990855742165"
## [1] "permutation_diffs[0.05 * n_permutations]: -1.65338713426272"
## [1] "reject null hypothesis"
## [1] "p-value: 0.00214"
```



5.1.3 Pipeline complexity

```
complexity_plot(filter(task_data, split == '5%'))
```



Summary statistics for the pipeline complexity of the selection schemes at the 5% selection set split:

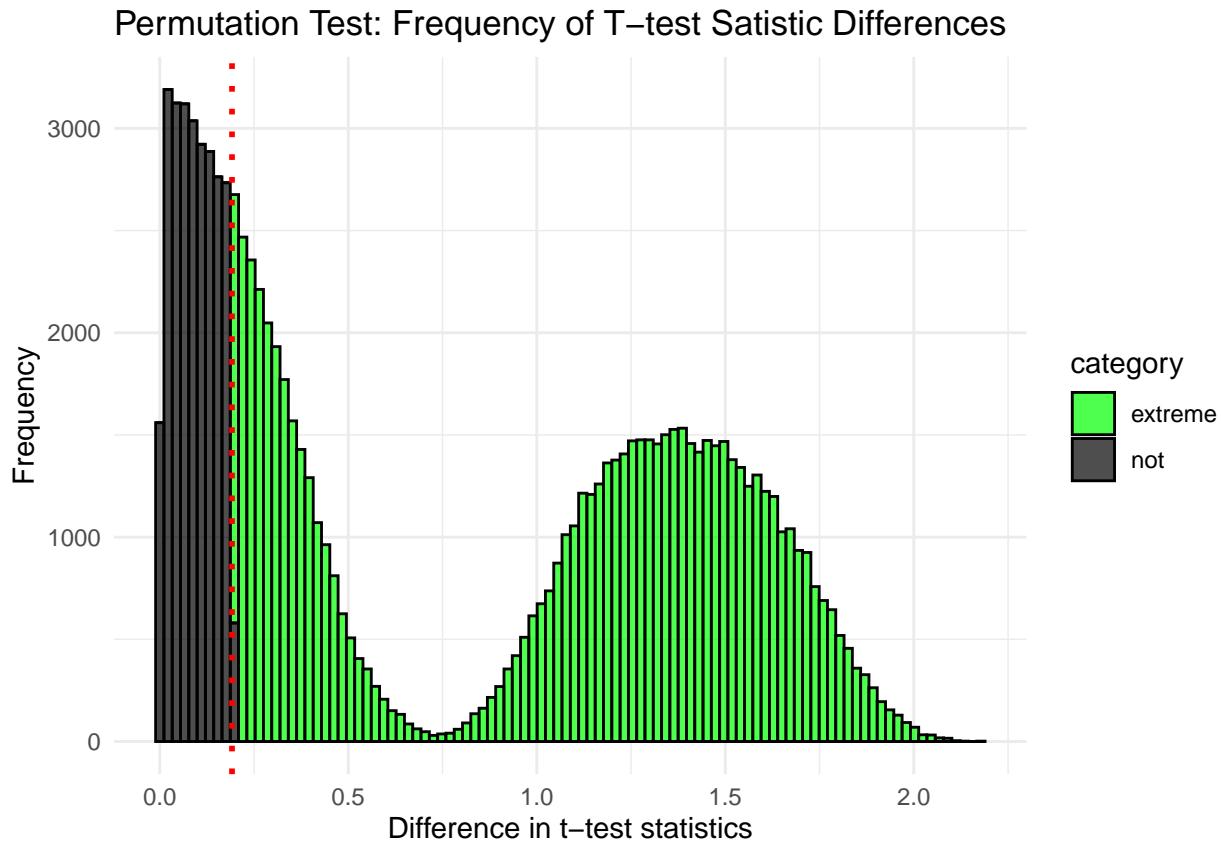
```
complexity_summary(filter(task_data, split == '5%'))
```

```
## # A tibble: 2 x 8
##   selection count na_cnt  min median  mean     max   IQR
##   <fct>     <int> <int> <dbl> <dbl> <dbl> <int> <dbl>
## 1 tournament     40     0     2  65.5  5254. 135131 2414
## 2 lexicase       40     0     2 256.   6245. 152431  974.
```

The permutation test revealed that the results are:

```
tournament_results <- filter(task_data, split == '5%' & selection == 'tournament')
lexicase_results <- filter(task_data, split == '5%' & selection == 'lexicase')
permutation_test(tournament_results$testing_complexity,
                 lexicase_results$testing_complexity,
                 seed = 210,
                 alternative = "t")

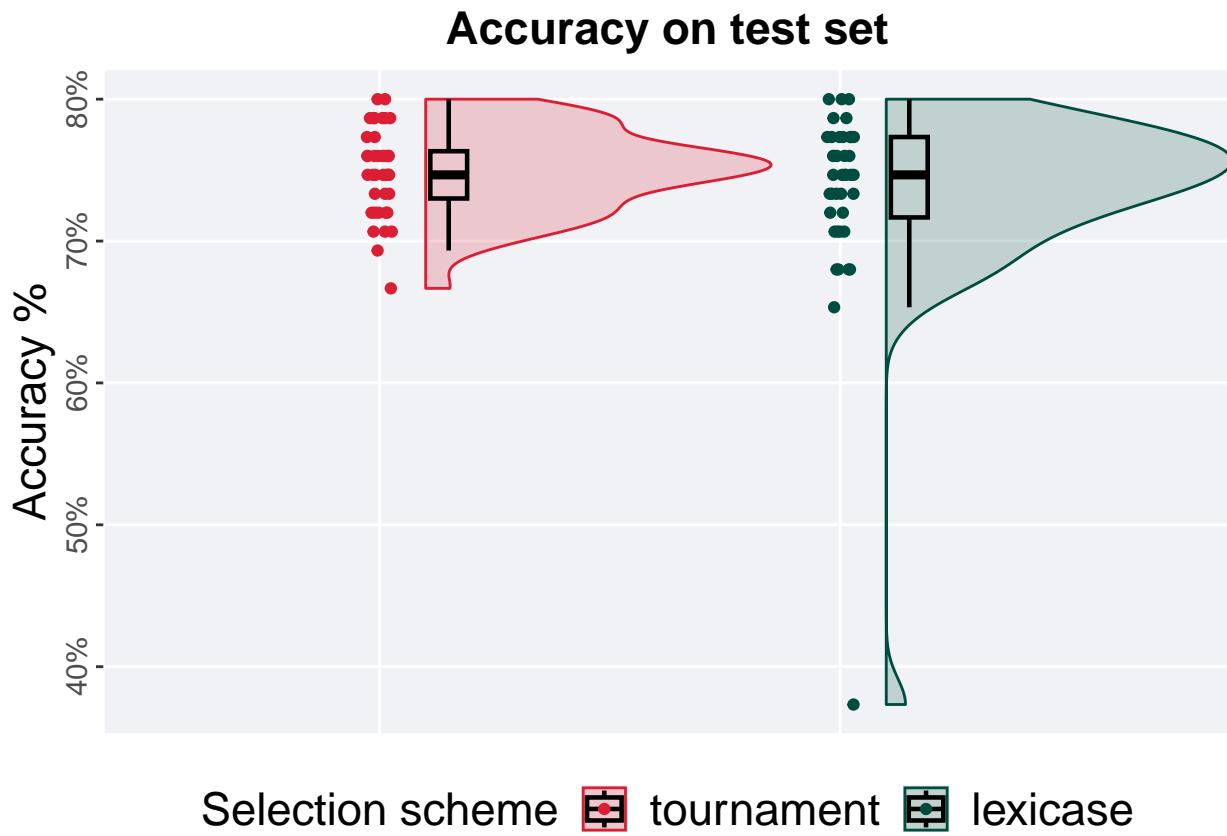
## [1] "observed_diff: -0.191602091268189"
## [1] "lower: -1.72058709434436"
## [1] "upper: 1.72169172866337"
## [1] "fail to reject null hypothesis"
## [1] "p-value: 0.74083"
```



5.2 10%

5.2.1 Test accuracy

```
test_plot(filter(task_data, split == '10%'))
```



Summary statistics for the testing performance of the selection schemes at the 5% selection set split:

```
test_results_summary(filter(task_data, split == '10%'))
```

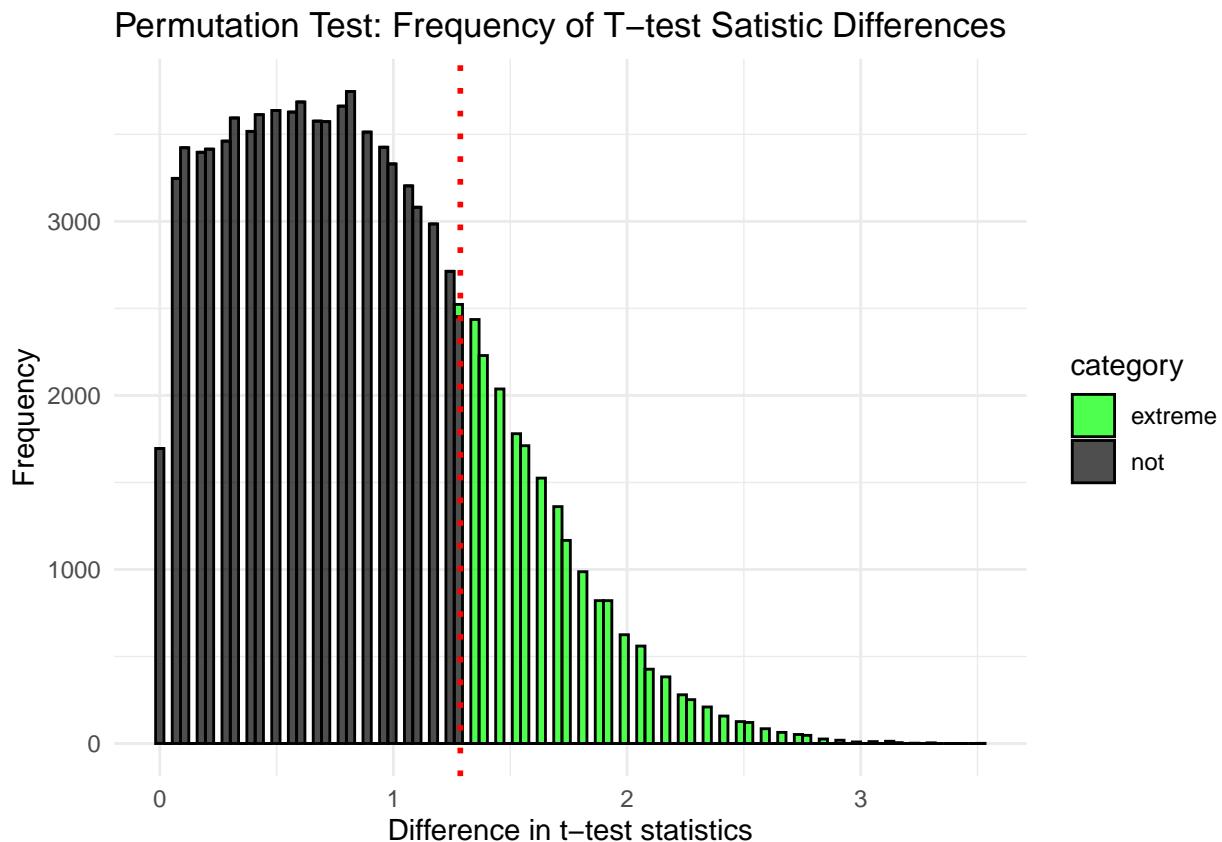
```
## # A tibble: 2 x 8
##   selection count na_cnt   min median   mean   max     IQR
##   <fct>     <int> <int> <dbl> <dbl> <dbl> <dbl> <dbl>
## 1 tournament     40     0 0.667  0.747  0.749   0.8  0.0333
## 2 lexicase       40     0 0.373  0.747  0.734   0.8  0.0567
```

The permutation test revealed that the results are:

```
tournament_results <- filter(task_data, split == '10%' & selection == 'tournament')
lexicase_results <- filter(task_data, split == '10%' & selection == 'lexicase')
```

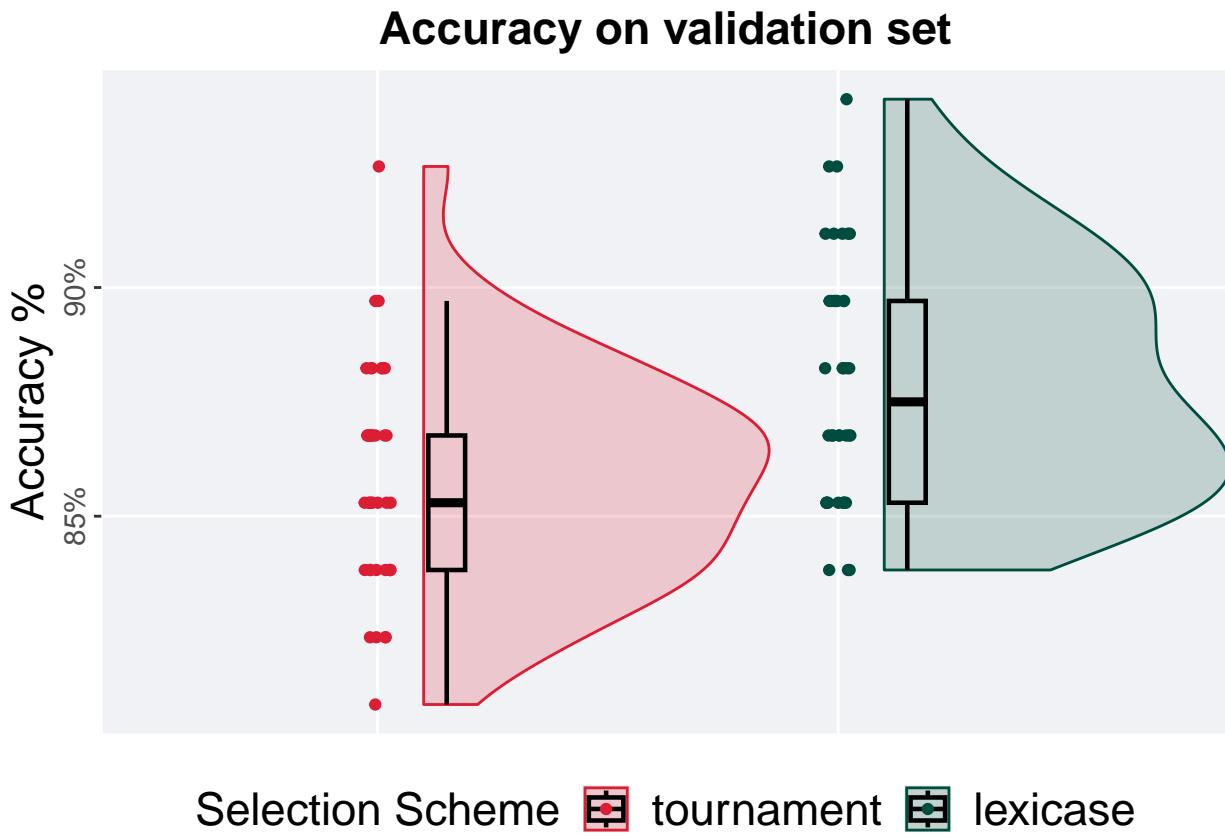
```
permutation_test(tournament_results$testing_performance,
                  lexicase_results$testing_performance,
                  seed = 33,
                  alternative = "t")
```

```
## [1] "observed_diff: 1.28638411846571"
## [1] "lower: -1.86676173626107"
## [1] "upper: 1.80778536386294"
## [1] "fail to reject null hypothesis"
## [1] "p-value: 0.20426"
```



5.2.2 Validation accuracy

```
validation_plot(filter(task_data, split == '10%'))
```



Summary statistics for the testing performance of the selection schemes at the 5% selection set split:

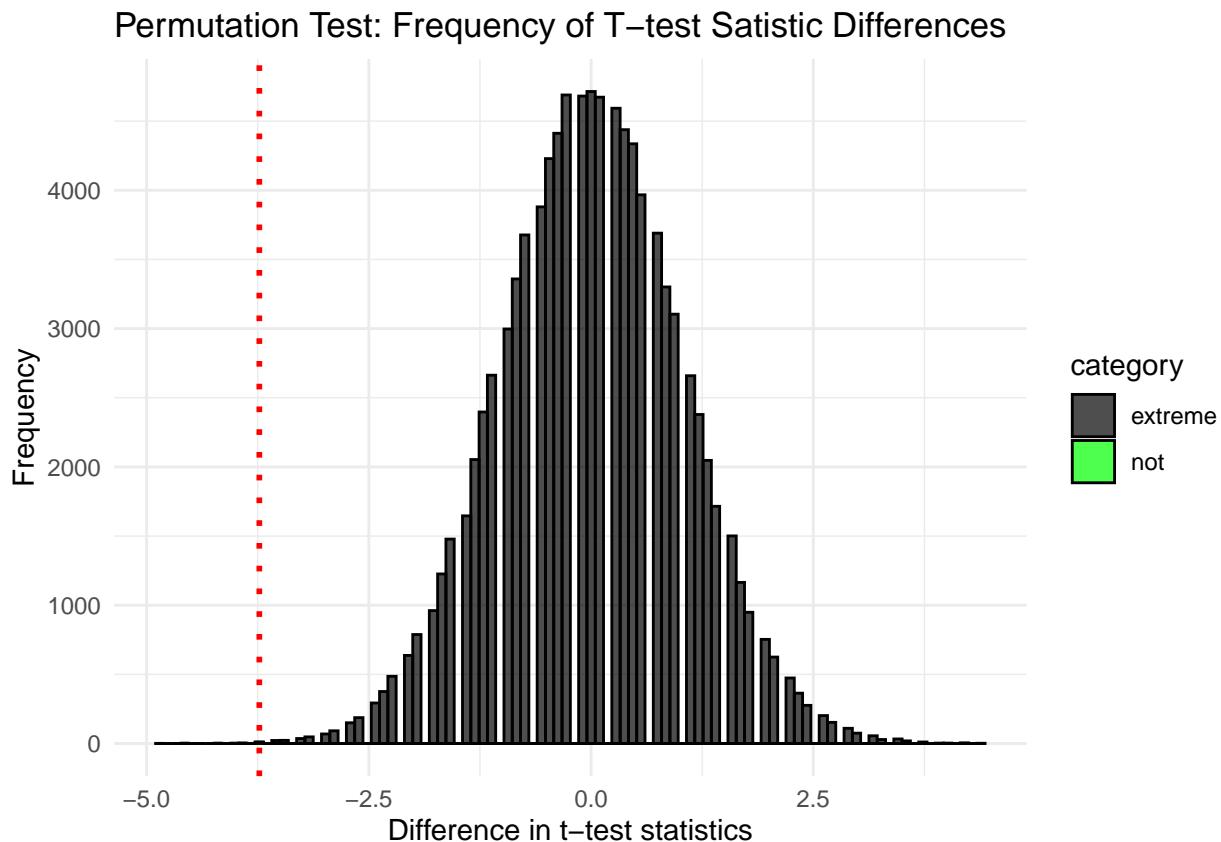
```
validation_accuracy_summary(filter(task_data, split == '10%'))
```

```
## # A tibble: 2 x 8
##   selection  count  na_cnt  min  median  mean  max    IQR
##   <fct>      <int>  <int>  <dbl>  <dbl>  <dbl> <dbl>  <dbl>
## 1 tournament    40      0  0.809  0.853  0.858  0.926  0.0294
## 2 lexicase      40      0  0.838  0.875  0.879  0.941  0.0441
```

The permutation test revealed that the results are:

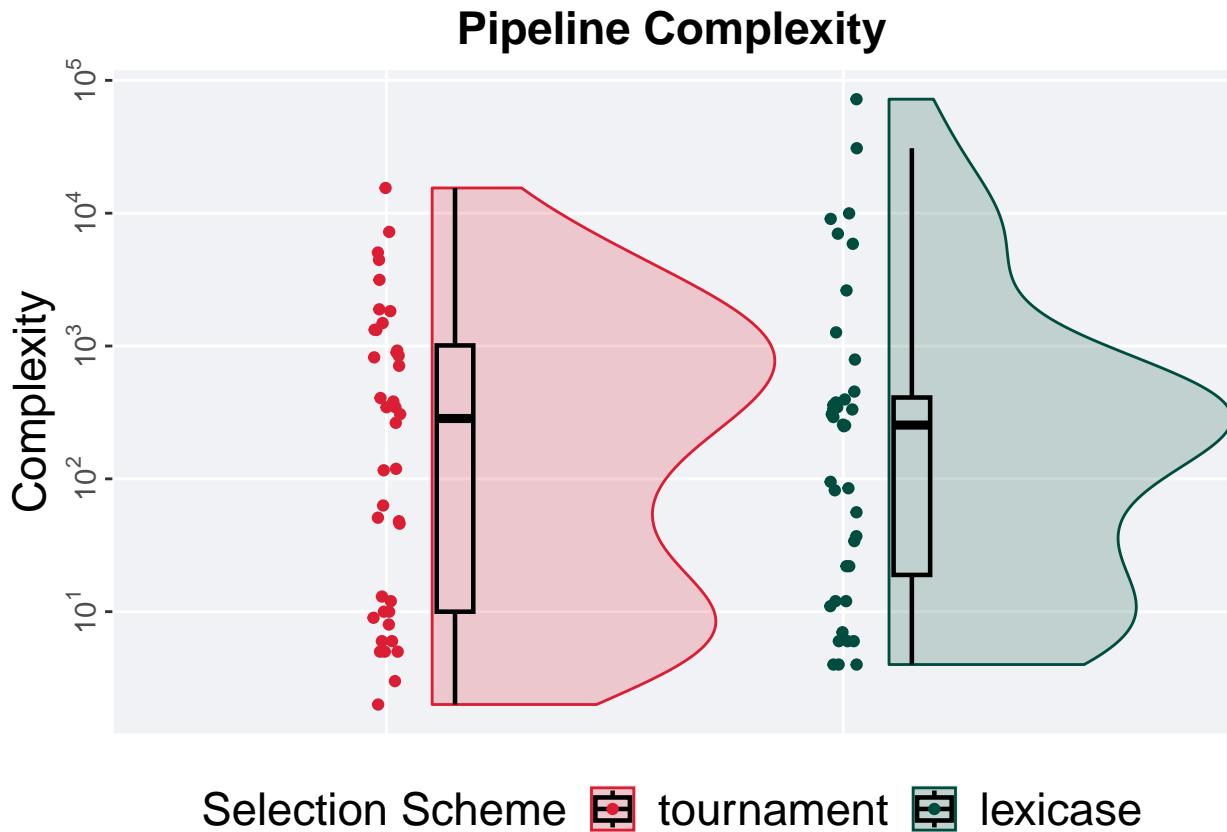
```
tournament_results <- filter(task_data, split == '10%' & selection == 'tournament')
lexicase_results <- filter(task_data, split == '10%' & selection == 'lexicase')
permutation_test(tournament_results$training_performance,
                 lexicase_results$training_performance,
                 seed = 34,
                 alternative = "1")
```

```
## [1] "observed_diff: -3.7319412515713"
## [1] "permutation_diffs[0.05 * n_permutations]: -1.68964535023223"
## [1] "reject null hypothesis"
## [1] "p-value: 0.00019"
```



5.2.3 Pipeline complexity

```
complexity_plot(filter(task_data, split == '10%'))
```



Summary statistics for the pipeline complexity of the selection schemes at the 5% selection set split:

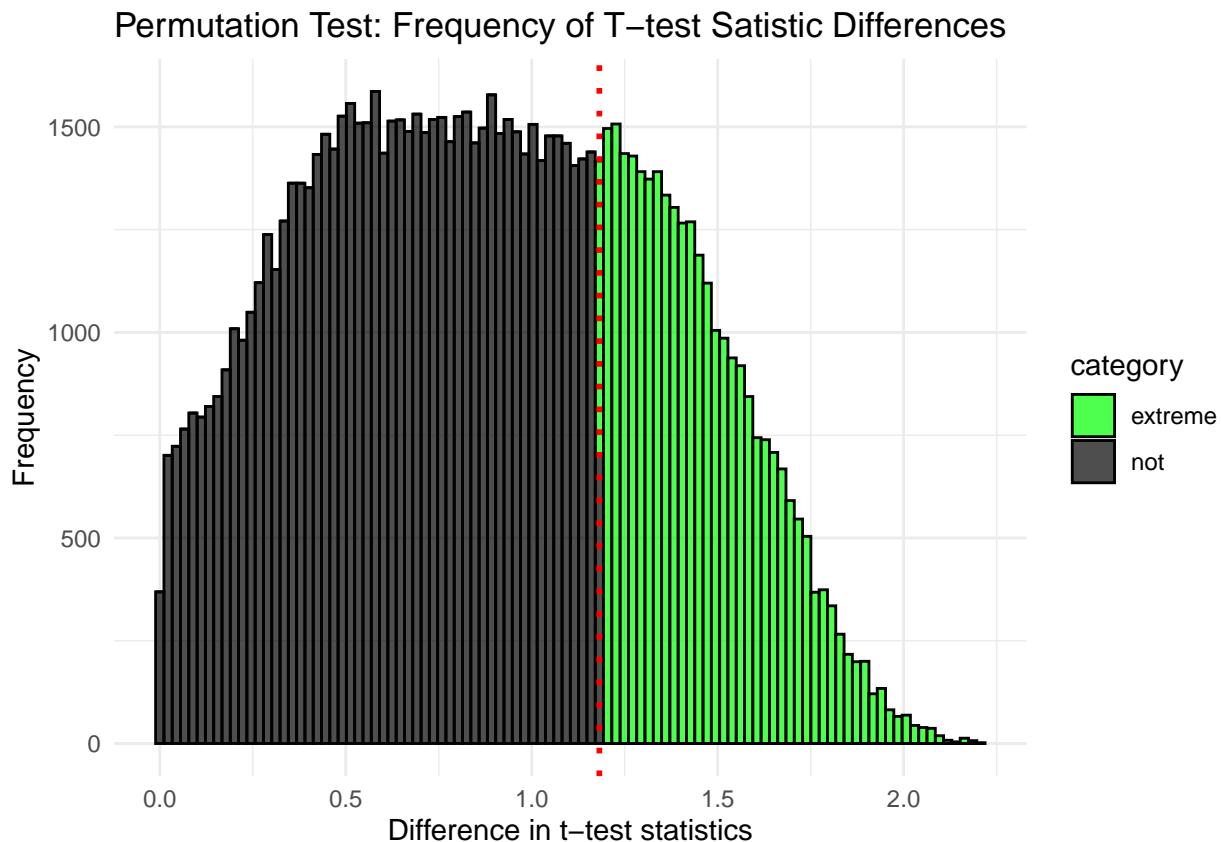
```
complexity_summary(filter(task_data, split == '10%'))
```

```
## # A tibble: 2 x 8
##   selection count na_cnt  min median  mean   max   IQR
##   <fct>     <int> <int> <dbl> <dbl> <dbl> <dbl>
## 1 tournament     40     0     2  286. 1251. 15491 1013.
## 2 lexicase       40     0     4   254  3610. 72171  391
```

The permutation test revealed that the results are:

```
tournament_results <- filter(task_data, split == '10%' & selection == 'tournament')
lexicase_results <- filter(task_data, split == '10%' & selection == 'lexicase')
permutation_test(tournament_results$testing_complexity,
                 lexicase_results$testing_complexity,
                 seed = 211,
                 alternative = "t")
```

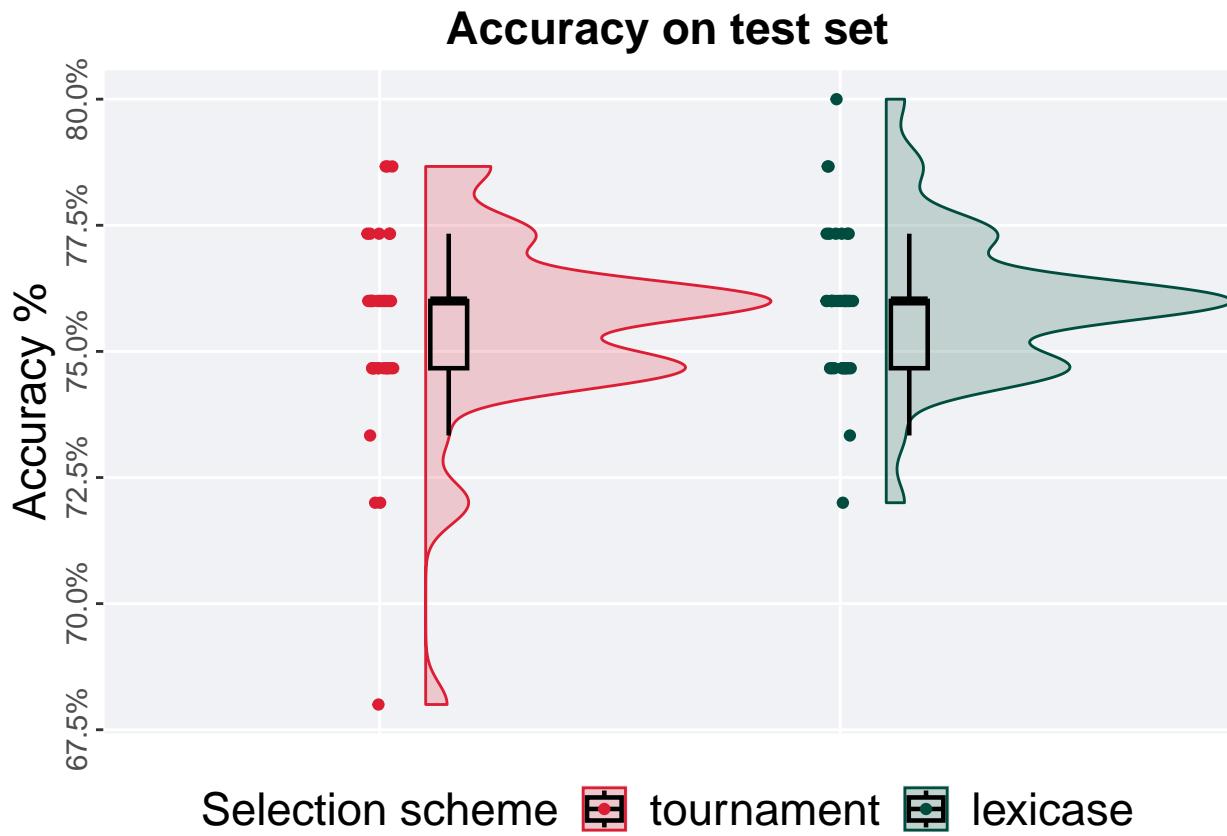
```
## [1] "observed_diff: -1.18178297989988"
## [1] "lower: -1.65760654953964"
## [1] "upper: 1.65865255679243"
## [1] "fail to reject null hypothesis"
## [1] "p-value: 0.30012"
```



5.3 50%

5.3.1 Test accuracy

```
test_plot(filter(task_data, split == '50%'))
```



Summary statistics for the testing performance of the selection schemes at the 5% selection set split:

```
test_results_summary(filter(task_data, split == '50%'))
```

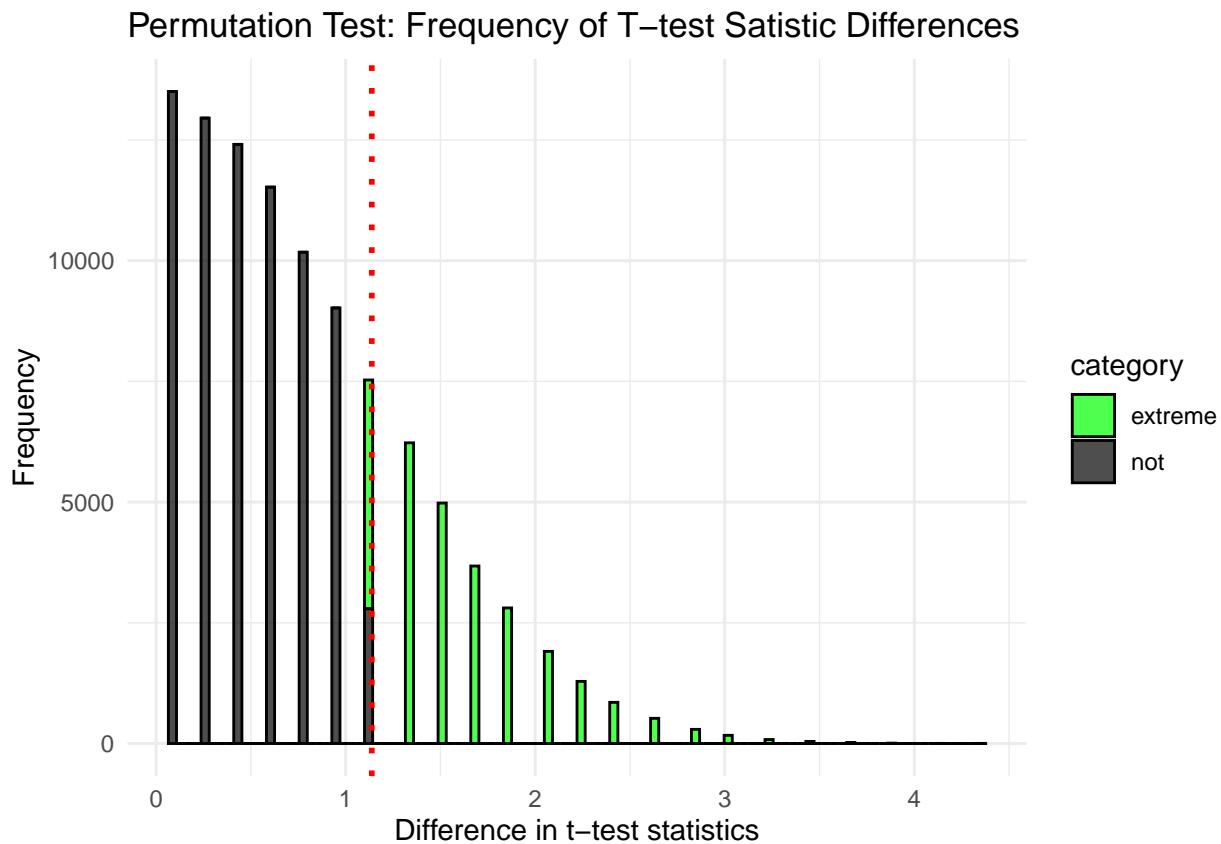
```
## # A tibble: 2 x 8
##   selection count na_cnt  min median  mean   max     IQR
##   <fct>     <int> <int> <dbl>  <dbl> <dbl> <dbl> <dbl>
## 1 tournament     40     0  0.68  0.76 0.755 0.787 0.0133
## 2 lexicase       40     0  0.72  0.76 0.759 0.8   0.0133
```

The permutation test revealed that the results are:

```
tournament_results <- filter(task_data, split == '50%' & selection == 'tournament')
lexicase_results <- filter(task_data, split == '50%' & selection == 'lexicase')
```

```
permutation_test(tournament_results$testing_performance,
                  lexicase_results$testing_performance,
                  seed = 35,
                  alternative = "t")
```

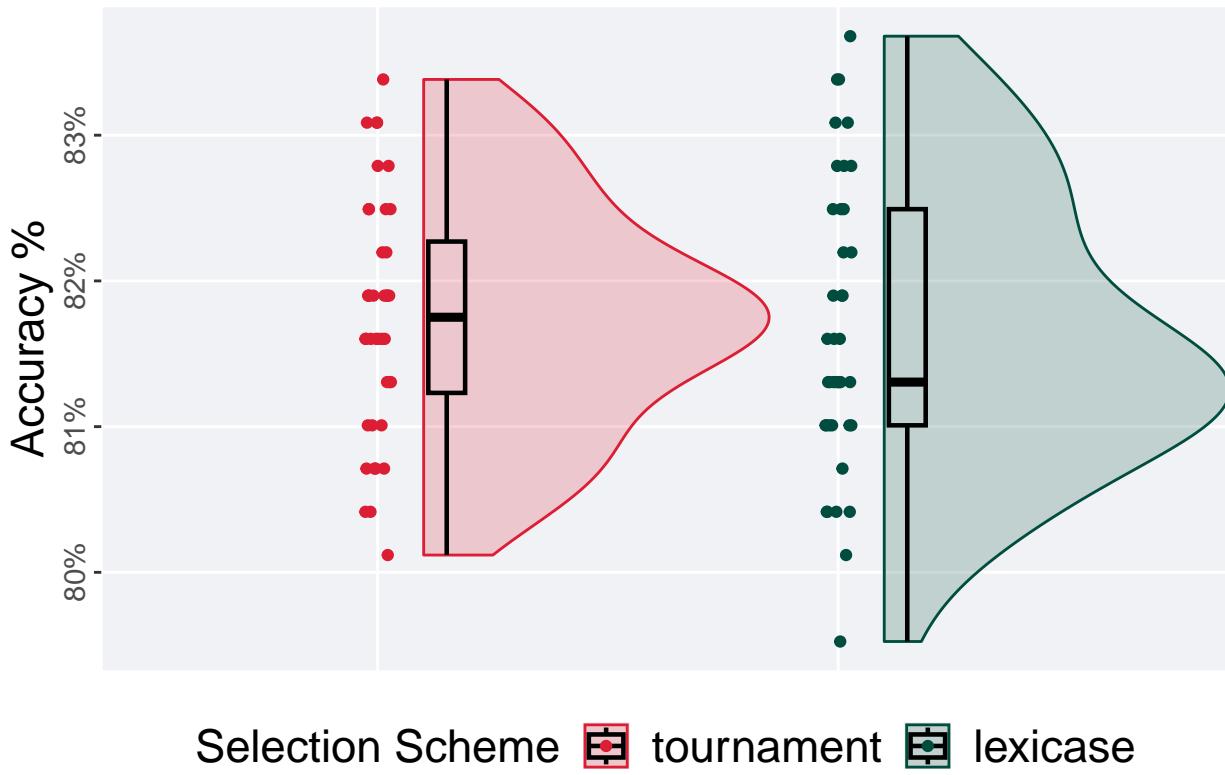
```
## [1] "observed_diff: -1.13782106538951"
## [1] "lower: -2.04962738612793"
## [1] "upper: 2.04962752699874"
## [1] "fail to reject null hypothesis"
## [1] "p-value: 0.27624"
```



5.3.2 Validation accuracy

```
validation_plot(filter(task_data, split == '50%'))
```

Accuracy on validation set



Summary statistics for the testing performance of the selection schemes at the 5% selection set split:

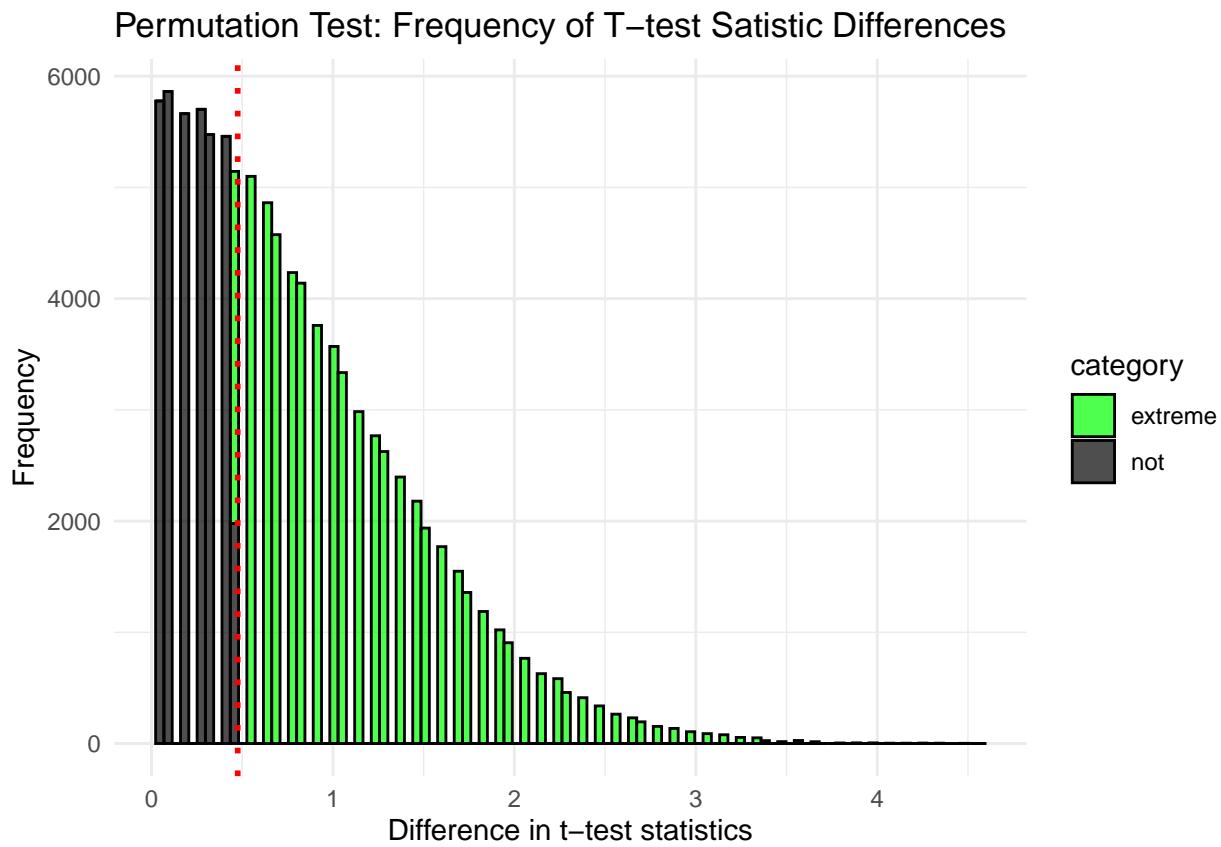
```
validation_accuracy_summary(filter(task_data, split == '50%'))
```

```
## # A tibble: 2 x 8
##   selection count na_cnt  min median  mean   max     IQR
##   <fct>     <int> <int> <dbl> <dbl> <dbl> <dbl> <dbl>
## 1 tournament     40     0 0.801 0.818 0.817 0.834 0.0104
## 2 lexicase       40     0 0.795 0.813 0.816 0.837 0.0148
```

The permutation test revealed that the results are:

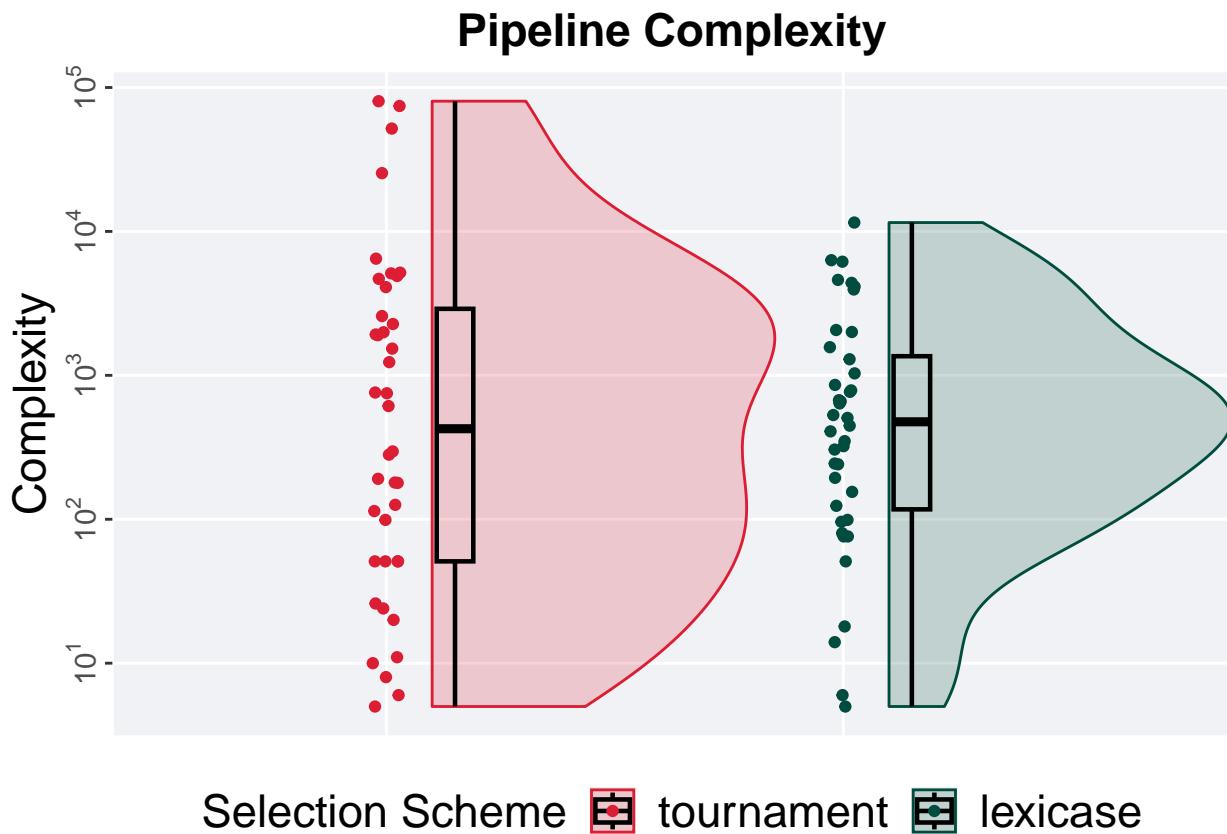
```
tournament_results <- filter(task_data, split == '50%' & selection == 'tournament')
lexicase_results <- filter(task_data, split == '50%' & selection == 'lexicase')
permutation_test(tournament_results$training_performance,
                 lexicase_results$training_performance,
                 seed = 36,
                 alternative = "t")

## [1] "observed_diff: 0.475399664870139"
## [1] "lower: -1.9835828407878"
## [1] "upper: 1.98358257581197"
## [1] "fail to reject null hypothesis"
## [1] "p-value: 0.64083"
```



5.3.3 Pipeline complexity

```
complexity_plot(filter(task_data, split == '50%'))
```



Summary statistics for the pipeline complexity of the selection schemes at the 5% selection set split:

```
complexity_summary(filter(task_data, split == '50%'))
```

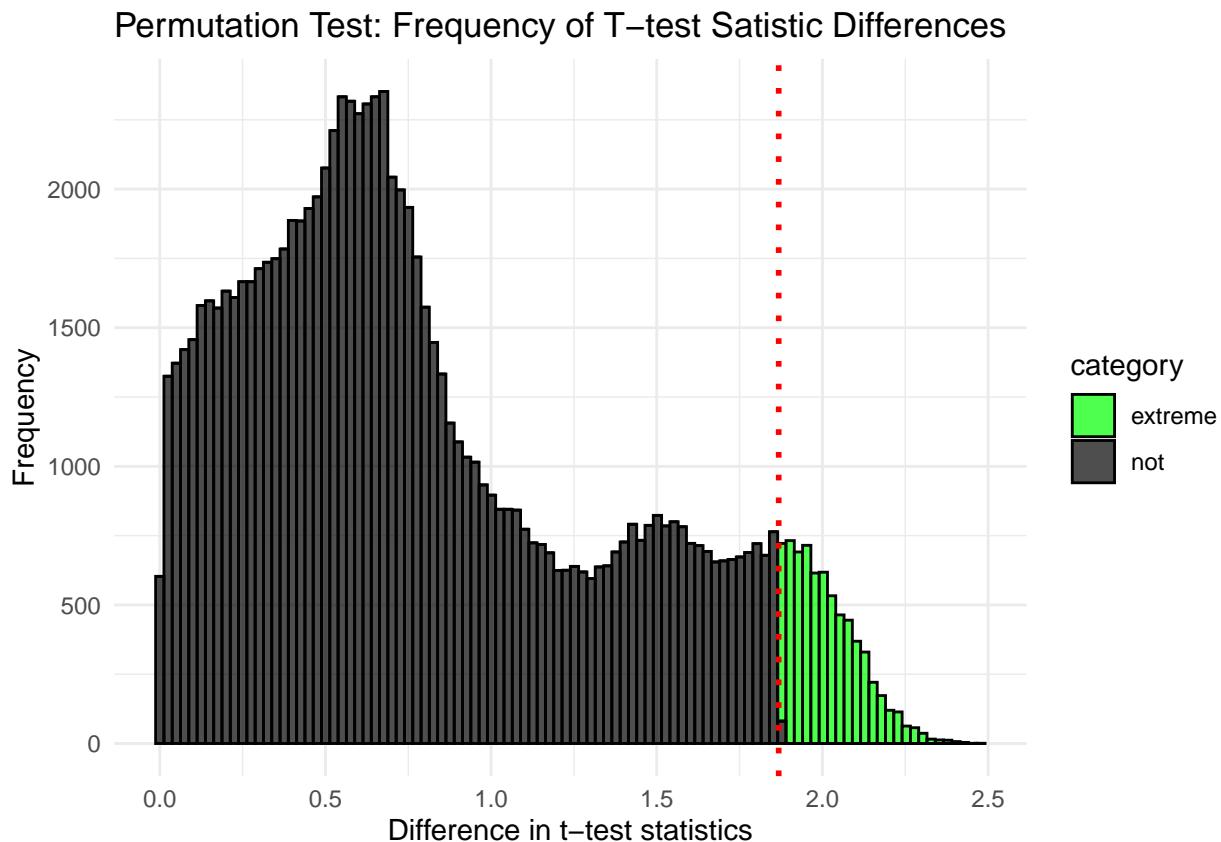
```
## # A tibble: 2 x 8
##   selection count na_cnt  min median  mean   max   IQR
##   <fct>     <int> <int> <int> <dbl> <dbl> <int> <dbl>
## 1 tournament     40     0     5    454   6993. 80322 2911.
## 2 lexicase       40     0     5   476.  1444. 11525 1245
```

The permutation test revealed that the results are:

```
tournament_results <- filter(task_data, split == '50%' & selection == 'tournament')
lexicase_results <- filter(task_data, split == '50%' & selection == 'lexicase')
```

```
permutation_test(tournament_results$testing_complexity,
                  lexicase_results$testing_complexity,
                  seed = 212,
                  alternative = "t")
```

```
## [1] "observed_diff: 1.86796164047676"
## [1] "lower: -1.93813478323364"
## [1] "upper: 1.93726895112346"
## [1] "fail to reject null hypothesis"
## [1] "p-value: 0.06992"
```

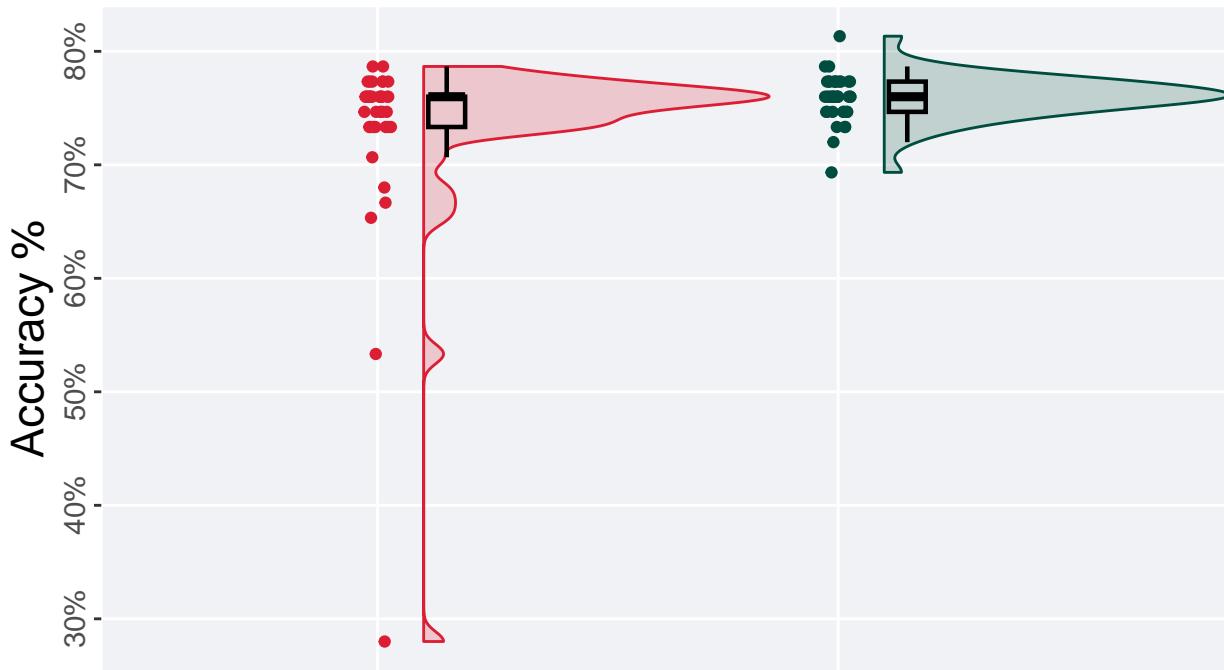


5.4 90%

5.4.1 Test accuracy

```
test_plot(filter(task_data, split == '90%'))
```

Accuracy on test set



Selection scheme tournament lexicase

Summary statistics for the testing performance of the selection schemes at the 5% selection set split:

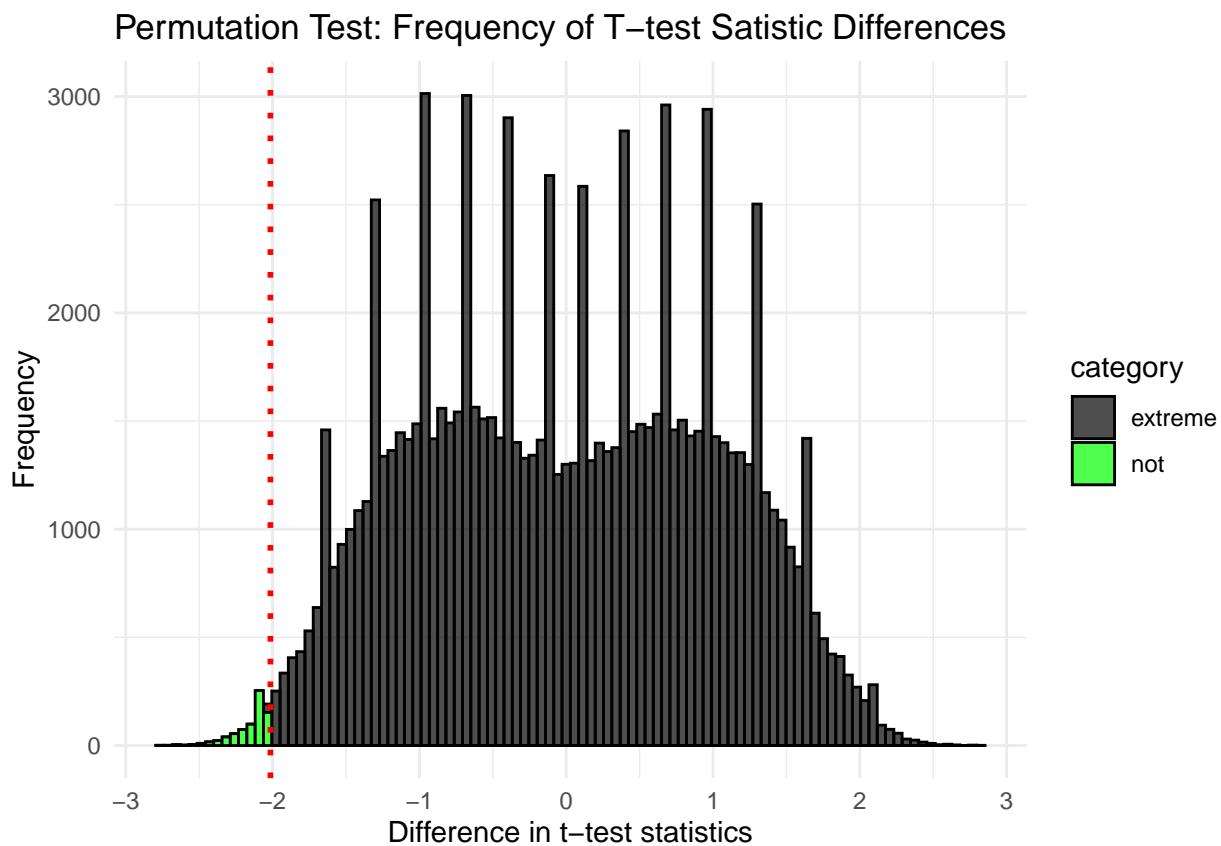
```
test_results_summary(filter(task_data, split == '90%'))
```

```
## # A tibble: 2 x 8
##   selection  count  na_cnt  min  median  mean   max    IQR
##   <fct>      <int>   <int>  <dbl>   <dbl>  <dbl> <dbl>   <dbl>
## 1 tournament     40      0  0.28    0.76  0.731  0.787  0.0267
## 2 lexicase       40      0  0.693   0.76  0.759  0.813  0.0267
```

The permutation test revealed that the results are:

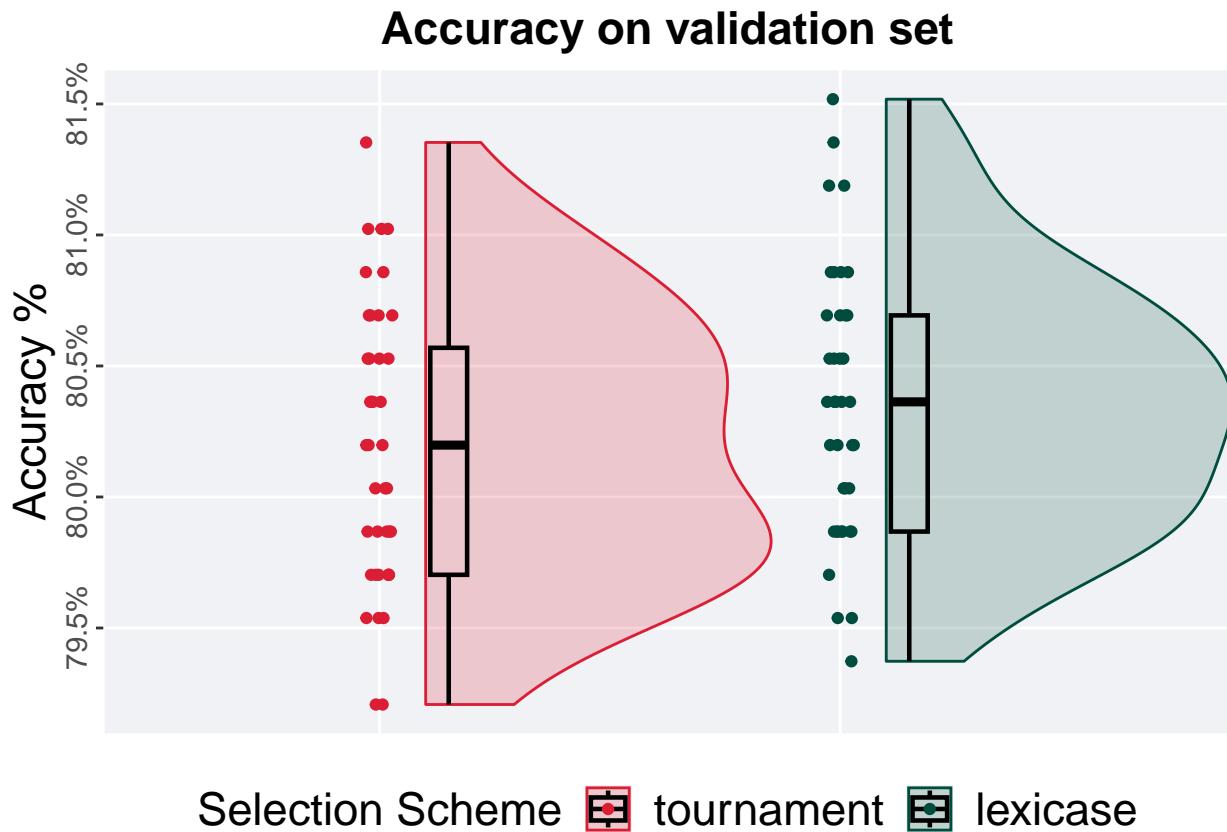
```
tournament_results <- filter(task_data, split == '90%' & selection == 'tournament')
lexicase_results <- filter(task_data, split == '90%' & selection == 'lexicase')
permutation_test(tournament_results$testing_performance,
                 lexicase_results$testing_performance,
                 seed = 37,
                 alternative = "1")
```

```
## [1] "observed_diff: -2.01490383284339"
## [1] "permutation_diffs[0.05 * n_permutations]: -1.56760535956015"
## [1] "reject null hypothesis"
## [1] "p-value: 0.00738"
```



5.4.2 Validation accuracy

```
validation_plot(filter(task_data, split == '90%'))
```



Summary statistics for the testing performance of the selection schemes at the 5% selection set split:

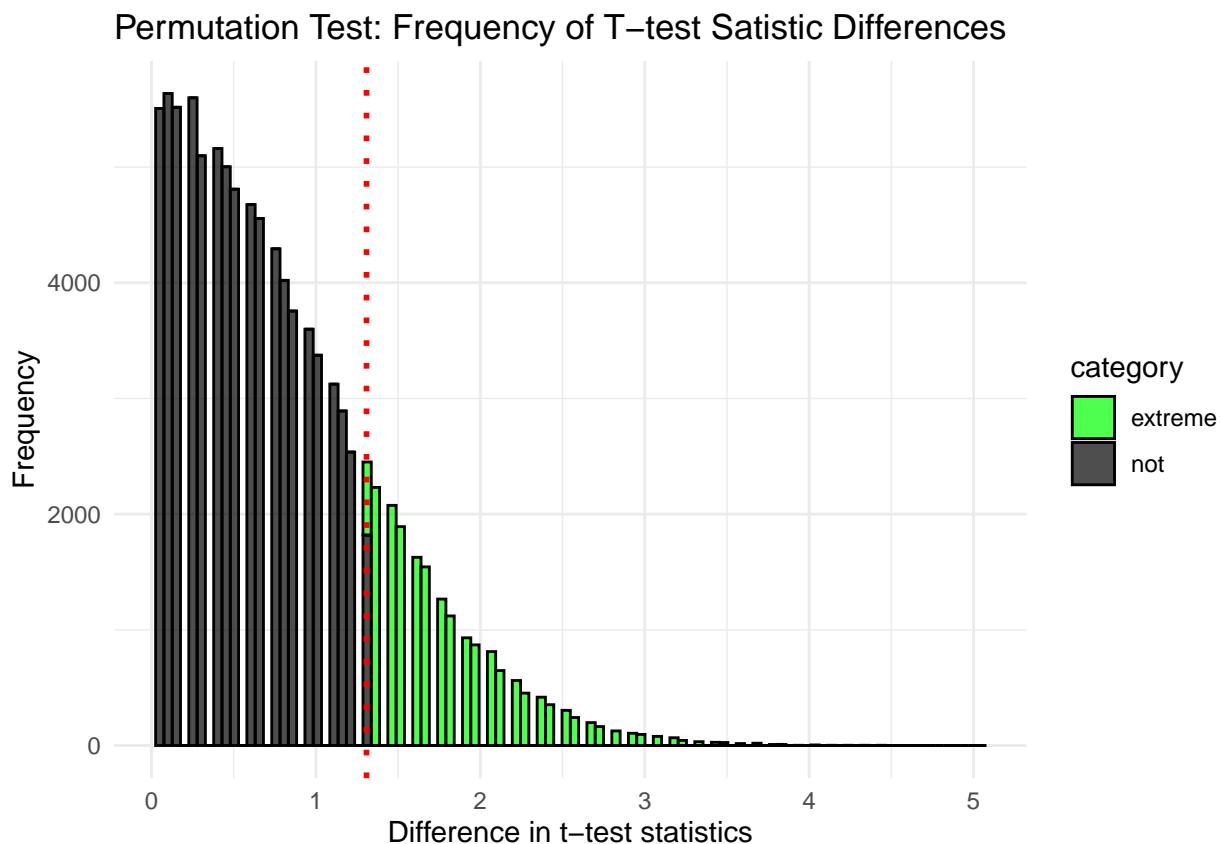
```
validation_accuracy_summary(filter(task_data, split == '90%'))
```

```
## # A tibble: 2 x 8
##   selection count na_cnt  min median  mean   max      IQR
##   <fct>     <int> <int> <dbl> <dbl> <dbl> <dbl> <dbl>
## 1 tournament     40     0 0.792 0.802 0.802 0.814 0.00866
## 2 lexicase       40     0 0.794 0.804 0.803 0.815 0.00825
```

The permutation test revealed that the results are:

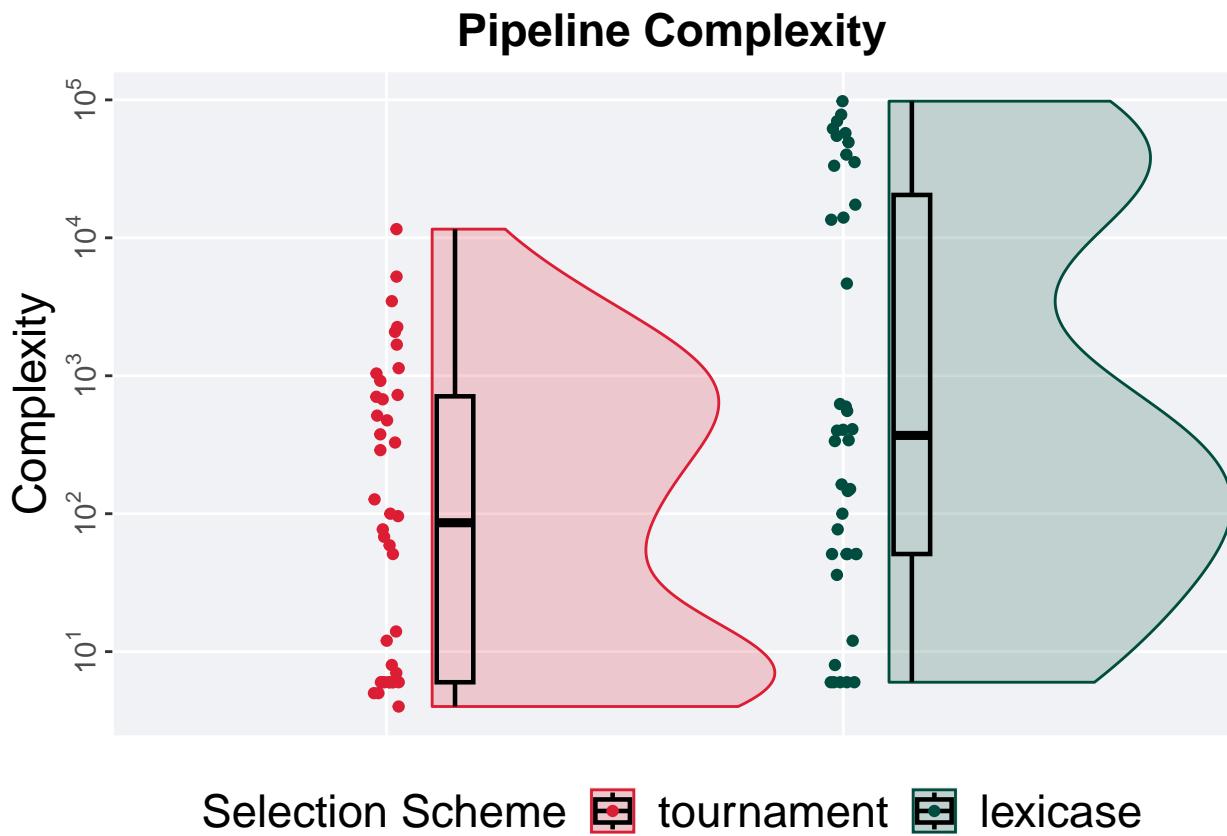
```
tournament_results <- filter(task_data, split == '90%' & selection == 'tournament')
lexicase_results <- filter(task_data, split == '90%' & selection == 'lexicase')
permutation_test(tournament_results$training_performance,
                 lexicase_results$training_performance,
                 seed = 38,
                 alternative = "t")
```

```
## [1] "observed_diff: -1.30766861645993"
## [1] "lower: -1.97012862430494"
## [1] "upper: 1.97013135879071"
## [1] "fail to reject null hypothesis"
## [1] "p-value: 0.19023"
```



5.4.3 Pipeline complexity

```
complexity_plot(filter(task_data, split == '90%'))
```



Summary statistics for the pipeline complexity of the selection schemes at the 5% selection set split:

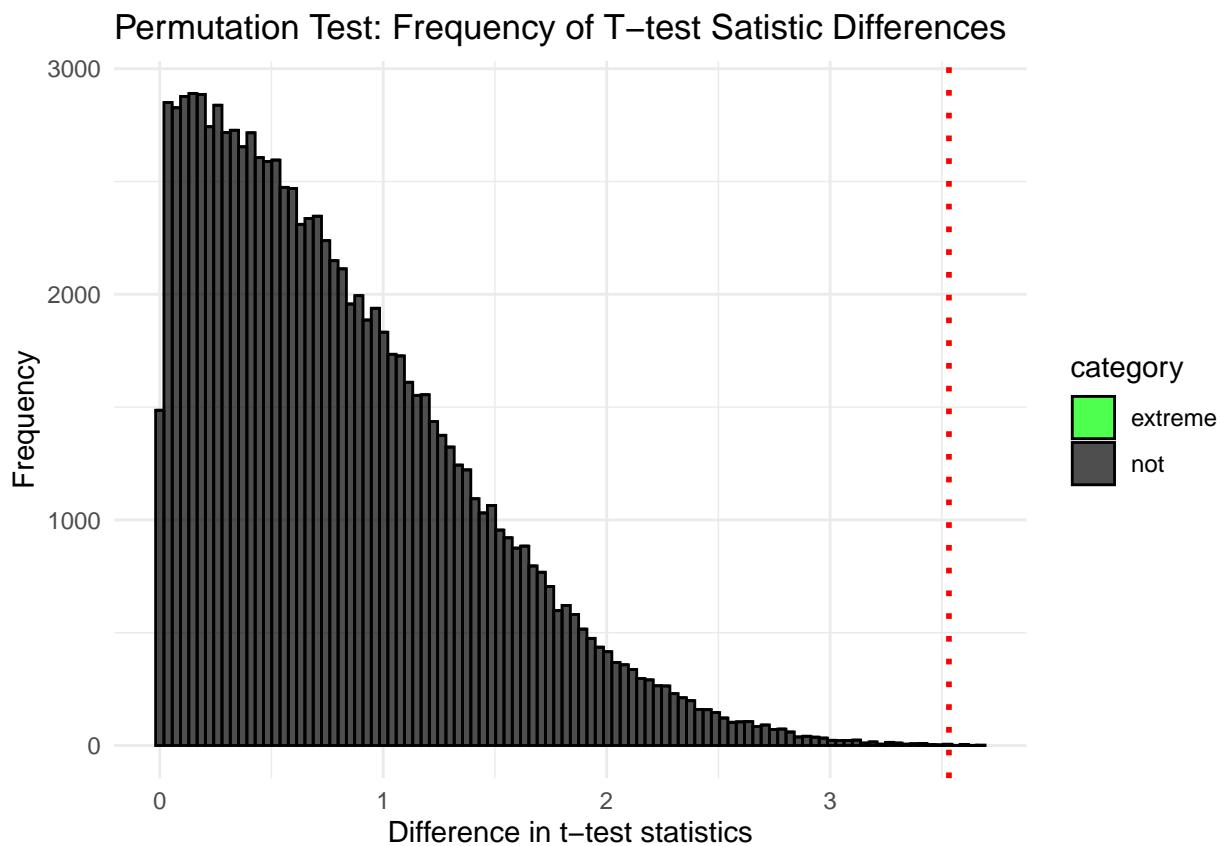
```
complexity_summary(filter(task_data, split == '90%'))
```

```
## # A tibble: 2 x 8
##   selection count na_cnt  min median   mean   max   IQR
##   <fct>     <int> <int> <dbl> <dbl> <dbl> <int> <dbl>
## 1 tournament     40     0     4  86.5  854. 11561   704.
## 2 lexicase       40     0     6 370.  15808. 97831 21312.
```

The permutation test revealed that the results are:

```
tournament_results <- filter(task_data, split == '90%' & selection == 'tournament')
lexicase_results <- filter(task_data, split == '90%' & selection == 'lexicase')
permutation_test(tournament_results$testing_complexity,
                 lexicase_results$testing_complexity,
                 seed = 213,
                 alternative = "t")
```

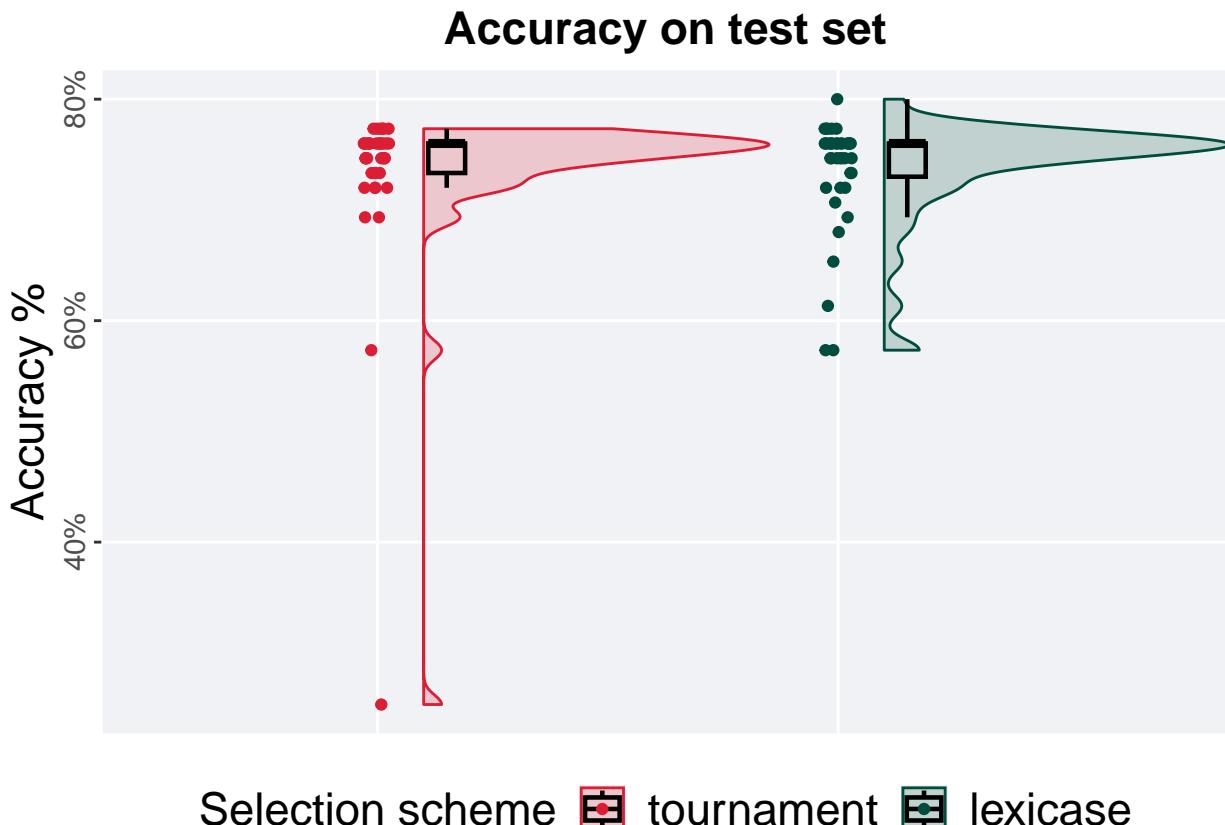
```
## [1] "observed_diff: -3.53172093670732"
## [1] "lower: -1.97015997010439"
## [1] "upper: 1.97825195772778"
## [1] "reject null hypothesis"
## [1] "p-value: 1e-04"
```



5.5 95%

5.5.1 Test accuracy

```
test_plot(filter(task_data, split == '95%'))
```



Summary statistics for the testing performance of the selection schemes at the 5% selection set split:

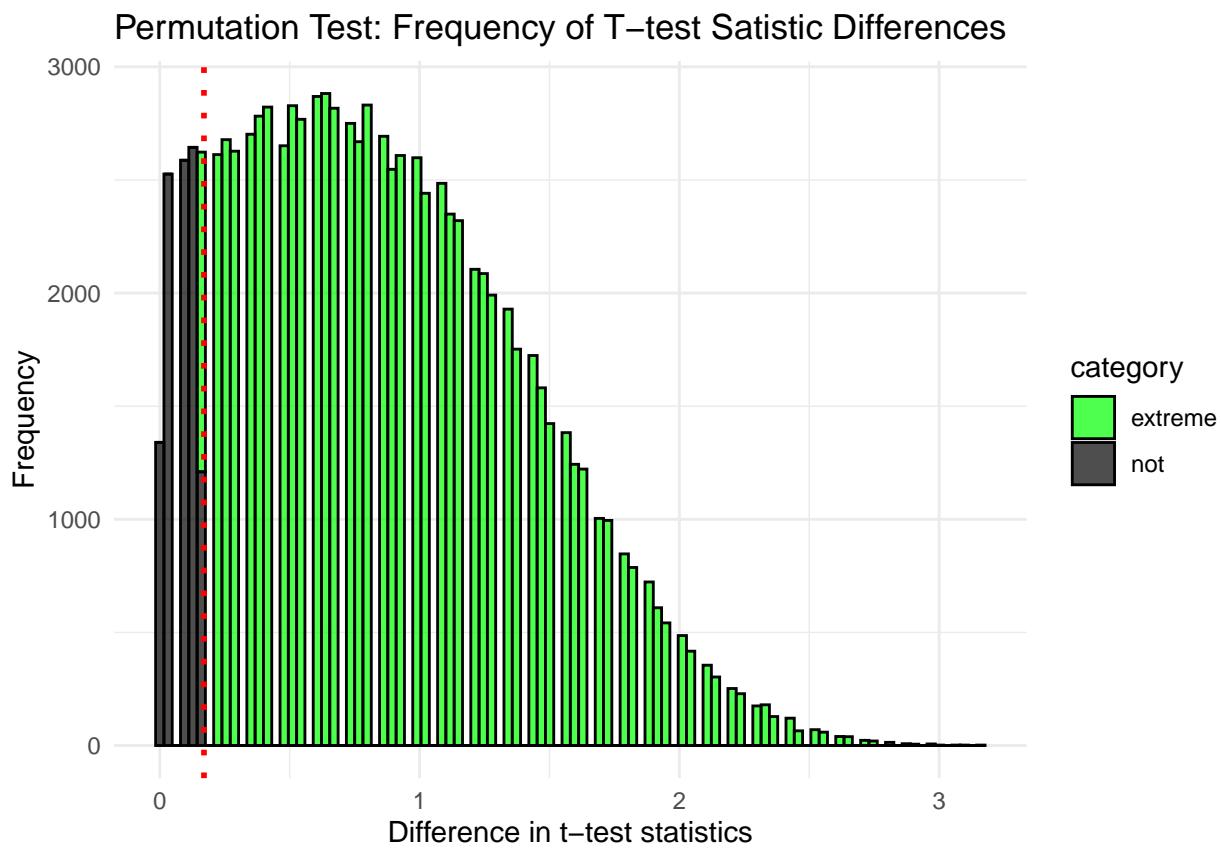
```
test_results_summary(filter(task_data, split == '95%'))
```

```
## # A tibble: 2 x 8
##   selection count na_cnt  min median  mean   max   IQR
##   <fct>     <int> <int> <dbl>  <dbl> <dbl> <dbl> <dbl>
## 1 tournament     40     0 0.253  0.76  0.733 0.773 0.0267
## 2 lexicase       40     0 0.573  0.76  0.735 0.8    0.0300
```

The permutation test revealed that the results are:

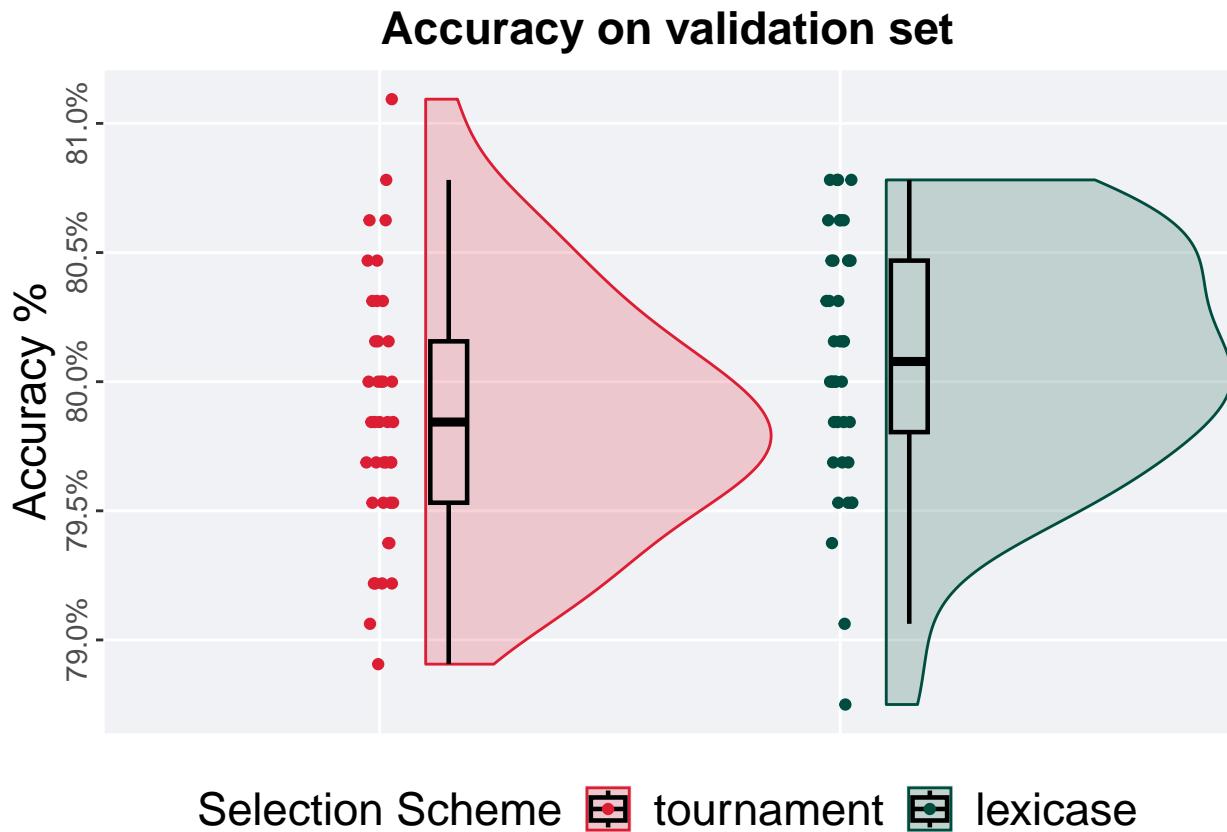
```
tournament_results <- filter(task_data, split == '95%' & selection == 'tournament')
lexicase_results <- filter(task_data, split == '95%' & selection == 'lexicase')
permutation_test(tournament_results$testing_performance,
                 lexicase_results$testing_performance,
                 seed = 39,
                 alternative = "t")

## [1] "observed_diff: -0.170085425997122"
## [1] "lower: -1.82321742524129"
## [1] "upper: 1.8232173573514"
## [1] "fail to reject null hypothesis"
## [1] "p-value: 0.89693"
```



5.5.2 Validation accuracy

```
validation_plot(filter(task_data, split == '95%'))
```



Summary statistics for the testing performance of the selection schemes at the 5% selection set split:

```
validation_accuracy_summary(filter(task_data, split == '95%'))
```

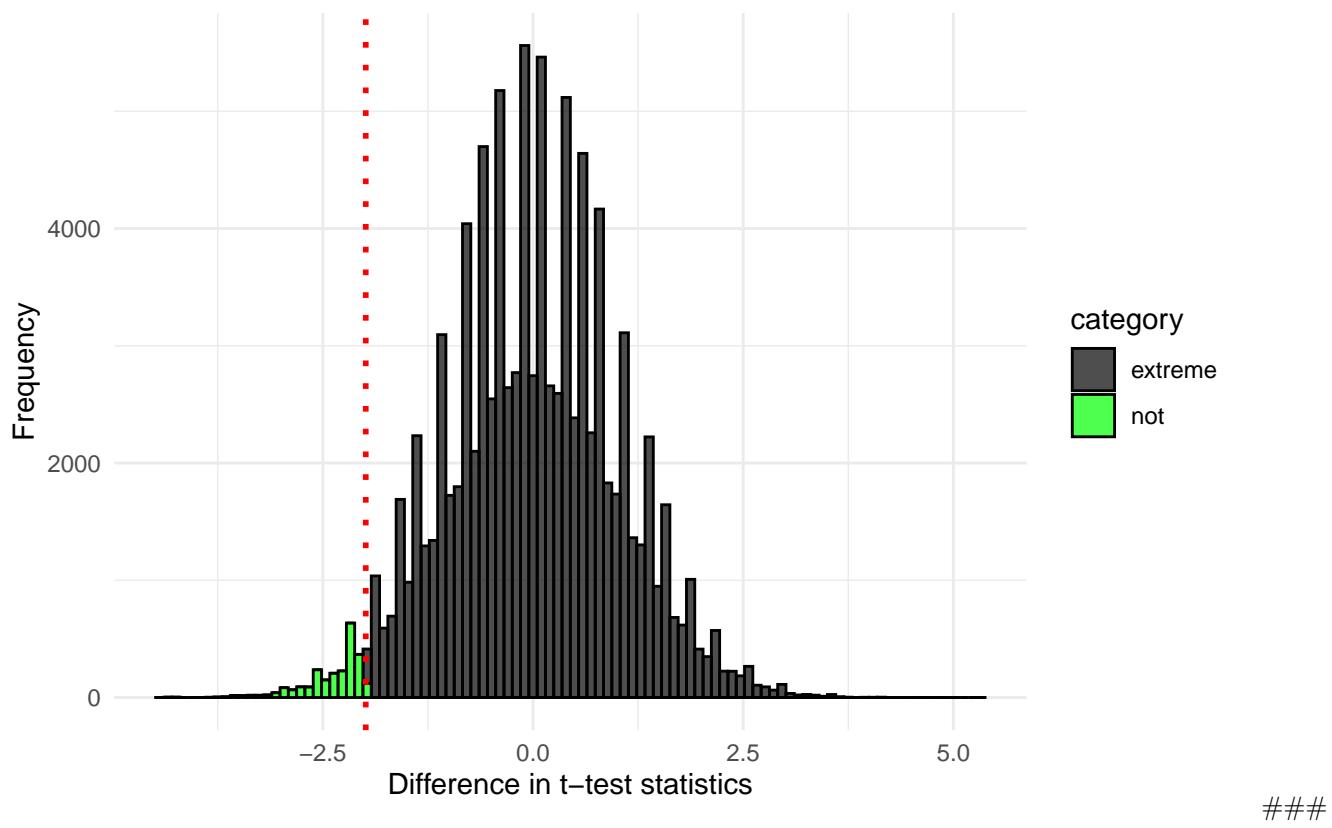
```
## # A tibble: 2 x 8
##   selection  count  na_cnt  min  median  mean   max      IQR
##   <fct>      <int>  <int>  <dbl>  <dbl>  <dbl>  <dbl>  <dbl>
## 1 tournament    40      0  0.789  0.798  0.799  0.811  0.00625
## 2 lexicase      40      0  0.788  0.801  0.801  0.808  0.00664
```

The permutation test revealed that the results are:

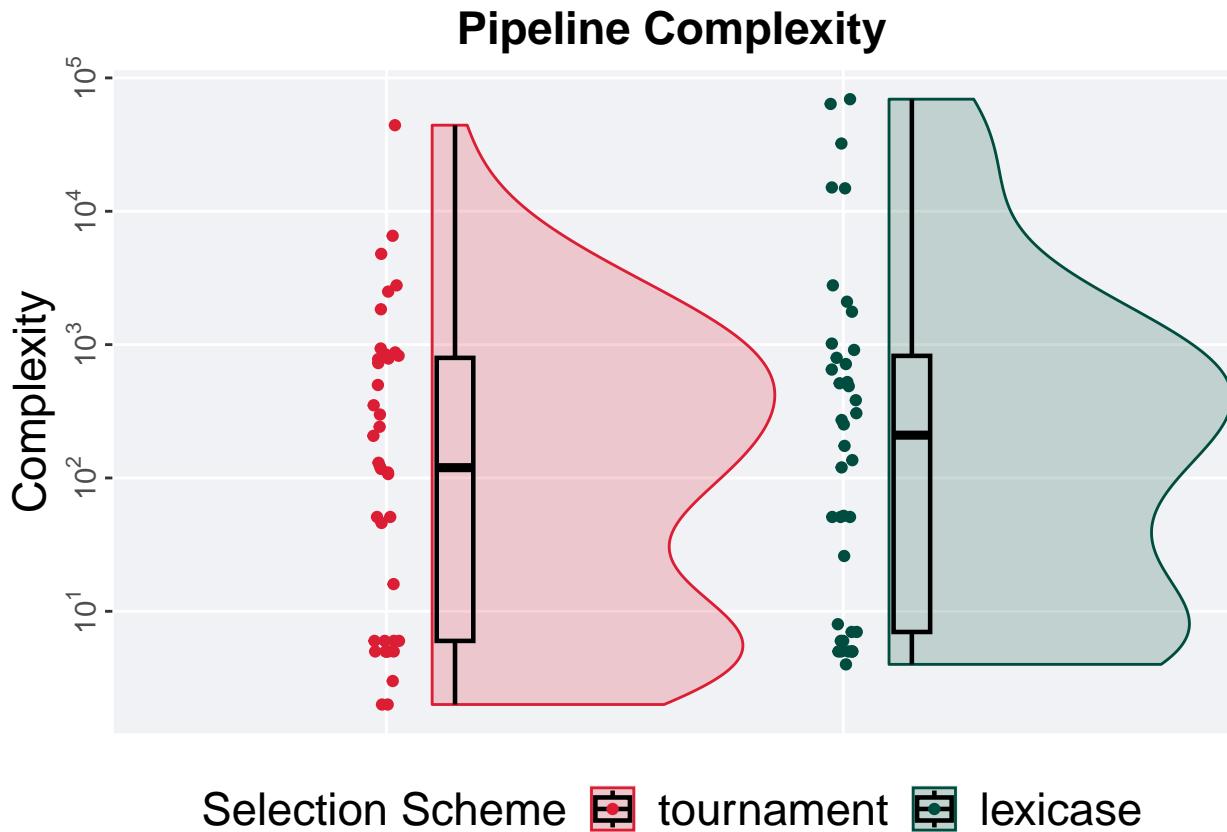
```
tournament_results <- filter(task_data, split == '95%' & selection == 'tournament')
lexicase_results <- filter(task_data, split == '95%' & selection == 'lexicase')
permutation_test(tournament_results$training_performance,
                 lexicase_results$training_performance,
                 seed = 40,
                 alternative = "1")
```

```
## [1] "observed_diff: -1.98968011416173"
## [1] "permutation_diffs[0.05 * n_permutations]: -1.69407255656074"
## [1] "reject null hypothesis"
## [1] "p-value: 0.0243"
```

Permutation Test: Frequency of T-test Statistic Differences



```
complexity_plot(filter(task_data, split == '95%'))
```



Summary statistics for the pipeline complexity of the selection schemes at the 5% selection set split:

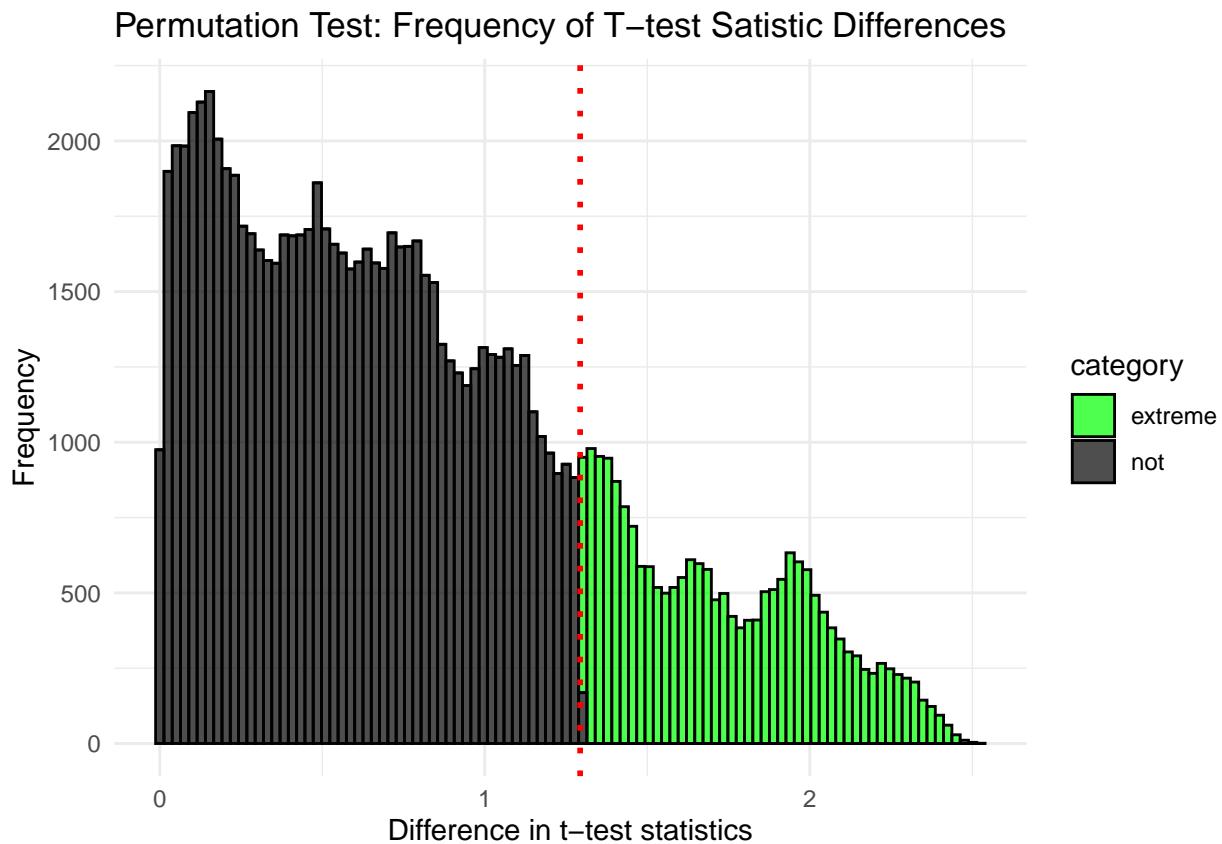
```
complexity_summary(filter(task_data, split == '95%'))
```

```
## # A tibble: 2 x 8
##   selection count na_cnt  min median  mean   max   IQR
##   <fct>     <int> <int> <dbl> <dbl> <dbl> <int> <dbl>
## 1 tournament     40     0     2  120. 1771. 44201   790.
## 2 lexicase       40     0     4  213  5235. 69251   818
```

The permutation test revealed that the results are:

```
tournament_results <- filter(task_data, split == '95%' & selection == 'tournament')
lexicase_results <- filter(task_data, split == '95%' & selection == 'lexicase')
permutation_test(tournament_results$testing_complexity,
                 lexicase_results$testing_complexity,
                 seed = 214,
                 alternative = "t")
```

```
## [1] "observed_diff: -1.29340009357794"
## [1] "lower: -1.9748060094316"
## [1] "upper: 1.9749649847374"
## [1] "fail to reject null hypothesis"
## [1] "p-value: 0.2142"
```



Chapter 6

Task 190146

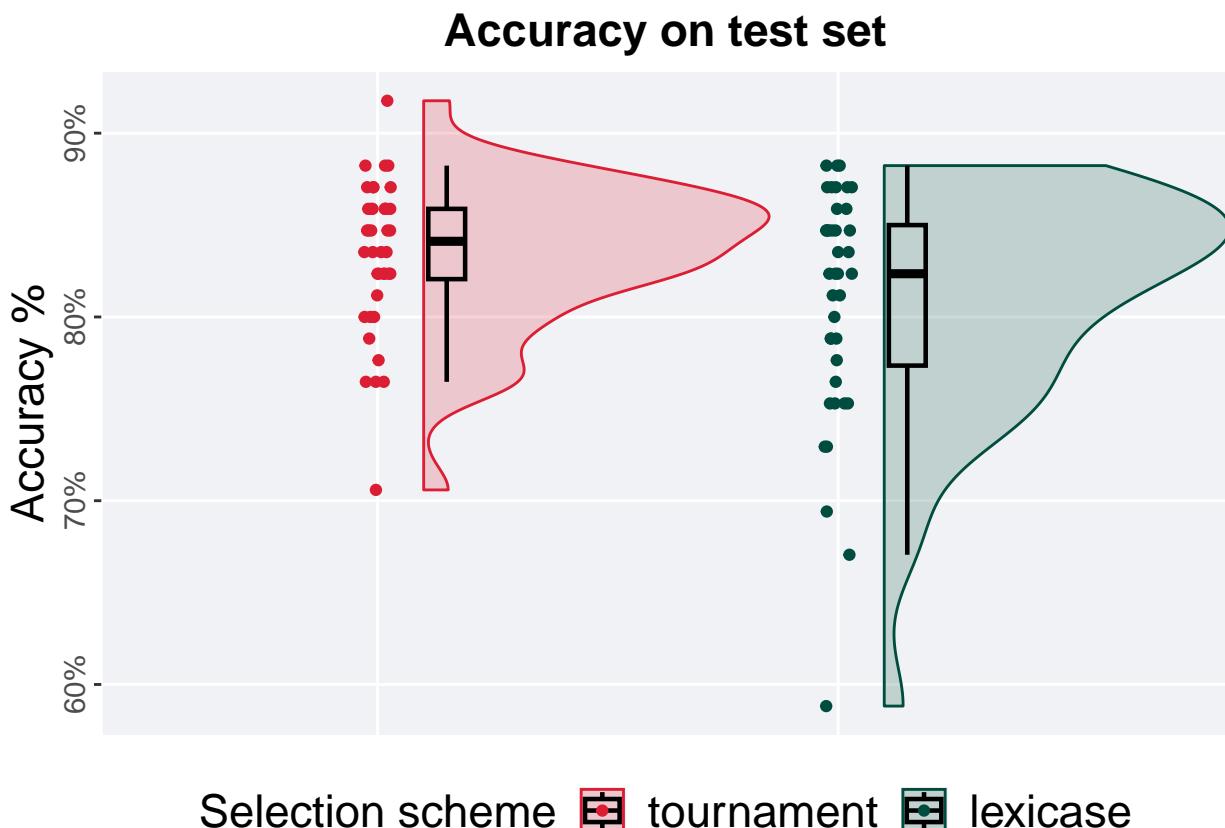
We present the results of our analysis of task 190146 with the different selection set splits used in our study.

```
task_data <- filter(results, task_id == 190146)
```

6.1 5%

6.1.1 Test accuracy

```
test_plot(filter(task_data, split == '5%'))
```



Summary statistics for the testing performance of the selection schemes at the 5% selection set split:

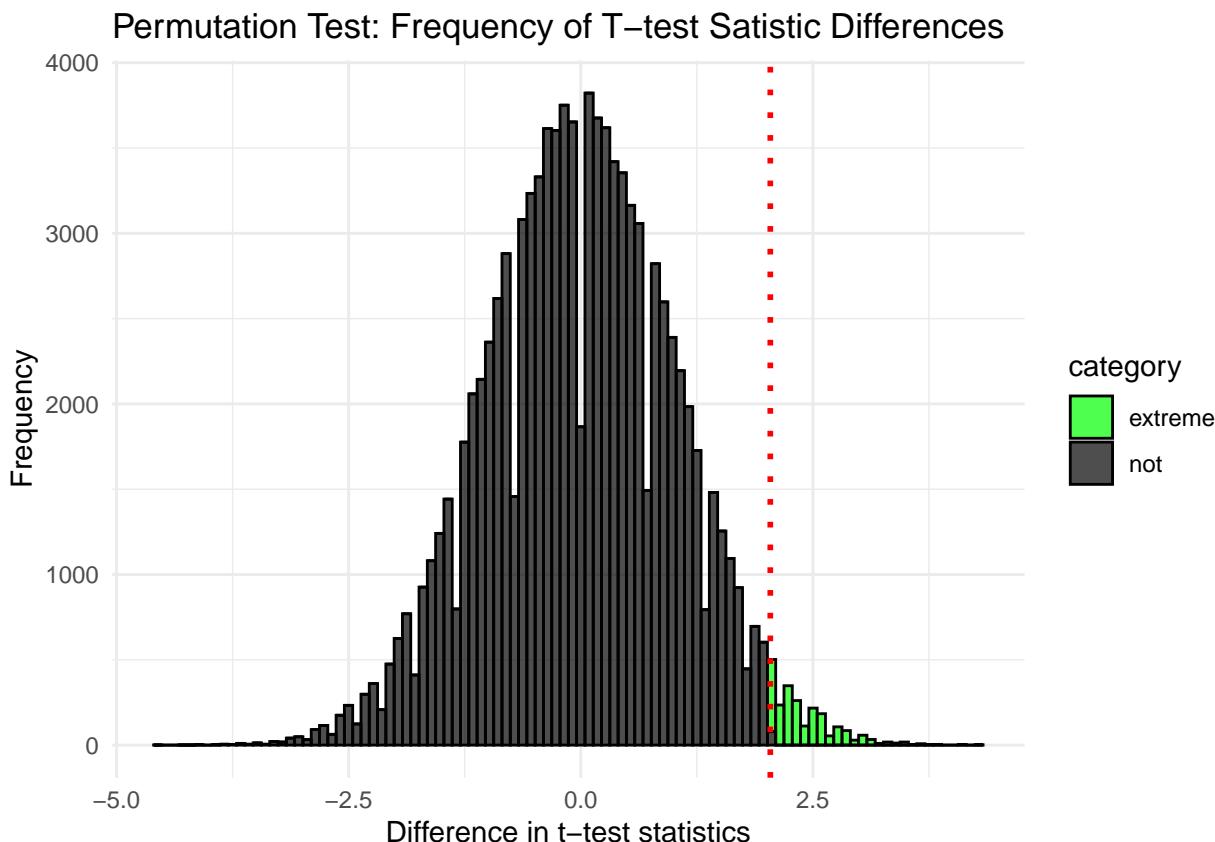
```
test_results_summary(filter(task_data, split == '5%'))
```

```
## # A tibble: 2 x 8
##   selection count na_cnt  min median  mean  max    IQR
##   <fct>     <int> <int> <dbl> <dbl> <dbl> <dbl> <dbl>
## 1 tournament     40     0 0.706 0.841 0.834 0.918 0.0382
## 2 lexicase       40     0 0.588 0.824 0.809 0.882 0.0765
```

The permutation test revealed that the results are:

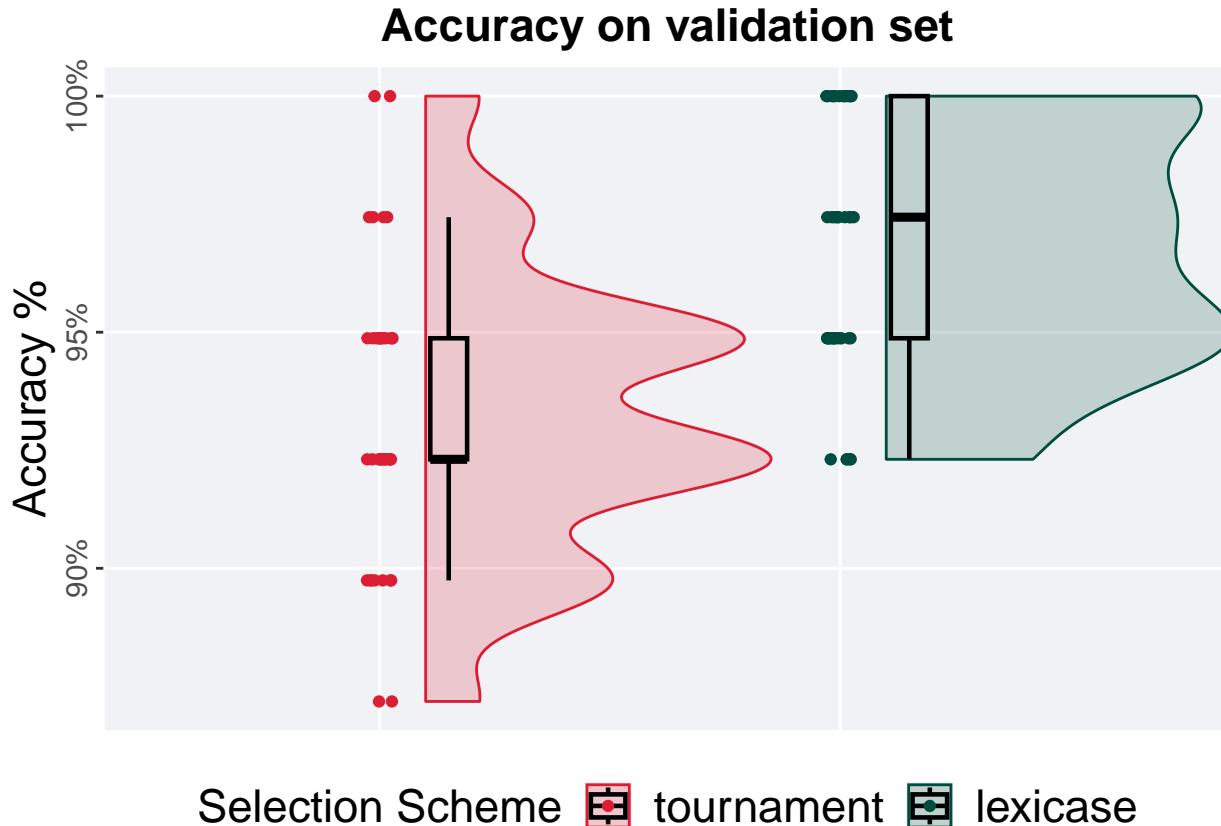
```
tournament_results <- filter(task_data, split == '5%' & selection == 'tournament')
lexicase_results <- filter(task_data, split == '5%' & selection == 'lexicase')
permutation_test(tournament_results$testing_performance,
                  lexicase_results$testing_performance,
                  seed = 41,
                  alternative = "g")
```

```
## [1] "observed_diff: 2.04108926085368"
## [1] "permutation_diffs[0.95 * n_permutations]: 1.63730880862319"
## [1] "reject null hypothesis"
## [1] "p-value: 0.02222"
```



6.1.2 Validation accuracy

```
validation_plot(filter(task_data, split == '5%'))
```



Summary statistics for the testing performance of the selection schemes at the 5% selection set split:

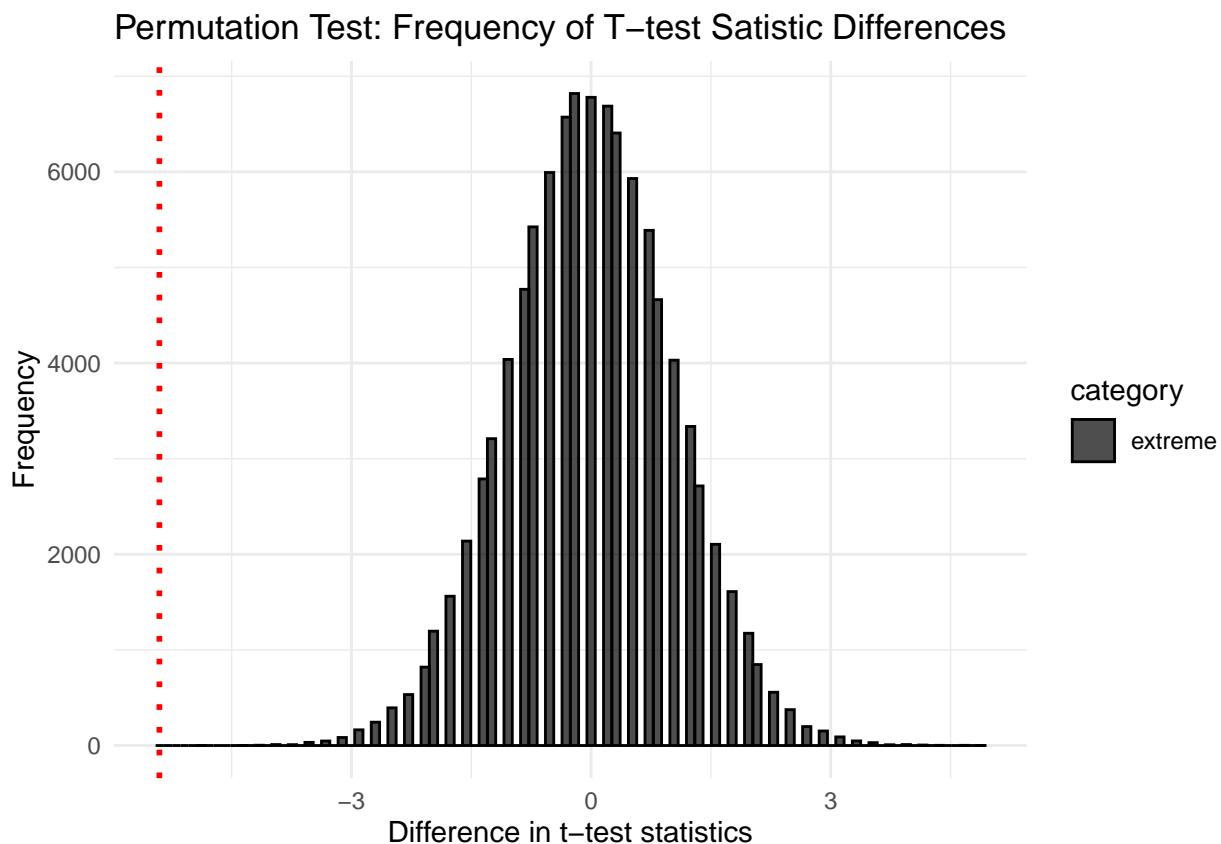
```
validation_accuracy_summary(filter(task_data, split == '5%'))
```

```
## # A tibble: 2 x 8
##   selection count na_cnt  min median  mean   max     IQR
##   <fct>     <int> <int> <dbl> <dbl> <dbl> <dbl> <dbl>
## 1 tournament     40     0 0.872  0.923  0.933     1  0.0256
## 2 lexicase       40     0 0.923  0.974  0.967     1  0.0513
```

The permutation test revealed that the results are:

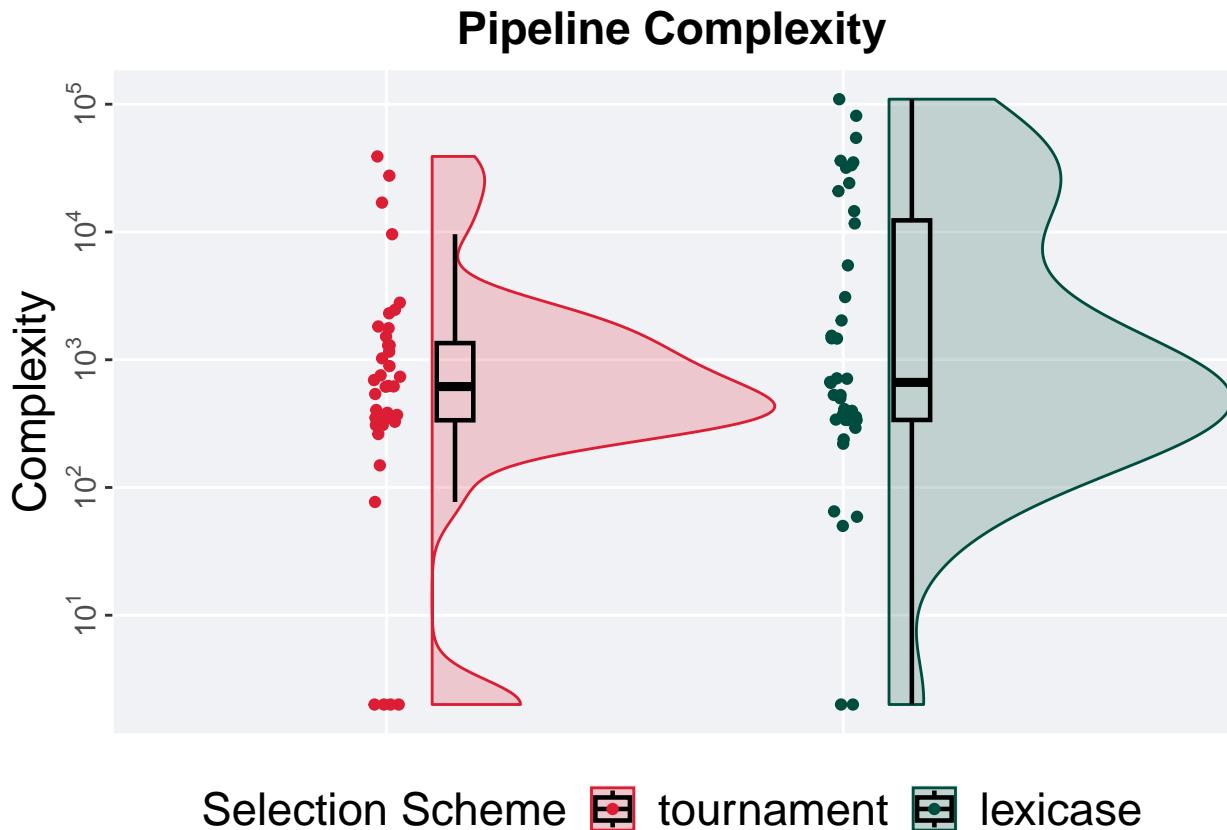
```
tournament_results <- filter(task_data, split == '5%' & selection == 'tournament')
lexicase_results <- filter(task_data, split == '5%' & selection == 'lexicase')
permutation_test(tournament_results$training_performance,
                 lexicase_results$training_performance,
                 seed = 42,
                 alternative = "1")
```

```
## [1] "observed_diff: -5.40415884185913"
## [1] "permutation_diffs[0.05 * n_permutations]: -1.74010014099775"
## [1] "reject null hypothesis"
## [1] "p-value: 1e-05"
```



6.1.3 Pipeline complexity

```
complexity_plot(filter(task_data, split == '5%'))
```



Summary statistics for the pipeline complexity of the selection schemes at the 5% selection set split:

```
complexity_summary(filter(task_data, split == '5%'))
```

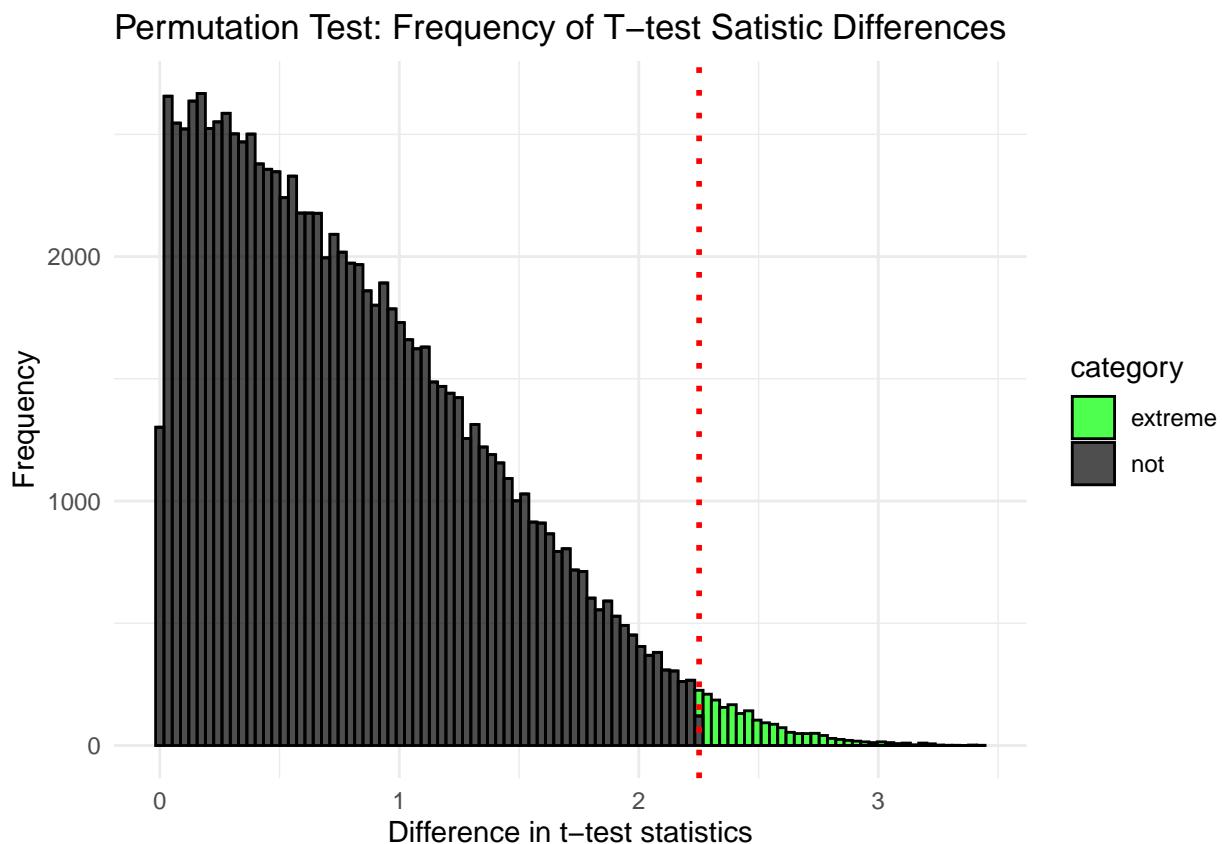
```
## # A tibble: 2 x 8
##   selection count na_cnt  min median  mean    max    IQR
##   <fct>     <int> <int> <int> <dbl> <dbl> <int> <dbl>
## 1 tournament     40     0     2    616.  3011. 39012 1017.
## 2 lexicase       40     0     2   664. 11899. 109501 12051.
```

The permutation test revealed that the results are:

```
tournament_results <- filter(task_data, split == '5%' & selection == 'tournament')
lexicase_results <- filter(task_data, split == '5%' & selection == 'lexicase')
```

```
permutation_test(tournament_results$testing_complexity,
                  lexicase_results$testing_complexity,
                  seed = 215,
                  alternative = "t")
```

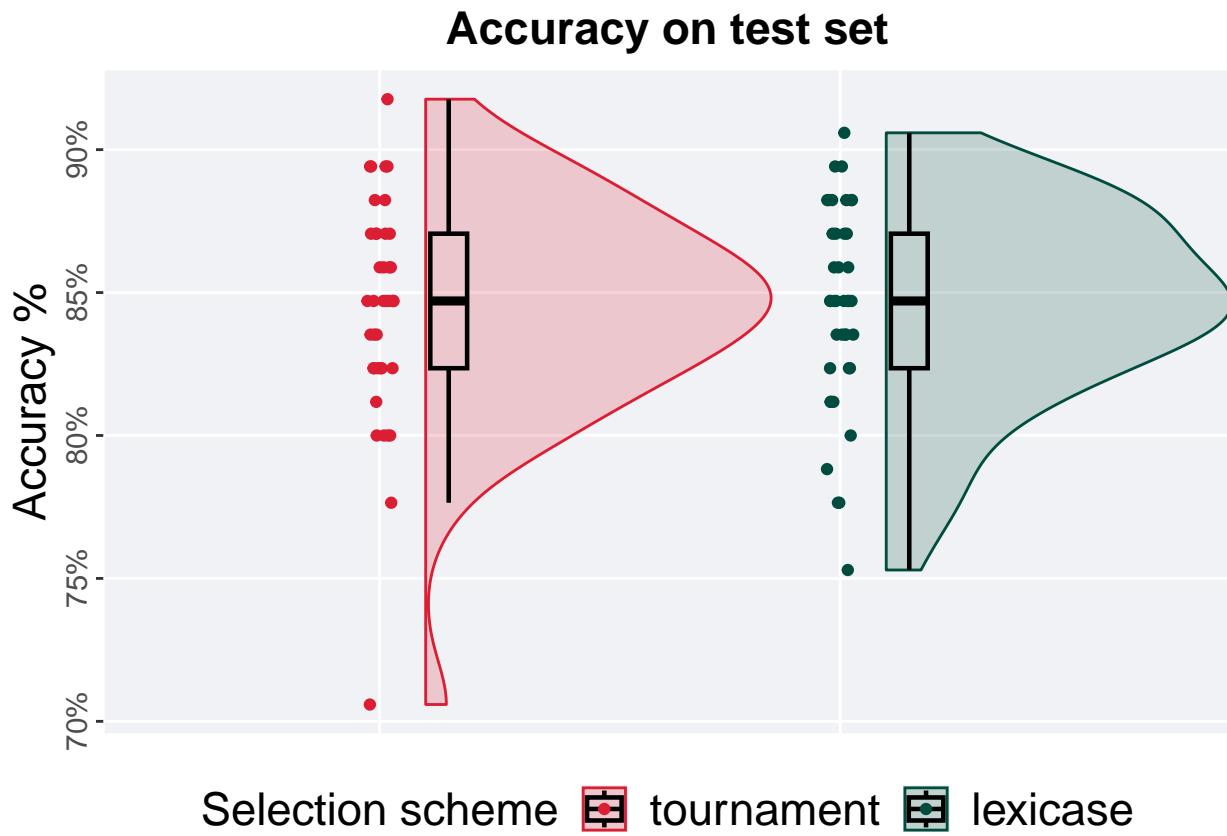
```
## [1] "observed_diff: -2.25177100524619"
## [1] "lower: -1.94027293992664"
## [1] "upper: 1.93915318045464"
## [1] "reject null hypothesis"
## [1] "p-value: 0.01889"
```



6.2 10%

6.2.1 Test accuracy

```
test_plot(filter(task_data, split == '10%'))
```



Summary statistics for the testing performance of the selection schemes at the 5% selection set split:

```
test_results_summary(filter(task_data, split == '10%'))
```

```
## # A tibble: 2 x 8
##   selection count na_cnt  min median  mean   max     IQR
##   <fct>     <int> <int> <dbl> <dbl> <dbl> <dbl> <dbl>
## 1 tournament     40     0 0.706 0.847 0.844 0.918 0.0471
## 2 lexicase       40     0 0.753 0.847 0.845 0.906 0.0471
```

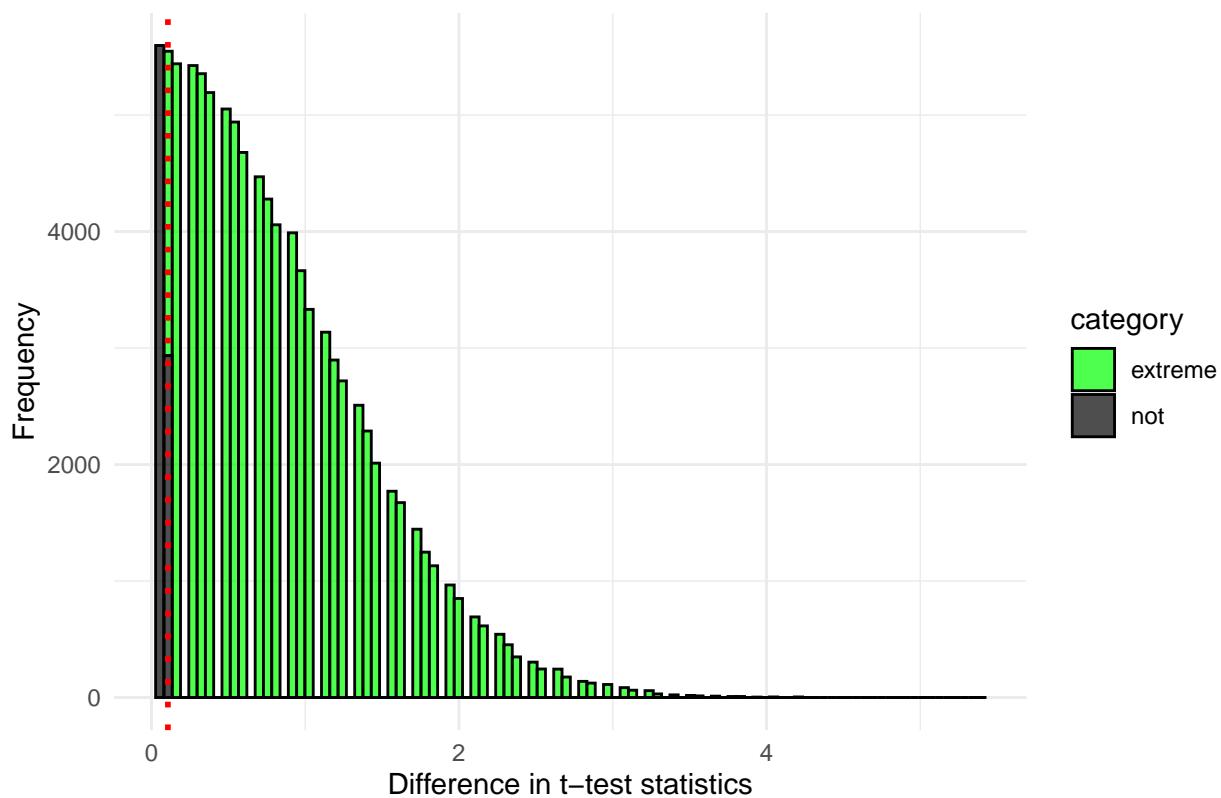
The permutation test revealed that the results are:

```
tournament_results <- filter(task_data, split == '10%' & selection == 'tournament')
lexicase_results <- filter(task_data, split == '10%' & selection == 'lexicase')
```

```
permutation_test(tournament_results$testing_performance,
                 lexicase_results$testing_performance,
                 seed = 43,
                 alternative = "t")
```

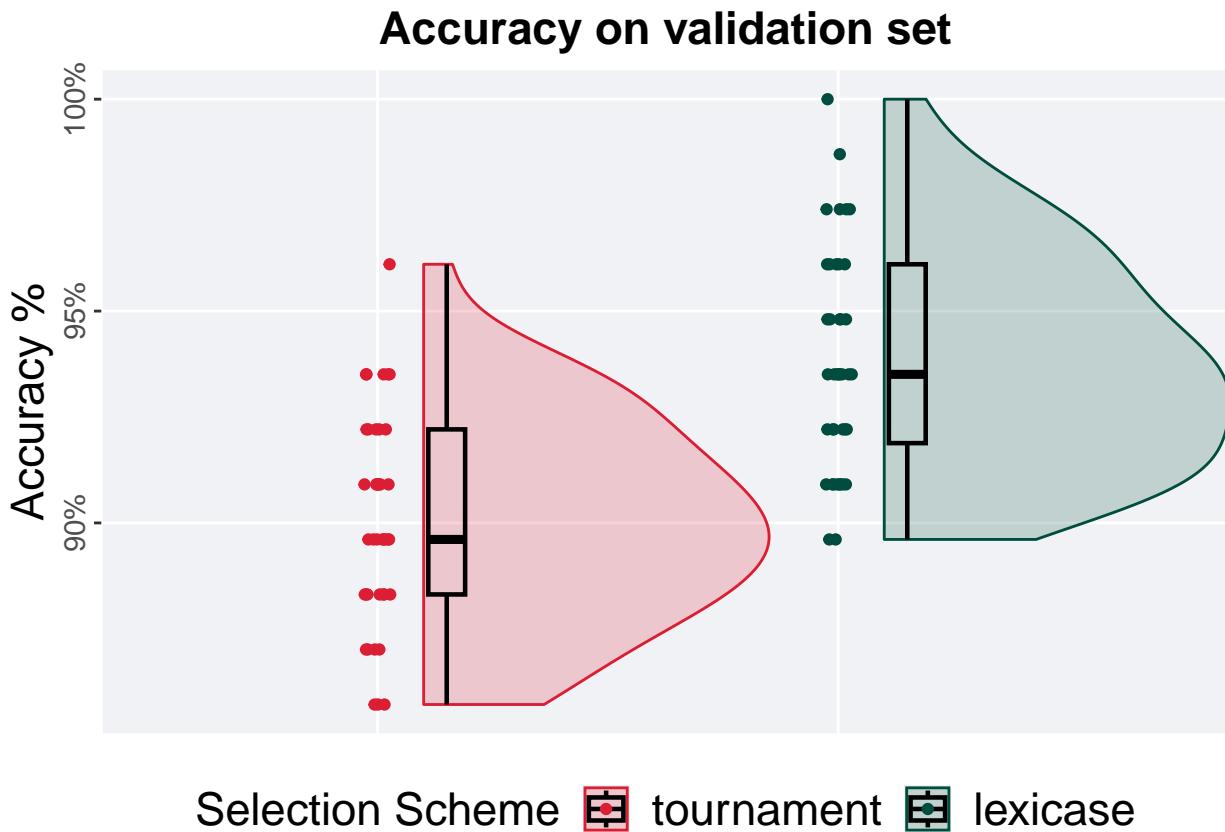
```
## [1] "observed_diff: -0.106802835929617"
## [1] "lower: -2.0078696403403"
## [1] "upper: 2.00786970560947"
## [1] "fail to reject null hypothesis"
## [1] "p-value: 0.91468"
```

Permutation Test: Frequency of T-test Statistic Differences



6.2.2 Validation accuracy

```
validation_plot(filter(task_data, split == '10%'))
```



Summary statistics for the testing performance of the selection schemes at the 5% selection set split:

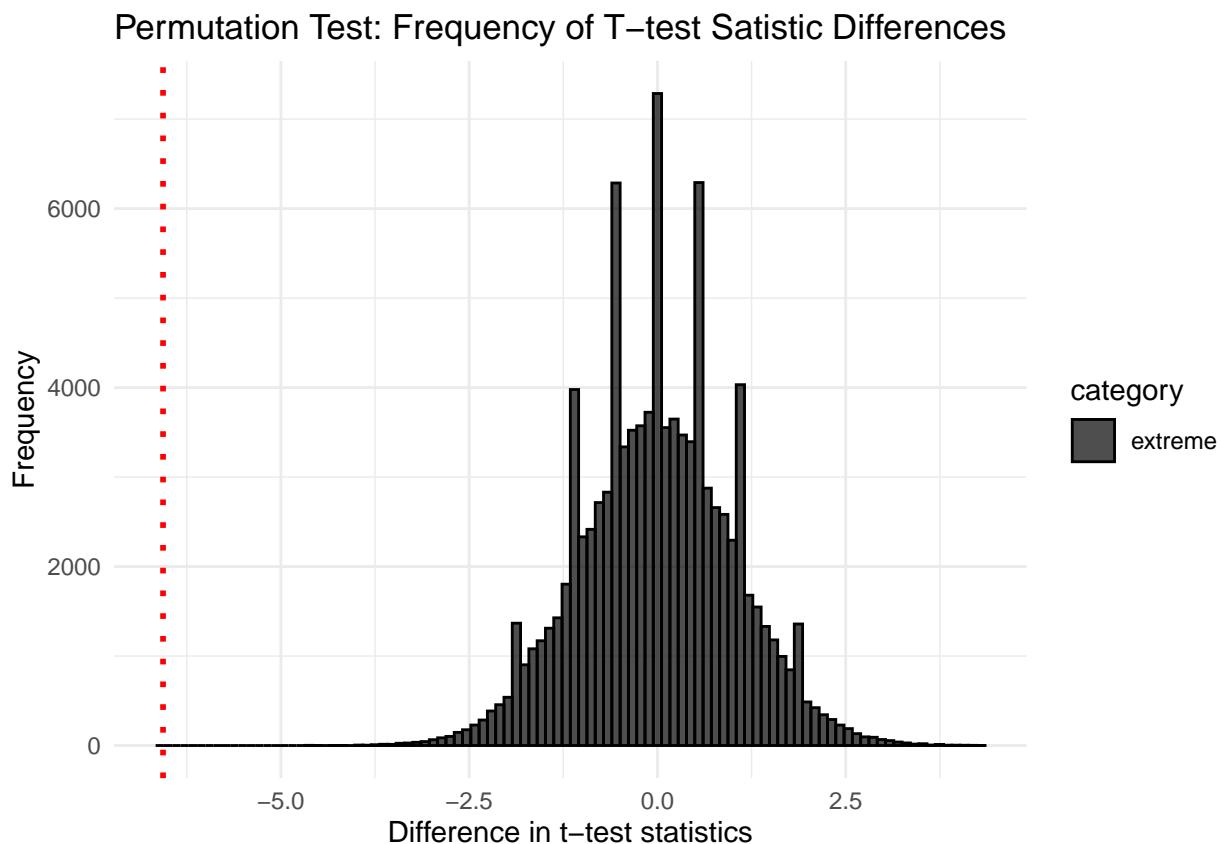
```
validation_accuracy_summary(filter(task_data, split == '10%'))
```

```
## # A tibble: 2 x 8
##   selection count na_cnt  min median  mean  max    IQR
##   <fct>     <int> <int> <dbl>  <dbl> <dbl> <dbl> <dbl>
## 1 tournament     40     0 0.857  0.896  0.900 0.961 0.0390
## 2 lexicase       40     0 0.896  0.935  0.938  1      0.0422
```

The permutation test revealed that the results are:

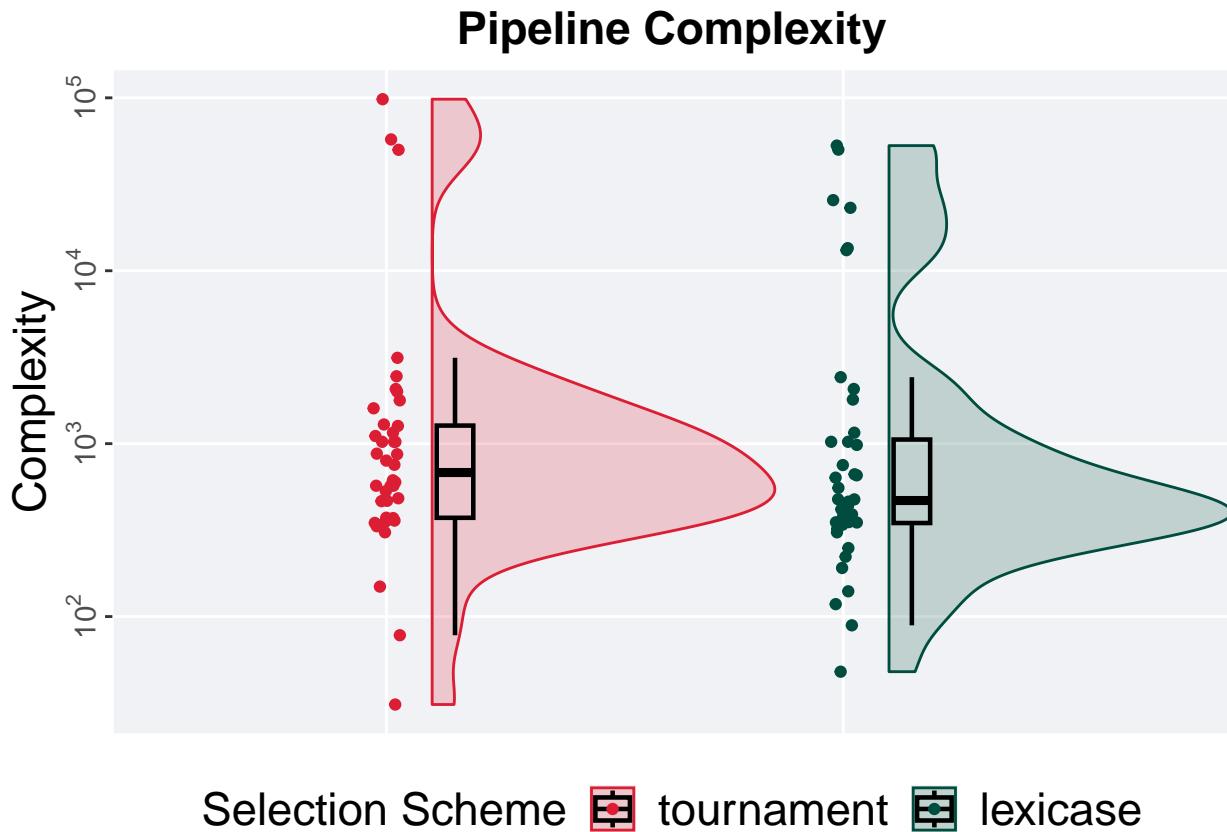
```
tournament_results <- filter(task_data, split == '10%' & selection == 'tournament')
lexicase_results <- filter(task_data, split == '10%' & selection == 'lexicase')
permutation_test(tournament_results$training_performance,
                 lexicase_results$training_performance,
                 seed = 44,
                 alternative = "1")
```

```
## [1] "observed_diff: -6.56504184677295"
## [1] "permutation_diffs[0.05 * n_permutations]: -1.63065789057766"
## [1] "reject null hypothesis"
## [1] "p-value: 1e-05"
```



6.2.3 Pipeline complexity

```
complexity_plot(filter(task_data, split == '10%'))
```



Summary statistics for the pipeline complexity of the selection schemes at the 5% selection set split:

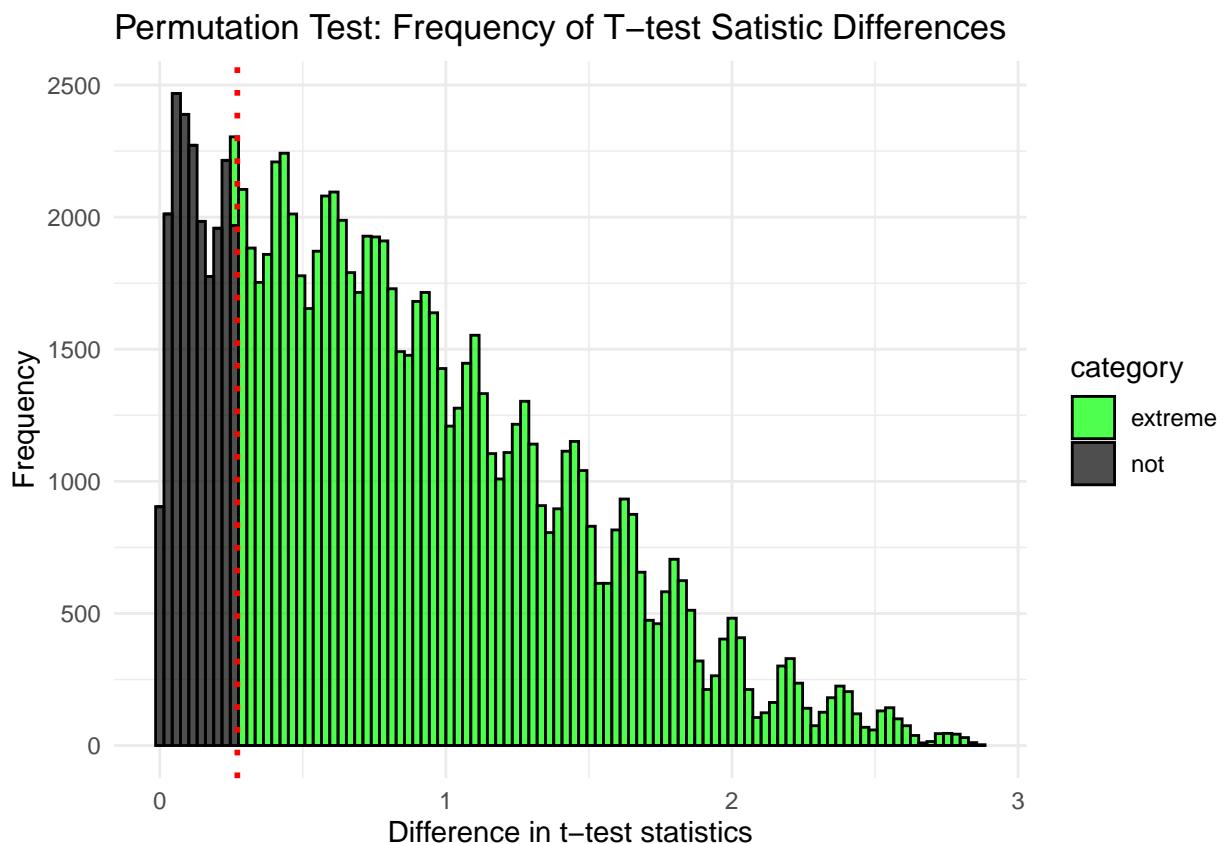
```
complexity_summary(filter(task_data, split == '10%'))
```

```
## # A tibble: 2 x 8
##   selection count na_cnt  min median  mean   max   IQR
##   <fct>     <int> <int> <dbl> <dbl> <dbl> <int> <dbl>
## 1 tournament     40     0    31   684.  5946. 98192  900.
## 2 lexicase       40     0    48   469   4976. 452921  710.
```

The permutation test revealed that the results are:

```
tournament_results <- filter(task_data, split == '10%' & selection == 'tournament')
lexicase_results <- filter(task_data, split == '10%' & selection == 'lexicase')
permutation_test(tournament_results$testing_complexity,
                 lexicase_results$testing_complexity,
                 seed = 216,
                 alternative = "t")

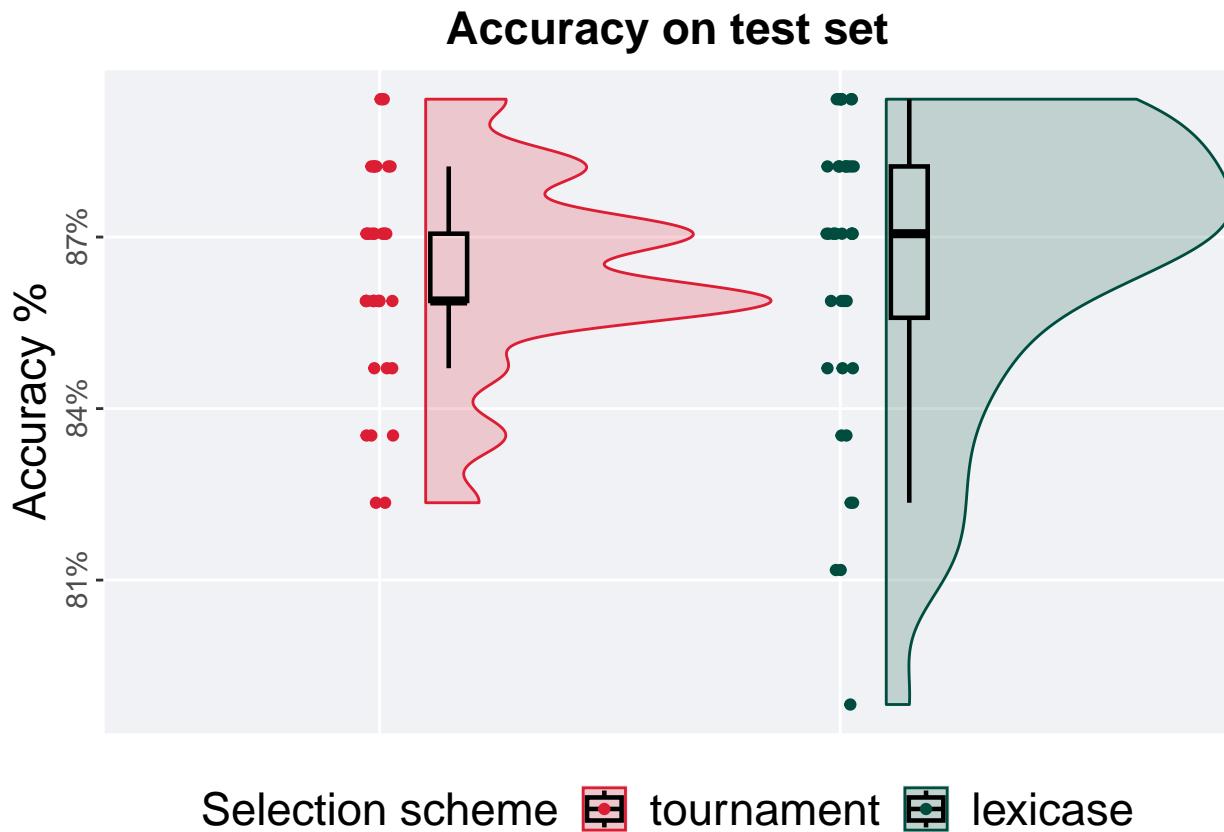
## [1] "observed_diff: 0.271013648398777"
## [1] "lower: -1.92742338558886"
## [1] "upper: 1.9081454040653"
## [1] "fail to reject null hypothesis"
## [1] "p-value: 0.80055"
```



6.3 50%

6.3.1 Test accuracy

```
test_plot(filter(task_data, split == '50%'))
```



Summary statistics for the testing performance of the selection schemes at the 5% selection set split:

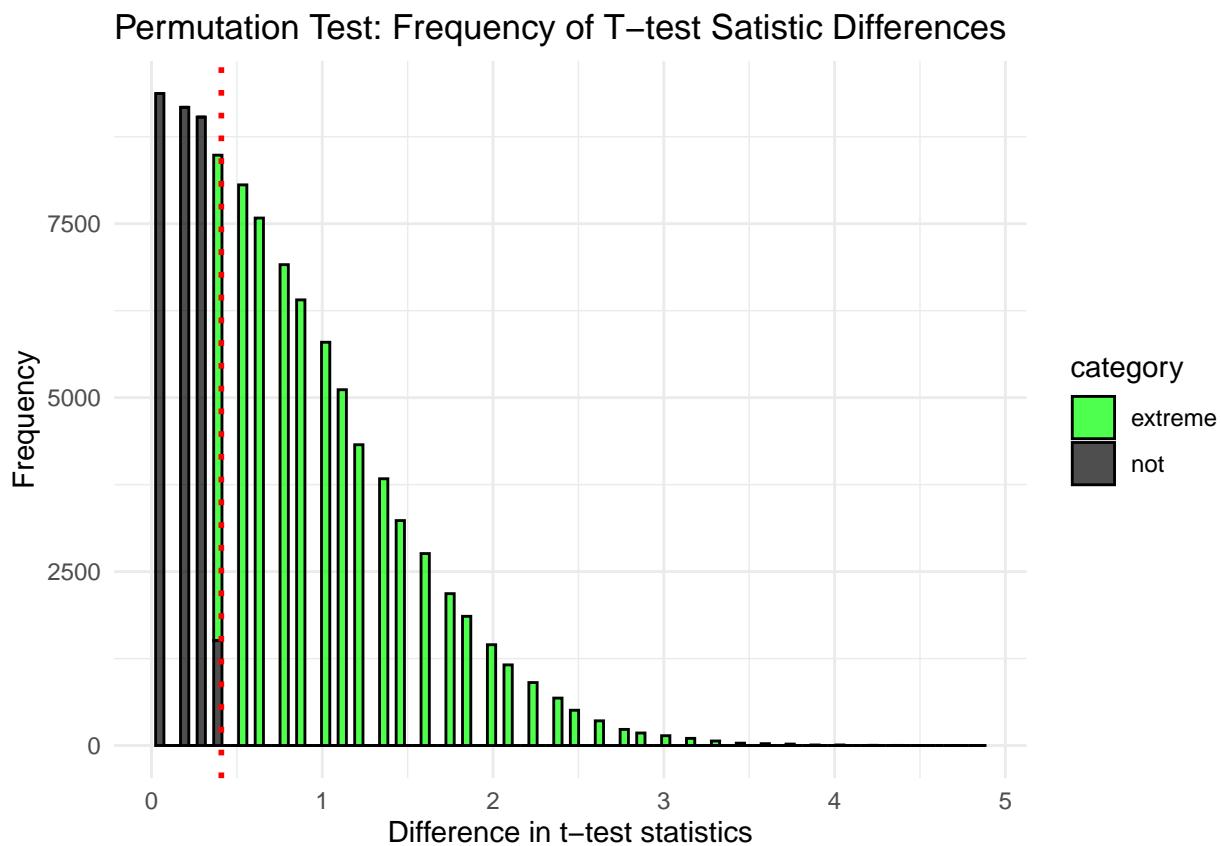
```
test_results_summary(filter(task_data, split == '50%'))
```

```
## # A tibble: 2 x 8
##   selection count na_cnt  min median  mean   max     IQR
##   <fct>     <int> <int> <dbl> <dbl> <dbl> <dbl> <dbl>
## 1 tournament     40     0 0.824 0.859 0.864 0.894 0.0118
## 2 lexicase       40     0 0.788 0.871 0.866 0.894 0.0265
```

The permutation test revealed that the results are:

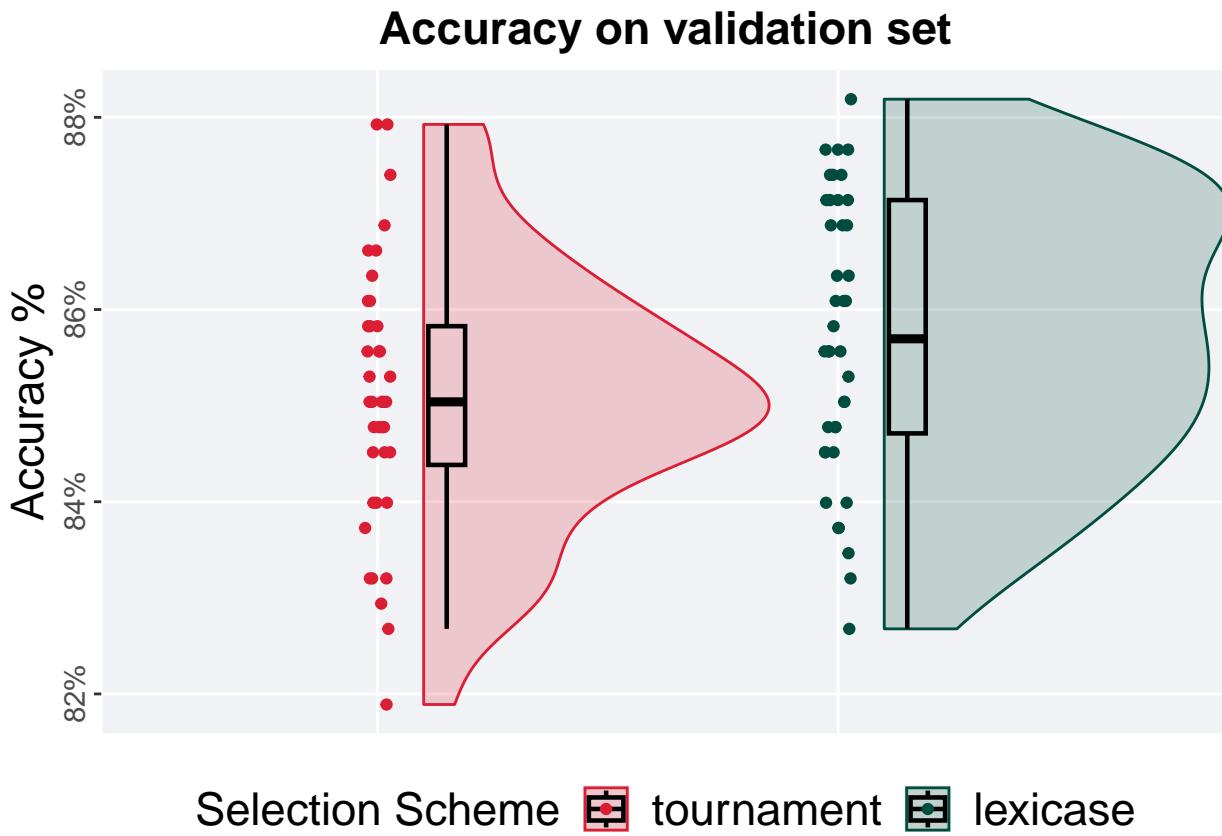
```
tournament_results <- filter(task_data, split == '50%' & selection == 'tournament')
lexicase_results <- filter(task_data, split == '50%' & selection == 'lexicase')
permutation_test(tournament_results$testing_performance,
                 lexicase_results$testing_performance,
                 seed = 45,
                 alternative = "t")
```

```
## [1] "observed_diff: -0.409158936906205"
## [1] "lower: -1.97438850833144"
## [1] "upper: 1.9743887219197"
## [1] "fail to reject null hypothesis"
## [1] "p-value: 0.70915"
```



6.3.2 Validation accuracy

```
validation_plot(filter(task_data, split == '50%'))
```



Summary statistics for the testing performance of the selection schemes at the 5% selection set split:

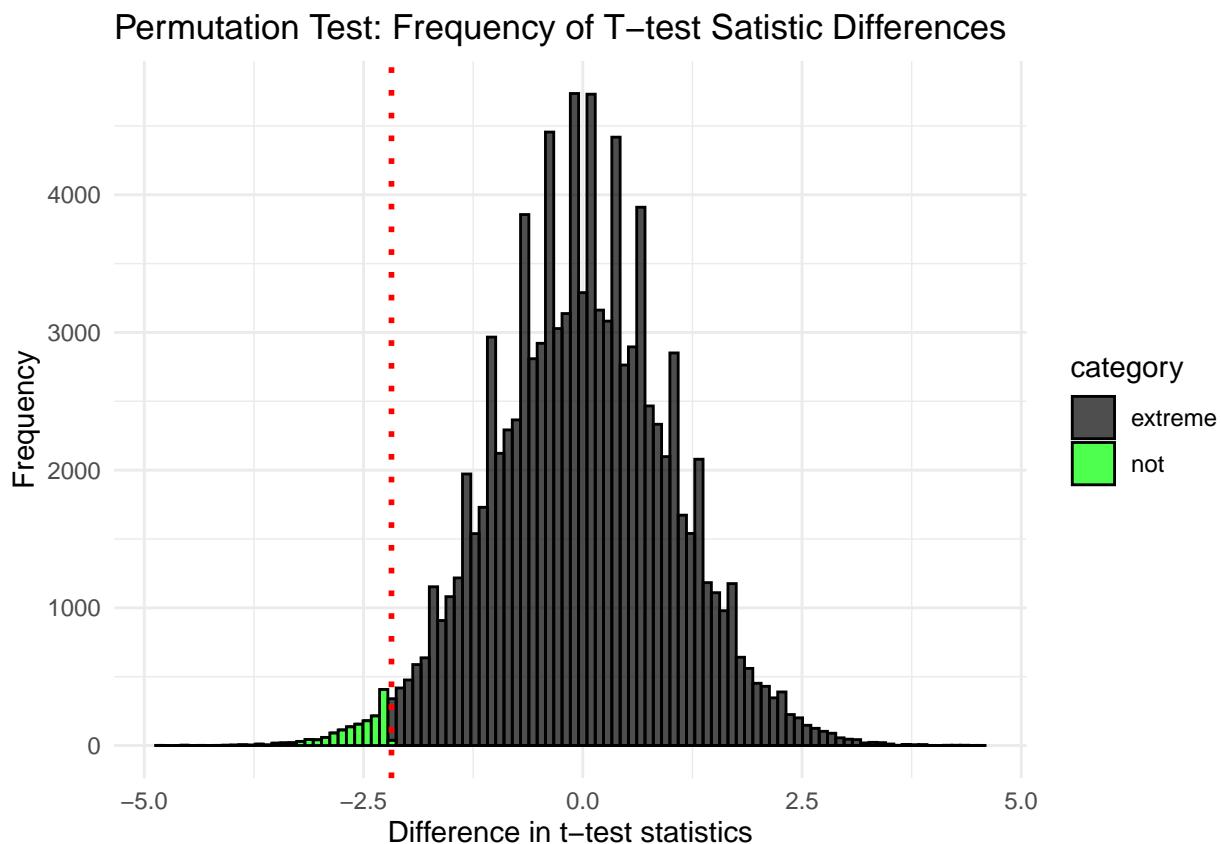
```
validation_accuracy_summary(filter(task_data, split == '50%'))
```

```
## # A tibble: 2 x 8
##   selection  count  na_cnt  min  median  mean  max    IQR
##   <fct>      <int>   <int>  <dbl>  <dbl>  <dbl> <dbl>  <dbl>
## 1 tournament    40      0  0.819  0.850  0.851  0.879  0.0144
## 2 lexicase      40      0  0.827  0.857  0.857  0.882  0.0243
```

The permutation test revealed that the results are:

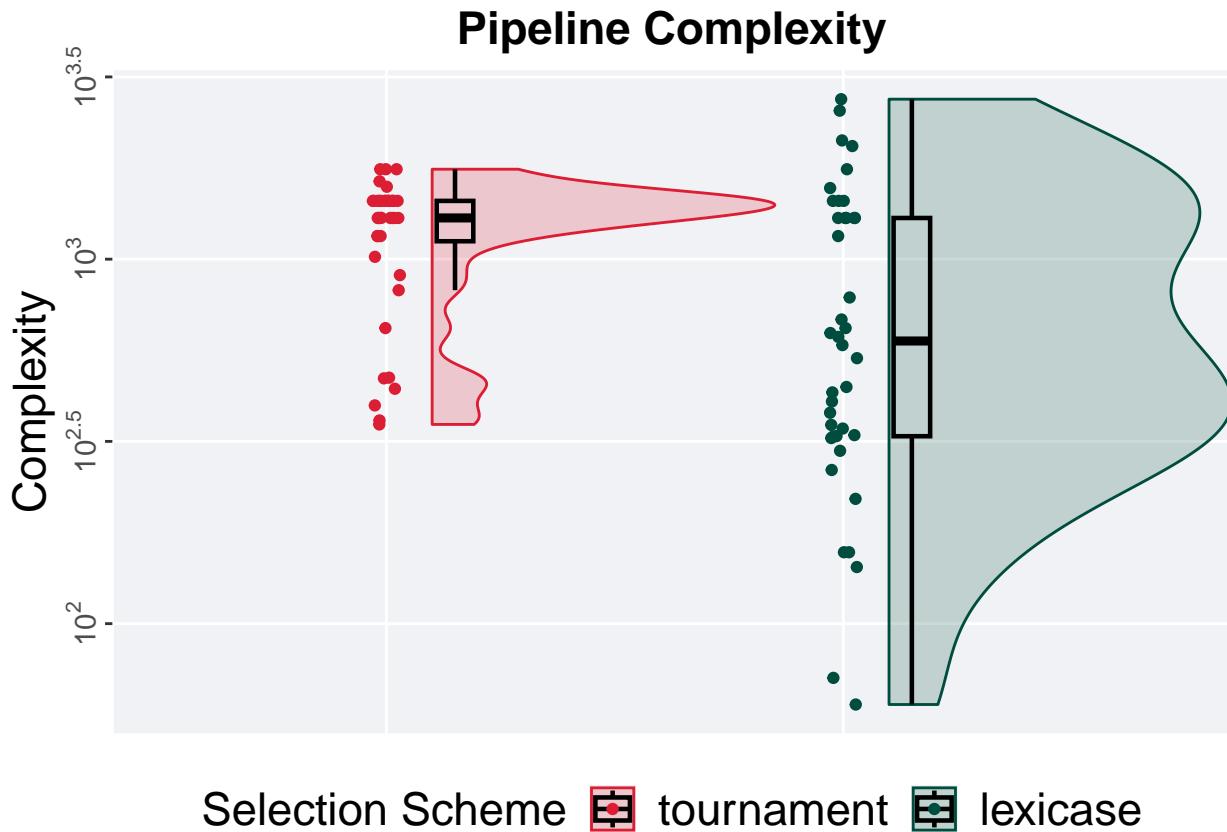
```
tournament_results <- filter(task_data, split == '50%' & selection == 'tournament')
lexicase_results <- filter(task_data, split == '50%' & selection == 'lexicase')
permutation_test(tournament_results$training_performance,
                 lexicase_results$training_performance,
                 seed = 46,
                 alternative = "1")
```

```
## [1] "observed_diff: -2.18154871539912"
## [1] "permutation_diffs[0.05 * n_permutations]: -1.66249763126621"
## [1] "reject null hypothesis"
## [1] "p-value: 0.01598"
```



6.3.3 Pipeline complexity

```
complexity_plot(filter(task_data, split == '50%'))
```



Summary statistics for the pipeline complexity of the selection schemes at the 5% selection set split:

```
complexity_summary(filter(task_data, split == '50%'))
```

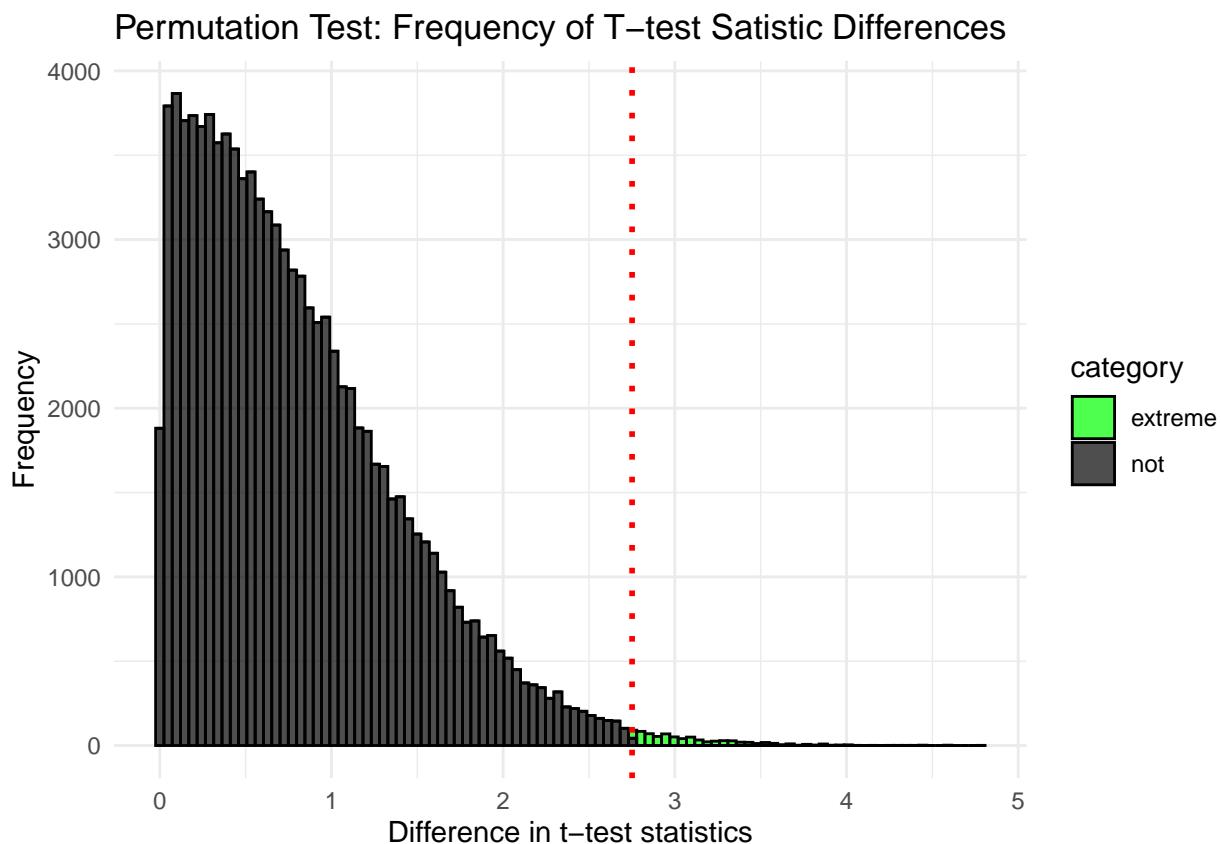
```
## # A tibble: 2 x 8
##   selection count na_cnt  min median  mean   max   IQR
##   <fct>     <int> <int> <dbl> <dbl> <dbl> <int> <dbl>
## 1 tournament     40     0   352  1298. 1212. 1765  324.
## 2 lexicase       40     0     60   596.  857.  2747  970.
```

The permutation test revealed that the results are:

```
tournament_results <- filter(task_data, split == '50%' & selection == 'tournament')
lexicase_results <- filter(task_data, split == '50%' & selection == 'lexicase')
```

```
permutation_test(tournament_results$testing_complexity,
                  lexicase_results$testing_complexity,
                  seed = 217,
                  alternative = "t")
```

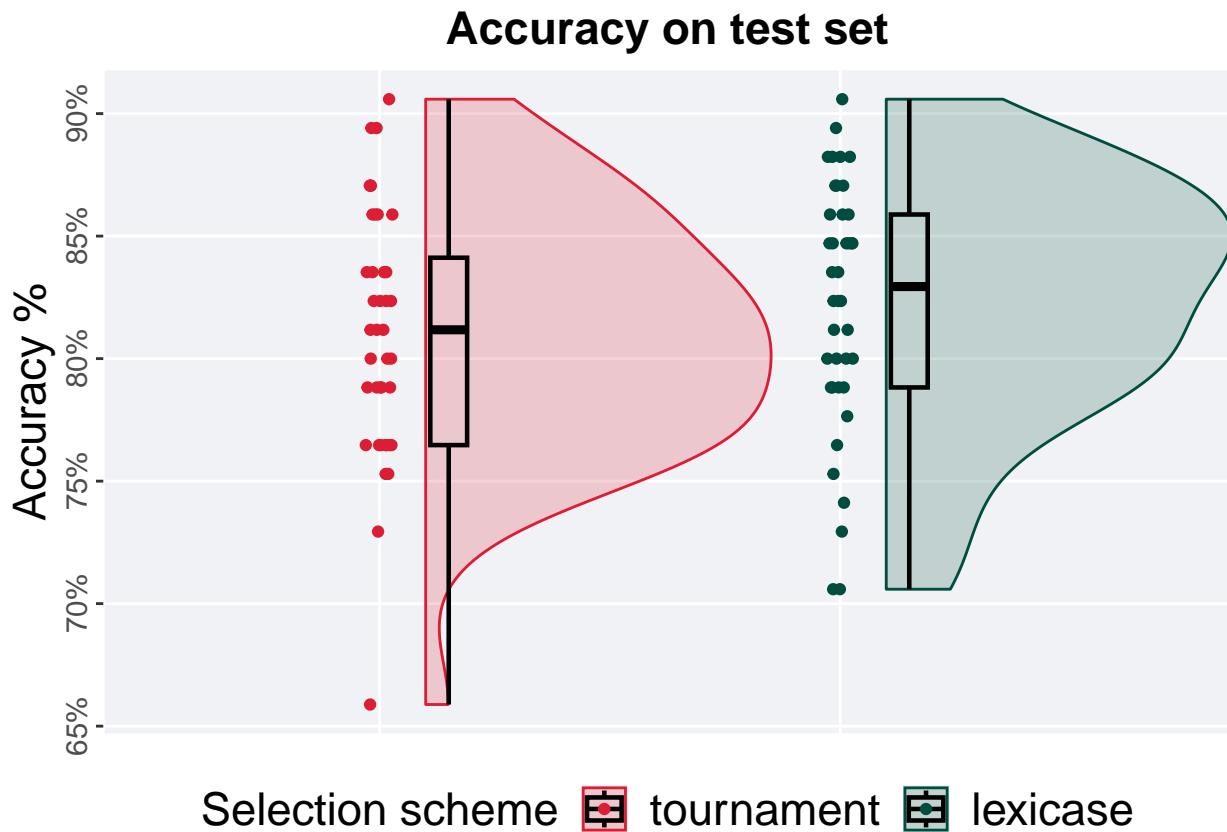
```
## [1] "observed_diff: 2.75110899703106"
## [1] "lower: -1.99045936769785"
## [1] "upper: 1.98527926322556"
## [1] "reject null hypothesis"
## [1] "p-value: 0.00743"
```



6.4 90%

6.4.1 Test accuracy

```
test_plot(filter(task_data, split == '90%'))
```



Summary statistics for the testing performance of the selection schemes at the 5% selection set split:

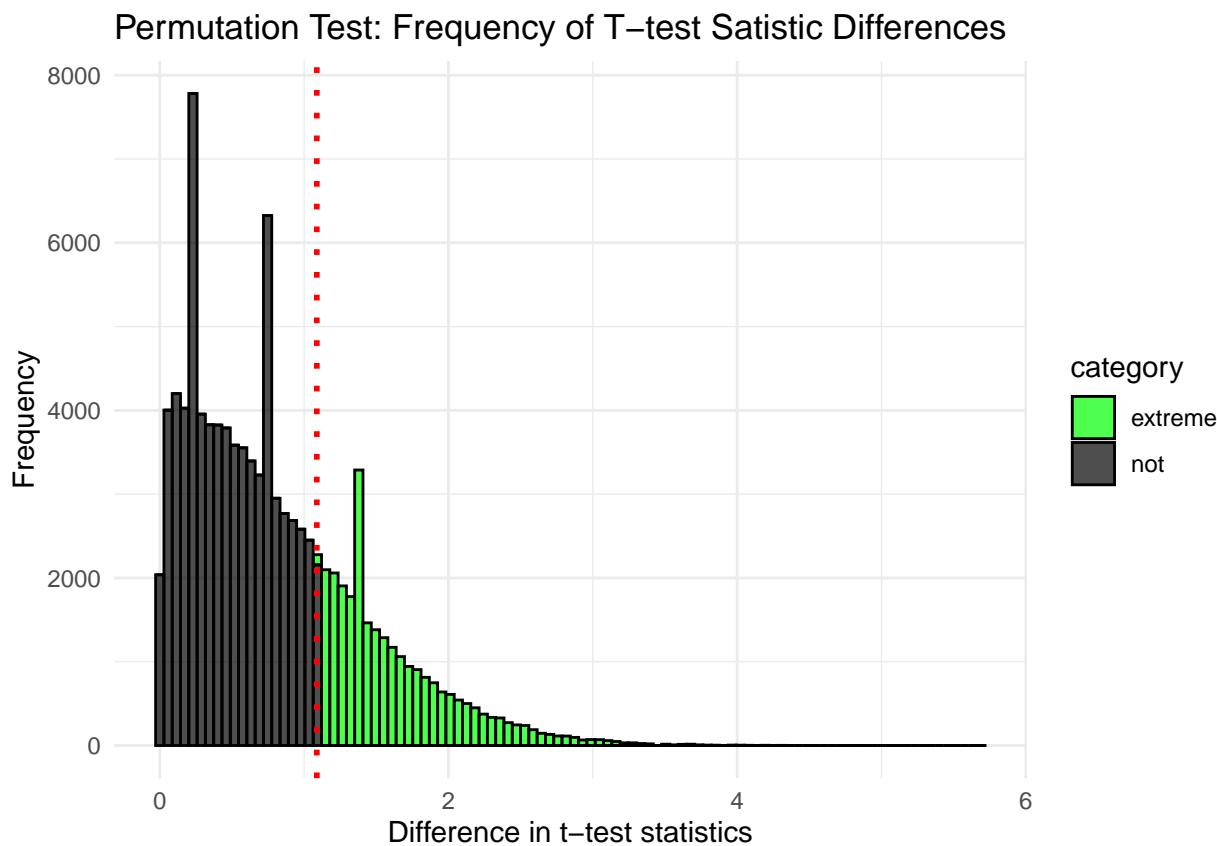
```
test_results_summary(filter(task_data, split == '90%'))
```

```
## # A tibble: 2 x 8
##   selection count na_cnt  min median  mean   max     IQR
##   <fct>     <int> <int> <dbl> <dbl> <dbl> <dbl> <dbl>
## 1 tournament     40     0 0.659 0.812 0.810 0.906 0.0765
## 2 lexicase       40     0 0.706 0.829 0.822 0.906 0.0706
```

The permutation test revealed that the results are:

```
tournament_results <- filter(task_data, split == '90%' & selection == 'tournament')
lexicase_results <- filter(task_data, split == '90%' & selection == 'lexicase')
permutation_test(tournament_results$testing_performance,
                 lexicase_results$testing_performance,
                 seed = 47,
                 alternative = "t")
```

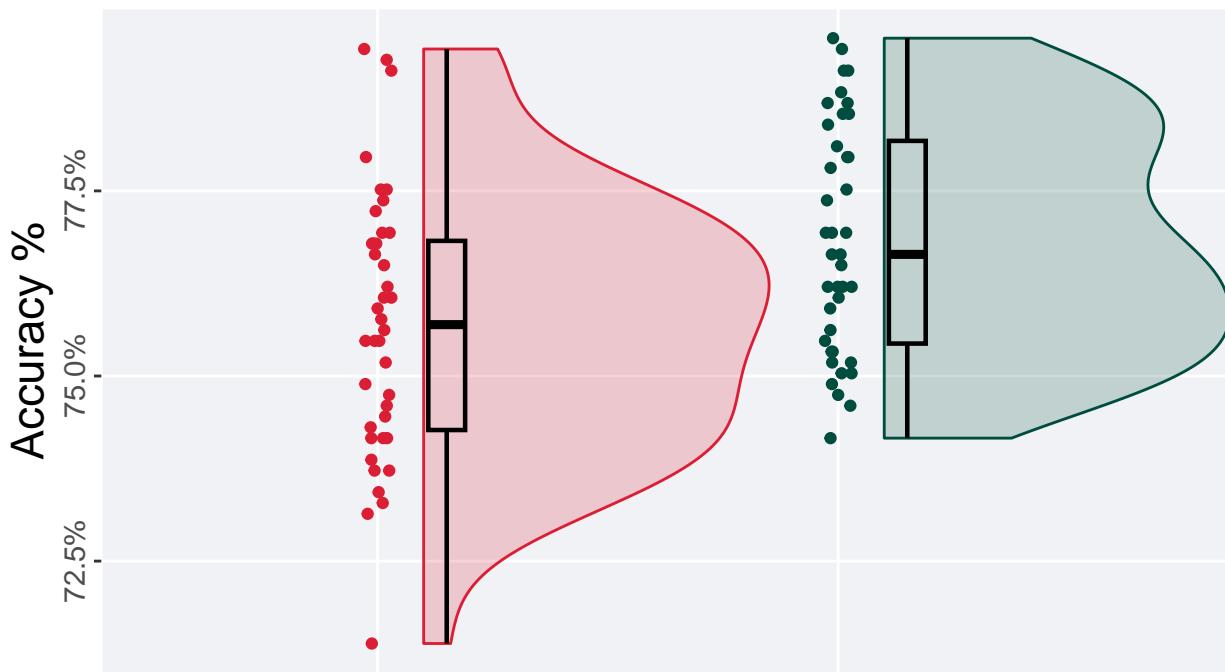
```
## [1] "observed_diff: -1.0880873800637"
## [1] "lower: -2.00381083912905"
## [1] "upper: 2.00381065061134"
## [1] "fail to reject null hypothesis"
## [1] "p-value: 0.26874"
```



6.4.2 Validation accuracy

```
validation_plot(filter(task_data, split == '90%'))
```

Accuracy on validation set



Selection Scheme  tournament  lexicase

Summary statistics for the testing performance of the selection schemes at the 5% selection set split:

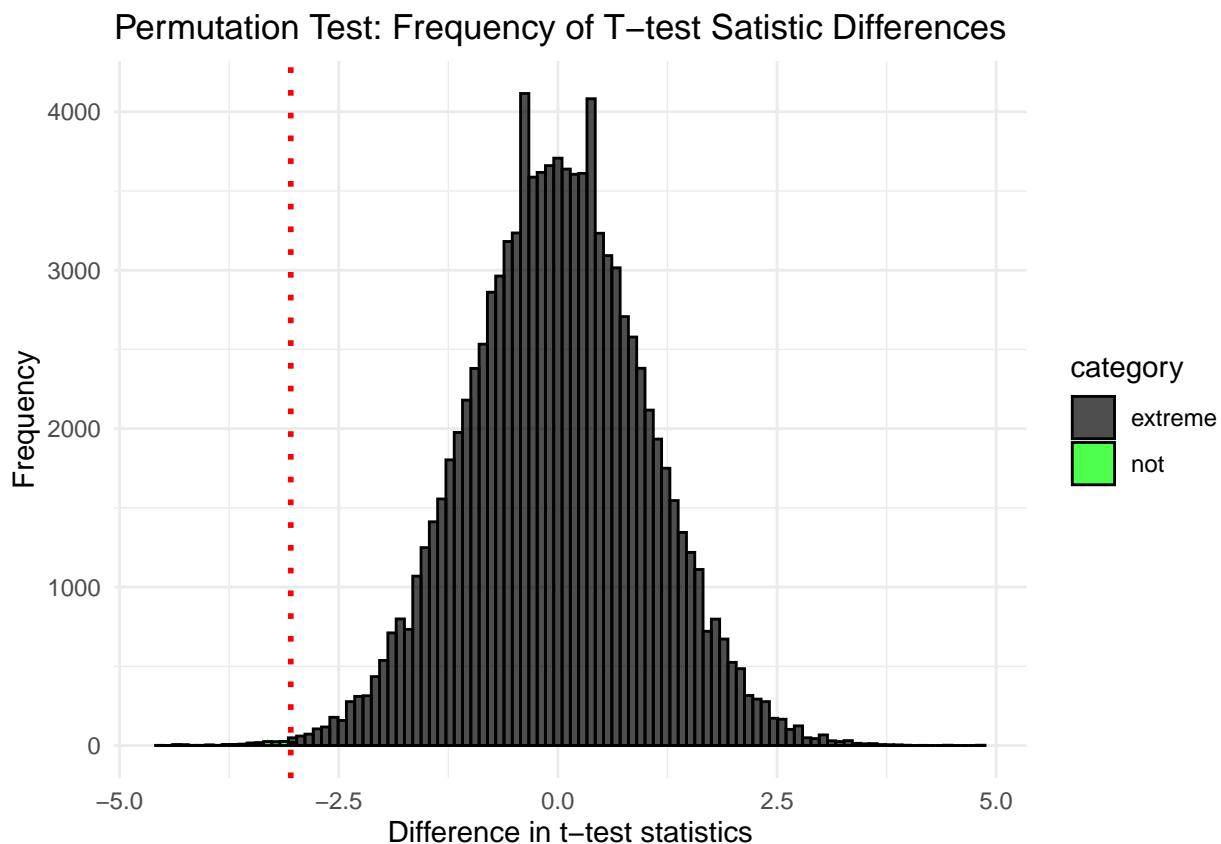
```
validation_accuracy_summary(filter(task_data, split == '90%'))
```

```
## # A tibble: 2 x 8
##   selection  count  na_cnt  min  median  mean  max    IQR
##   <fct>      <int>  <int>  <dbl>  <dbl>  <dbl> <dbl>  <dbl>
## 1 tournament    40      0  0.714  0.757  0.757  0.794  0.0255
## 2 lexicase      40      0  0.742  0.766  0.768  0.796  0.0274
```

The permutation test revealed that the results are:

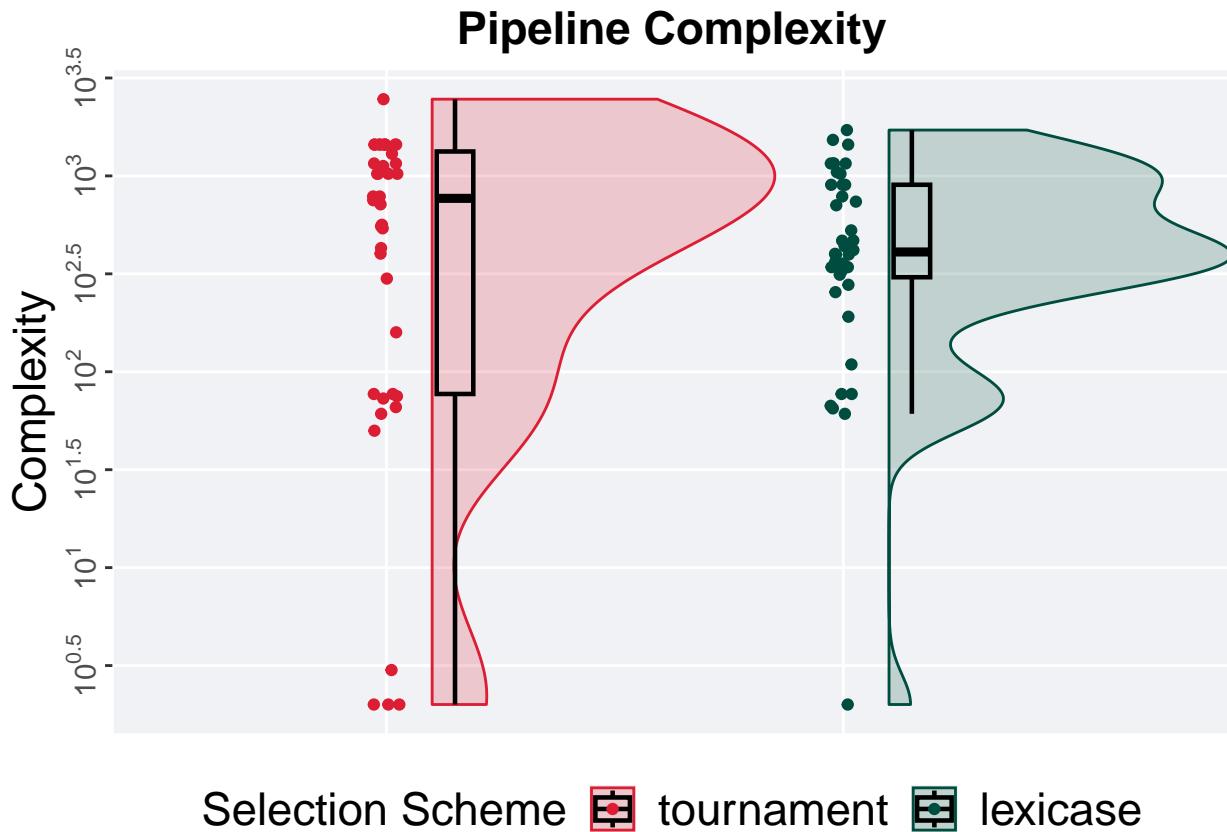
```
tournament_results <- filter(task_data, split == '90%' & selection == 'tournament')
lexicase_results <- filter(task_data, split == '90%' & selection == 'lexicase')
permutation_test(tournament_results$training_performance,
                 lexicase_results$training_performance,
                 seed = 48,
                 alternative = "1")
```

```
## [1] "observed_diff: -3.04756272712672"
## [1] "permutation_diffs[0.05 * n_permutations]: -1.65333887825538"
## [1] "reject null hypothesis"
## [1] "p-value: 0.00159"
```



6.4.3 Pipeline complexity

```
complexity_plot(filter(task_data, split == '90%'))
```



Summary statistics for the pipeline complexity of the selection schemes at the 5% selection set split:

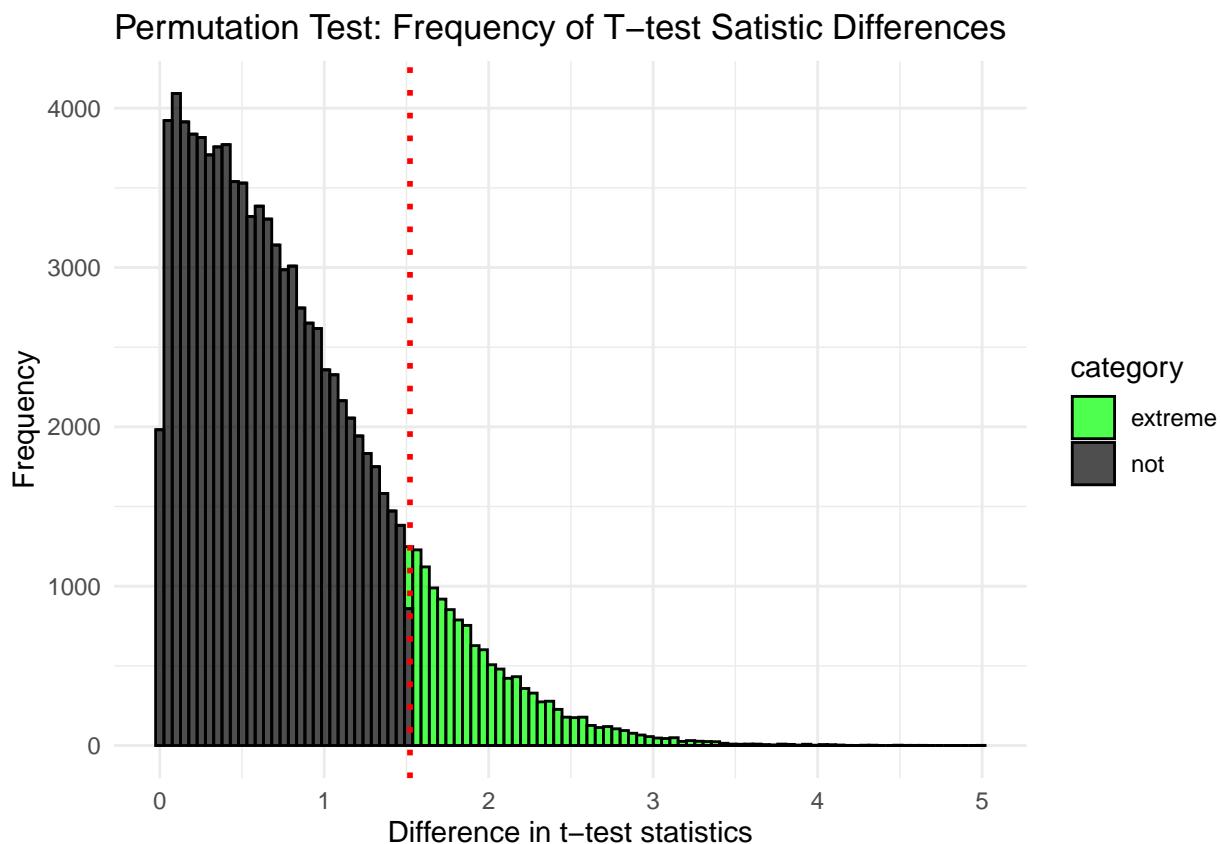
```
complexity_summary(filter(task_data, split == '90%'))
```

```
## # A tibble: 2 x 8
##   selection count na_cnt  min median  mean   max   IQR
##   <fct>     <int> <int> <int> <dbl> <dbl> <int> <dbl>
## 1 tournament     40     0     2    768   769.  2464 1257
## 2 lexicase       40     0     2    409   586.  1714  597.
```

The permutation test revealed that the results are:

```
tournament_results <- filter(task_data, split == '90%' & selection == 'tournament')
lexicase_results <- filter(task_data, split == '90%' & selection == 'lexicase')
permutation_test(tournament_results$testing_complexity,
                 lexicase_results$testing_complexity,
                 seed = 218,
                 alternative = "t")
```

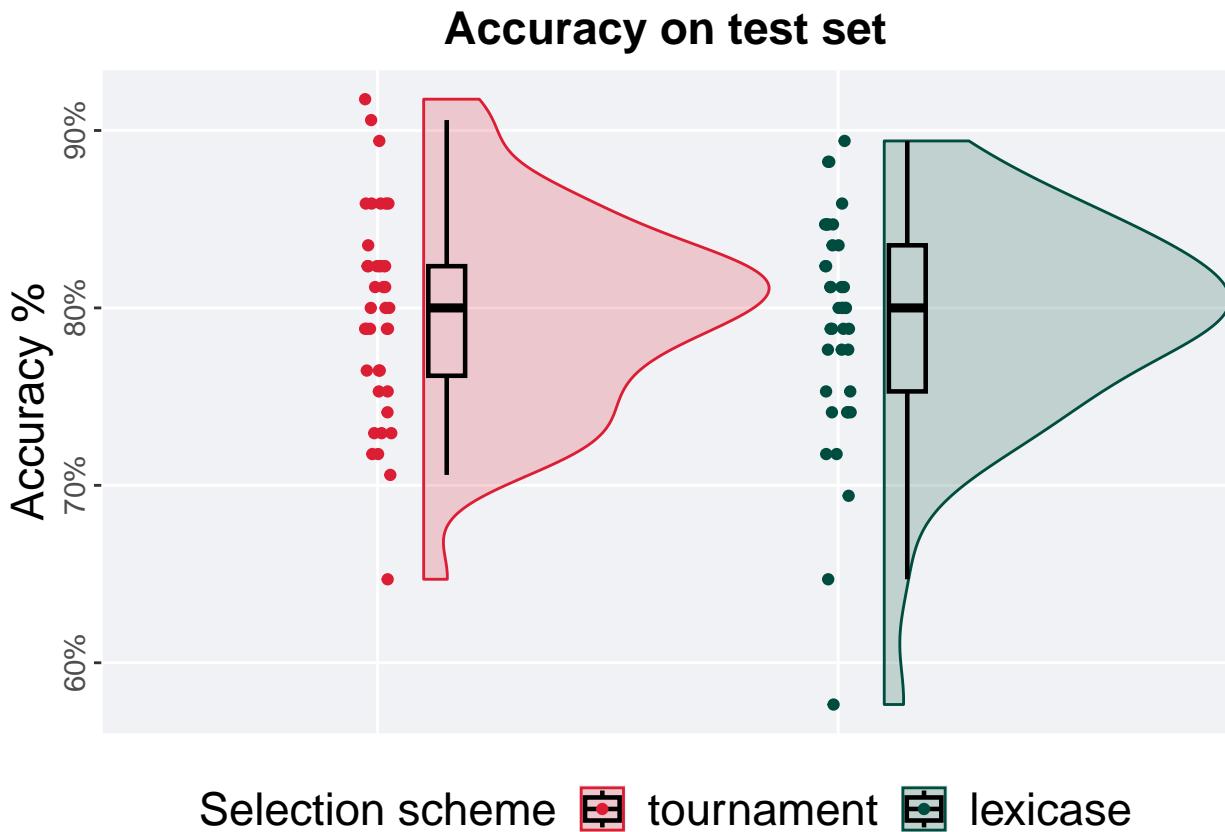
```
## [1] "observed_diff: 1.52147127484176"
## [1] "lower: -1.98927656647669"
## [1] "upper: 1.99192459942517"
## [1] "fail to reject null hypothesis"
## [1] "p-value: 0.13249"
```



6.5 95%

6.5.1 Test accuracy

```
test_plot(filter(task_data, split == '95%'))
```



Summary statistics for the testing performance of the selection schemes at the 5% selection set split:

```
test_results_summary(filter(task_data, split == '95%'))
```

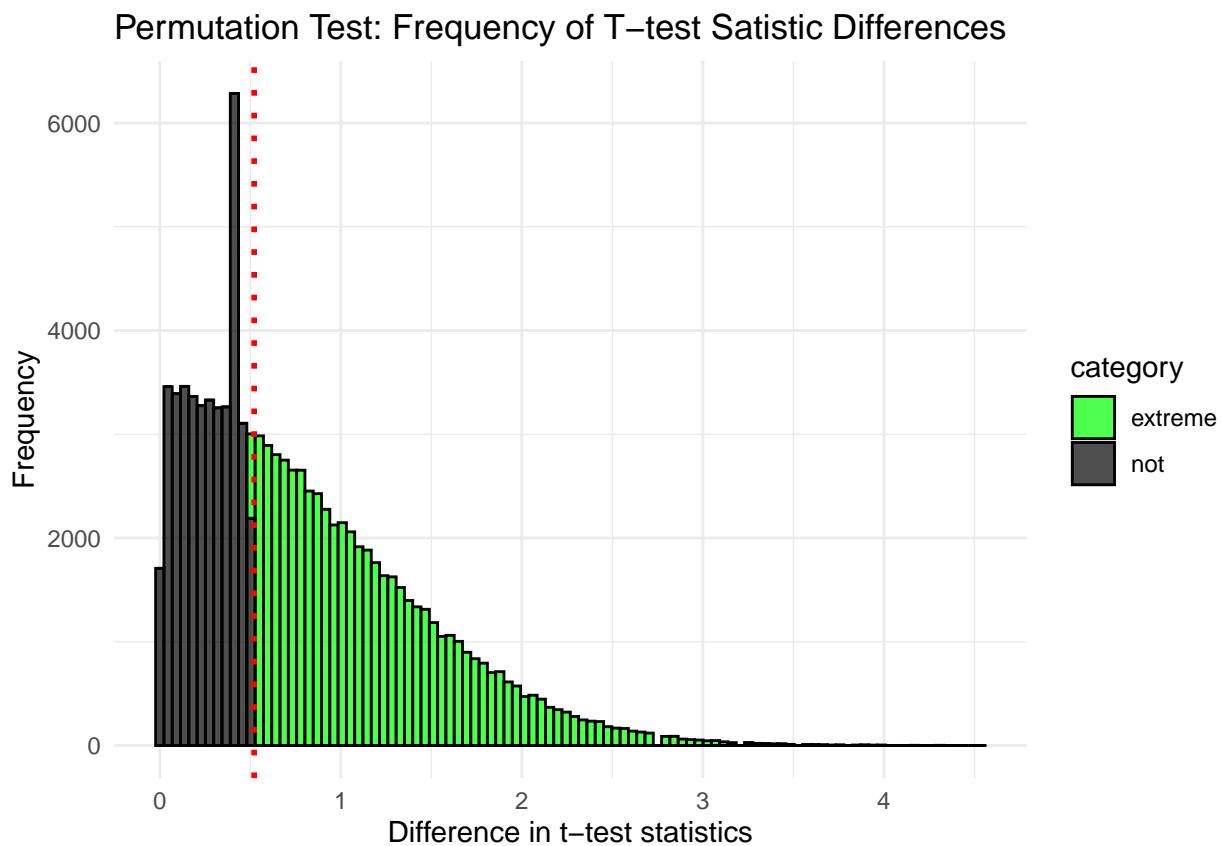
```
## # A tibble: 2 x 8
##   selection count na_cnt  min median  mean   max     IQR
##   <fct>     <int> <int> <dbl>  <dbl> <dbl> <dbl> <dbl>
## 1 tournament     40     0 0.647    0.8 0.798 0.918 0.0618
## 2 lexicase       40     0 0.576    0.8 0.791 0.894 0.0824
```

The permutation test revealed that the results are:

```
tournament_results <- filter(task_data, split == '95%' & selection == 'tournament')
lexicase_results <- filter(task_data, split == '95%' & selection == 'lexicase')
```

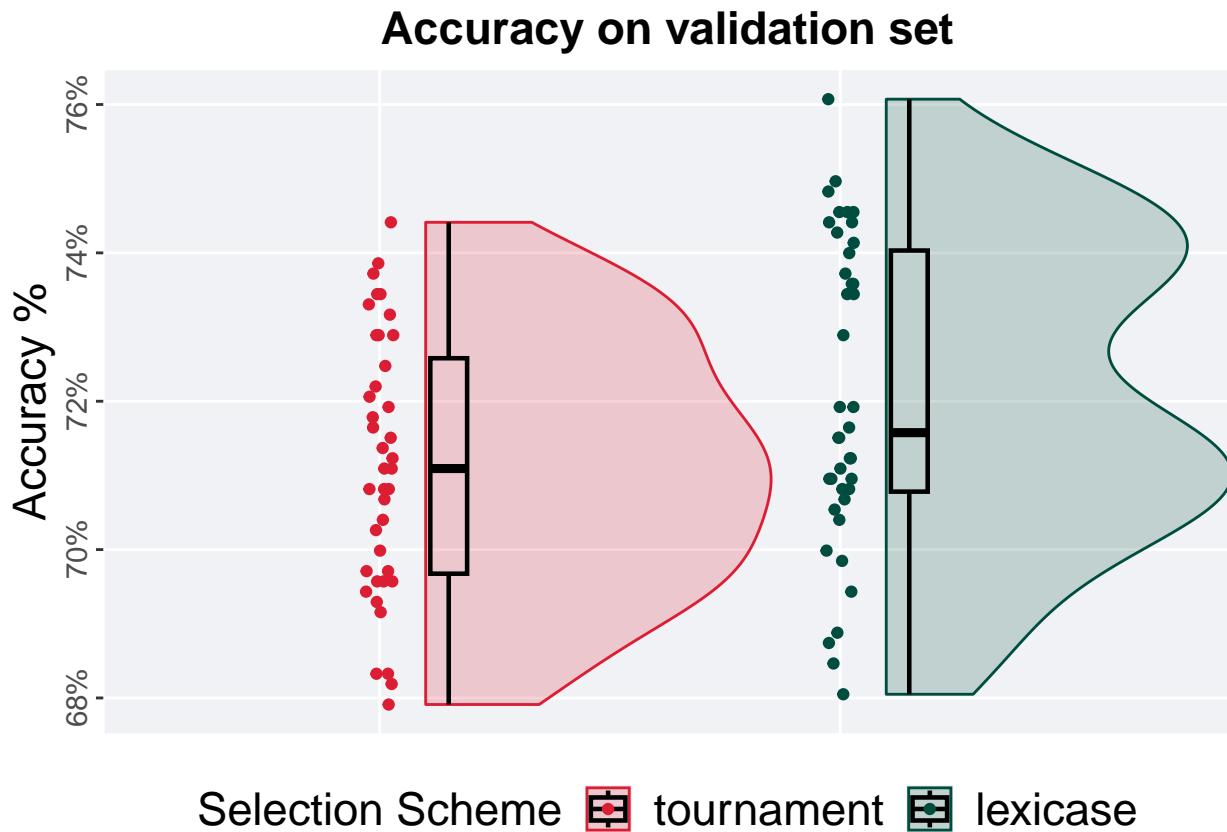
```
permutation_test(tournament_results$testing_performance,
                  lexicase_results$testing_performance,
                  seed = 49,
                  alternative = "t")
```

```
## [1] "observed_diff: 0.521524422177645"
## [1] "lower: -1.95514751807407"
## [1] "upper: 2.00183834772485"
## [1] "fail to reject null hypothesis"
## [1] "p-value: 0.59908"
```



6.5.2 Validation accuracy

```
validation_plot(filter(task_data, split == '95%'))
```



Summary statistics for the testing performance of the selection schemes at the 5% selection set split:

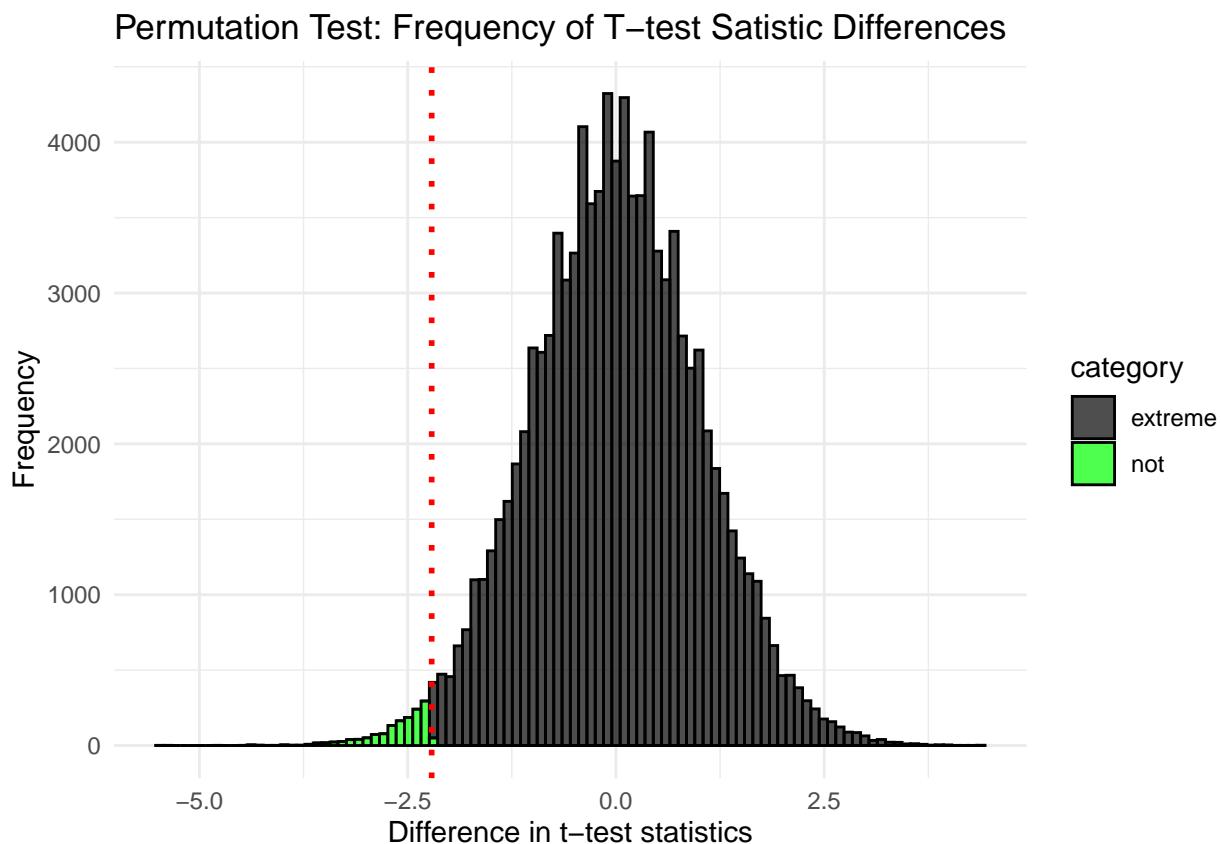
```
validation_accuracy_summary(filter(task_data, split == '95%'))
```

```
## # A tibble: 2 x 8
##   selection  count  na_cnt  min  median  mean  max    IQR
##   <fct>      <int>   <int>  <dbl>   <dbl>  <dbl> <dbl>  <dbl>
## 1 tournament     40      0  0.679  0.711  0.711  0.744  0.0290
## 2 lexicase       40      0  0.680  0.716  0.721  0.761  0.0325
```

The permutation test revealed that the results are:

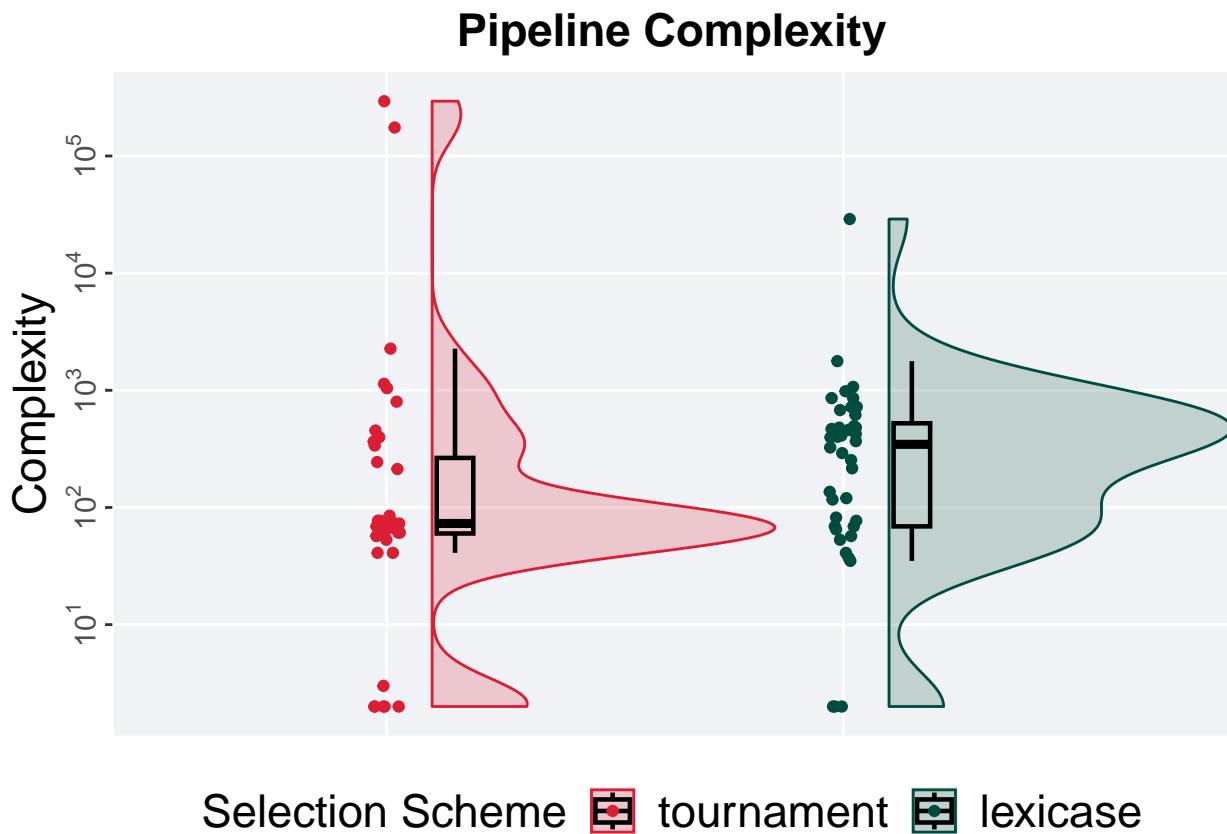
```
tournament_results <- filter(task_data, split == '95%' & selection == 'tournament')
lexicase_results <- filter(task_data, split == '95%' & selection == 'lexicase')
permutation_test(tournament_results$training_performance,
                 lexicase_results$training_performance,
                 seed = 50,
                 alternative = "1")
```

```
## [1] "observed_diff: -2.21192504456209"
## [1] "permutation_diffs[0.05 * n_permutations]: -1.65922504718008"
## [1] "reject null hypothesis"
## [1] "p-value: 0.01463"
```



6.5.3 Pipeline complexity

```
complexity_plot(filter(task_data, split == '95%'))
```



Summary statistics for the pipeline complexity of the selection schemes at the 5% selection set split:

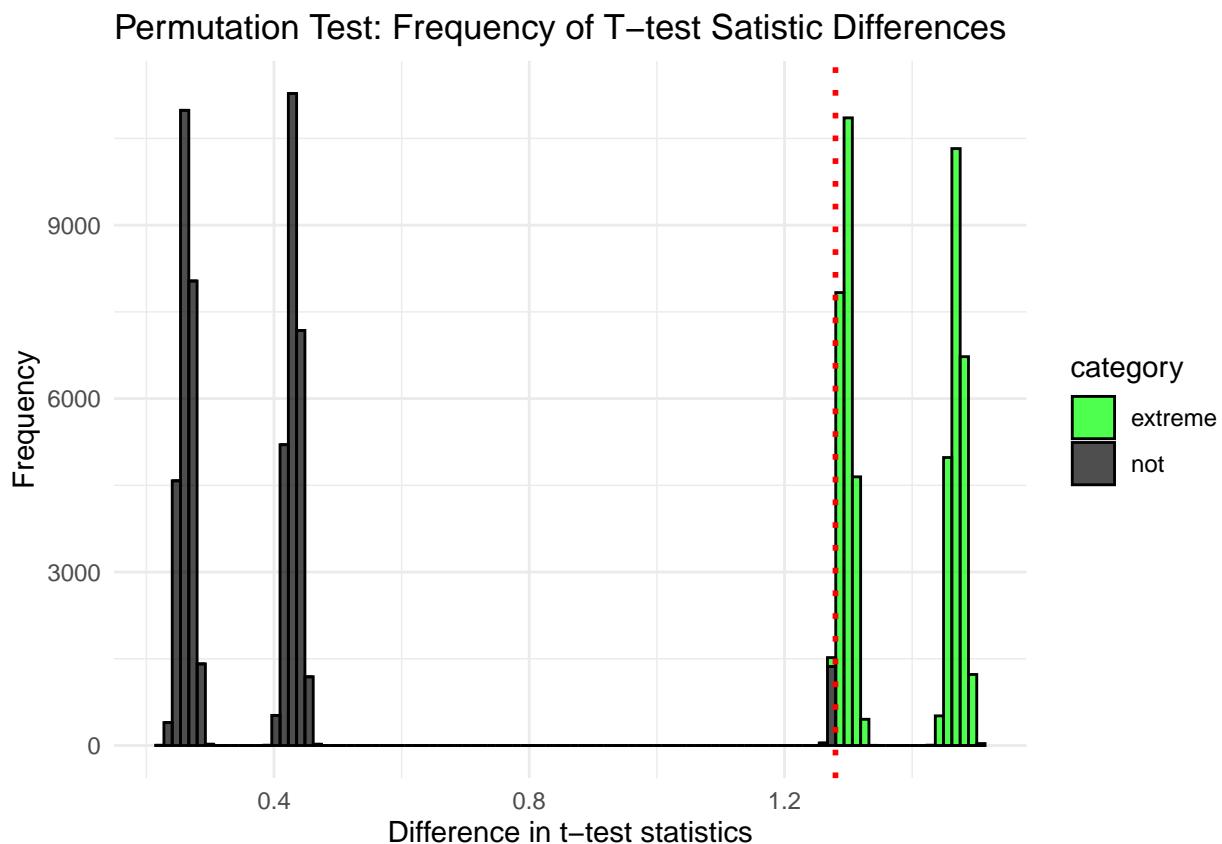
```
complexity_summary(filter(task_data, split == '95%'))
```

```
## # A tibble: 2 x 8
##   selection count na_cnt  min median   mean   max   IQR
##   <fct>     <int> <int> <int> <dbl> <dbl> <int> <dbl>
## 1 tournament     40     0     2     73  11933. 293581  208.
## 2 lexicase       40     0     2    346.  1092.  28966  456.
```

The permutation test revealed that the results are:

```
tournament_results <- filter(task_data, split == '95%' & selection == 'tournament')
lexicase_results <- filter(task_data, split == '95%' & selection == 'lexicase')
permutation_test(tournament_results$testing_complexity,
                 lexicase_results$testing_complexity,
                 seed = 219,
                 alternative = "t")

## [1] "observed_diff: 1.27984938289203"
## [1] "lower: -1.47960907514976"
## [1] "upper: 1.47962734375373"
## [1] "fail to reject null hypothesis"
## [1] "p-value: 0.47761"
```



Chapter 7

Task 168757

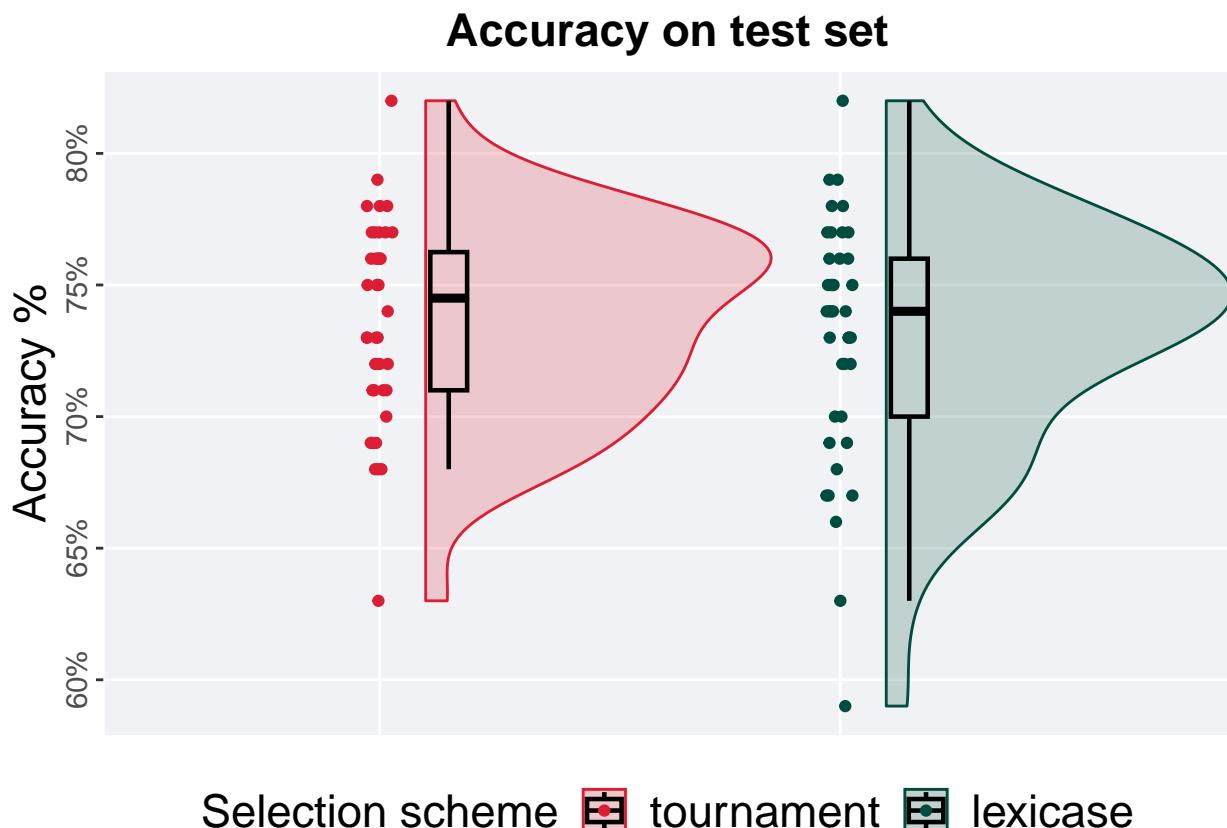
We present the results of our analysis of task 168757 with the different selection set splits used in our study.

```
task_data <- filter(results, task_id == 168757)
```

7.1 5%

7.1.1 Test accuracy

```
test_plot(filter(task_data, split == '5%'))
```



Summary statistics for the testing performance of the selection schemes at the 5% selection set split:

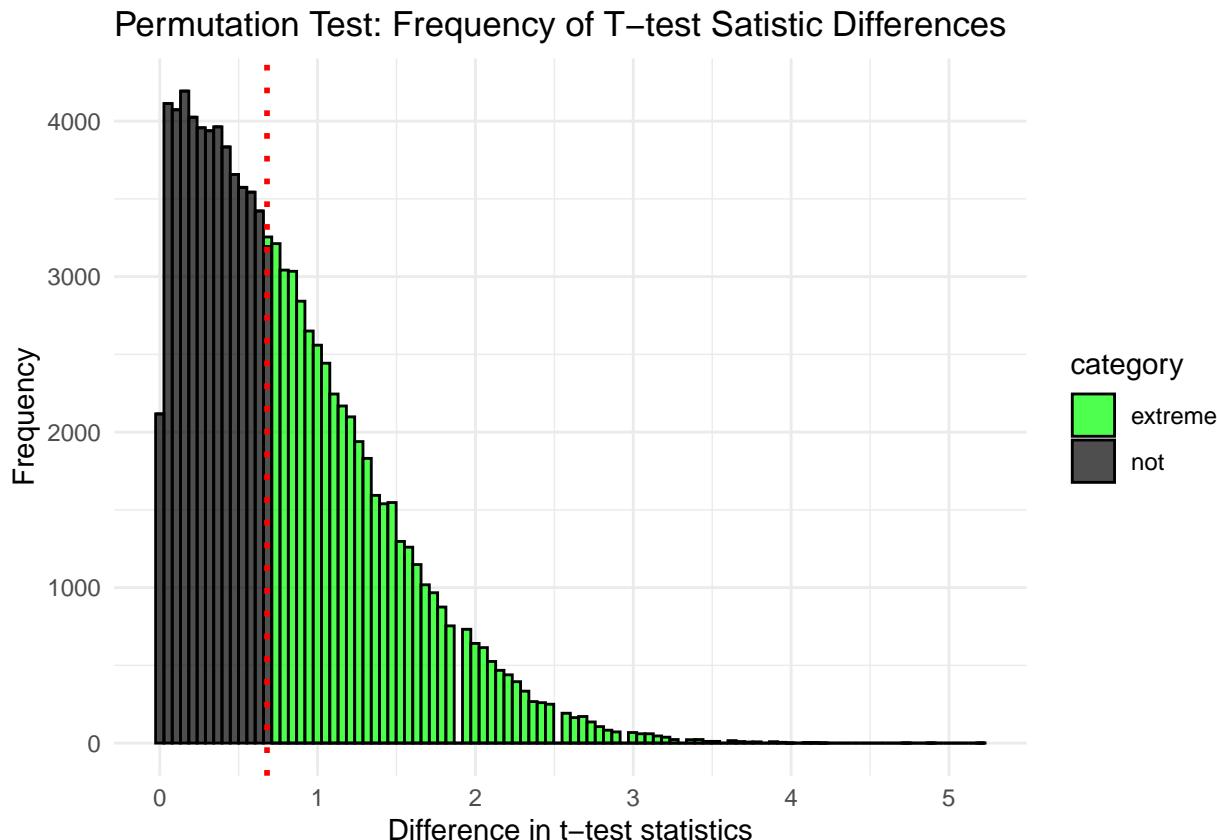
```
test_results_summary(filter(task_data, split == '5%'))
```

```
## # A tibble: 2 x 8
##   selection count na_cnt  min median  mean   max   IQR
##   <fct>     <int> <int> <dbl> <dbl> <dbl> <dbl> <dbl>
## 1 tournament     40     0  0.63  0.745  0.736  0.82  0.0525
## 2 lexicase       40     0  0.59  0.74   0.73   0.82  0.0600
```

The permutation test revealed that the results are:

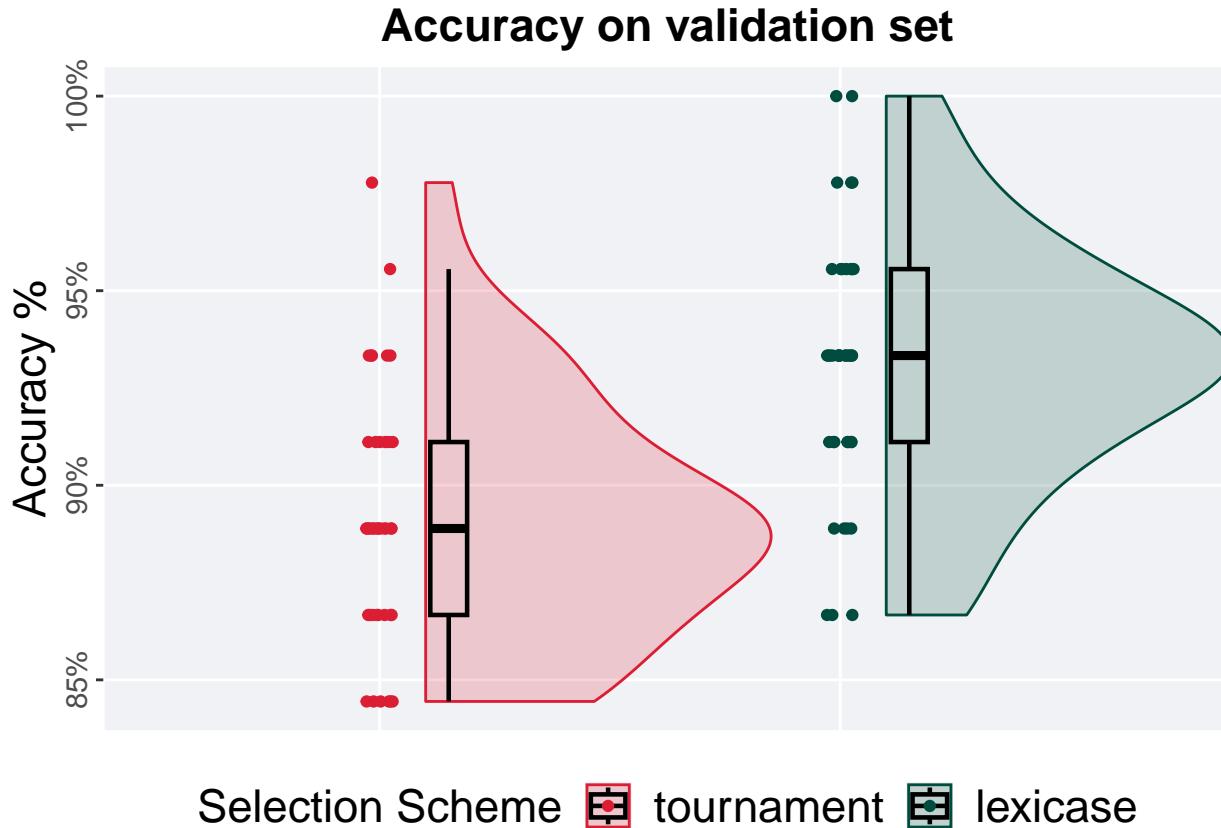
```
tournament_results <- filter(task_data, split == '5%' & selection == 'tournament')
lexicase_results <- filter(task_data, split == '5%' & selection == 'lexicase')
permutation_test(tournament_results$testing_performance,
                  lexicase_results$testing_performance,
                  seed = 51,
                  alternative = "t")
```

```
## [1] "observed_diff: 0.679592305041863"
## [1] "lower: -1.97621361006578"
## [1] "upper: 1.97621361006577"
## [1] "fail to reject null hypothesis"
## [1] "p-value: 0.48399"
```



7.1.2 Validation accuracy

```
validation_plot(filter(task_data, split == '5%'))
```



Summary statistics for the testing performance of the selection schemes at the 5% selection set split:

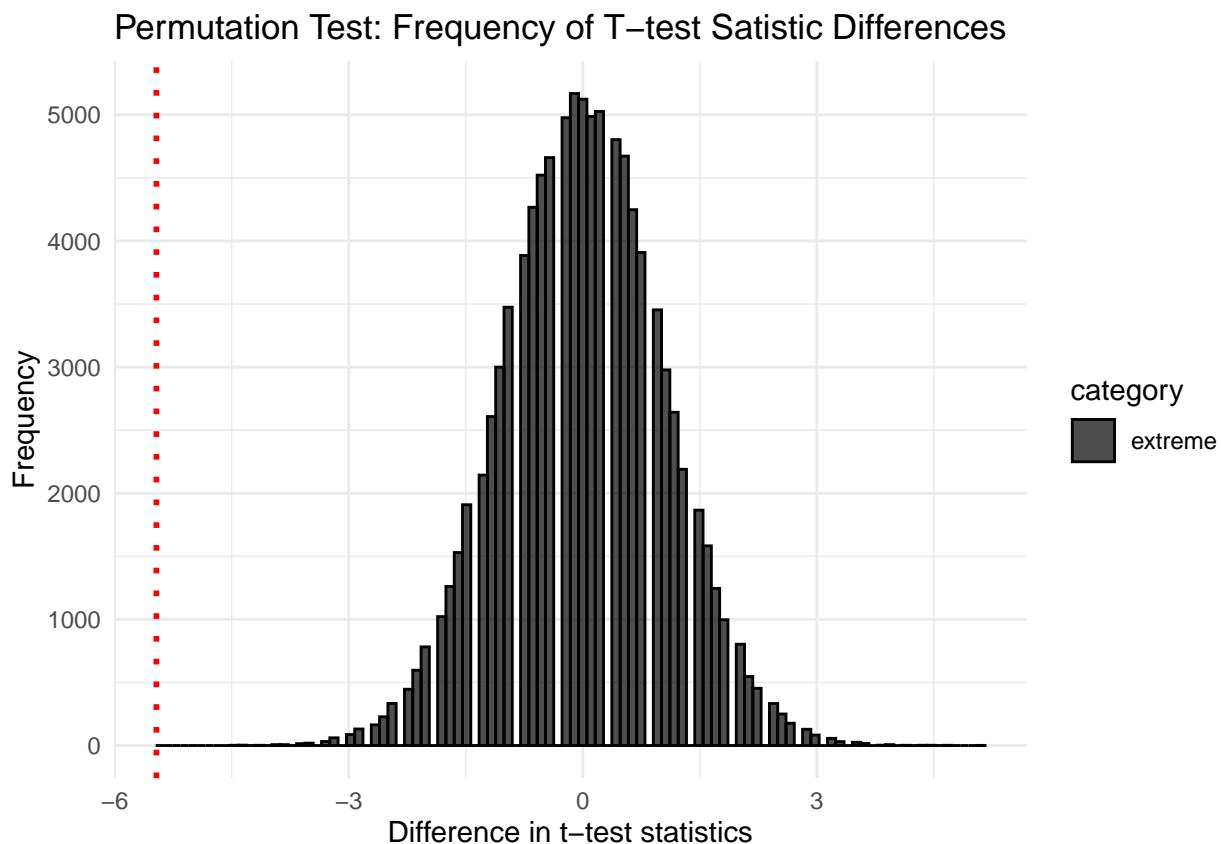
```
validation_accuracy_summary(filter(task_data, split == '5%'))
```

```
## # A tibble: 2 x 8
##   selection count na_cnt  min median  mean   max     IQR
##   <fct>     <int>  <int> <dbl>  <dbl>  <dbl> <dbl>  <dbl>
## 1 tournament     40      0 0.844  0.889  0.891  0.978  0.0444
## 2 lexicase       40      0 0.867  0.933  0.931  1       0.0444
```

The permutation test revealed that the results are:

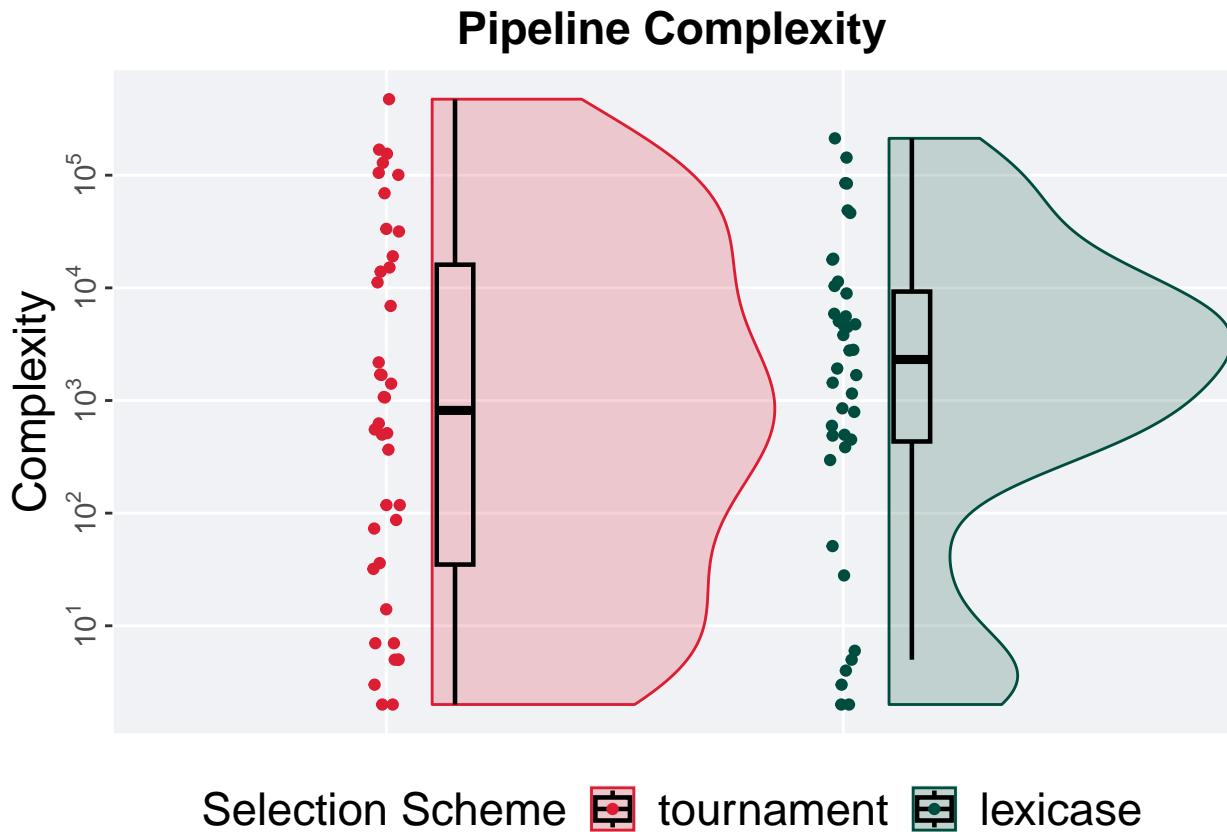
```
tournament_results <- filter(task_data, split == '5%' & selection == 'tournament')
lexicase_results <- filter(task_data, split == '5%' & selection == 'lexicase')
permutation_test(tournament_results$training_performance,
                 lexicase_results$training_performance,
                 seed = 52,
                 alternative = "1")
```

```
## [1] "observed_diff: -5.46636715299929"
## [1] "permutation_diffs[0.05 * n_permutations]: -1.70963063314534"
## [1] "reject null hypothesis"
## [1] "p-value: 1e-05"
```



7.1.3 Pipeline complexity

```
complexity_plot(filter(task_data, split == '5%'))
```



Summary statistics for the pipeline complexity of the selection schemes at the 5% selection set split:

```
complexity_summary(filter(task_data, split == '5%'))
```

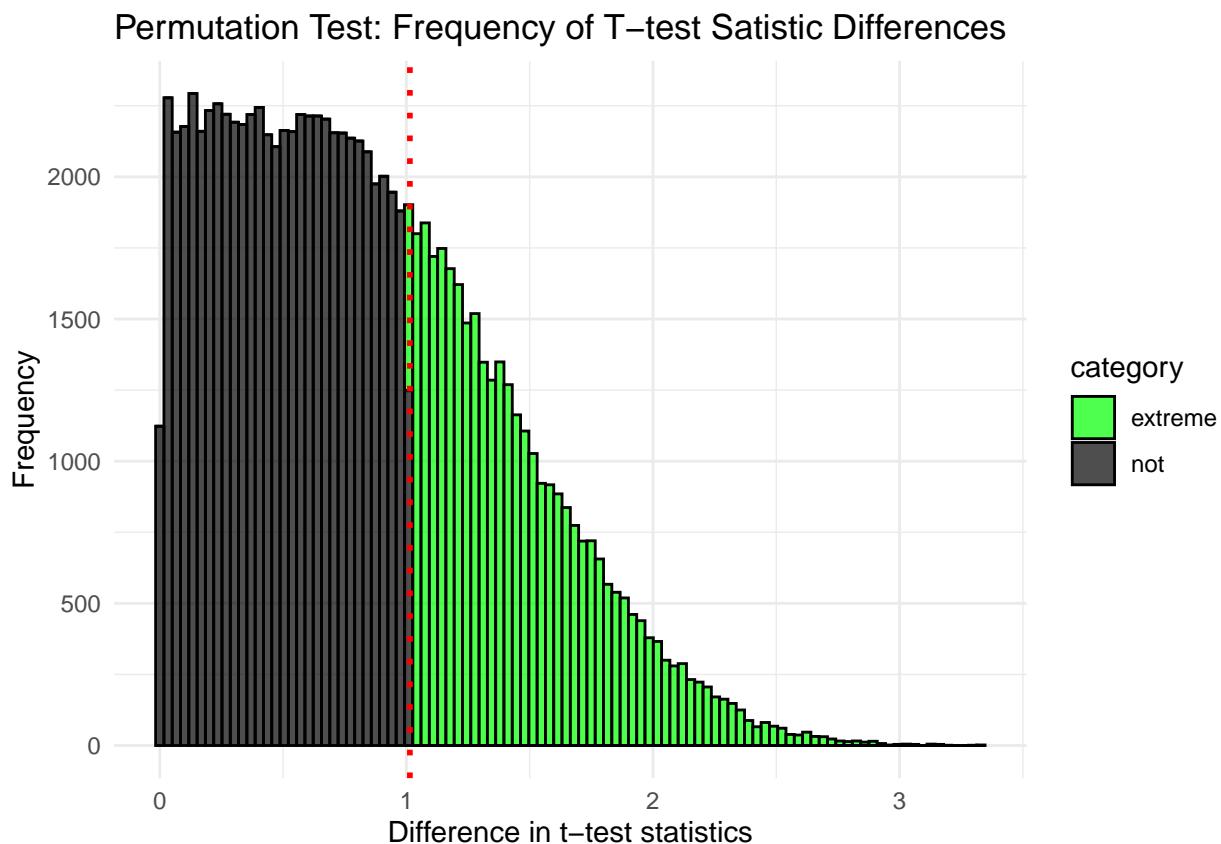
```
## # A tibble: 2 x 8
##   selection count na_cnt  min median   mean   max   IQR
##   <fct>     <int> <int> <dbl> <dbl> <dbl> <int> <dbl>
## 1 tournament     40     0     2  846. 33603. 473071 16106
## 2 lexicase       40     0     2 2352  18433. 212511  8858.
```

The permutation test revealed that the results are:

```
tournament_results <- filter(task_data, split == '5%' & selection == 'tournament')
lexicase_results <- filter(task_data, split == '5%' & selection == 'lexicase')
```

```
permutation_test(tournament_results$testing_complexity,
                 lexicase_results$testing_complexity,
                 seed = 218,
                 alternative = "t")
```

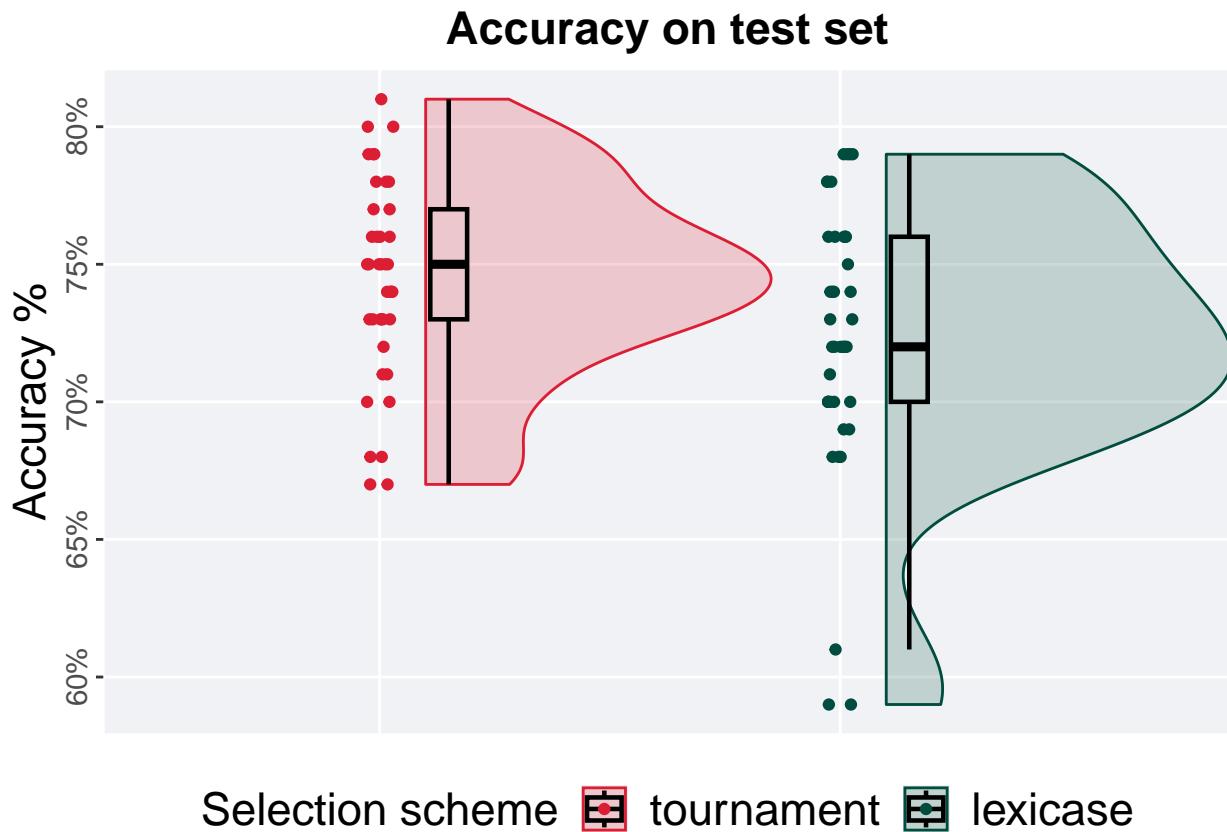
```
## [1] "observed_diff: 1.0143239643335"
## [1] "lower: -1.8618441853886"
## [1] "upper: 1.86870922733726"
## [1] "fail to reject null hypothesis"
## [1] "p-value: 0.35127"
```



7.2 10%

7.2.1 Test accuracy

```
test_plot(filter(task_data, split == '10%'))
```



Summary statistics for the testing performance of the selection schemes at the 5% selection set split:

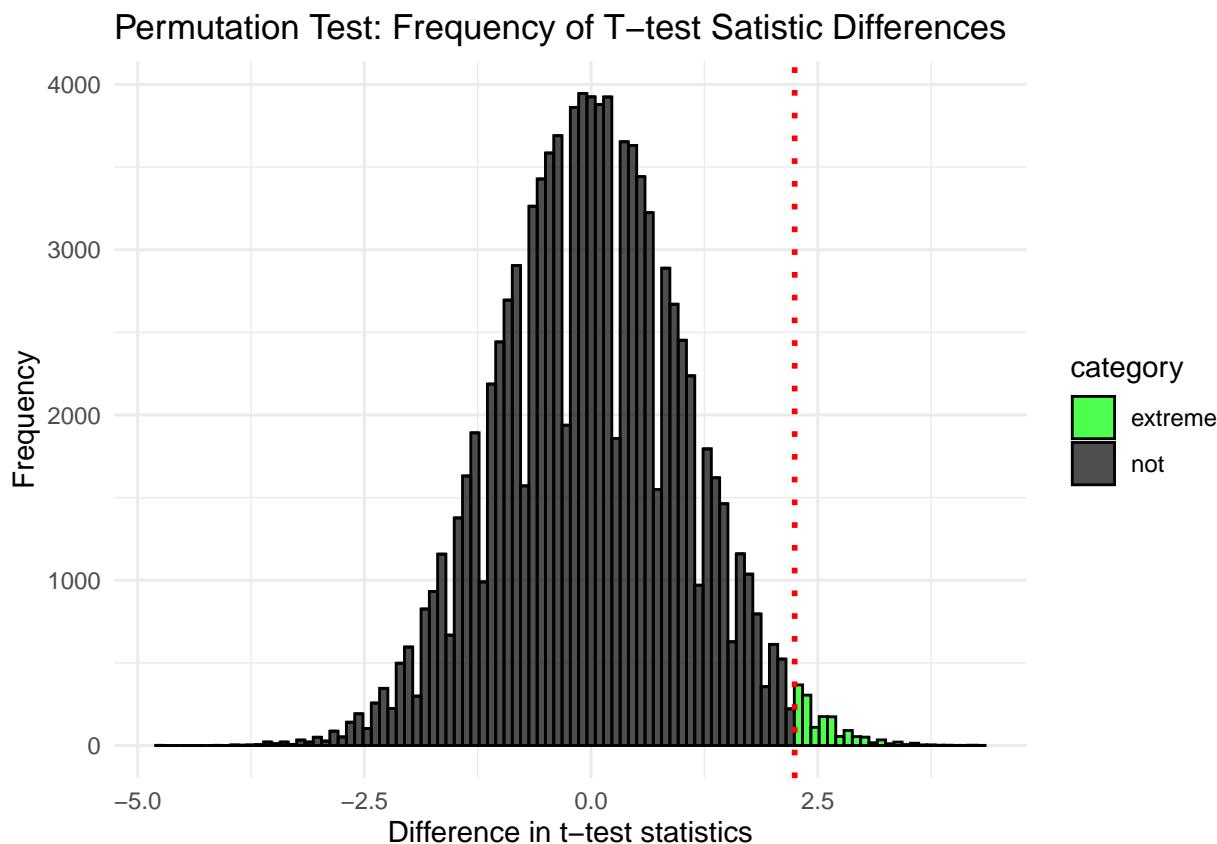
```
test_results_summary(filter(task_data, split == '10%'))
```

```
## # A tibble: 2 x 8
##   selection  count  na_cnt  min  median  mean  max  IQR
##   <fct>      <int>   <int> <dbl>  <dbl> <dbl> <dbl> <dbl>
## 1 tournament     40      0  0.67  0.754 0.744 0.81  0.0400
## 2 lexicase       40      0  0.59  0.720 0.723 0.79  0.0600
```

The permutation test revealed that the results are:

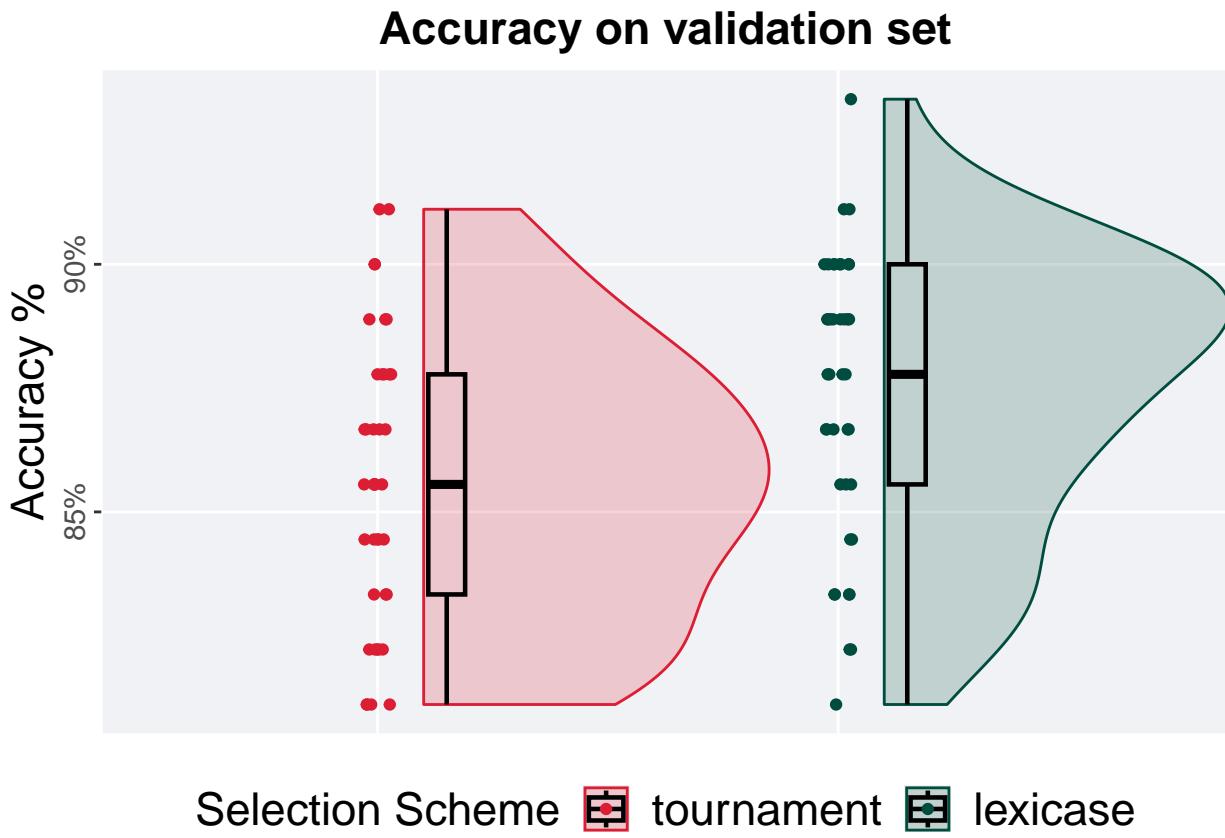
```
tournament_results <- filter(task_data, split == '10%' & selection == 'tournament')
lexicase_results <- filter(task_data, split == '10%' & selection == 'lexicase')
permutation_test(tournament_results$testing_performance,
                 lexicase_results$testing_performance,
                 seed = 53,
                 alternative = "g")
```

```
## [1] "observed_diff: 2.24527511755549"
## [1] "permutation_diffs[0.95 * n_permutations]: 1.70682083833002"
## [1] "reject null hypothesis"
## [1] "p-value: 0.01398"
```



7.2.2 Validation accuracy

```
validation_plot(filter(task_data, split == '10%'))
```



Summary statistics for the testing performance of the selection schemes at the 5% selection set split:

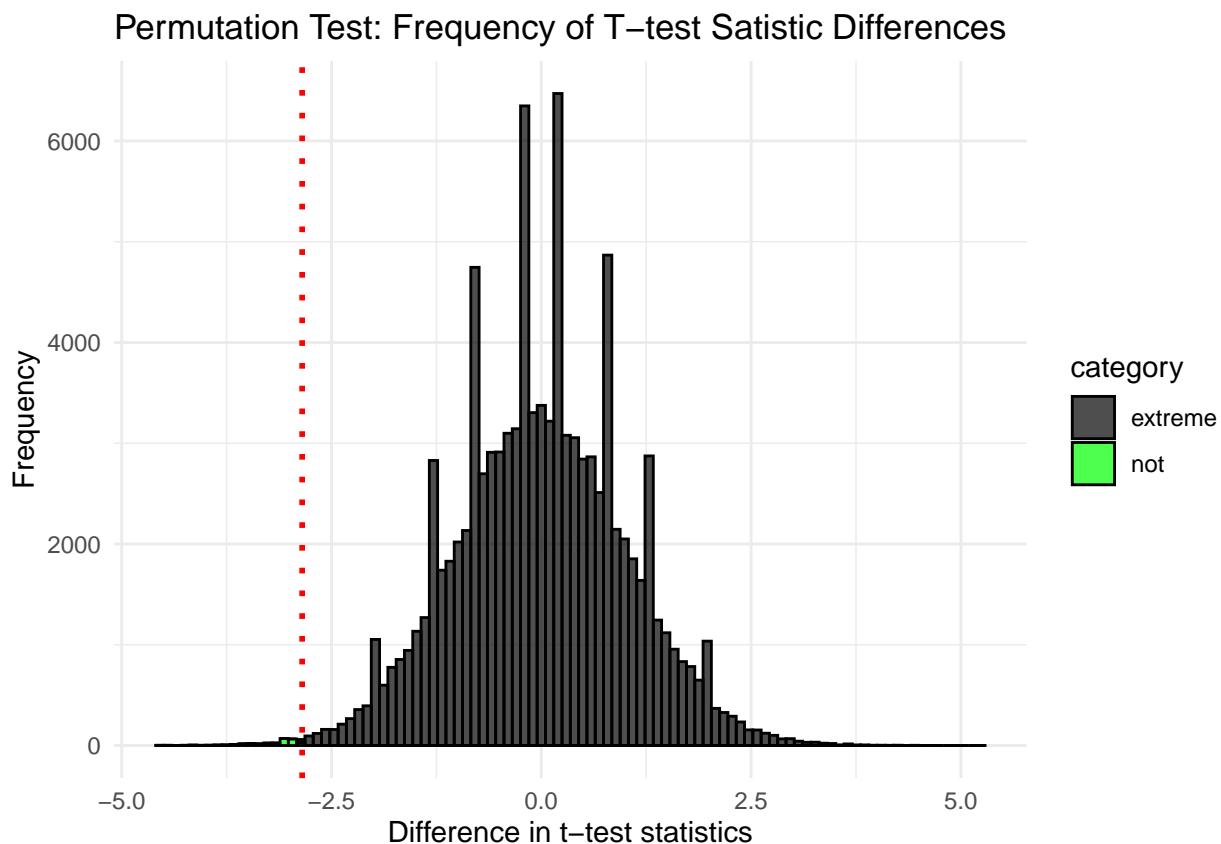
```
validation_accuracy_summary(filter(task_data, split == '10%'))
```

```
## # A tibble: 2 x 8
##   selection  count  na_cnt  min  median  mean   max     IQR
##   <fct>      <int>  <int>  <dbl>  <dbl>  <dbl>  <dbl>  <dbl>
## 1 tournament    40      0  0.811  0.856  0.856  0.911  0.0444
## 2 lexicase      40      0  0.811  0.878  0.874  0.933  0.0444
```

The permutation test revealed that the results are:

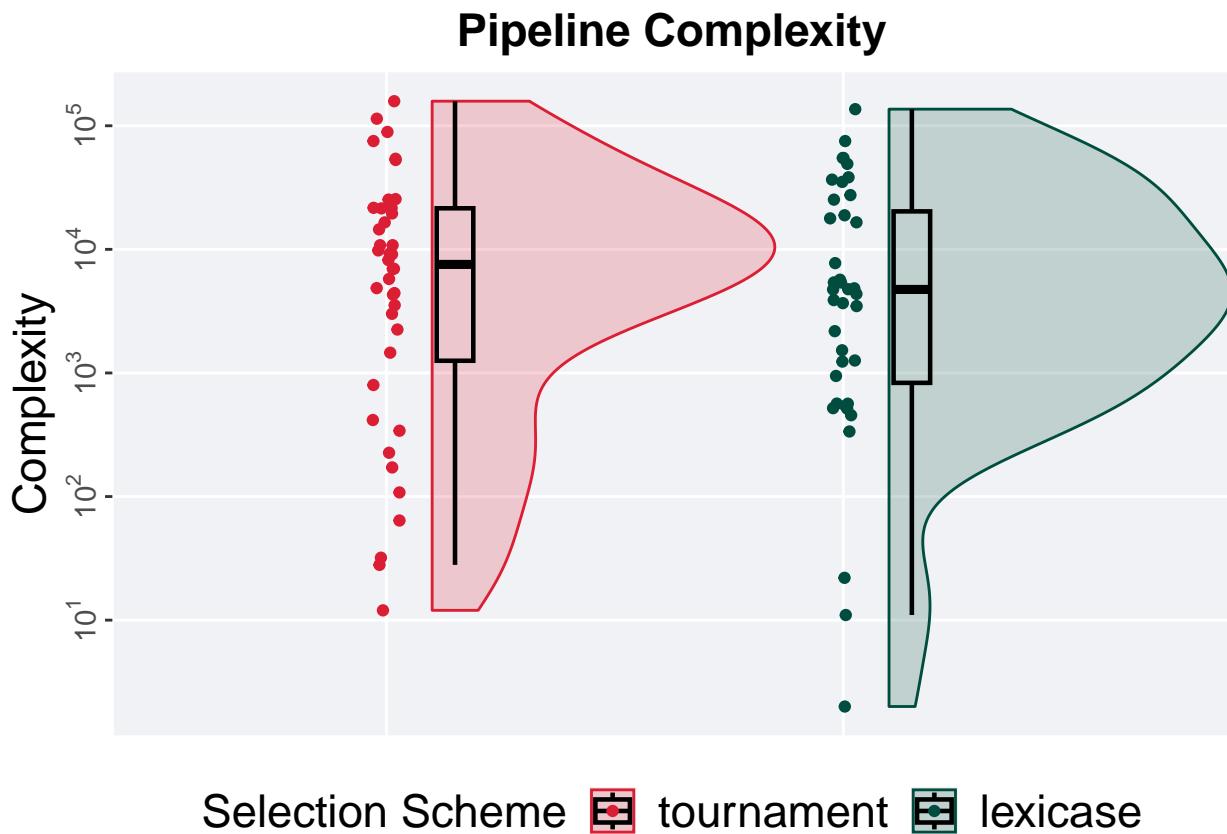
```
tournament_results <- filter(task_data, split == '10%' & selection == 'tournament')
lexicase_results <- filter(task_data, split == '10%' & selection == 'lexicase')
permutation_test(tournament_results$training_performance,
                 lexicase_results$training_performance,
                 seed = 54,
                 alternative = "1")
```

```
## [1] "observed_diff: -2.8494031342686"
## [1] "permutation_diffs[0.05 * n_permutations]: -1.67270885820756"
## [1] "reject null hypothesis"
## [1] "p-value: 0.00303"
```



7.2.3 Pipeline complexity

```
complexity_plot(filter(task_data, split == '10%'))
```



Summary statistics for the pipeline complexity of the selection schemes at the 5% selection set split:

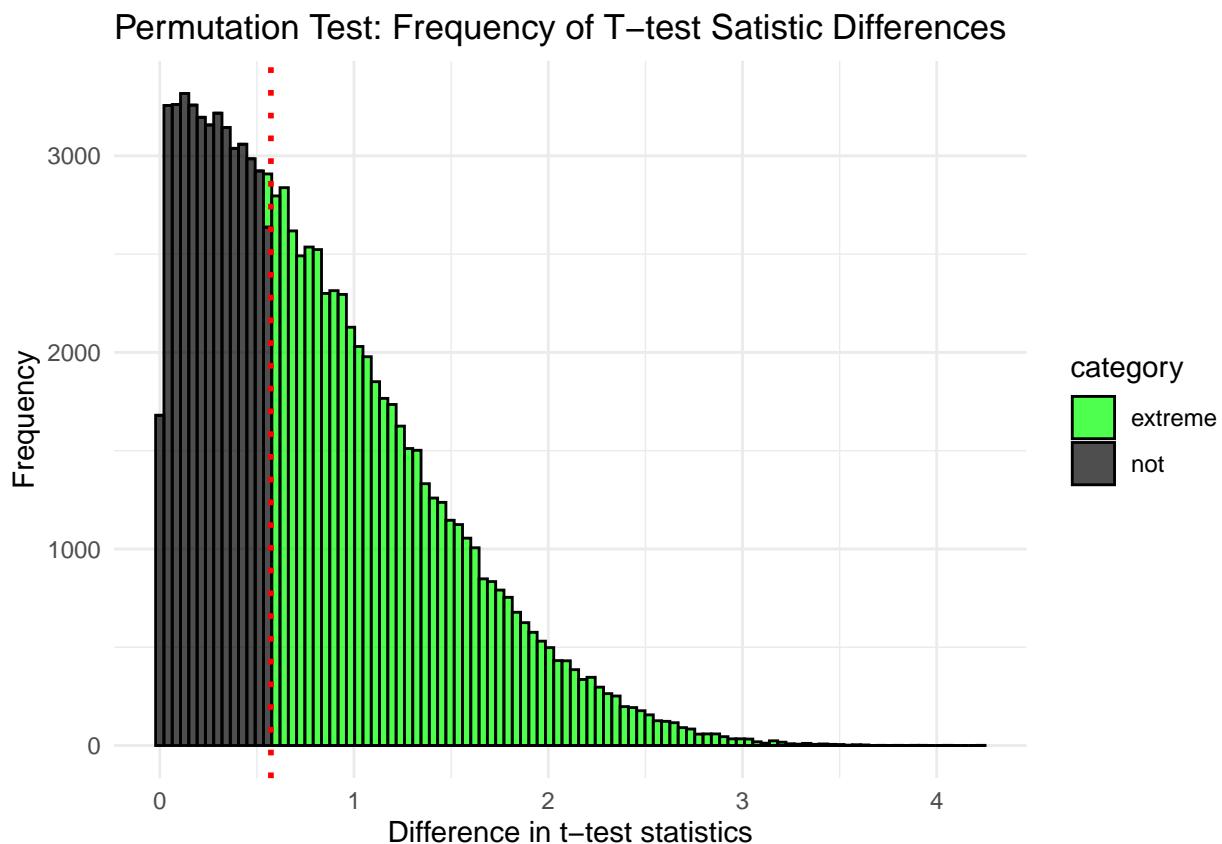
```
complexity_summary(filter(task_data, split == '10%'))
```

```
## # A tibble: 2 x 8
##   selection count na_cnt  min median  mean    max    IQR
##   <fct>     <int> <int> <dbl> <dbl> <dbl> <int> <dbl>
## 1 tournament     40     0    12  7578 20332. 158091 20206.
## 2 lexicase       40     0     2  4739 16398. 136281 19618.
```

The permutation test revealed that the results are:

```
tournament_results <- filter(task_data, split == '10%' & selection == 'tournament')
lexicase_results <- filter(task_data, split == '10%' & selection == 'lexicase')
permutation_test(tournament_results$testing_complexity,
                 lexicase_results$testing_complexity,
                 seed = 219,
                 alternative = "t")
```

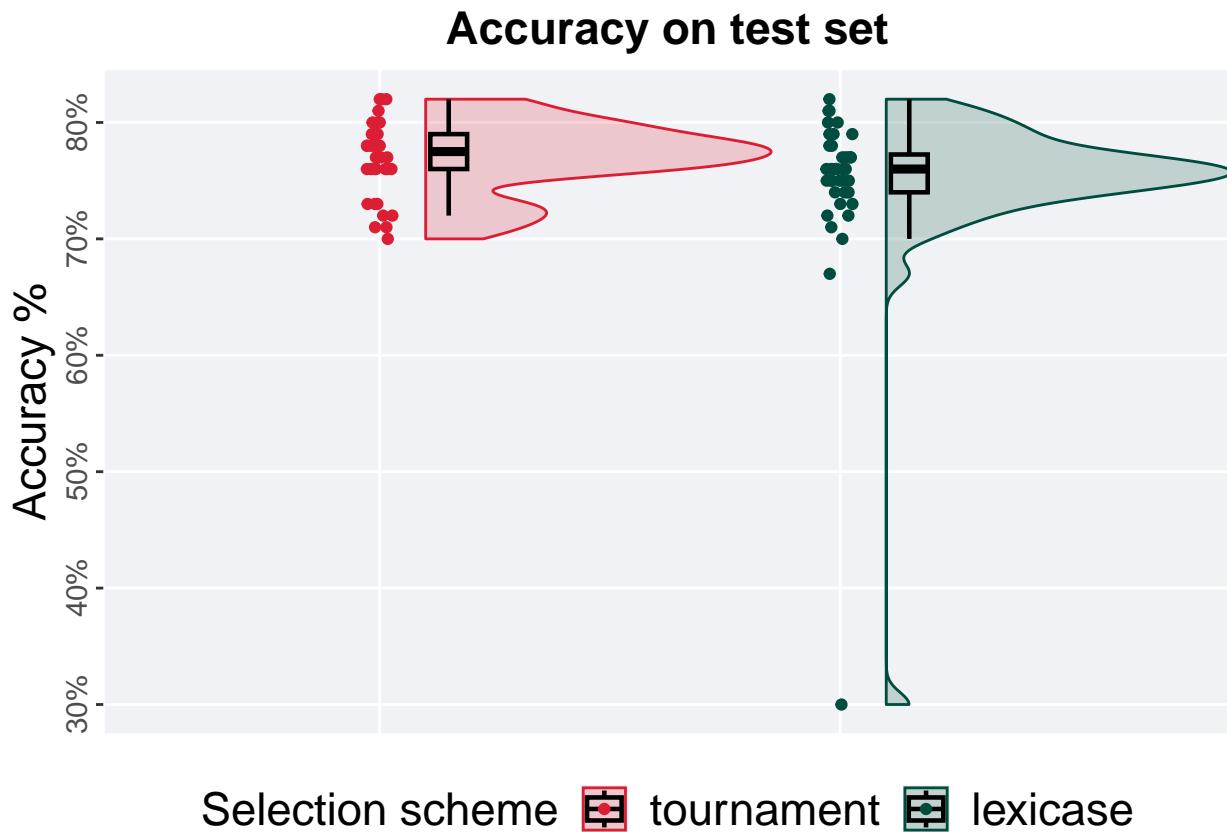
```
## [1] "observed_diff: 0.572258805899218"
## [1] "lower: -2.00170394256905"
## [1] "upper: 1.96814430290368"
## [1] "fail to reject null hypothesis"
## [1] "p-value: 0.57872"
```



7.3 50%

7.3.1 Test accuracy

```
test_plot(filter(task_data, split == '50%'))
```



Summary statistics for the testing performance of the selection schemes at the 5% selection set split:

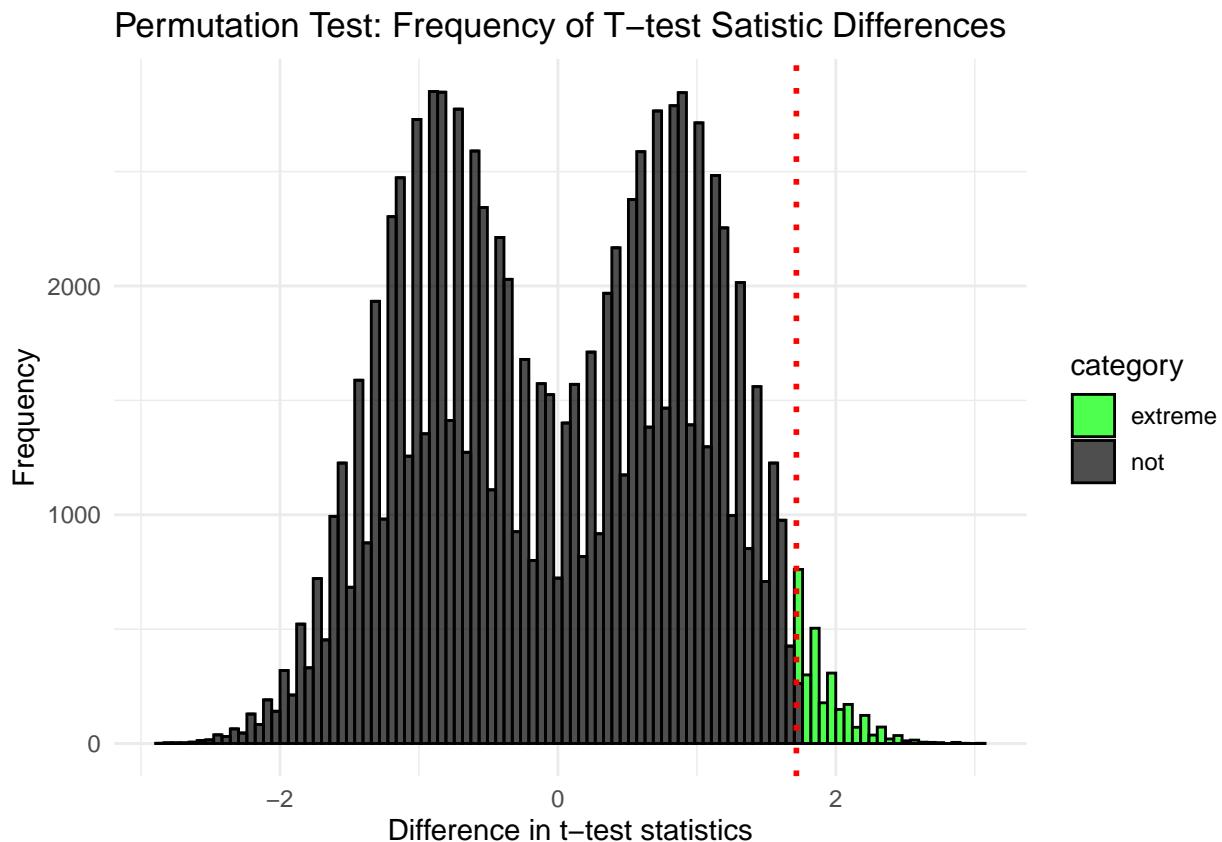
```
test_results_summary(filter(task_data, split == '50%'))
```

```
## # A tibble: 2 x 8
##   selection  count na_cnt  min median  mean  max    IQR
##   <fct>      <int> <int> <dbl>  <dbl> <dbl> <dbl> <dbl>
## 1 tournament    40     0    0.7  0.775  0.77   0.82  0.0300
## 2 lexicase     40     0    0.3  0.76   0.747  0.82  0.0325
```

The permutation test revealed that the results are:

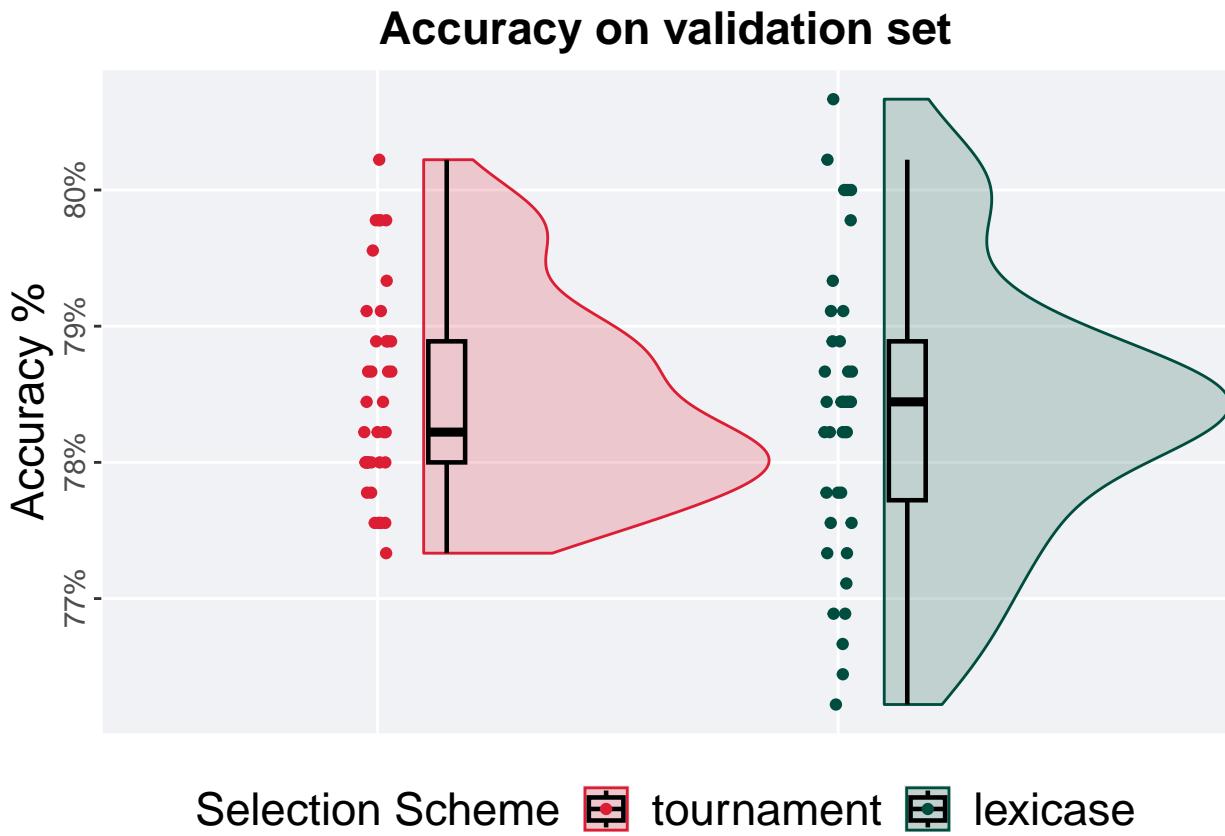
```
tournament_results <- filter(task_data, split == '50%' & selection == 'tournament')
lexicase_results <- filter(task_data, split == '50%' & selection == 'lexicase')
permutation_test(tournament_results$testing_performance,
                 lexicase_results$testing_performance,
                 seed = 55,
                 alternative = "g")
```

```
## [1] "observed_diff: 1.71603077853694"
## [1] "permutation_diffs[0.95 * n_permutations]: 1.52360461800033"
## [1] "reject null hypothesis"
## [1] "p-value: 0.02379"
```



7.3.2 Validation accuracy

```
validation_plot(filter(task_data, split == '50%'))
```



Summary statistics for the testing performance of the selection schemes at the 5% selection set split:

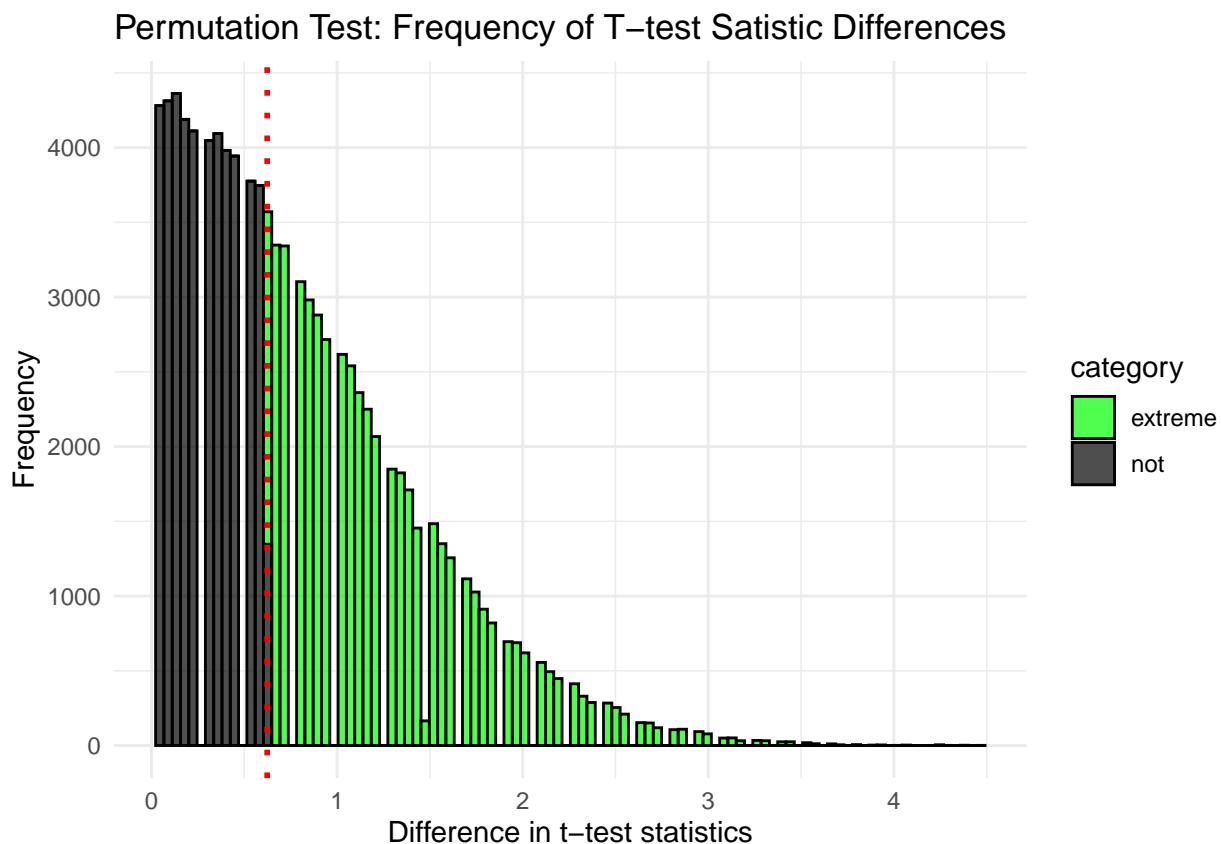
```
validation_accuracy_summary(filter(task_data, split == '50%'))
```

```
## # A tibble: 2 x 8
##   selection count na_cnt  min median  mean   max      IQR
##   <fct>     <int> <int> <dbl> <dbl> <dbl> <dbl> <dbl>
## 1 tournament     40     0 0.773 0.782 0.785 0.802 0.00889
## 2 lexicase       40     0 0.762 0.784 0.784 0.807 0.0117
```

The permutation test revealed that the results are:

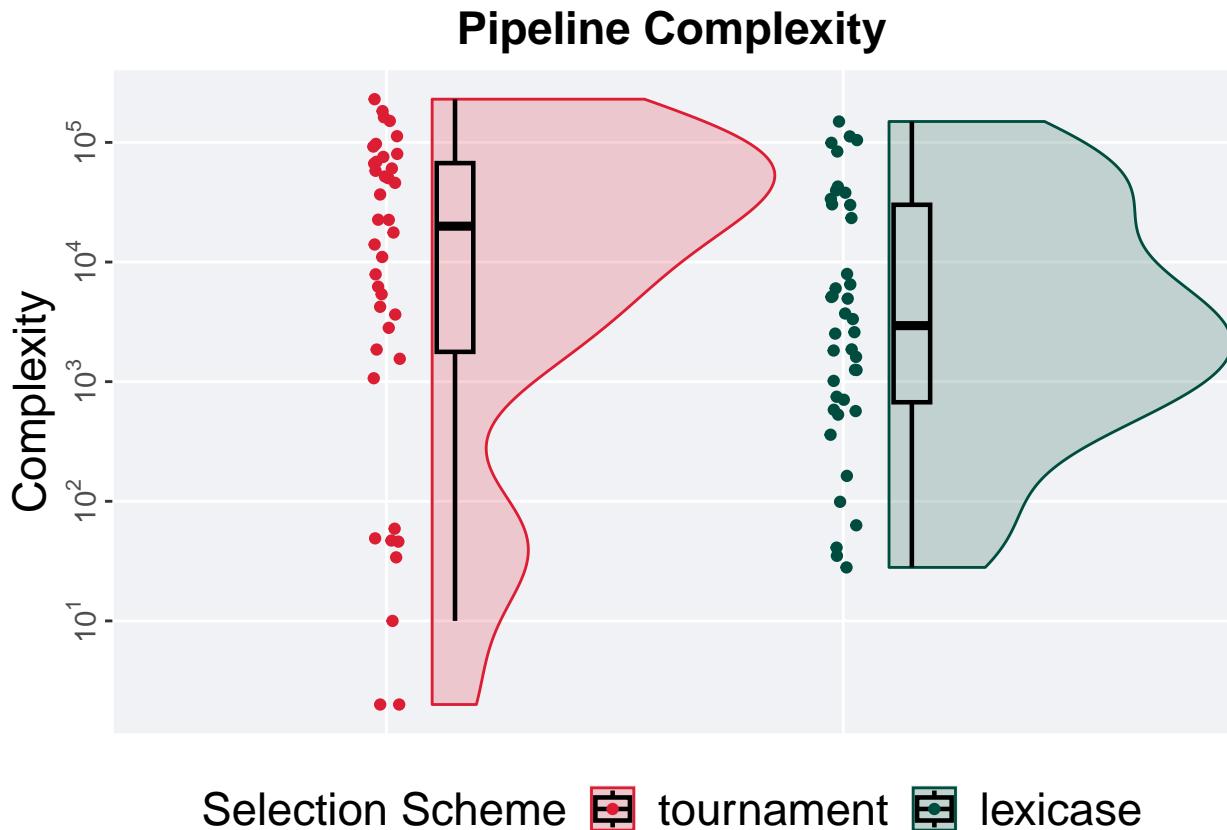
```
tournament_results <- filter(task_data, split == '50%' & selection == 'tournament')
lexicase_results <- filter(task_data, split == '50%' & selection == 'lexicase')
permutation_test(tournament_results$training_performance,
                 lexicase_results$training_performance,
                 seed = 56,
                 alternative = "t")
```

```
## [1] "observed_diff: 0.623468562343347"
## [1] "lower: -2.02515196524075"
## [1] "upper: 2.02515196524075"
## [1] "fail to reject null hypothesis"
## [1] "p-value: 0.53808"
```



7.3.3 Pipeline complexity

```
complexity_plot(filter(task_data, split == '50%'))
```



Summary statistics for the pipeline complexity of the selection schemes at the 5% selection set split:

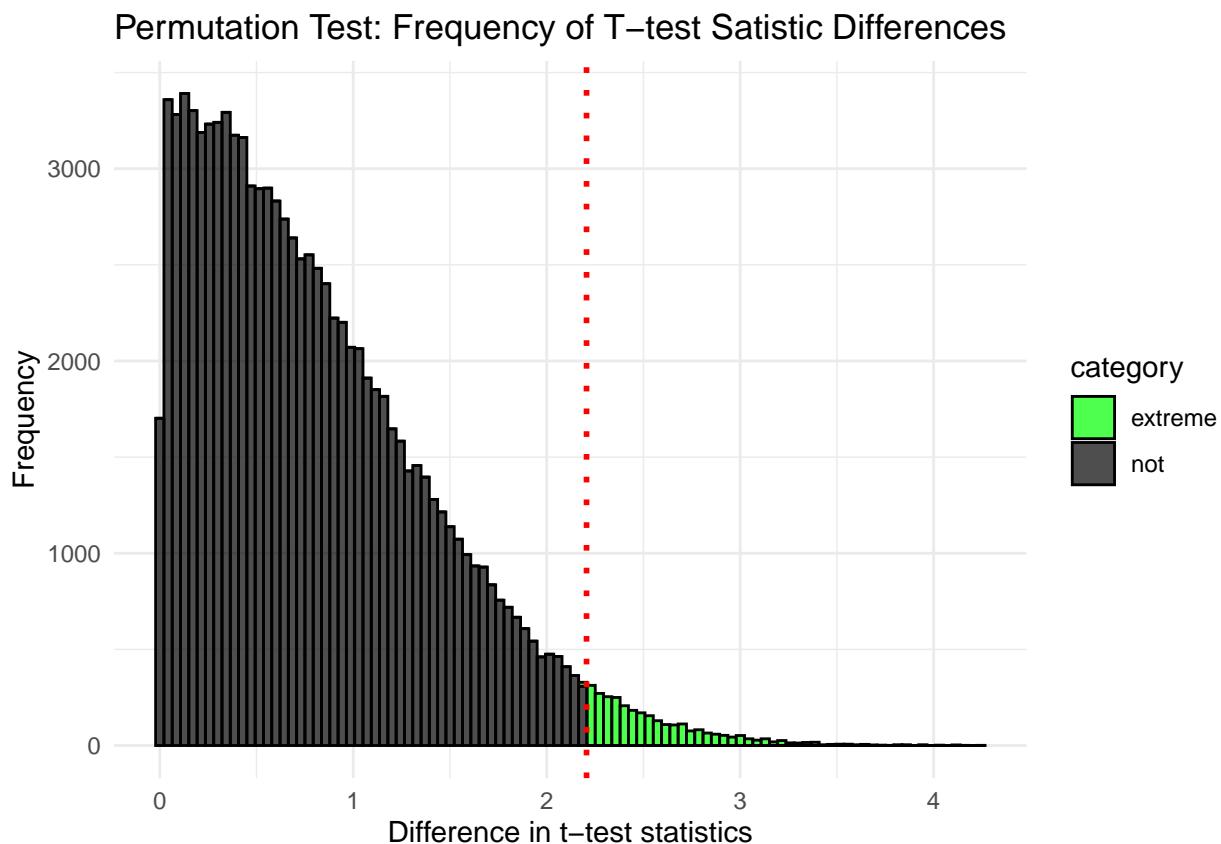
```
complexity_summary(filter(task_data, split == '50'))
```

```
## # A tibble: 2 x 8
##   selection count na_cnt  min median   mean   max   IQR
##   <fct>     <int>  <int> <dbl>  <dbl>  <dbl>  <dbl>
## 1 tournament     40      0     2 20086  44914. 229791 65358.
## 2 lexicase       40      0    28 2964.  21201. 149551 29425.
```

The permutation test revealed that the results are:

```
tournament_results <- filter(task_data, split == '50' & selection == 'tournament')
lexicase_results <- filter(task_data, split == '50' & selection == 'lexicase')
permutation_test(tournament_results$testing_complexity,
                 lexicase_results$testing_complexity,
                 seed = 220,
                 alternative = "t")
```

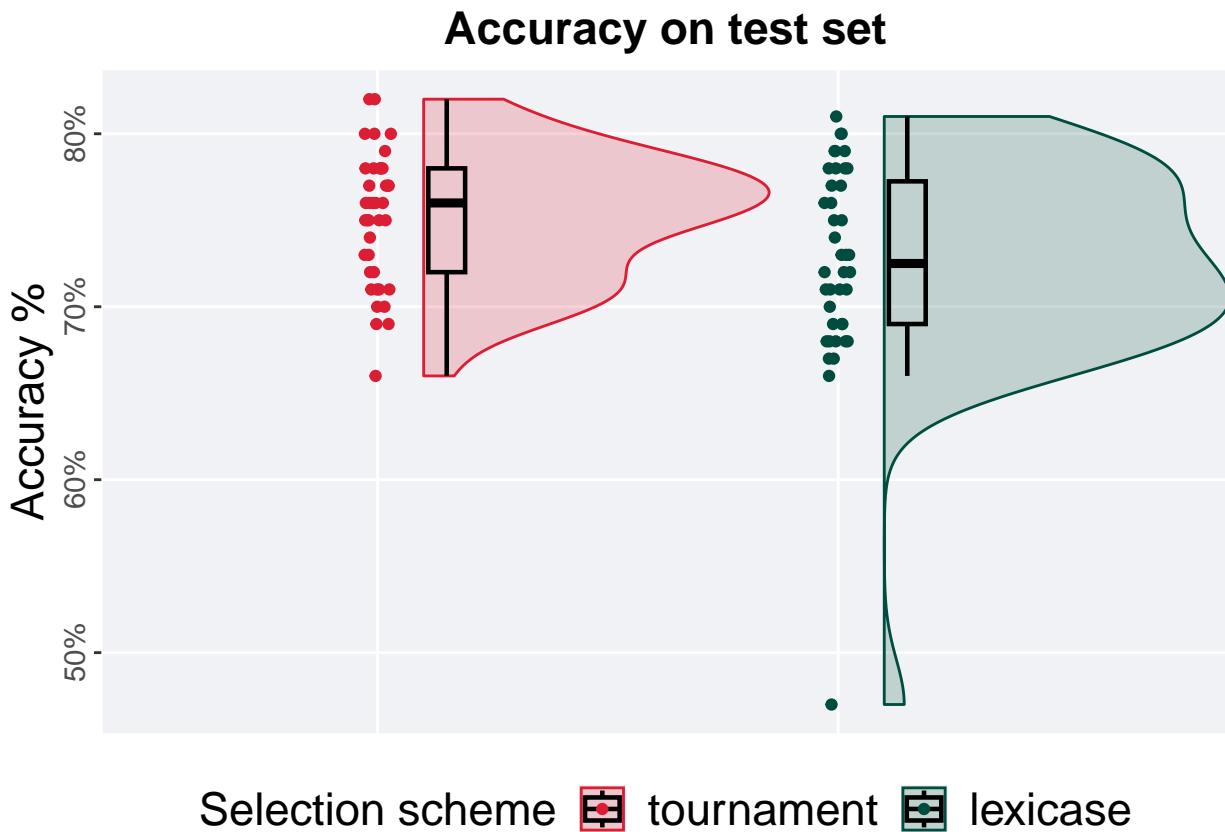
```
## [1] "observed_diff: 2.20585693087363"
## [1] "lower: -1.98853335019783"
## [1] "upper: 1.99761960468652"
## [1] "reject null hypothesis"
## [1] "p-value: 0.02974"
```



7.4 90%

7.4.1 Test accuracy

```
test_plot(filter(task_data, split == '90%'))
```



Summary statistics for the testing performance of the selection schemes at the 5% selection set split:

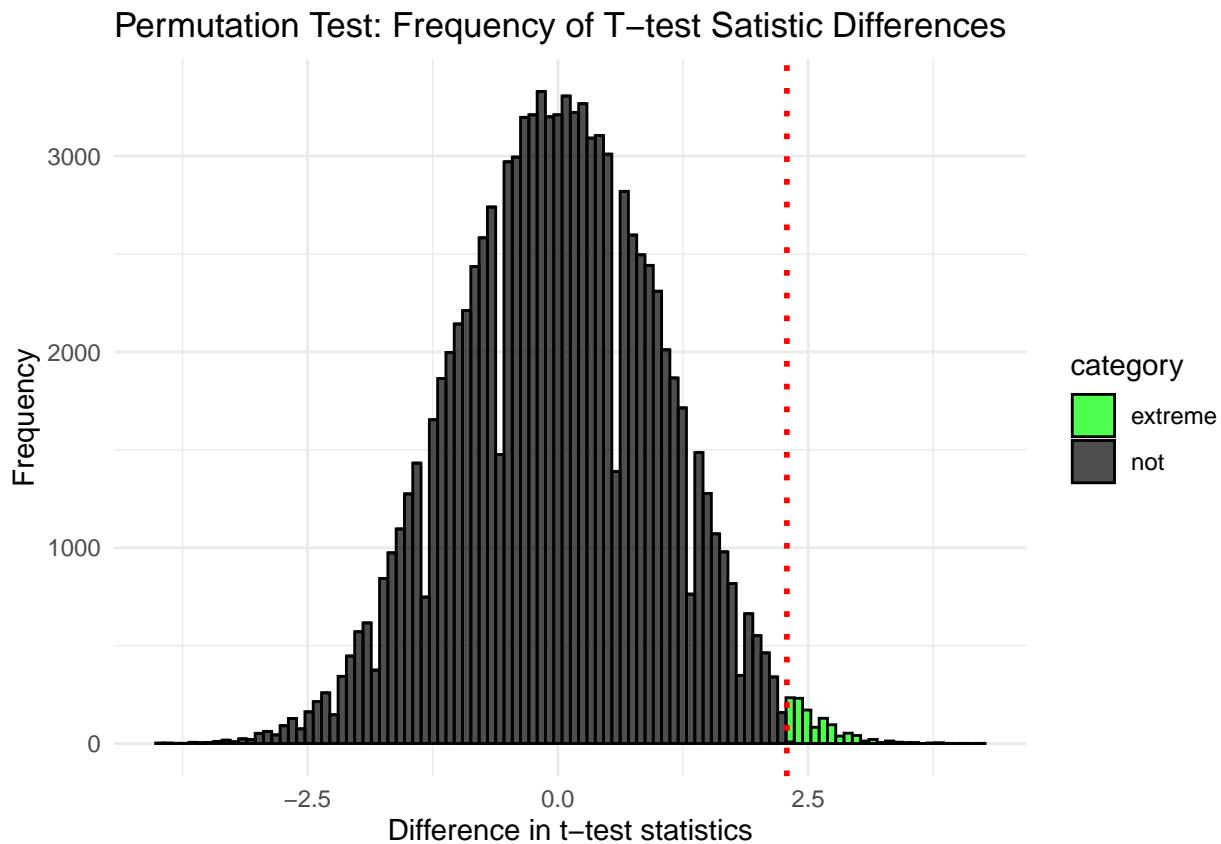
```
test_results_summary(filter(task_data, split == '90%'))
```

```
## # A tibble: 2 x 8
##   selection  count na_cnt   min median   mean   max   IQR
##   <fct>      <int> <int> <dbl>  <dbl> <dbl> <dbl> <dbl>
## 1 tournament     40     0  0.66  0.76  0.752  0.82  0.0600
## 2 lexicase       40     0  0.47  0.725 0.726  0.81  0.0825
```

The permutation test revealed that the results are:

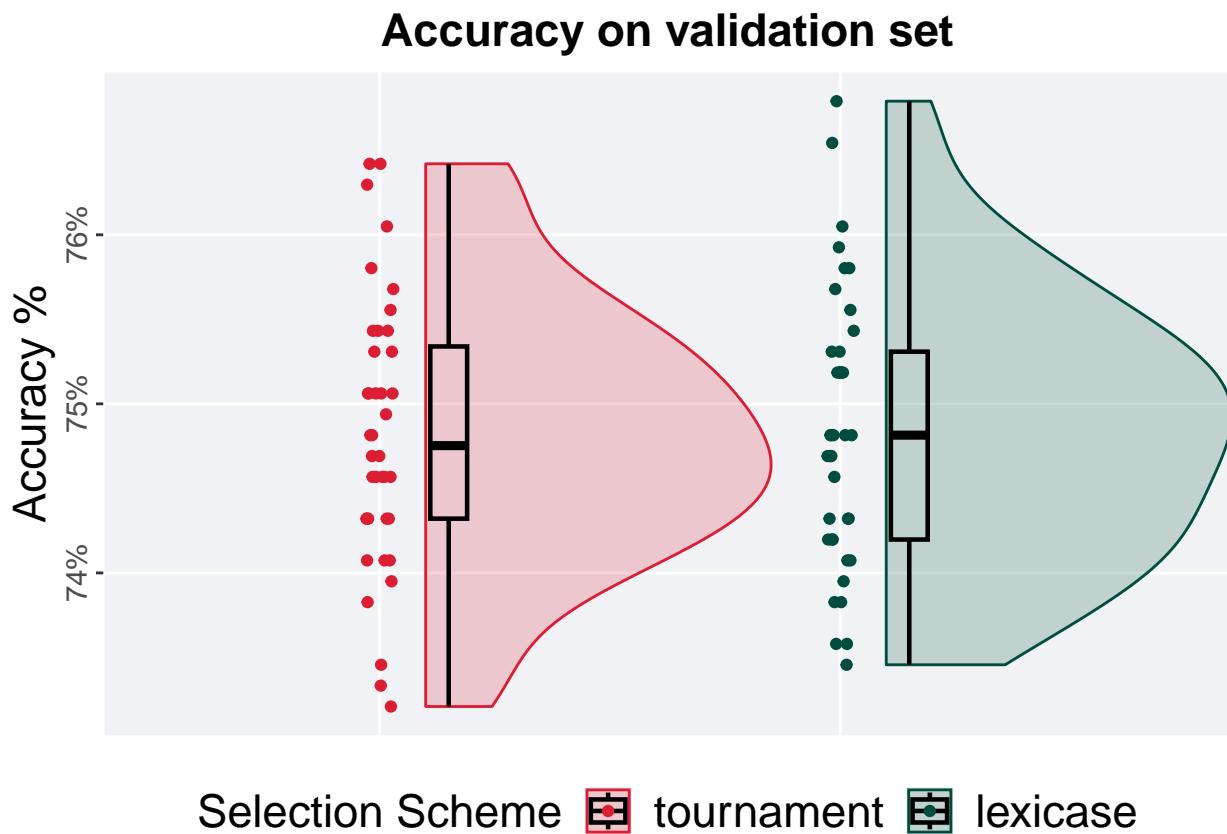
```
tournament_results <- filter(task_data, split == '90%' & selection == 'tournament')
lexicase_results <- filter(task_data, split == '90%' & selection == 'lexicase')
permutation_test(tournament_results$testing_performance,
                 lexicase_results$testing_performance,
                 seed = 57,
                 alternative = "g")
```

```
## [1] "observed_diff: 2.28744114118623"
## [1] "permutation_diffs[0.95 * n_permutations]: 1.63997158080026"
## [1] "reject null hypothesis"
## [1] "p-value: 0.01148"
```



7.4.2 Validation accuracy

```
validation_plot(filter(task_data, split == '90%'))
```



Summary statistics for the testing performance of the selection schemes at the 5% selection set split:

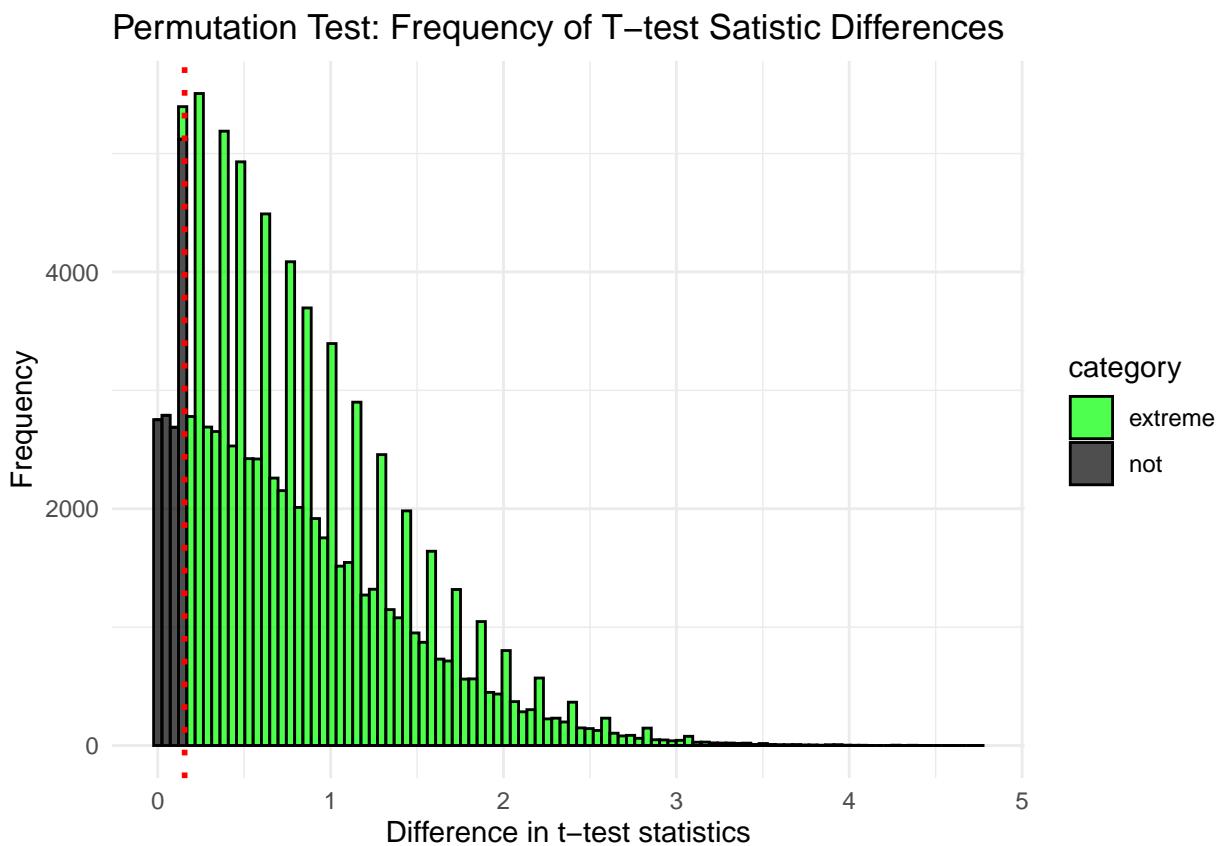
```
validation_accuracy_summary(filter(task_data, split == '90%'))
```

```
## # A tibble: 2 x 8
##   selection count na_cnt  min median  mean  max    IQR
##   <fct>     <int> <int> <dbl> <dbl> <dbl> <dbl> <dbl>
## 1 tournament     40     0 0.732 0.748 0.748 0.764 0.0102
## 2 lexicase       40     0 0.735 0.748 0.748 0.768 0.0111
```

The permutation test revealed that the results are:

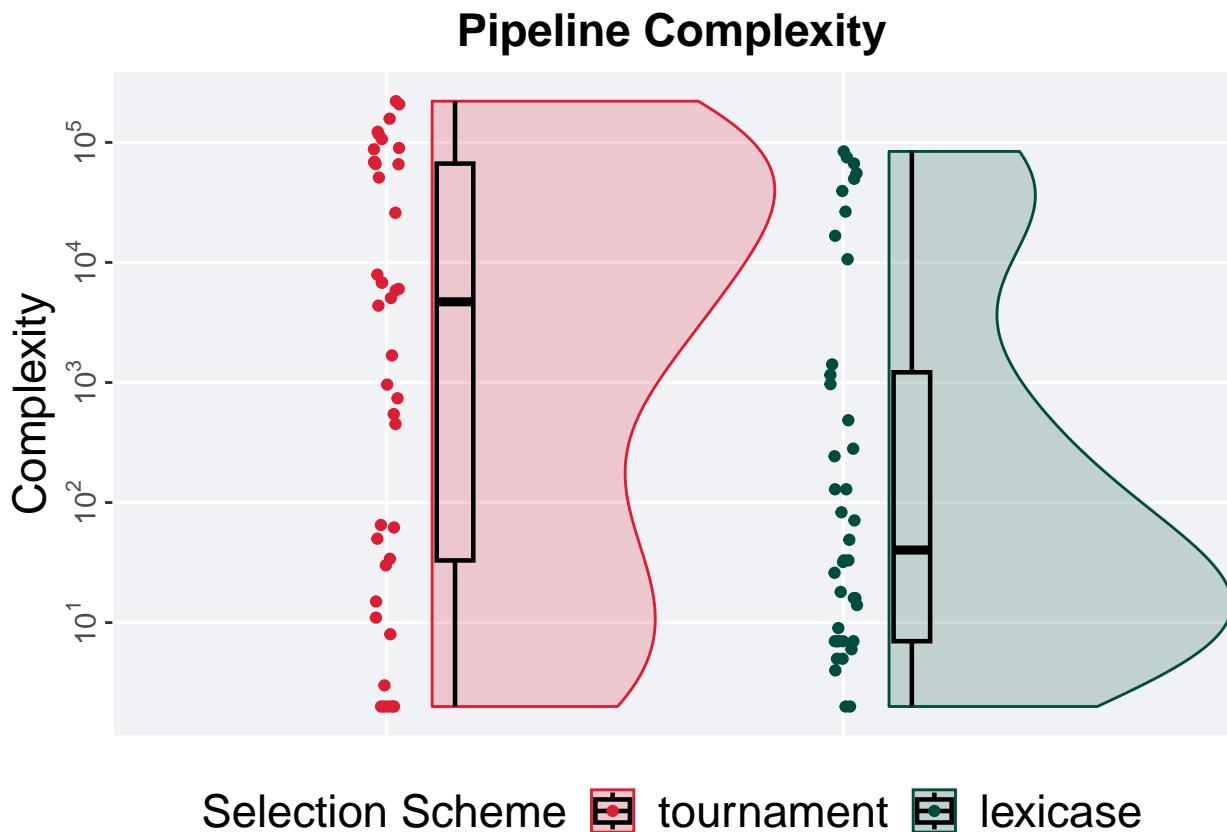
```
tournament_results <- filter(task_data, split == '90%' & selection == 'tournament')
lexicase_results <- filter(task_data, split == '90%' & selection == 'lexicase')
permutation_test(tournament_results$training_performance,
                 lexicase_results$training_performance,
                 seed = 58,
                 alternative = "t")
```

```
## [1] "observed_diff: -0.155413318723722"
## [1] "lower: -2.0003726169214"
## [1] "upper: 1.96320799741225"
## [1] "fail to reject null hypothesis"
## [1] "p-value: 0.8665"
```



7.4.3 Pipeline complexity

```
complexity_plot(filter(task_data, split == '90%'))
```



Summary statistics for the pipeline complexity of the selection schemes at the 5% selection set split:

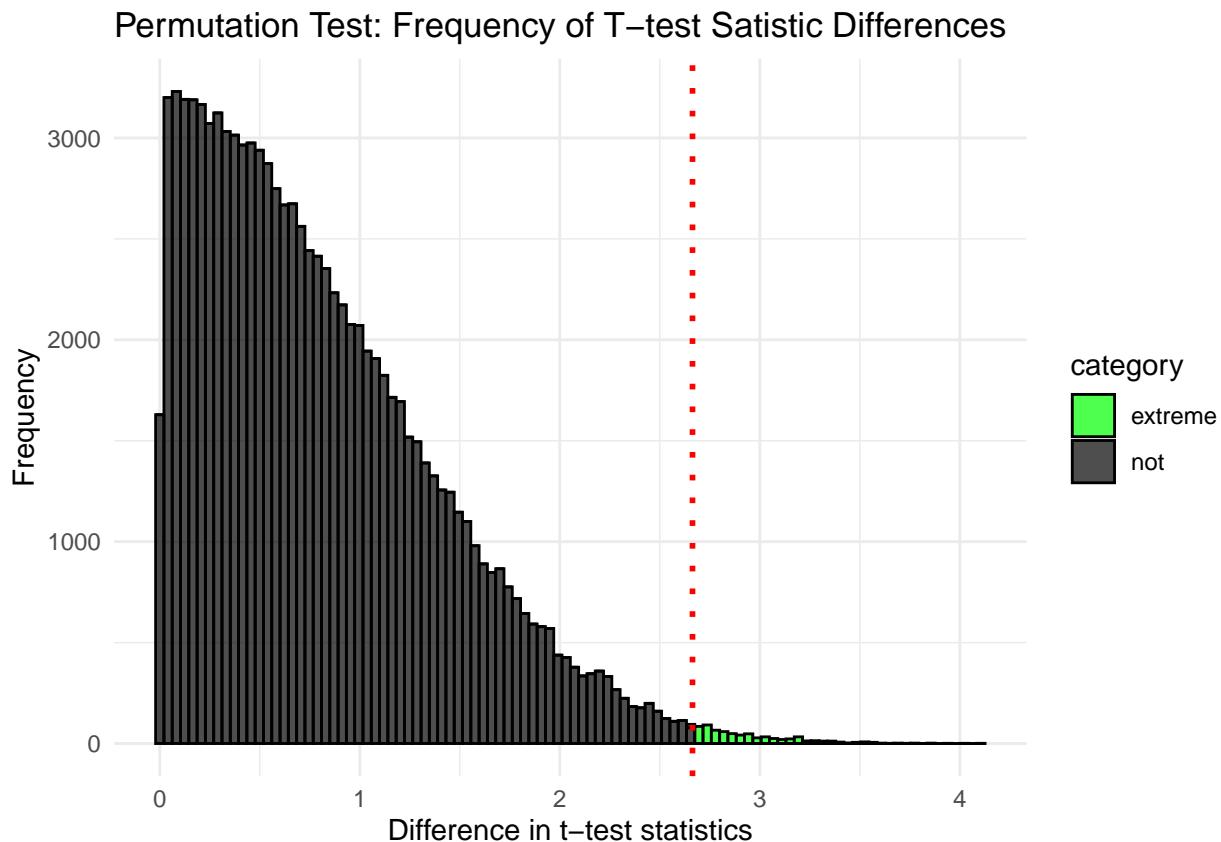
```
complexity_summary(filter(task_data, split == '90%'))
```

```
## # A tibble: 2 x 8
##   selection count na_cnt  min median   mean   max   IQR
##   <fct>     <int> <int> <dbl> <dbl> <dbl> <int> <dbl>
## 1 tournament     40     0     2  4713 37579. 220471 66590.
## 2 lexicase       40     0     2     41 10736.  84131  1214.
```

The permutation test revealed that the results are:

```
tournament_results <- filter(task_data, split == '90%' & selection == 'tournament')
lexicase_results <- filter(task_data, split == '90%' & selection == 'lexicase')
permutation_test(tournament_results$testing_complexity,
                 lexicase_results$testing_complexity,
                 seed = 220,
                 alternative = "t")
```

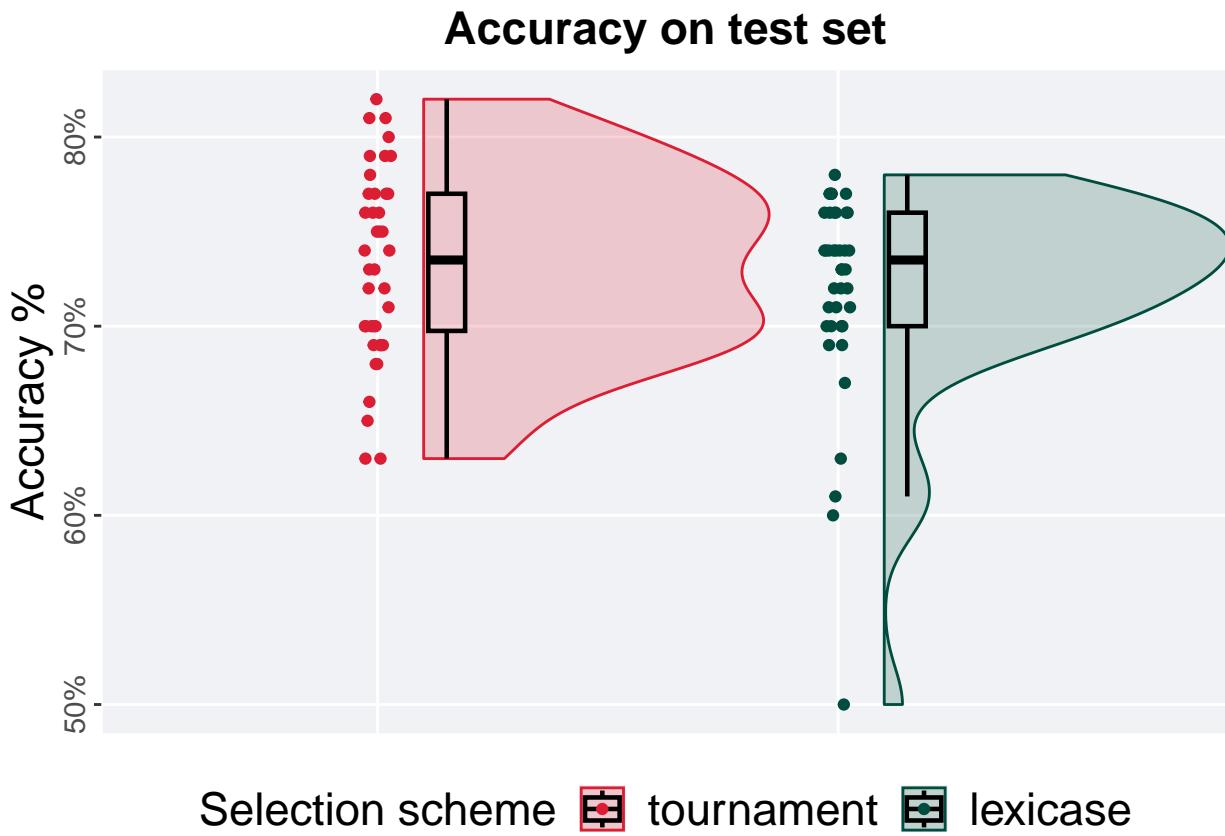
```
## [1] "observed_diff: 2.66362976028605"
## [1] "lower: -1.97715552372964"
## [1] "upper: 1.95915905592898"
## [1] "reject null hypothesis"
## [1] "p-value: 0.00714"
```



7.5 95%

7.5.1 Test accuracy

```
test_plot(filter(task_data, split == '95%'))
```



Summary statistics for the testing performance of the selection schemes at the 5% selection set split:

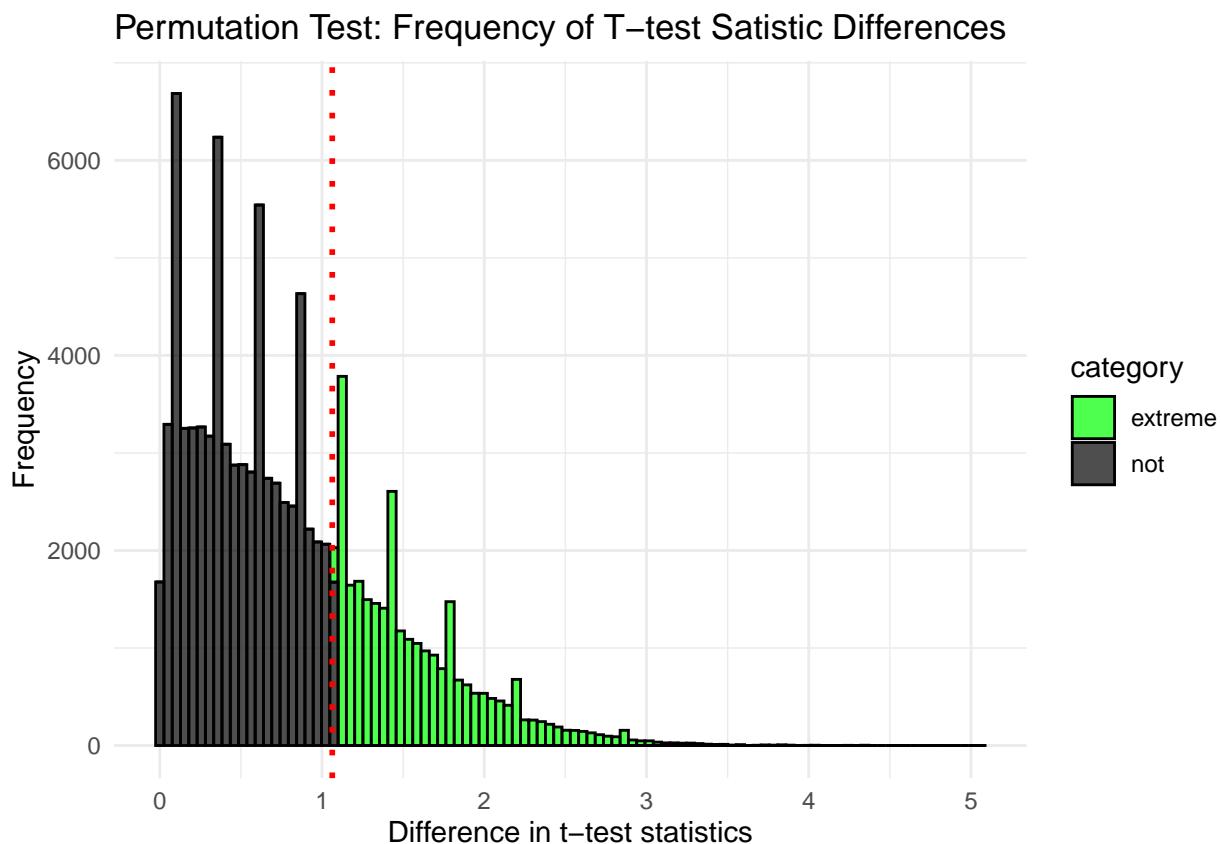
```
test_results_summary(filter(task_data, split == '95%'))
```

```
## # A tibble: 2 x 8
##   selection count na_cnt  min median  mean   max     IQR
##   <fct>     <int> <int> <dbl> <dbl> <dbl> <dbl> <dbl>
## 1 tournament     40     0  0.63  0.735 0.732  0.82  0.0725
## 2 lexicase       40     0  0.5   0.735 0.720  0.78  0.0600
```

The permutation test revealed that the results are:

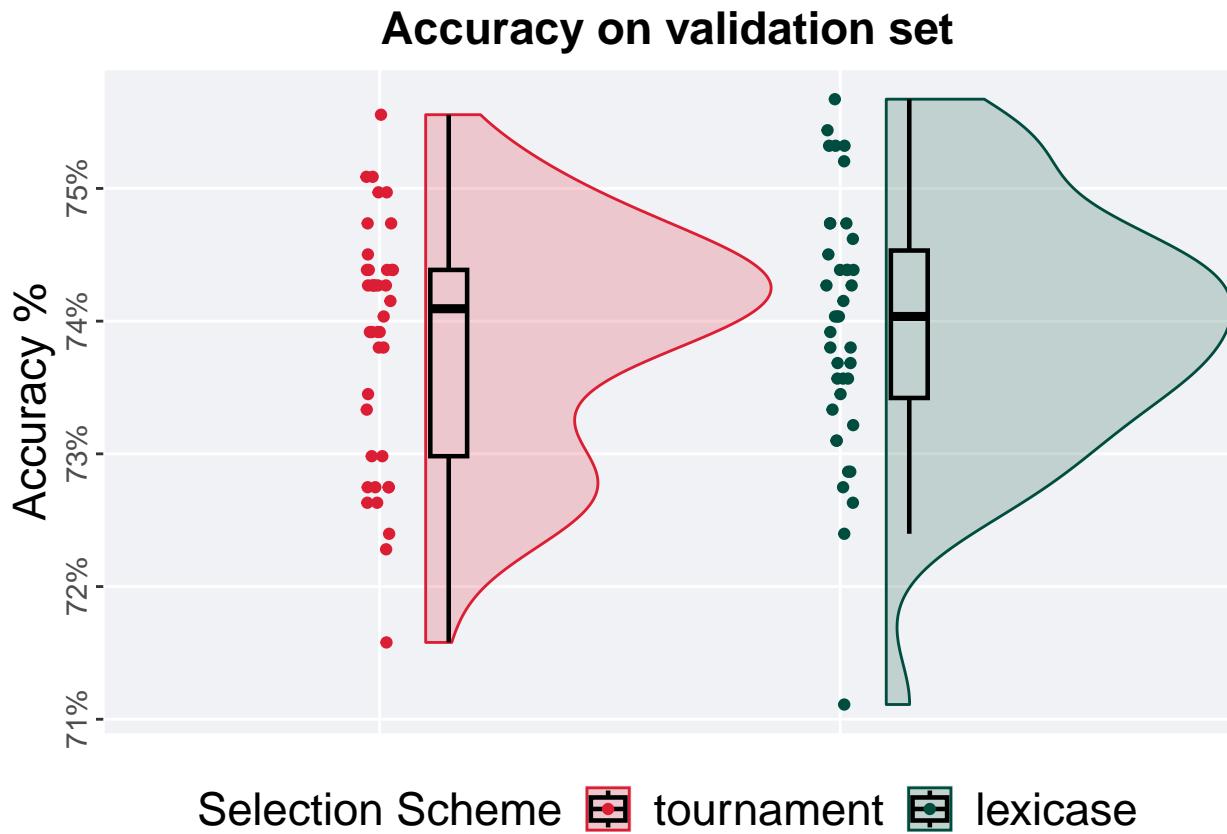
```
tournament_results <- filter(task_data, split == '95%' & selection == 'tournament')
lexicase_results <- filter(task_data, split == '95%' & selection == 'lexicase')
permutation_test(tournament_results$testing_performance,
                 lexicase_results$testing_performance,
                 seed = 59,
                 alternative = "t")
```

```
## [1] "observed_diff: 1.06320673917919"
## [1] "lower: -1.99102103585612"
## [1] "upper: 1.99102103585612"
## [1] "fail to reject null hypothesis"
## [1] "p-value: 0.28918"
```



7.5.2 Validation accuracy

```
validation_plot(filter(task_data, split == '95%'))
```



Summary statistics for the testing performance of the selection schemes at the 5% selection set split:

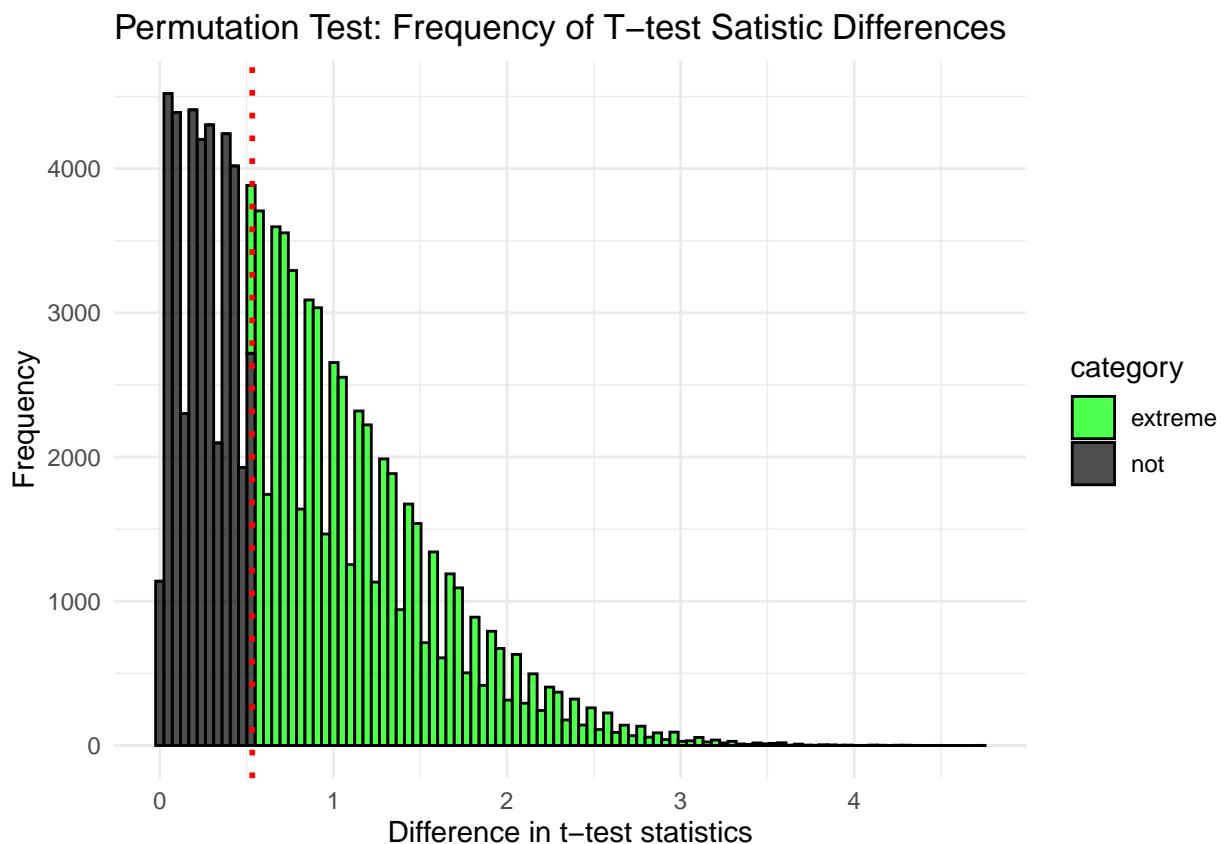
```
validation_accuracy_summary(filter(task_data, split == '95%'))
```

```
## # A tibble: 2 x 8
##   selection count na_cnt  min median  mean   max     IQR
##   <fct>     <int> <int> <dbl> <dbl> <dbl> <dbl> <dbl>
## 1 tournament     40     0 0.716 0.741 0.738 0.756 0.0140
## 2 lexicase       40     0 0.711 0.740 0.740 0.757 0.0111
```

The permutation test revealed that the results are:

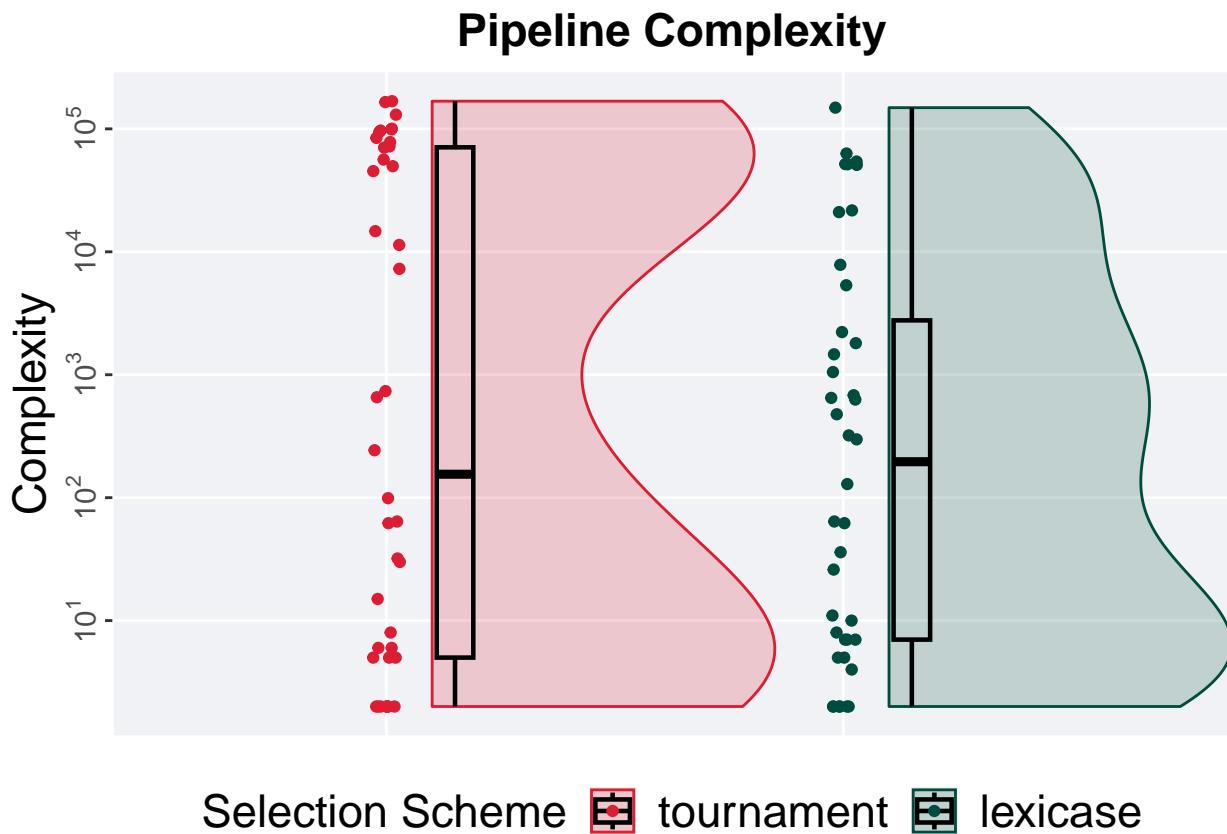
```
tournament_results <- filter(task_data, split == '95%' & selection == 'tournament')
lexicase_results <- filter(task_data, split == '95%' & selection == 'lexicase')
permutation_test(tournament_results$training_performance,
                 lexicase_results$training_performance,
                 seed = 60,
                 alternative = "t")
```

```
## [1] "observed_diff: -0.532630909404782"
## [1] "lower: -2.00879367960193"
## [1] "upper: 2.00879574403038"
## [1] "fail to reject null hypothesis"
## [1] "p-value: 0.59734"
```



7.5.3 Pipeline complexity

```
complexity_plot(filter(task_data, split == '95%'))
```



Summary statistics for the pipeline complexity of the selection schemes at the 5% selection set split:

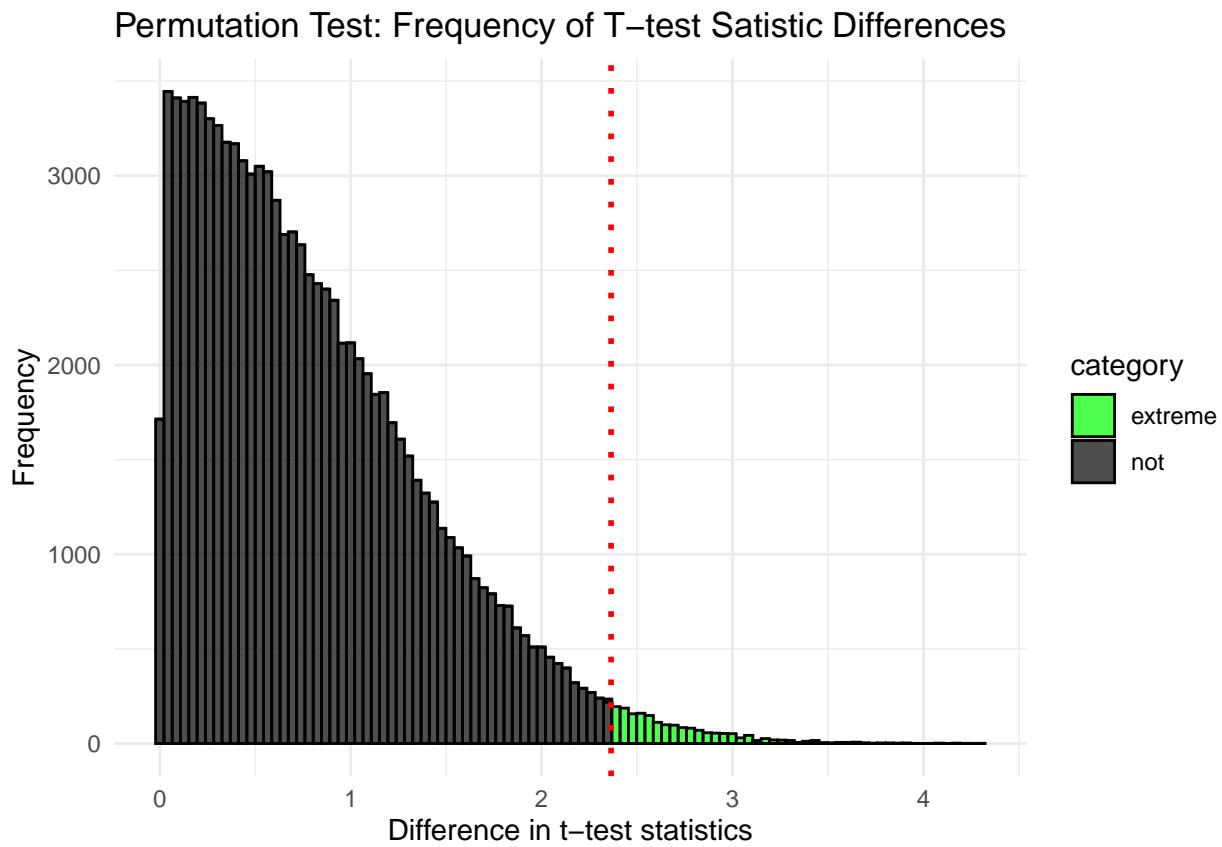
```
complexity_summary(filter(task_data, split == '95%'))
```

```
## # A tibble: 2 x 8
##   selection count na_cnt  min median   mean   max    IQR
##   <fct>     <int> <int> <dbl> <dbl> <dbl> <int> <dbl>
## 1 tournament     40     0     2   171  33642. 167911 70771
## 2 lexicase       40     0     2  214.  12153. 148767  2996.
```

The permutation test revealed that the results are:

```
tournament_results <- filter(task_data, split == '95%' & selection == 'tournament')
lexicase_results <- filter(task_data, split == '95%' & selection == 'lexicase')
permutation_test(tournament_results$testing_complexity,
                 lexicase_results$testing_complexity,
                 seed = 221,
                 alternative = "t")
```

```
## [1] "observed_diff: 2.36386426978814"
## [1] "lower: -1.9863320715242"
## [1] "upper: 1.96770422150421"
## [1] "reject null hypothesis"
## [1] "p-value: 0.01879"
```



Chapter 8

Task 359956

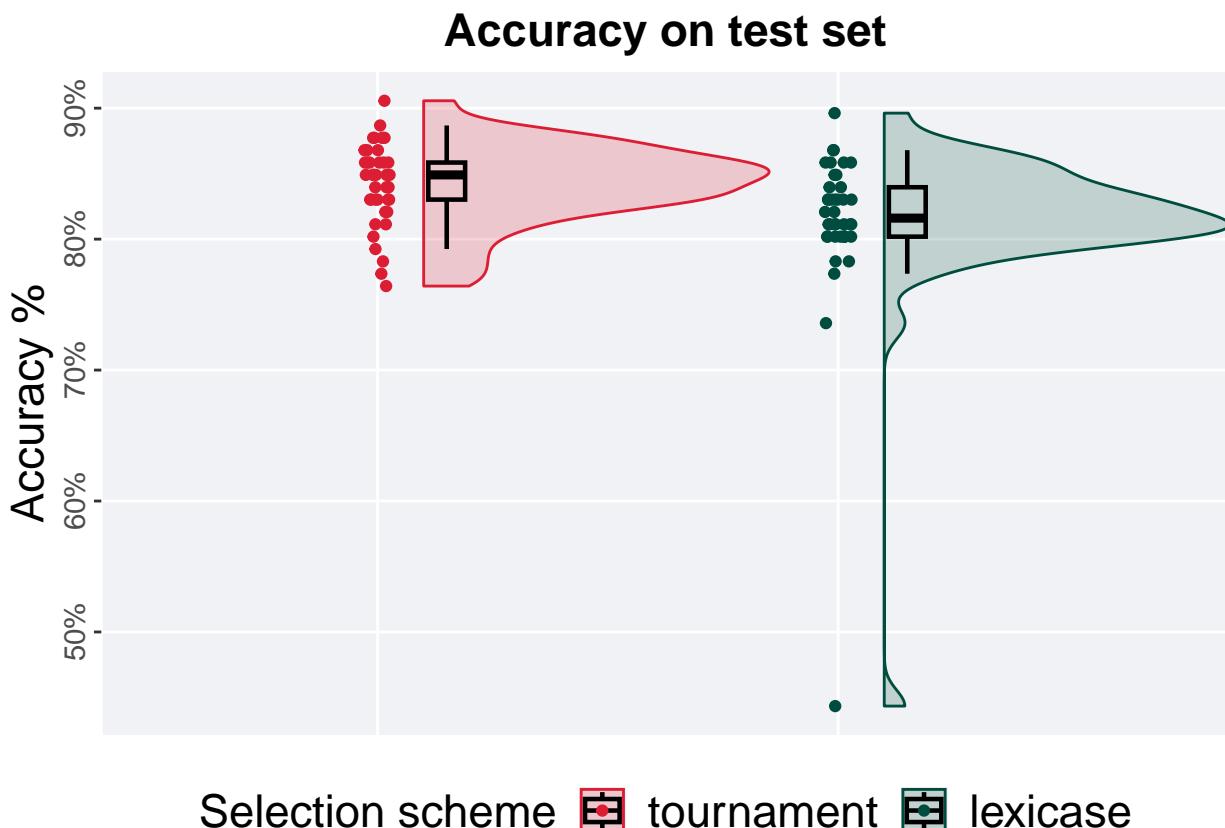
We present the results of our analysis of task 359956 with the different selection set splits used in our study.

```
task_data <- filter(results, task_id == 359956)
```

8.1 5%

8.1.1 Test accuracy

```
test_plot(filter(task_data, split == '5%'))
```



Summary statistics for the testing performance of the selection schemes at the 5% selection set split:

```
test_results_summary(filter(task_data, split == '5%'))
```

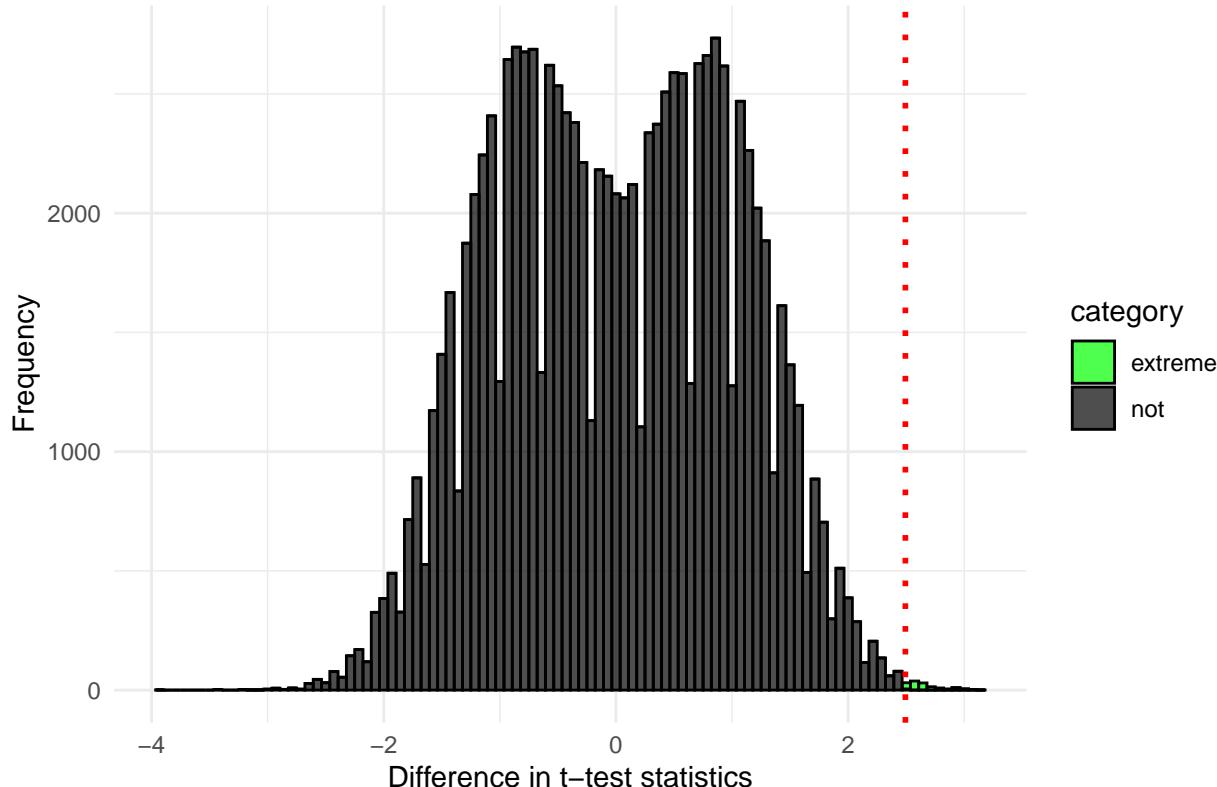
```
## # A tibble: 2 x 8
##   selection count na_cnt  min median  mean  max    IQR
##   <fct>     <int> <int> <dbl> <dbl> <dbl> <dbl> <dbl>
## 1 tournament     40     0 0.764 0.849 0.842 0.906 0.0283
## 2 lexicase       40     0 0.443 0.816 0.813 0.896 0.0377
```

The permutation test revealed that the results are:

```
tournament_results <- filter(task_data, split == '5%' & selection == 'tournament')
lexicase_results <- filter(task_data, split == '5%' & selection == 'lexicase')
permutation_test(tournament_results$testing_performance,
                  lexicase_results$testing_performance,
                  seed = 61,
                  alternative = "g")
```

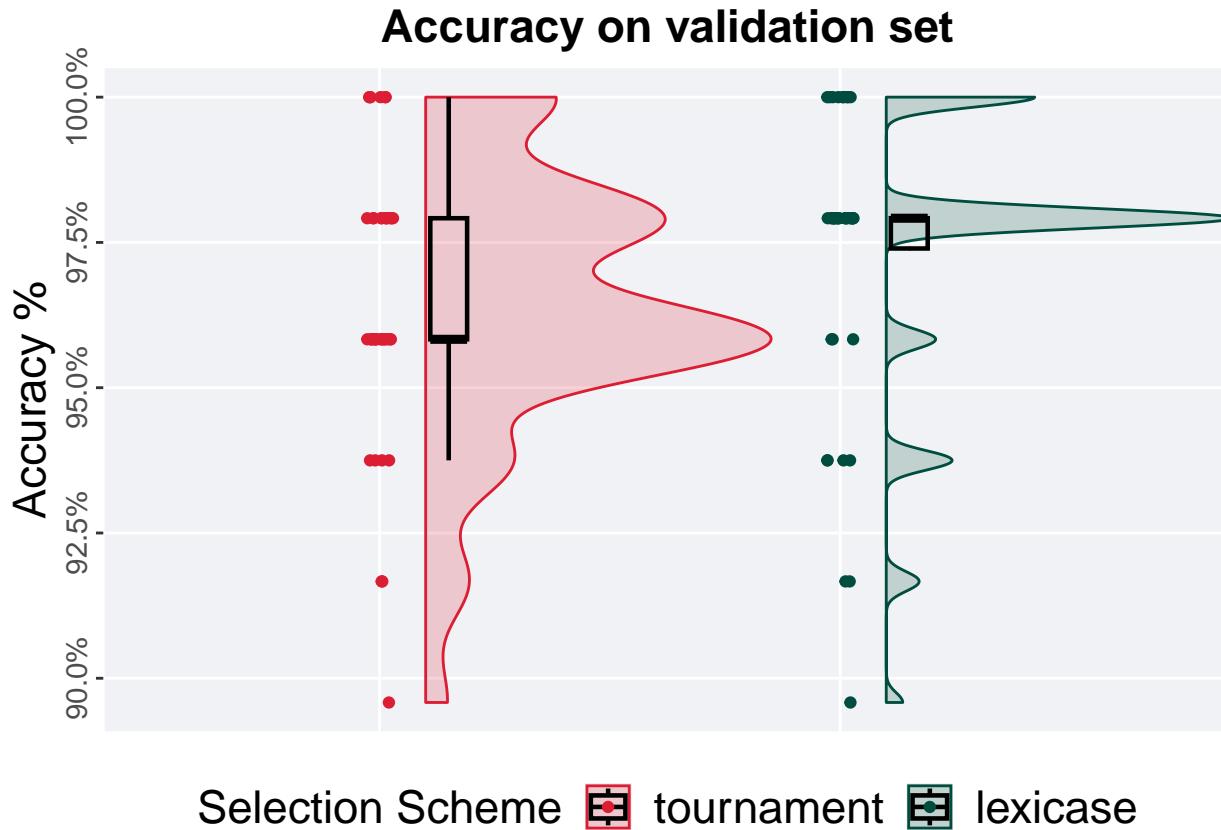
```
## [1] "observed_diff: 2.49264287194845"
## [1] "permutation_diffs[0.95 * n_permutations]: 1.5647718135028"
## [1] "reject null hypothesis"
## [1] "p-value: 0.00143"
```

Permutation Test: Frequency of T-test Statistic Differences



8.1.2 Validation accuracy

```
validation_plot(filter(task_data, split == '5%'))
```



Summary statistics for the testing performance of the selection schemes at the 5% selection set split:

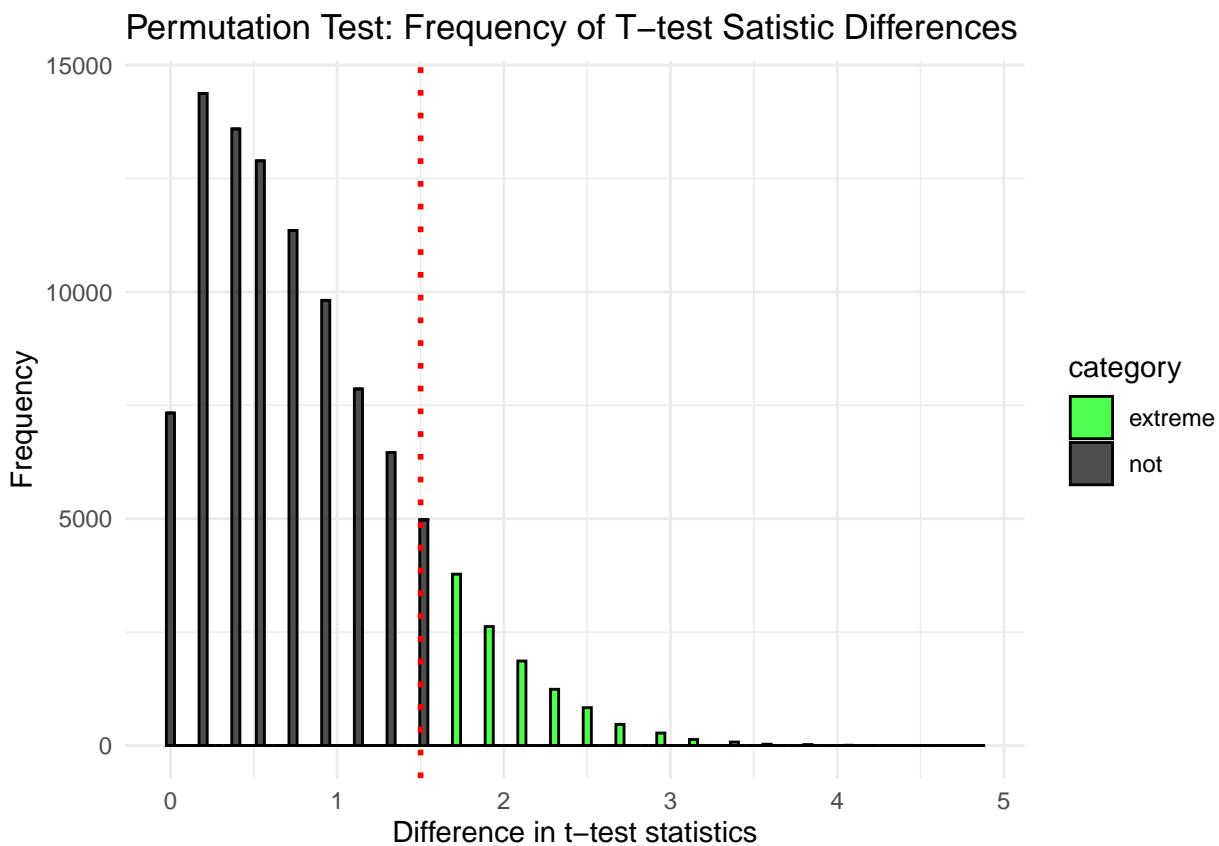
```
validation_accuracy_summary(filter(task_data, split == '5%'))
```

```
## # A tibble: 2 x 8
##   selection  count na_cnt   min median   mean   max     IQR
##   <fct>     <int> <int> <dbl>  <dbl>  <dbl> <dbl>  <dbl>
## 1 tournament     40     0 0.896  0.958  0.965     1 0.0208
## 2 lexicase      40     0 0.896  0.979  0.973     1 0.00521
```

The permutation test revealed that the results are:

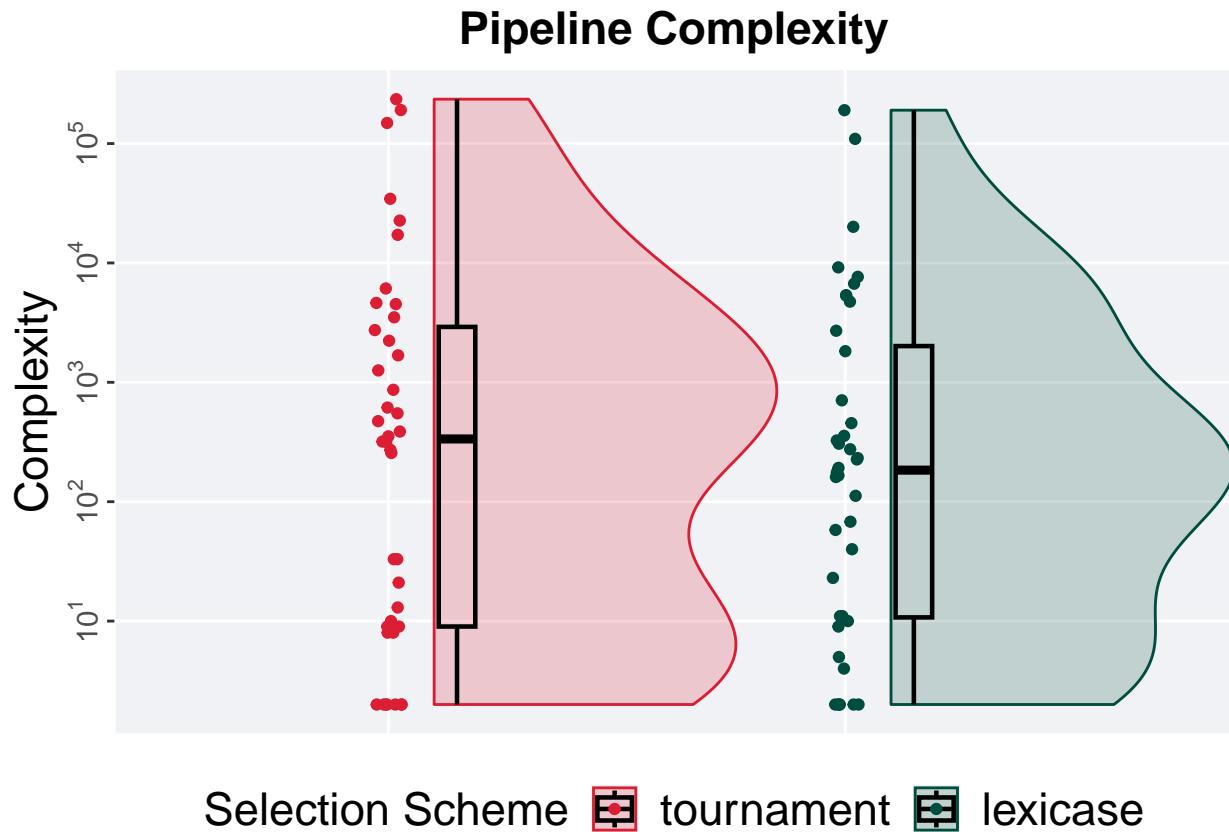
```
tournament_results <- filter(task_data, split == '5%' & selection == 'tournament')
lexicase_results <- filter(task_data, split == '5%' & selection == 'lexicase')
permutation_test(tournament_results$training_performance,
                 lexicase_results$training_performance,
                 seed = 62,
                 alternative = "t")

## [1] "observed_diff: -1.50090531983546"
## [1] "lower: -2.09080286669461"
## [1] "upper: 1.89155678516992"
## [1] "fail to reject null hypothesis"
## [1] "p-value: 0.11353"
```



8.1.3 Pipeline complexity

```
complexity_plot(filter(task_data, split == '5%'))
```



Summary statistics for the pipeline complexity of the selection schemes at the 5% selection set split:

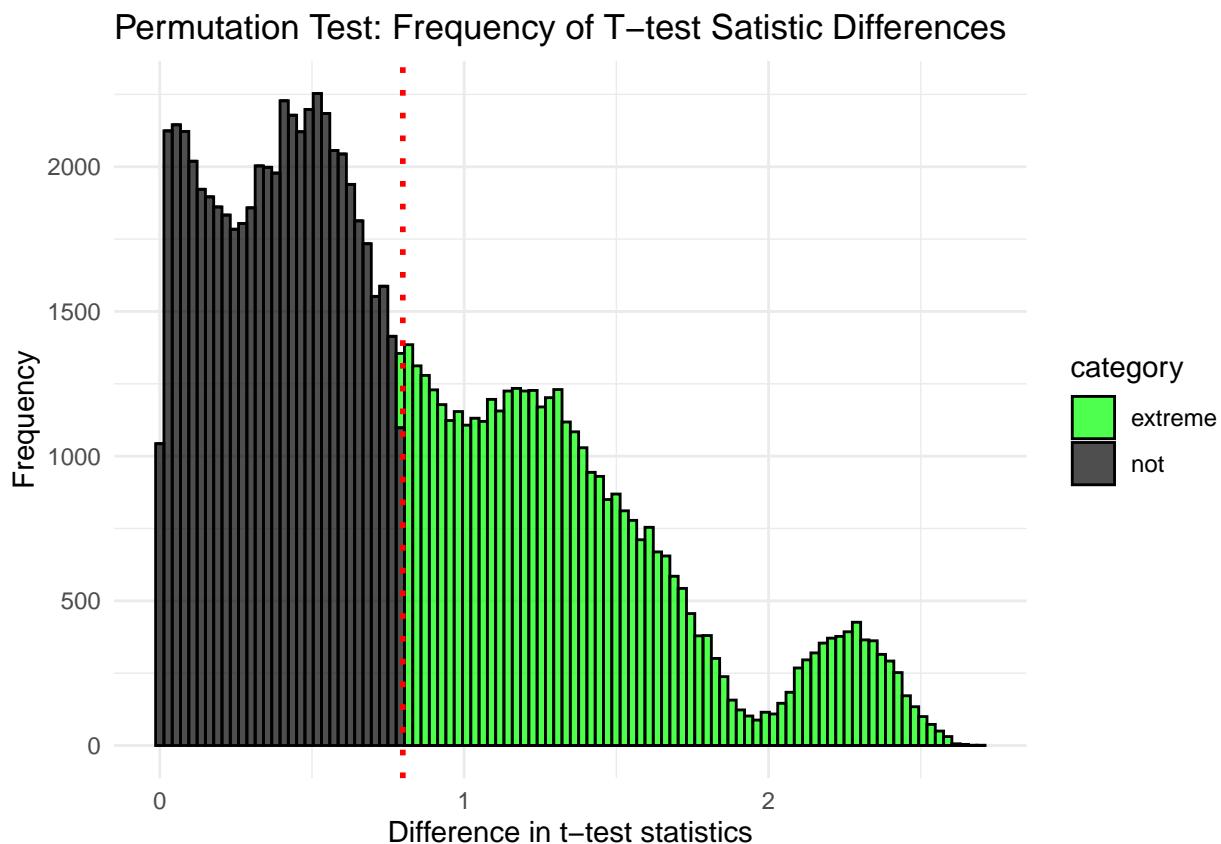
```
complexity_summary(filter(task_data, split == '5%'))
```

```
## # A tibble: 2 x 8
##   selection count na_cnt  min median   mean   max   IQR
##   <fct>     <int> <int> <int> <dbl> <dbl> <int> <dbl>
## 1 tournament     40     0     2    336 17015. 235362 2921
## 2 lexicase       40     0     2    184  9185. 190371 2033.
```

The permutation test revealed that the results are:

```
tournament_results <- filter(task_data, split == '5%' & selection == 'tournament')
lexicase_results <- filter(task_data, split == '5%' & selection == 'lexicase')
permutation_test(tournament_results$testing_complexity,
                 lexicase_results$testing_complexity,
                 seed = 220,
                 alternative = "t")

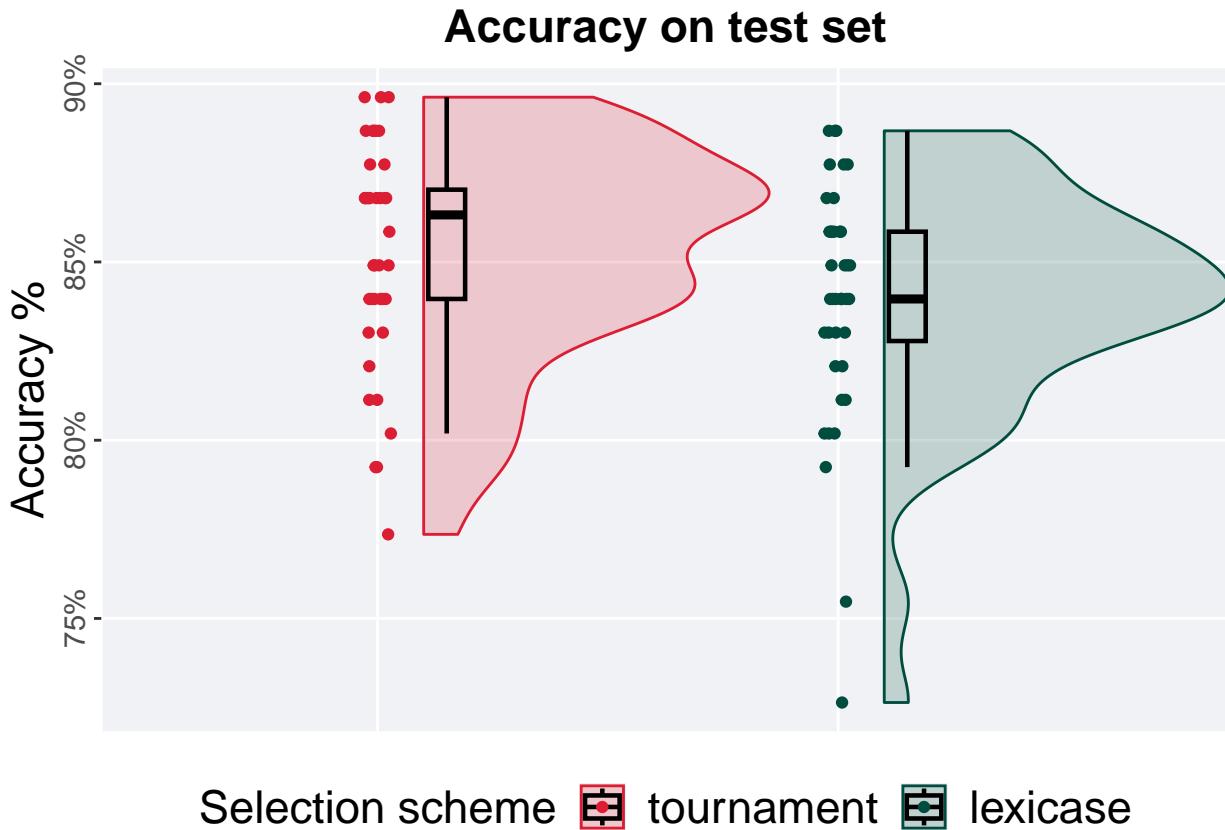
## [1] "observed_diff: 0.798407638425608"
## [1] "lower: -2.09336753329772"
## [1] "upper: 2.06435980018571"
## [1] "fail to reject null hypothesis"
## [1] "p-value: 0.43211"
```



8.2 10%

8.2.1 Test accuracy

```
test_plot(filter(task_data, split == '10%'))
```



Summary statistics for the testing performance of the selection schemes at the 5% selection set split:

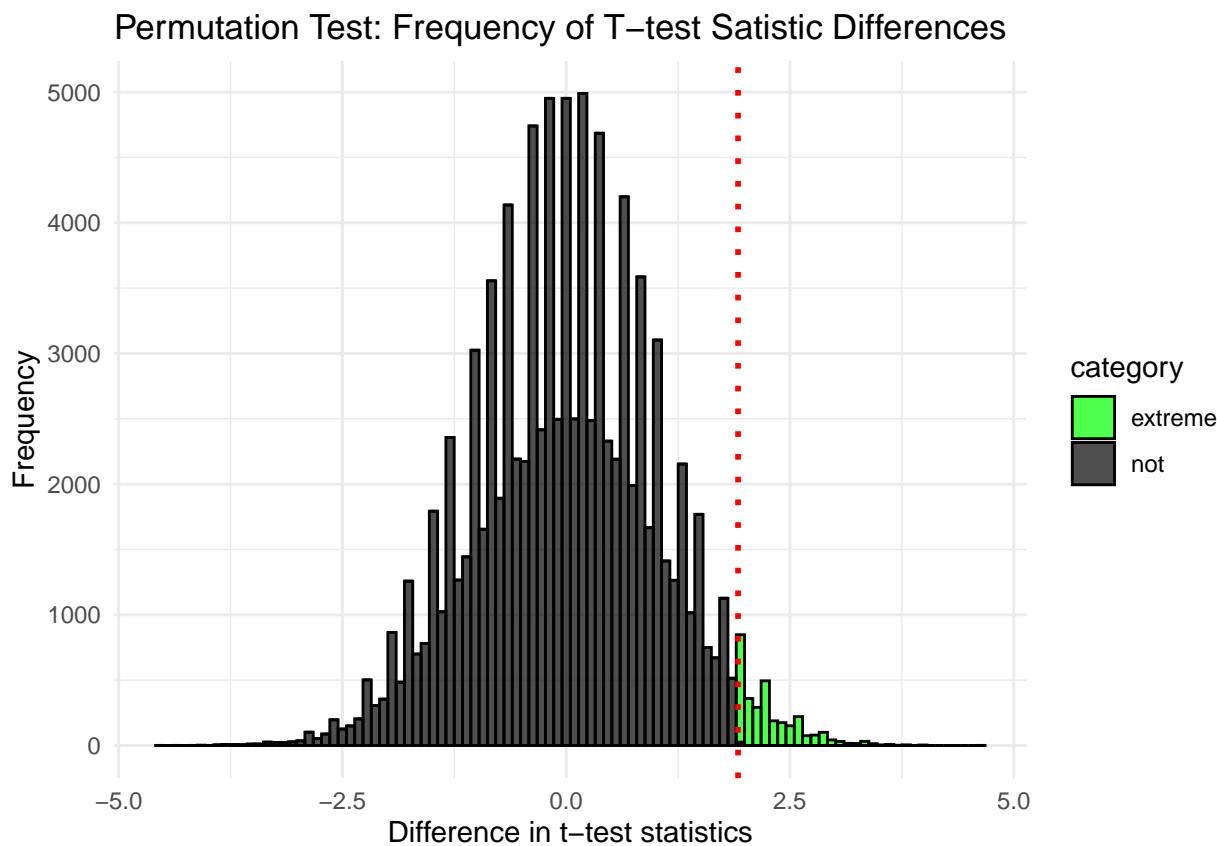
```
test_results_summary(filter(task_data, split == '10%'))
```

```
## # A tibble: 2 x 8
##   selection  count  na_cnt  min  median  mean  max  IQR
##   <fct>      <int>   <int> <dbl>   <dbl> <dbl> <dbl> <dbl>
## 1 tournament     40      0 0.774  0.863  0.853 0.896 0.0307
## 2 lexicase       40      0 0.726  0.840  0.840  0.839 0.0307
```

The permutation test revealed that the results are:

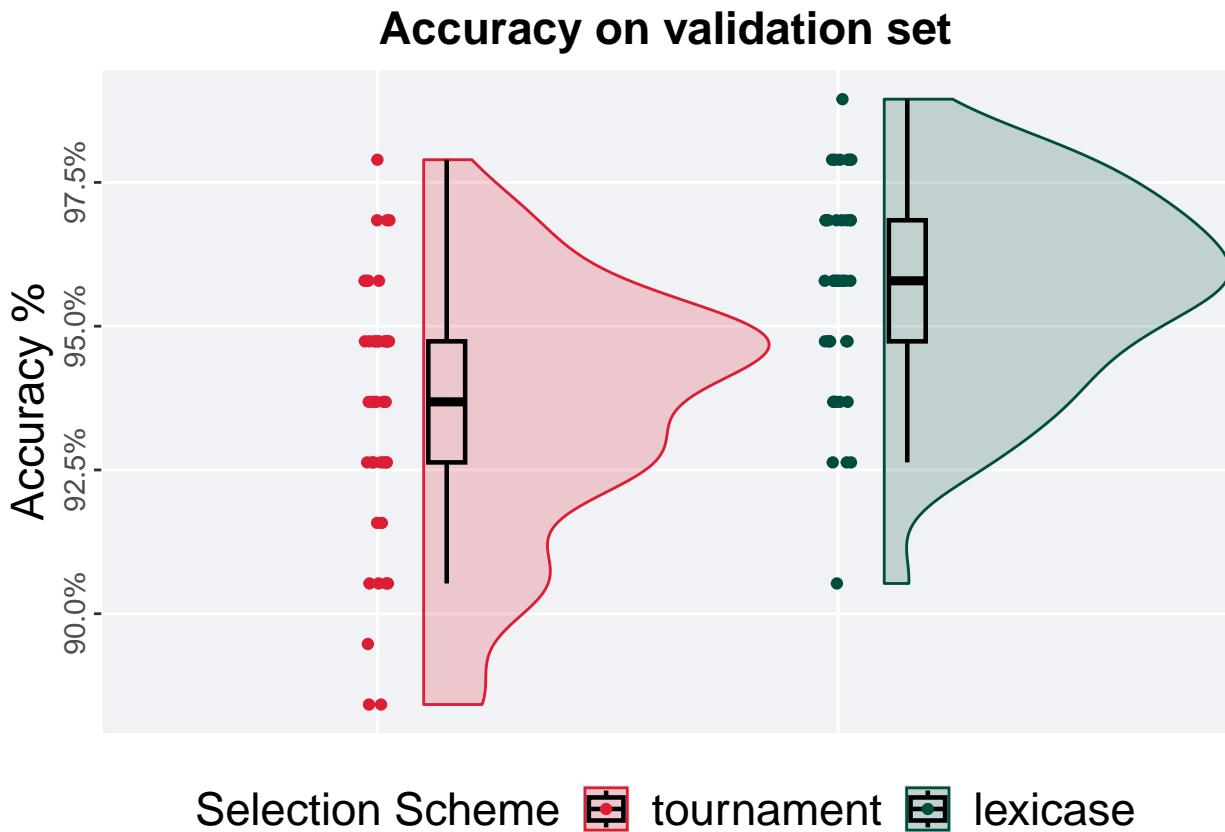
```
tournament_results <- filter(task_data, split == '10%' & selection == 'tournament')
lexicase_results <- filter(task_data, split == '10%' & selection == 'lexicase')
permutation_test(tournament_results$testing_performance,
                 lexicase_results$testing_performance,
                 seed = 63,
                 alternative = "g")
```

```
## [1] "observed_diff: 1.92031097297846"
## [1] "permutation_diffs[0.95 * n_permutations]: 1.65010001663041"
## [1] "reject null hypothesis"
## [1] "p-value: 0.03129"
```



8.2.2 Validation accuracy

```
validation_plot(filter(task_data, split == '10%'))
```



Summary statistics for the testing performance of the selection schemes at the 5% selection set split:

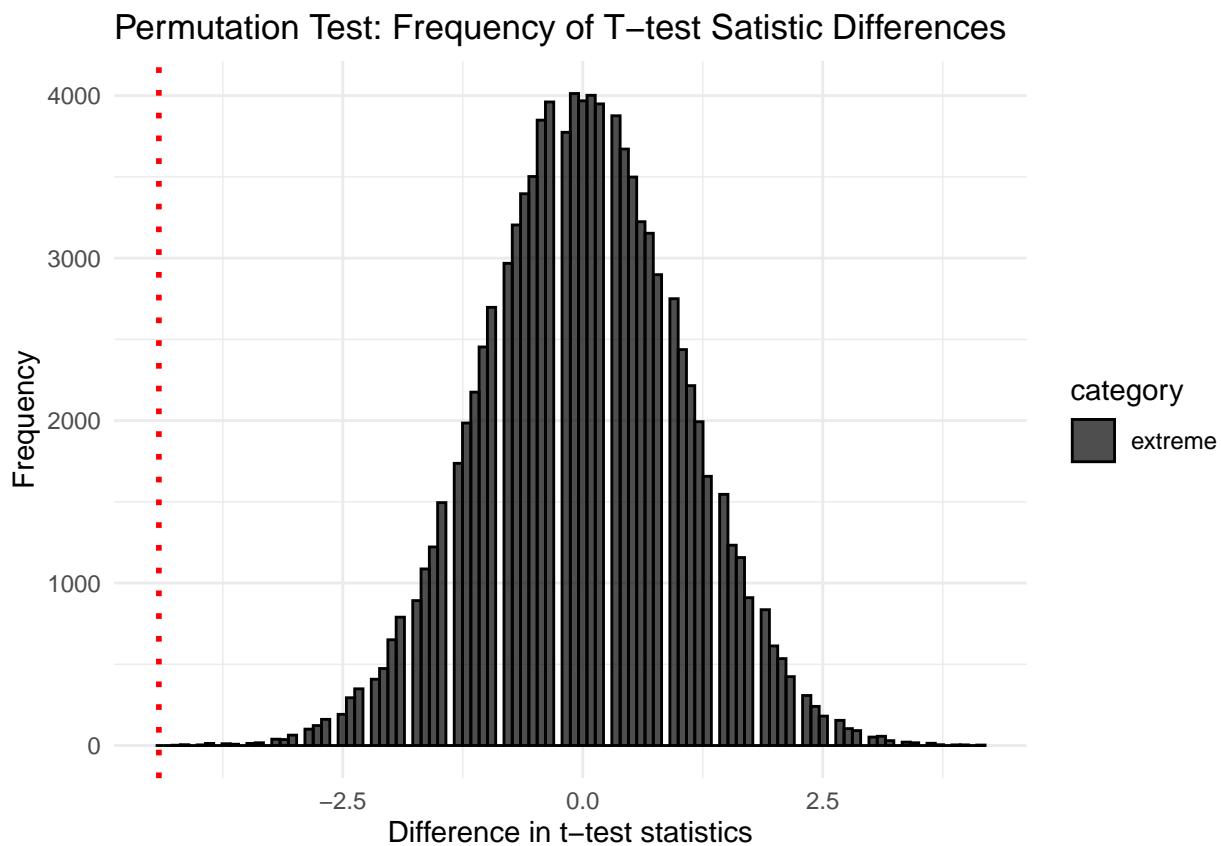
```
validation_accuracy_summary(filter(task_data, split == '10%'))
```

```
## # A tibble: 2 x 8
##   selection count na_cnt  min median  mean  max    IQR
##   <fct>     <int> <int> <dbl>  <dbl> <dbl> <dbl> <dbl>
## 1 tournament     40     0 0.884  0.937  0.936  0.979  0.0211
## 2 lexicase       40     0 0.905  0.958  0.956  0.989  0.0211
```

The permutation test revealed that the results are:

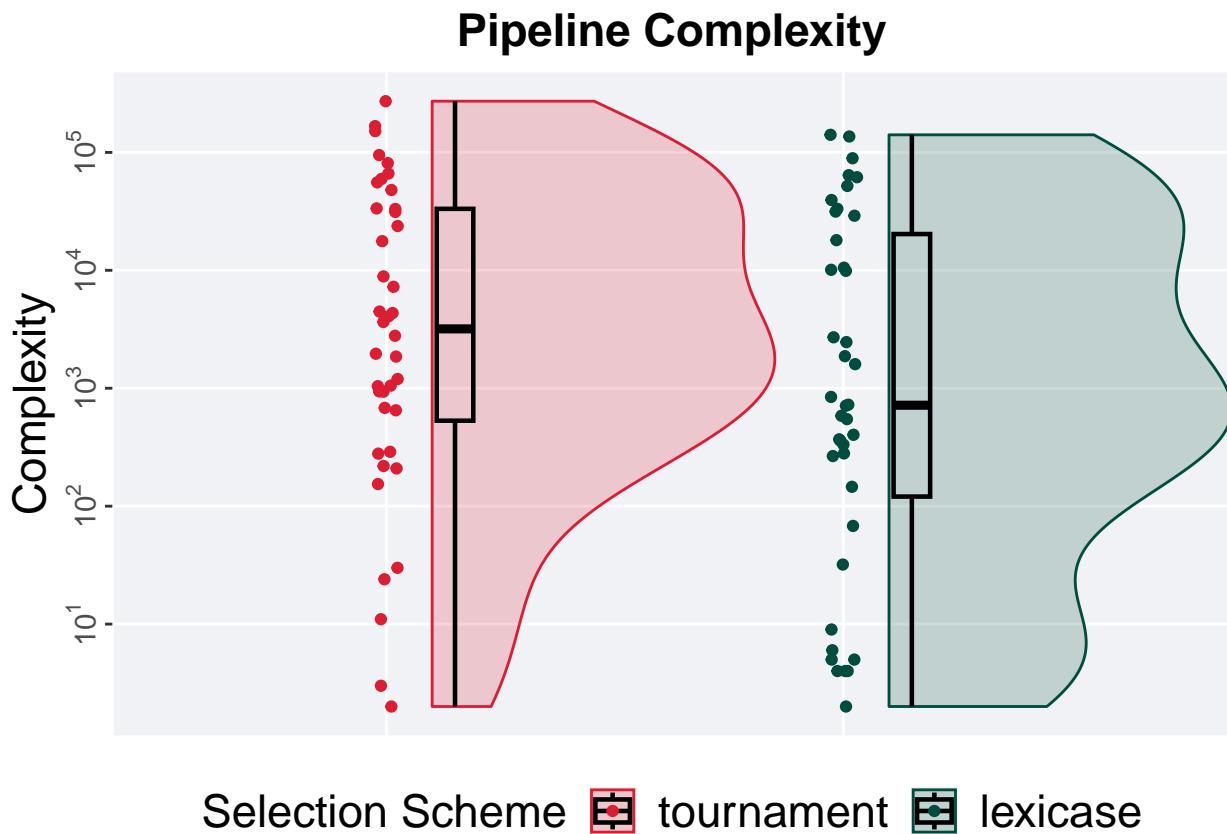
```
tournament_results <- filter(task_data, split == '10%' & selection == 'tournament')
lexicase_results <- filter(task_data, split == '10%' & selection == 'lexicase')
permutation_test(tournament_results$training_performance,
                 lexicase_results$training_performance,
                 seed = 64,
                 alternative = "1")
```

```
## [1] "observed_diff: -4.41624262918063"
## [1] "permutation_diffs[0.05 * n_permutations]: -1.64847676079247"
## [1] "reject null hypothesis"
## [1] "p-value: 1e-05"
```



8.2.3 Pipeline complexity

```
complexity_plot(filter(task_data, split == '10%'))
```



Summary statistics for the pipeline complexity of the selection schemes at the 5% selection set split:

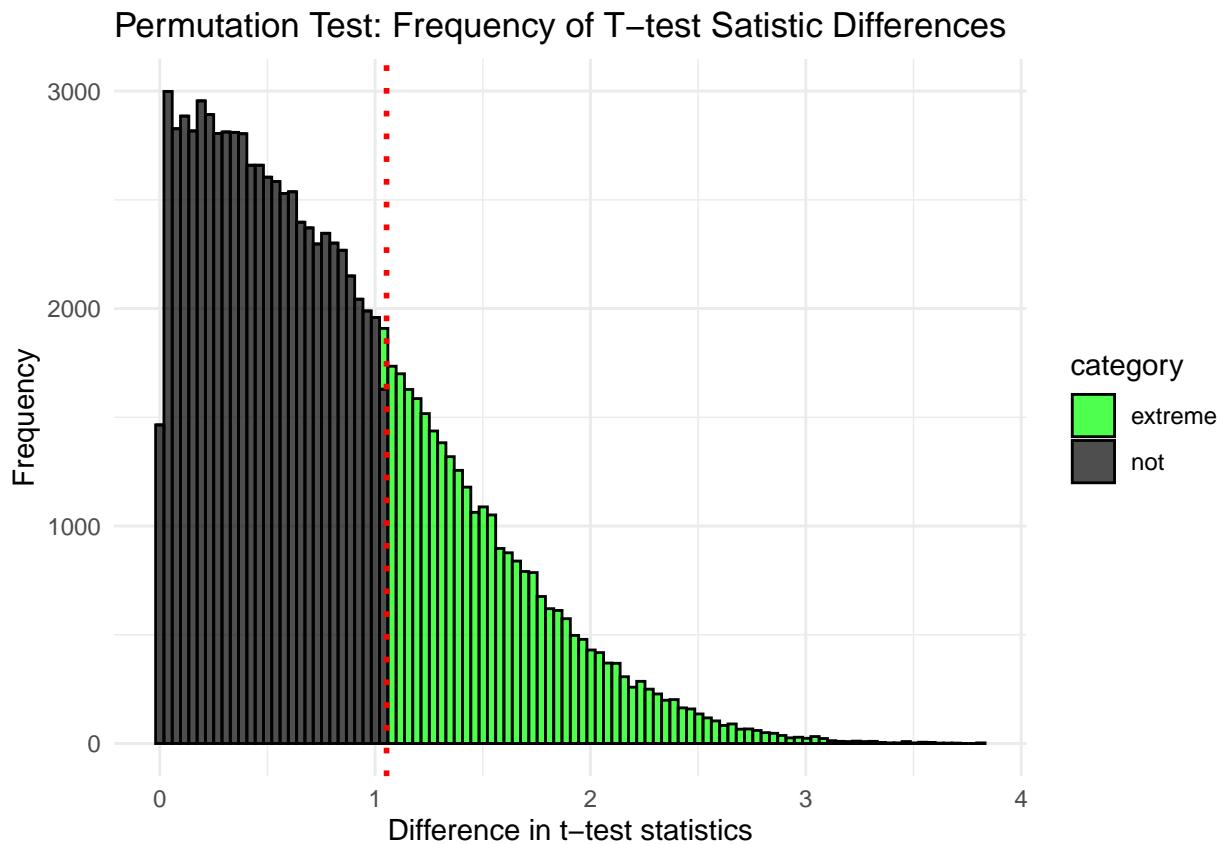
```
complexity_summary(filter(task_data, split == '10%'))
```

```
## # A tibble: 2 x 8
##   selection count na_cnt  min median   mean   max   IQR
##   <fct>     <int> <int> <dbl> <dbl> <dbl> <int> <dbl>
## 1 tournament     40     0     2 3216. 29553. 271801 32662.
## 2 lexicase       40     0     2   718 18511. 141001 20650.
```

The permutation test revealed that the results are:

```
tournament_results <- filter(task_data, split == '10%' & selection == 'tournament')
lexicase_results <- filter(task_data, split == '10%' & selection == 'lexicase')
permutation_test(tournament_results$testing_complexity,
                 lexicase_results$testing_complexity,
                 seed = 221,
                 alternative = "t")
```

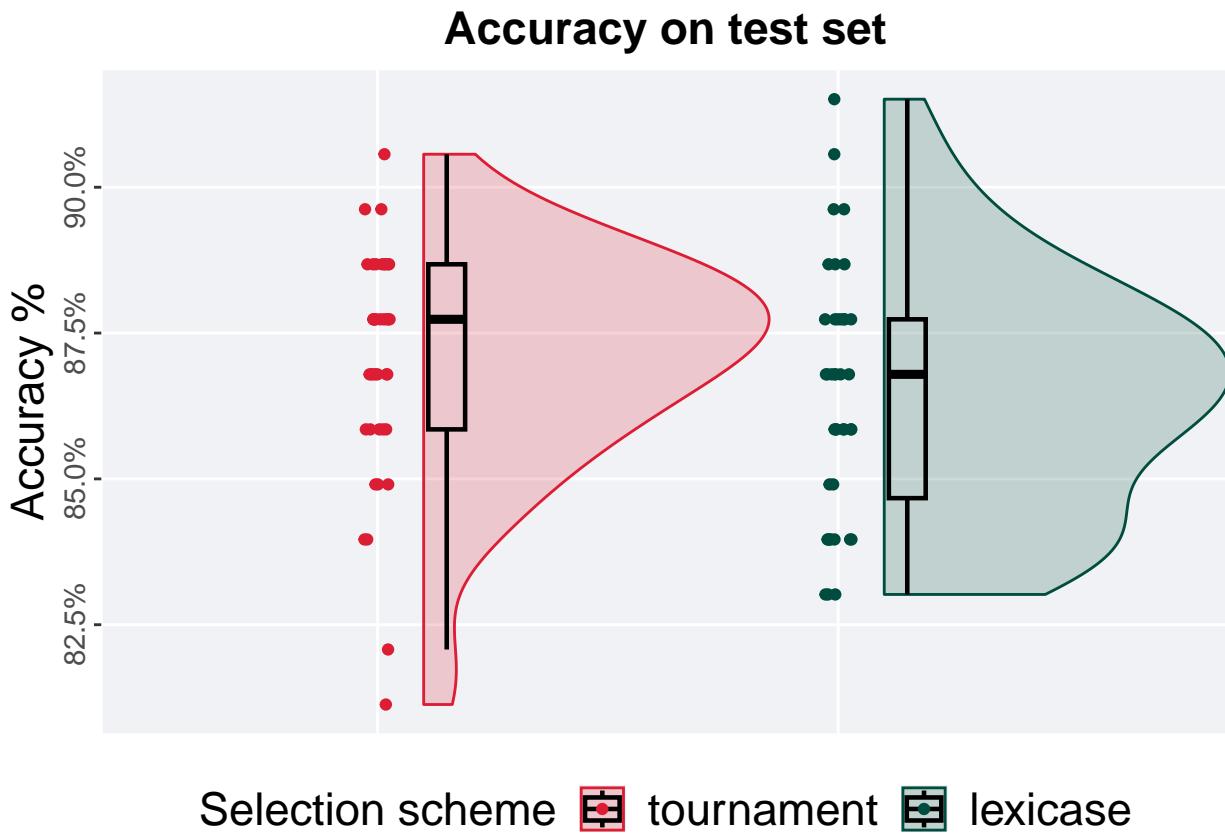
```
## [1] "observed_diff: 1.05307947019019"
## [1] "lower: -1.97261888798967"
## [1] "upper: 1.95253738290975"
## [1] "fail to reject null hypothesis"
## [1] "p-value: 0.30608"
```



8.3 50%

8.3.1 Test accuracy

```
test_plot(filter(task_data, split == '50%'))
```



Summary statistics for the testing performance of the selection schemes at the 5% selection set split:

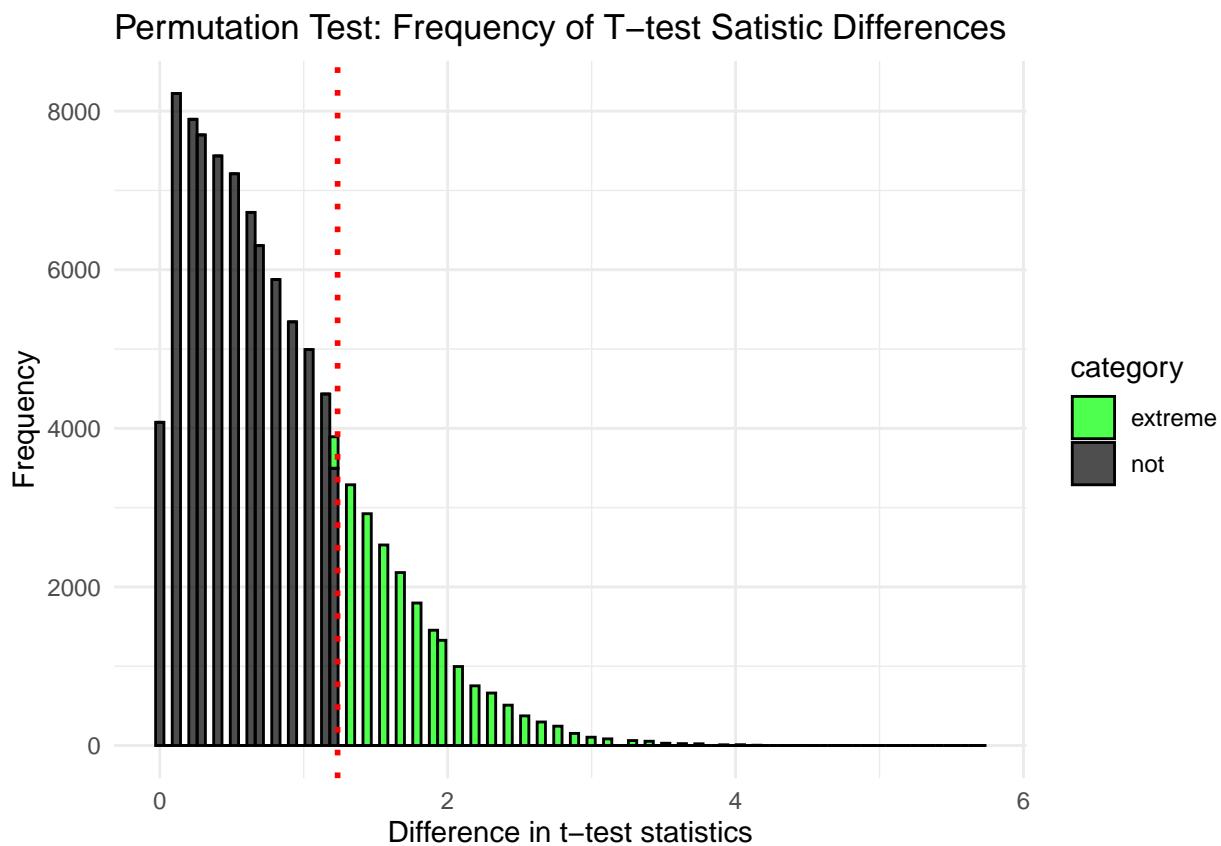
```
test_results_summary(filter(task_data, split == '50%'))
```

```
## # A tibble: 2 x 8
##   selection count na_cnt  min median  mean  max     IQR
##   <fct>     <int> <int> <dbl> <dbl> <dbl> <dbl> <dbl>
## 1 tournament     40     0 0.811 0.877 0.870 0.906 0.0283
## 2 lexicase       40     0 0.830 0.868 0.864 0.915 0.0307
```

The permutation test revealed that the results are:

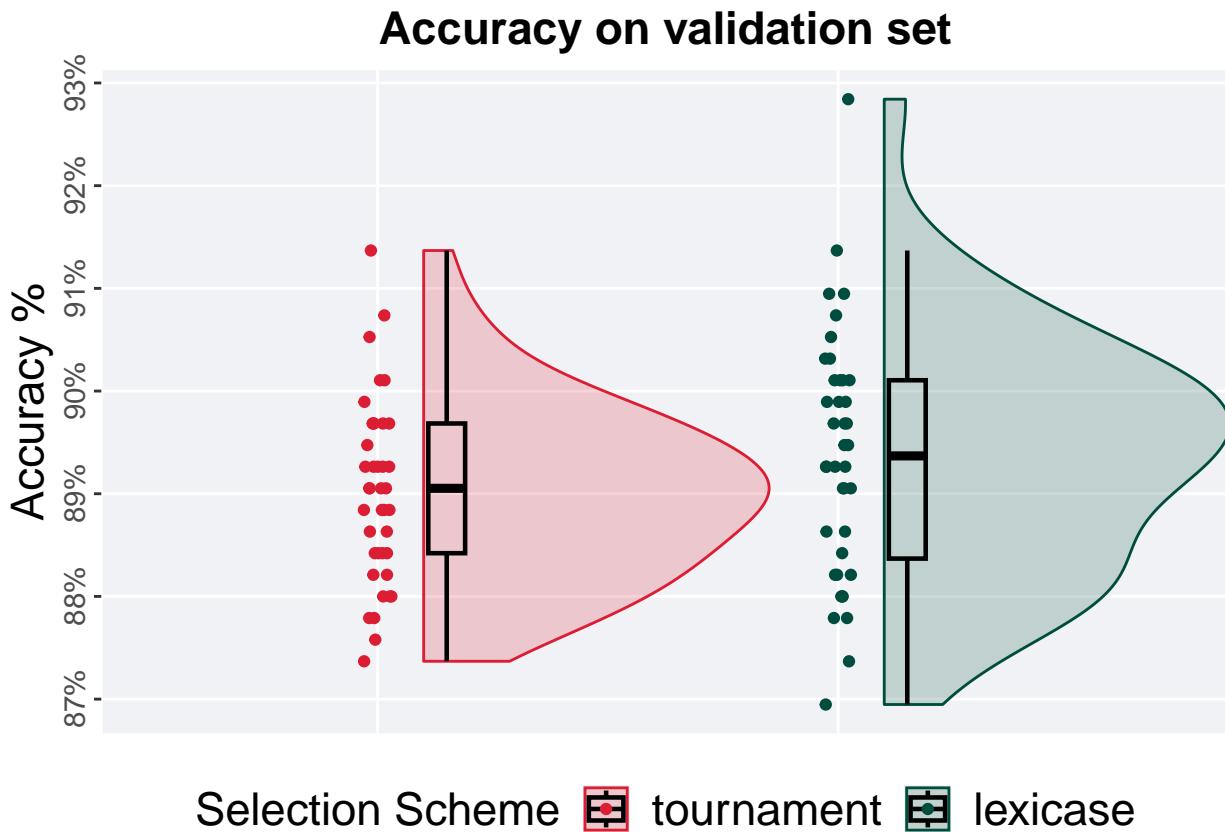
```
tournament_results <- filter(task_data, split == '50%' & selection == 'tournament')
lexicase_results <- filter(task_data, split == '50%' & selection == 'lexicase')
permutation_test(tournament_results$testing_performance,
                 lexicase_results$testing_performance,
                 seed = 65,
                 alternative = "t")
```

```
## [1] "observed_diff: 1.23535069775879"
## [1] "lower: -1.98545740292316"
## [1] "upper: 1.98545740292316"
## [1] "fail to reject null hypothesis"
## [1] "p-value: 0.20292"
```



8.3.2 Validation accuracy

```
validation_plot(filter(task_data, split == '50%'))
```



Summary statistics for the testing performance of the selection schemes at the 5% selection set split:

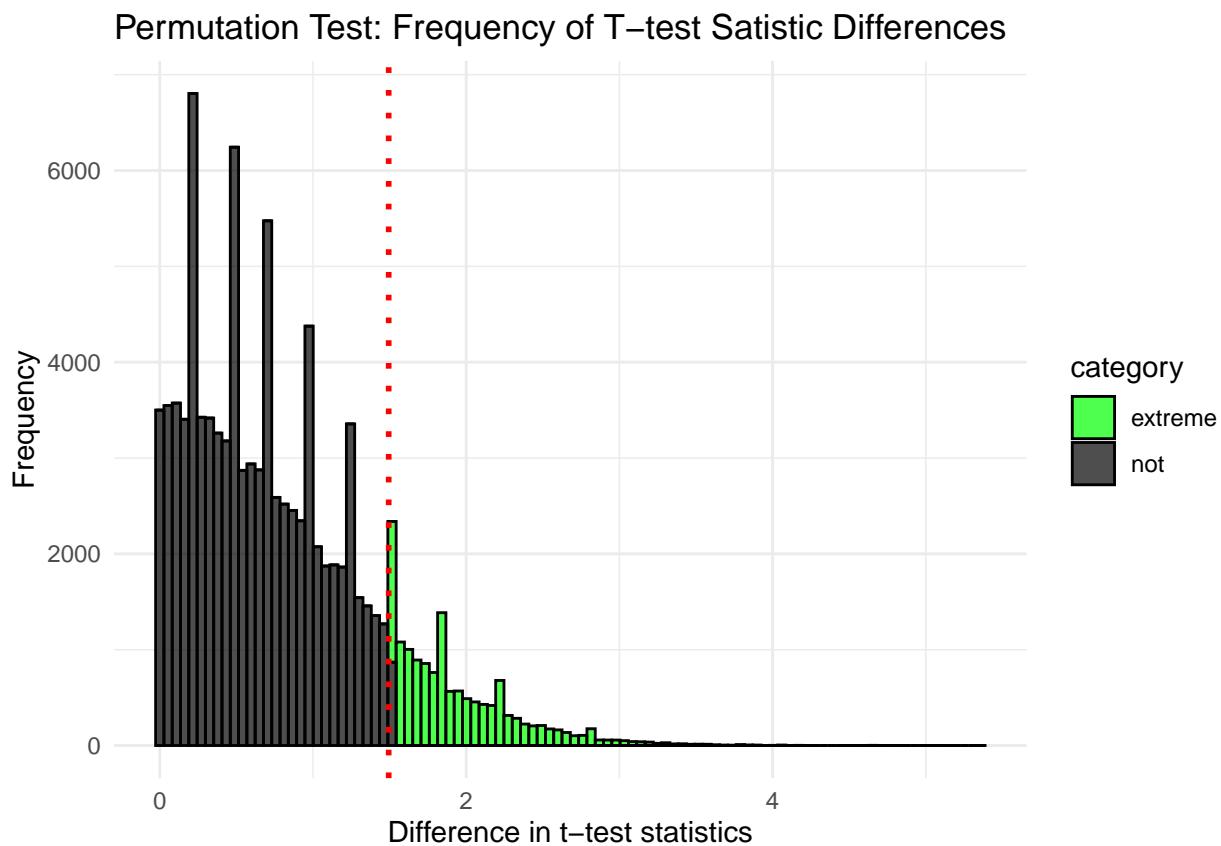
```
validation_accuracy_summary(filter(task_data, split == '50%'))
```

```
## # A tibble: 2 x 8
##   selection count na_cnt   min median   mean   max     IQR
##   <fct>     <int>  <int> <dbl>  <dbl> <dbl> <dbl> <dbl>
## 1 tournament     40      0 0.874  0.891  0.890  0.914  0.0126
## 2 lexicase       40      0 0.869  0.894  0.894  0.928  0.0174
```

The permutation test revealed that the results are:

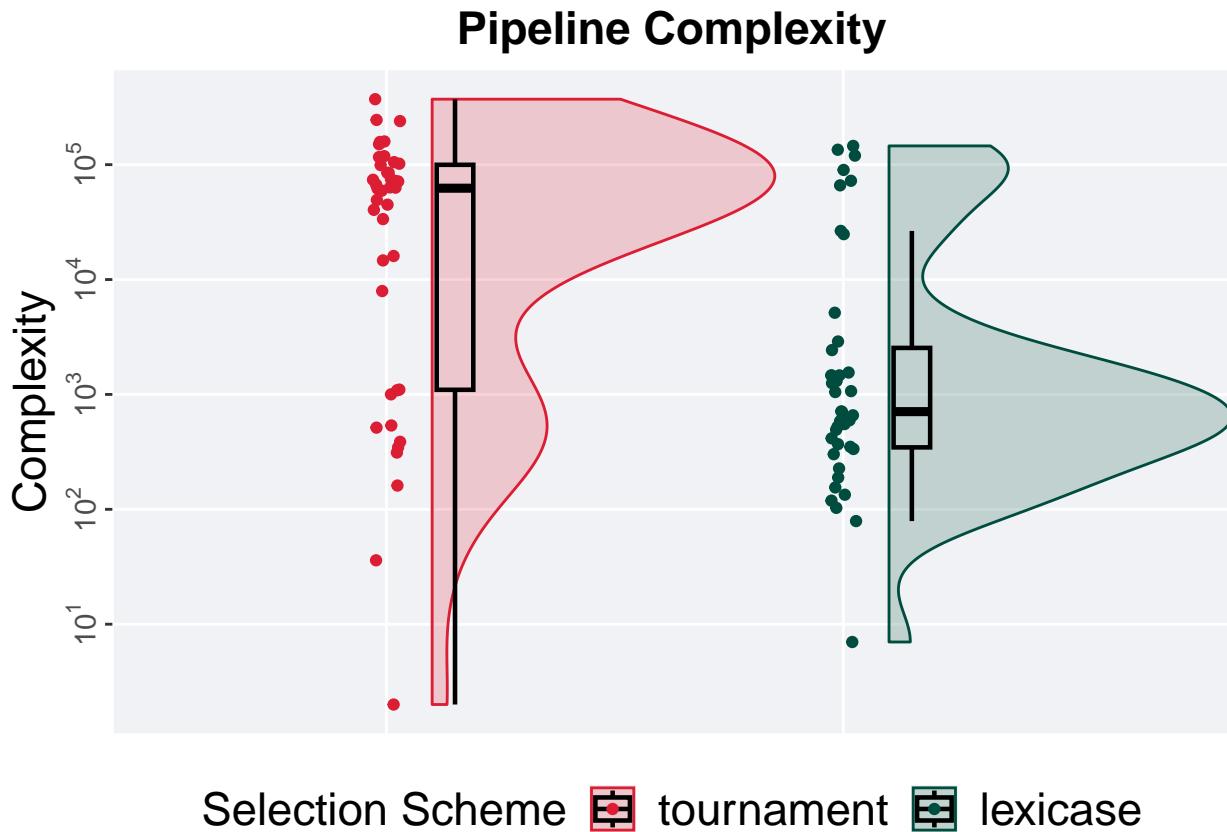
```
tournament_results <- filter(task_data, split == '50%' & selection == 'tournament')
lexicase_results <- filter(task_data, split == '50%' & selection == 'lexicase')
permutation_test(tournament_results$training_performance,
                 lexicase_results$training_performance,
                 seed = 66,
                 alternative = "t")

## [1] "observed_diff: -1.49419701600044"
## [1] "lower: -2.00690565895584"
## [1] "upper: 2.00690565895579"
## [1] "fail to reject null hypothesis"
## [1] "p-value: 0.13664"
```



8.3.3 Pipeline complexity

```
complexity_plot(filter(task_data, split == '50%'))
```



Summary statistics for the pipeline complexity of the selection schemes at the 5% selection set split:

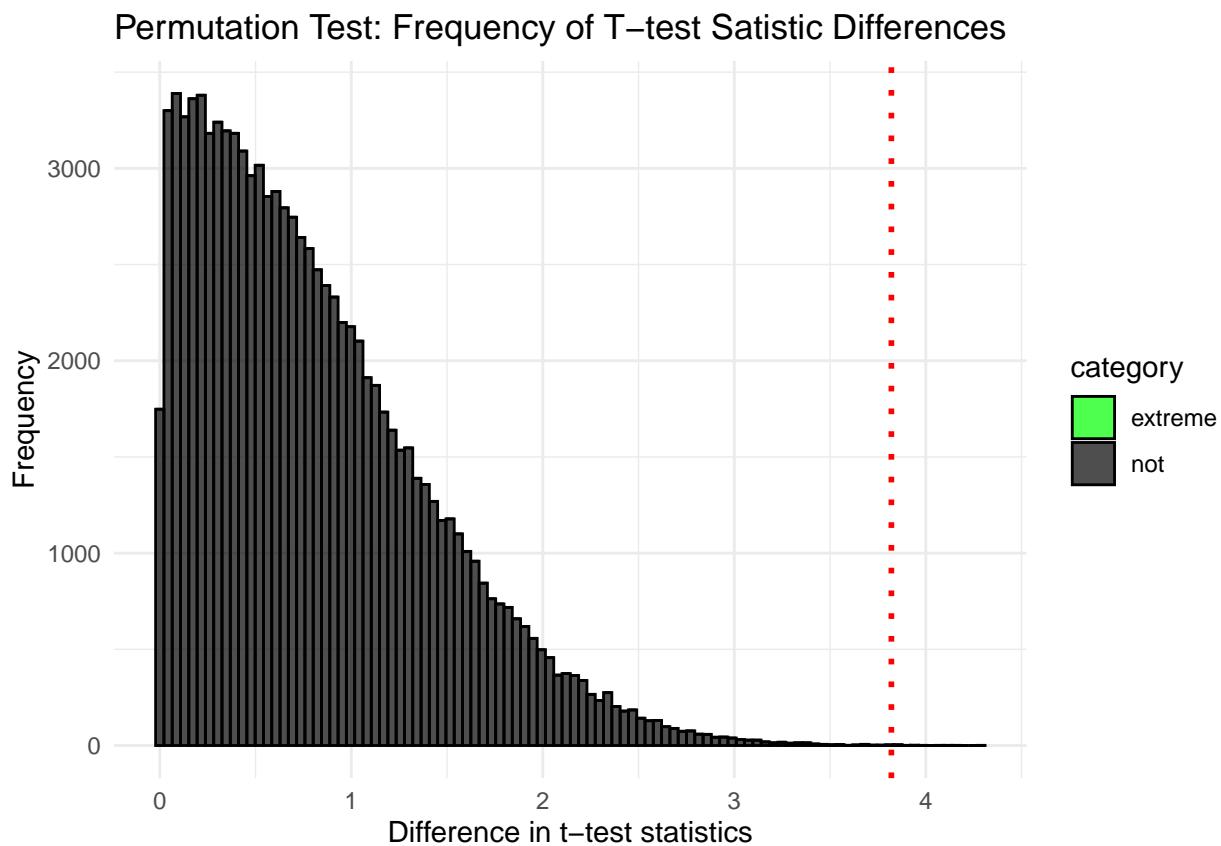
```
complexity_summary(filter(task_data, split == '50%'))
```

```
## # A tibble: 2 x 8
##   selection count na_cnt  min median   mean   max   IQR
##   <fct>     <int> <int> <dbl> <dbl> <dbl> <int> <dbl>
## 1 tournament     40     0     2 62368. 71300. 371231 98584.
## 2 lexicase       40     0     7  708. 17713. 145541  2200
```

The permutation test revealed that the results are:

```
tournament_results <- filter(task_data, split == '50%' & selection == 'tournament')
lexicase_results <- filter(task_data, split == '50%' & selection == 'lexicase')
permutation_test(tournament_results$testing_complexity,
                 lexicase_results$testing_complexity,
                 seed = 222,
                 alternative = "t")
```

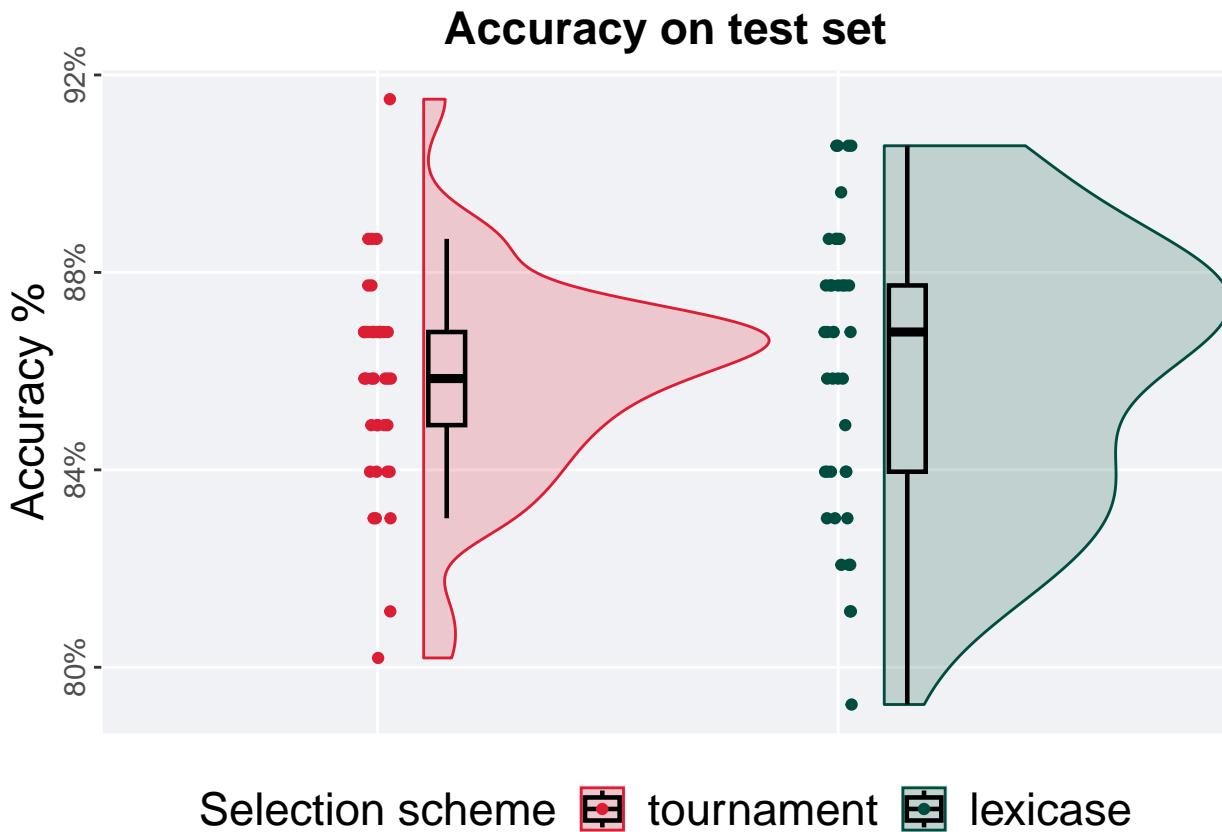
```
## [1] "observed_diff: 3.82033488918495"
## [1] "lower: -1.95987890693742"
## [1] "upper: 1.9744931144761"
## [1] "reject null hypothesis"
## [1] "p-value: 0.00014"
```



8.4 90%

8.4.1 Test accuracy

```
test_plot(filter(task_data, split == '90%'))
```



Summary statistics for the testing performance of the selection schemes at the 5% selection set split:

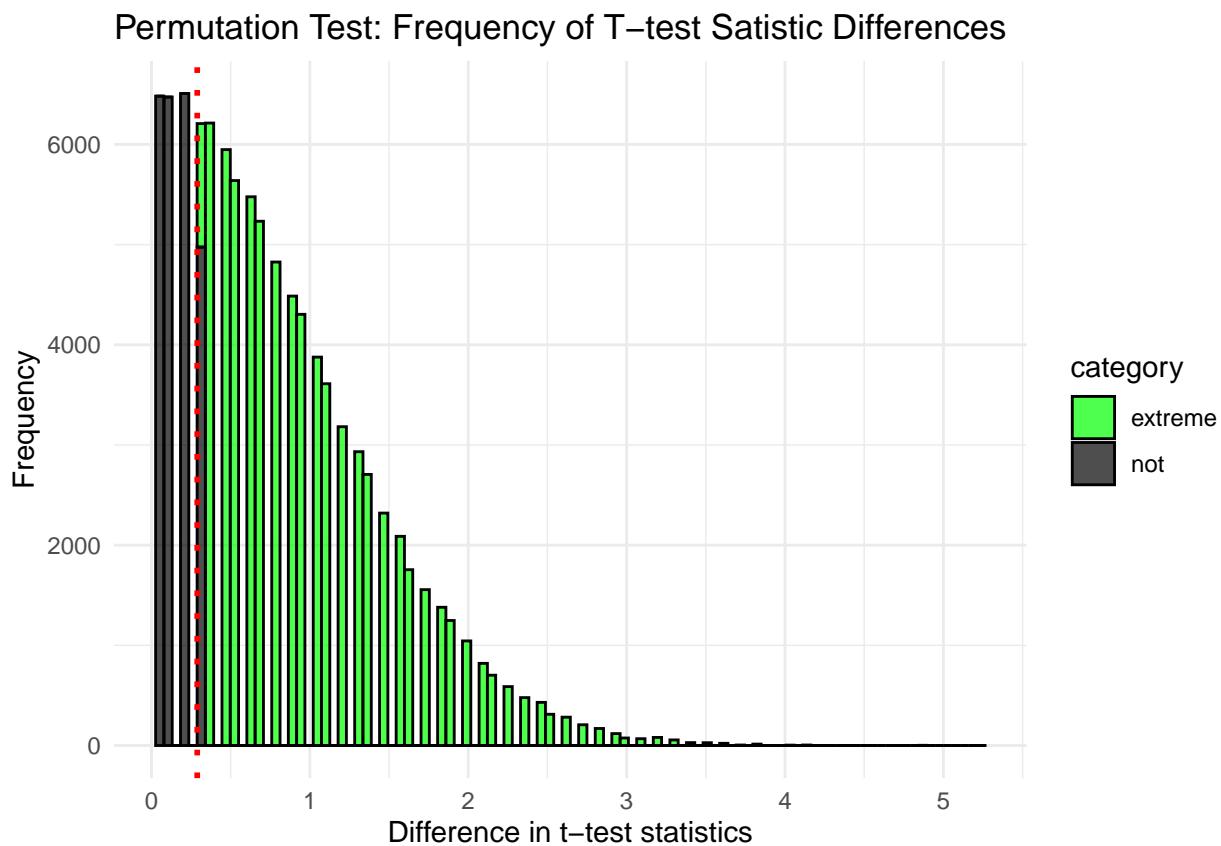
```
test_results_summary(filter(task_data, split == '90%'))
```

```
## # A tibble: 2 x 8
##   selection count na_cnt  min median  mean   max     IQR
##   <fct>     <int> <int> <dbl> <dbl> <dbl> <dbl> <dbl>
## 1 tournament     40     0 0.802 0.858 0.858 0.915 0.0189
## 2 lexicase       40     0 0.792 0.868 0.860 0.906 0.0377
```

The permutation test revealed that the results are:

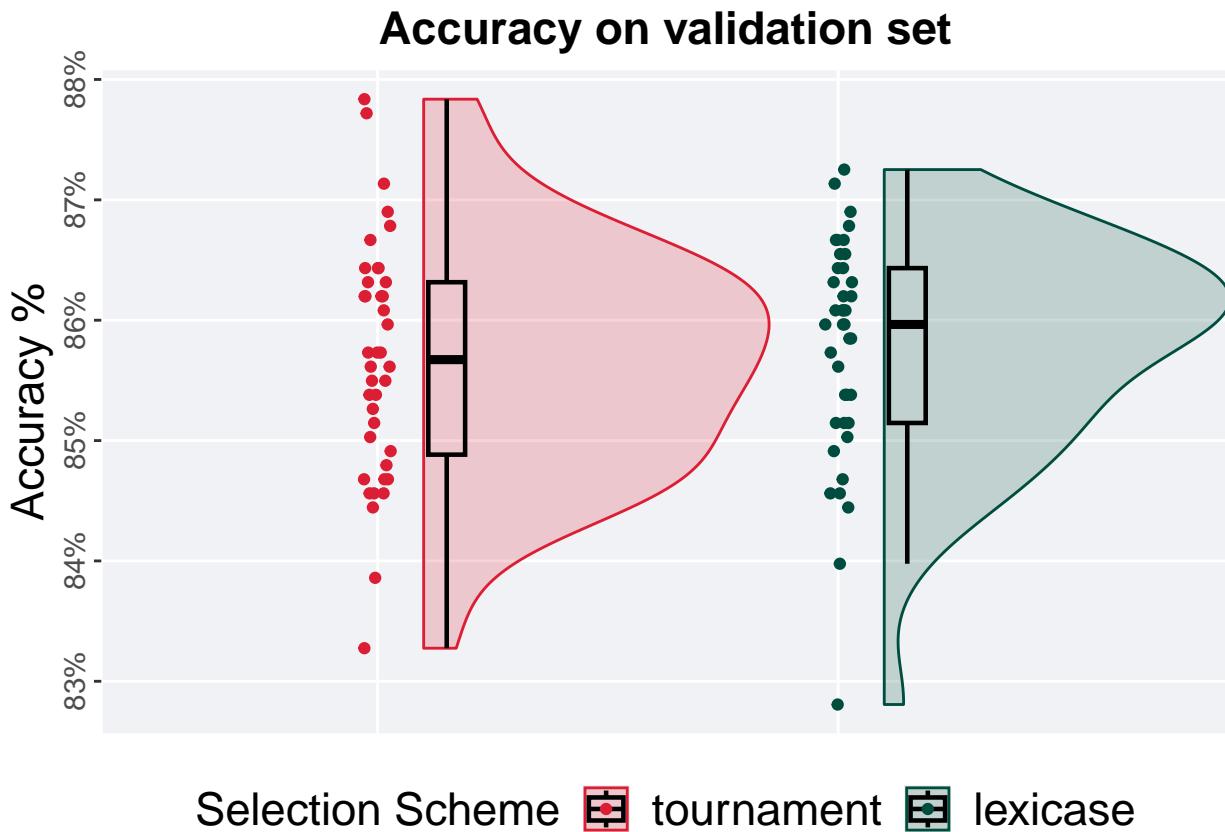
```
tournament_results <- filter(task_data, split == '90%' & selection == 'tournament')
lexicase_results <- filter(task_data, split == '90%' & selection == 'lexicase')
permutation_test(tournament_results$testing_performance,
                 lexicase_results$testing_performance,
                 seed = 67,
                 alternative = "t")
```

```
## [1] "observed_diff: -0.288985153074226"
## [1] "lower: -1.98779654792212"
## [1] "upper: 1.98779748996659"
## [1] "fail to reject null hypothesis"
## [1] "p-value: 0.75562"
```



8.4.2 Validation accuracy

```
validation_plot(filter(task_data, split == '90%'))
```



Summary statistics for the testing performance of the selection schemes at the 5% selection set split:

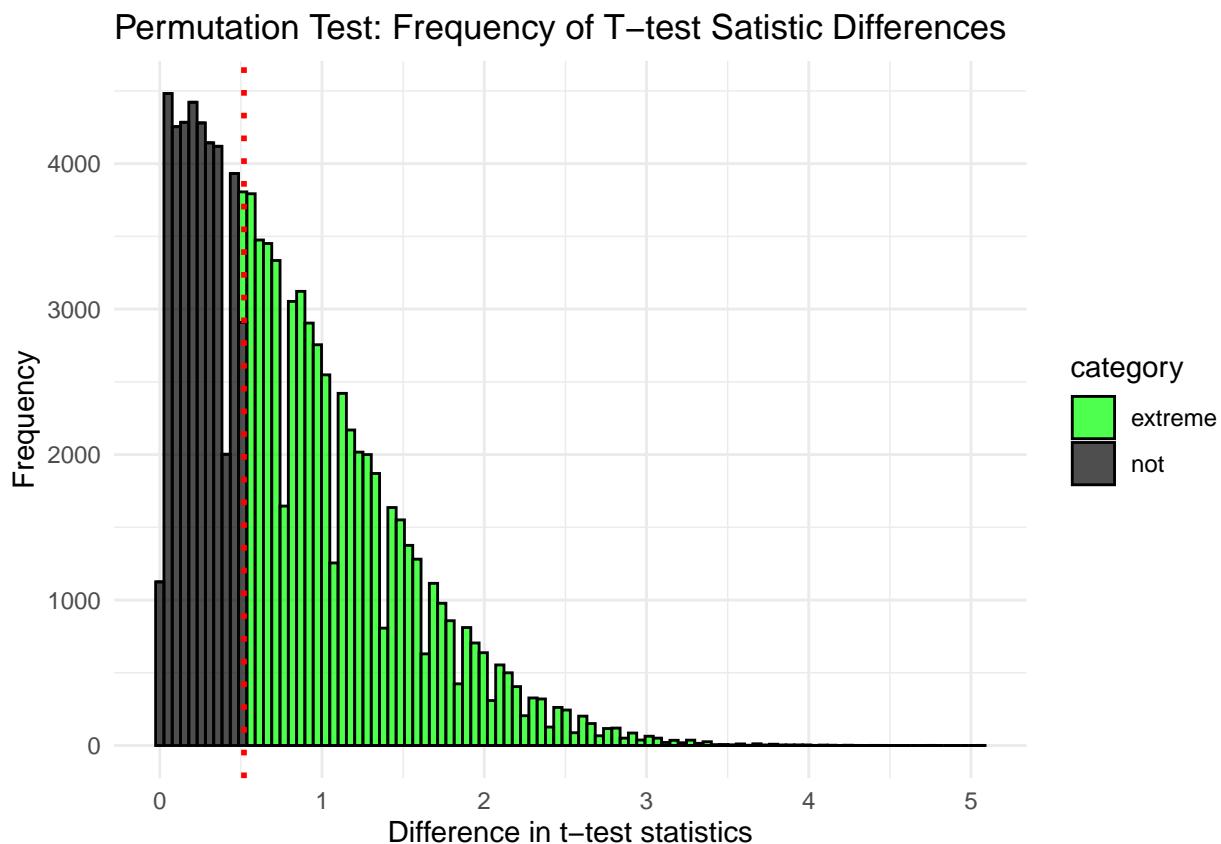
```
validation_accuracy_summary(filter(task_data, split == '90%'))
```

```
## # A tibble: 2 x 8
##   selection count na_cnt   min median   mean   max     IQR
##   <fct>     <int>  <int> <dbl>  <dbl> <dbl> <dbl> <dbl>
## 1 tournament     40      0 0.833  0.857  0.857  0.878  0.0143
## 2 lexicase       40      0 0.828  0.860  0.858  0.873  0.0129
```

The permutation test revealed that the results are:

```
tournament_results <- filter(task_data, split == '90%' & selection == 'tournament')
lexicase_results <- filter(task_data, split == '90%' & selection == 'lexicase')
permutation_test(tournament_results$training_performance,
                 lexicase_results$training_performance,
                 seed = 68,
                 alternative = "t")
```

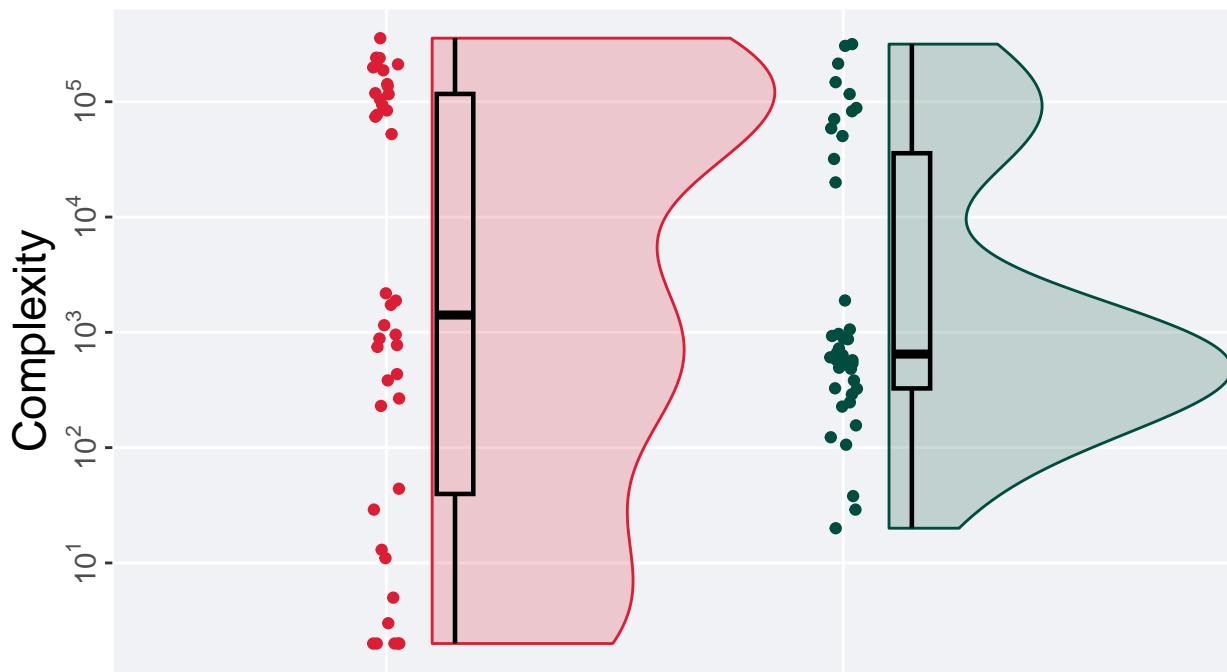
```
## [1] "observed_diff: -0.519105003995363"
## [1] "lower: -1.98476786090786"
## [1] "upper: 1.98476961848034"
## [1] "fail to reject null hypothesis"
## [1] "p-value: 0.60049"
```



8.4.3 Pipeline complexity

```
complexity_plot(filter(task_data, split == '90%'))
```

Pipeline Complexity



Selection Scheme tournament lexicase

Summary statistics for the pipeline complexity of the selection schemes at the 5% selection set split:

```
complexity_summary(filter(task_data, split == '90%'))
```

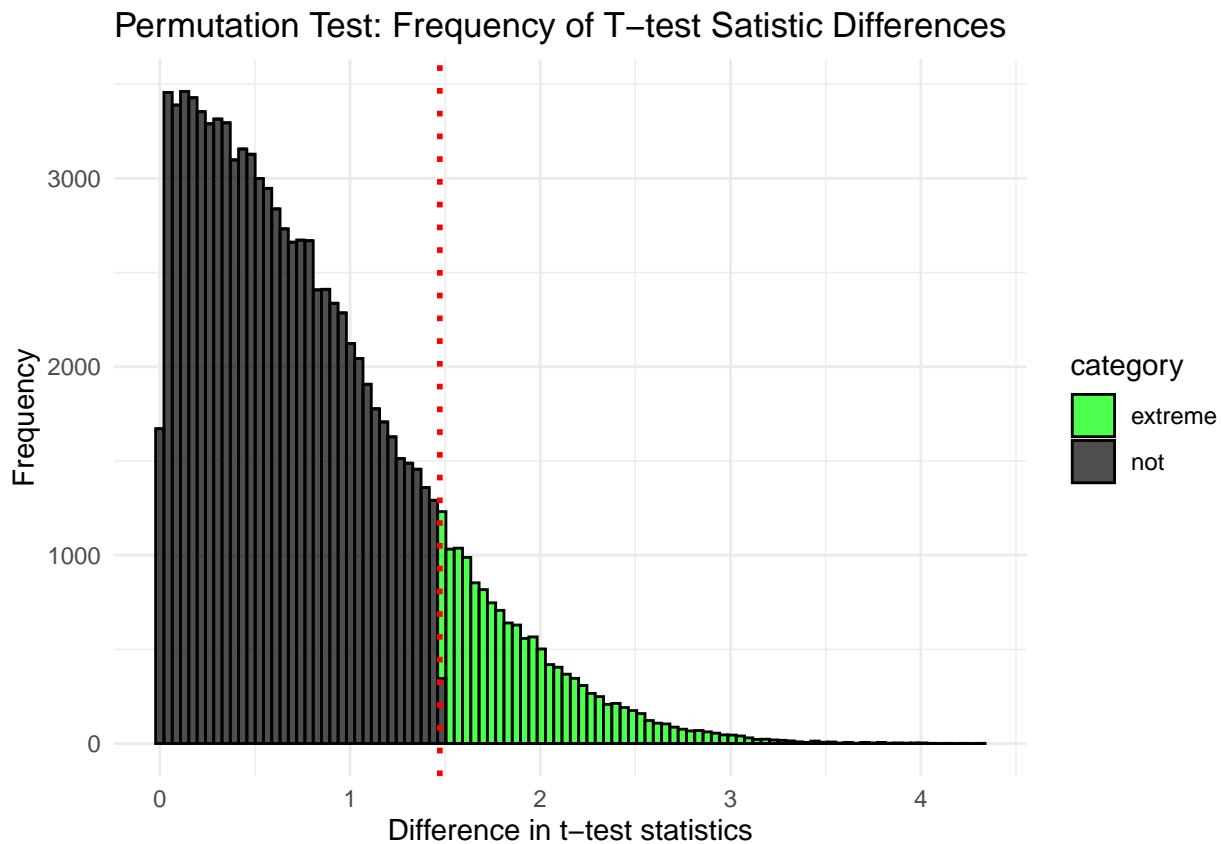
```
## # A tibble: 2 x 8
##   selection  count  na_cnt  min  median  mean  max      IQR
##   <fct>     <int>   <int> <dbl>   <dbl>  <dbl> <dbl>    <dbl>
## 1 tournament     40      0      2  1442  66310. 355841 116831.
## 2 lexicase       40      0     20    648  37939. 316081  36198.
```

The permutation test revealed that the results are:

```
tournament_results <- filter(task_data, split == '90%' & selection == 'tournament')
lexicase_results <- filter(task_data, split == '90%' & selection == 'lexicase')
```

```
permutation_test(tournament_results$testing_complexity,
                  lexicase_results$testing_complexity,
                  seed = 223,
                  alternative = "t")
```

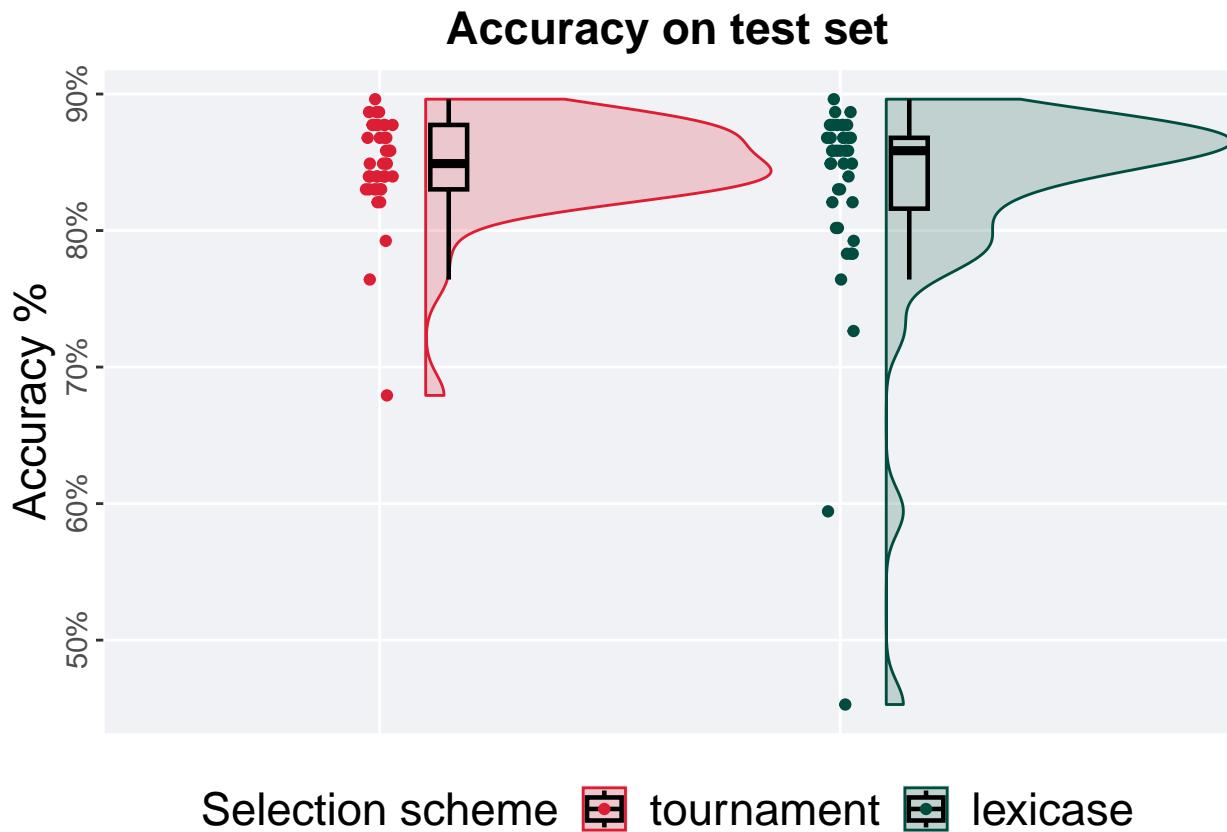
```
## [1] "observed_diff: 1.47197882300349"
## [1] "lower: -1.95879060294153"
## [1] "upper: 1.99322182046029"
## [1] "fail to reject null hypothesis"
## [1] "p-value: 0.14362"
```



8.5 95%

8.5.1 Test accuracy

```
test_plot(filter(task_data, split == '95%'))
```



Summary statistics for the testing performance of the selection schemes at the 5% selection set split:

```
test_results_summary(filter(task_data, split == '95%'))
```

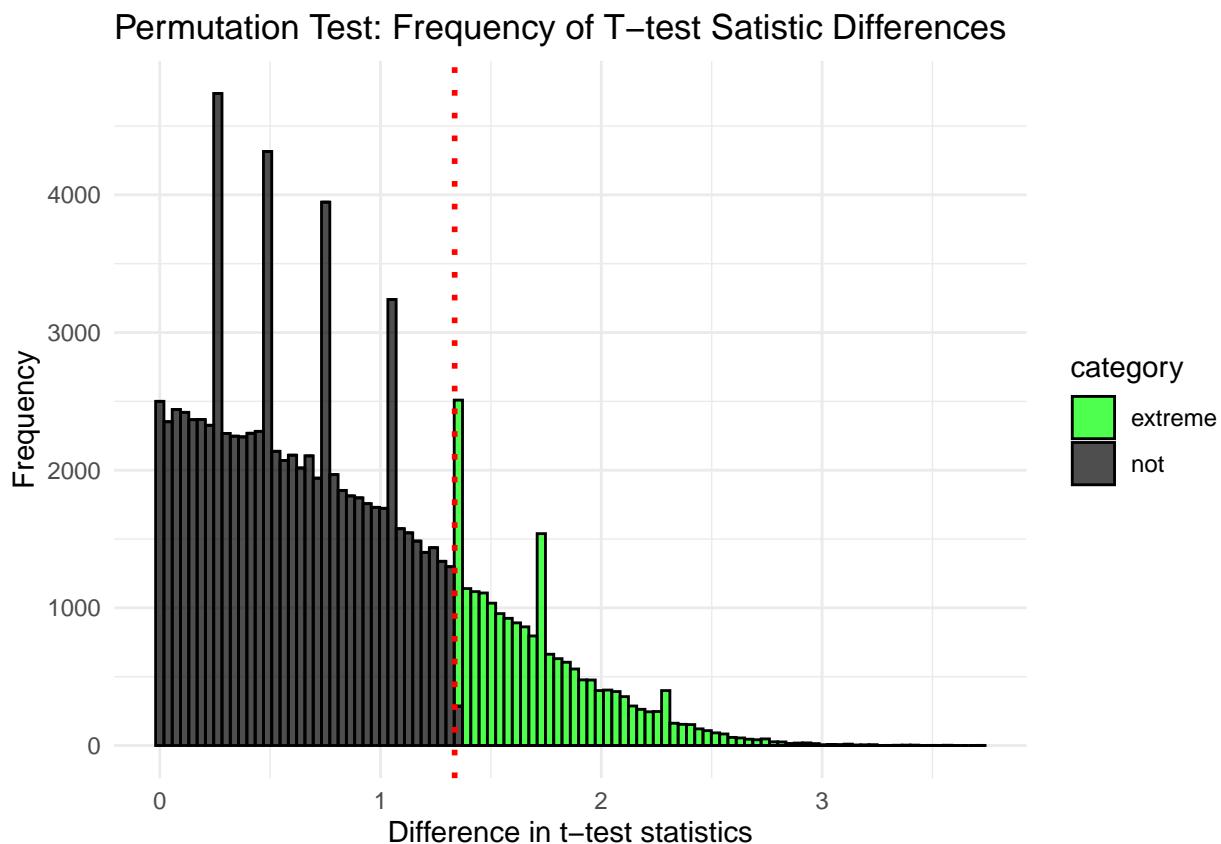
```
## # A tibble: 2 x 8
##   selection count na_cnt  min median  mean   max     IQR
##   <fct>     <int> <int> <dbl> <dbl> <dbl> <dbl> <dbl>
## 1 tournament     40     0 0.679 0.849 0.847 0.896 0.0472
## 2 lexicase       40     0 0.453 0.858 0.828 0.896 0.0519
```

The permutation test revealed that the results are:

```
tournament_results <- filter(task_data, split == '95%' & selection == 'tournament')
lexicase_results <- filter(task_data, split == '95%' & selection == 'lexicase')
```

```
permutation_test(tournament_results$testing_performance,
                  lexicase_results$testing_performance,
                  seed = 69,
                  alternative = "t")
```

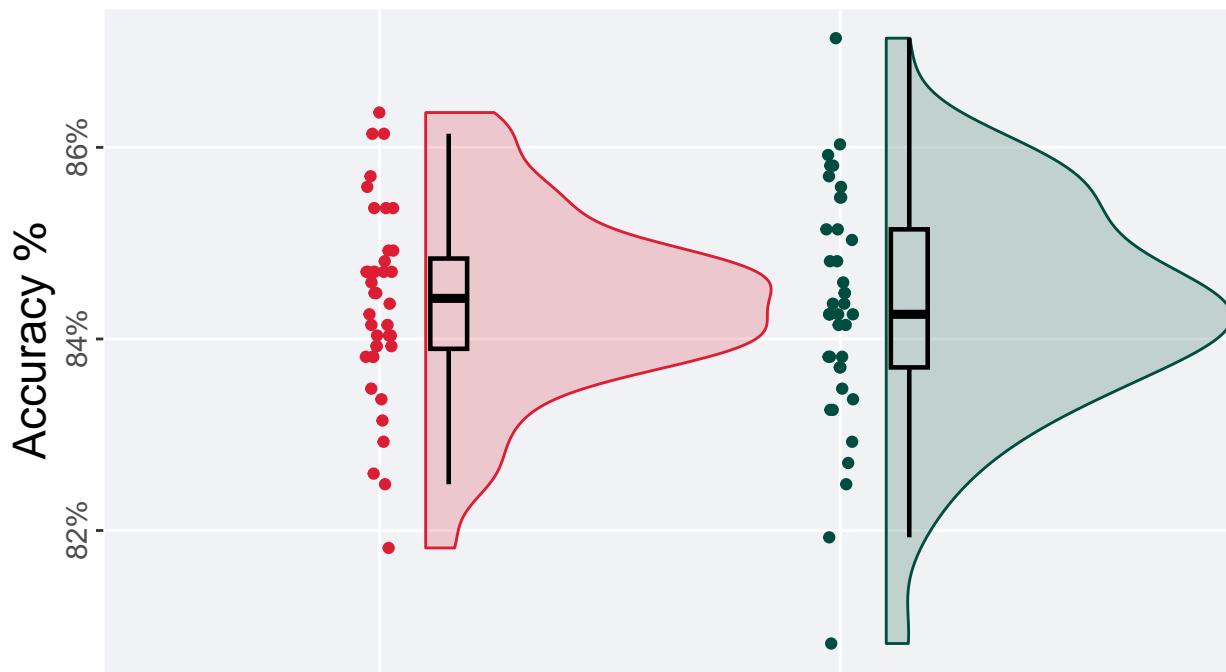
```
## [1] "observed_diff: 1.33590135470976"
## [1] "lower: -1.91908070055885"
## [1] "upper: 1.91908070055885"
## [1] "fail to reject null hypothesis"
## [1] "p-value: 0.20293"
```



8.5.2 Validation accuracy

```
validation_plot(filter(task_data, split == '95%'))
```

Accuracy on validation set



Selection Scheme tournament lexicase

Summary statistics for the testing performance of the selection schemes at the 5% selection set split:

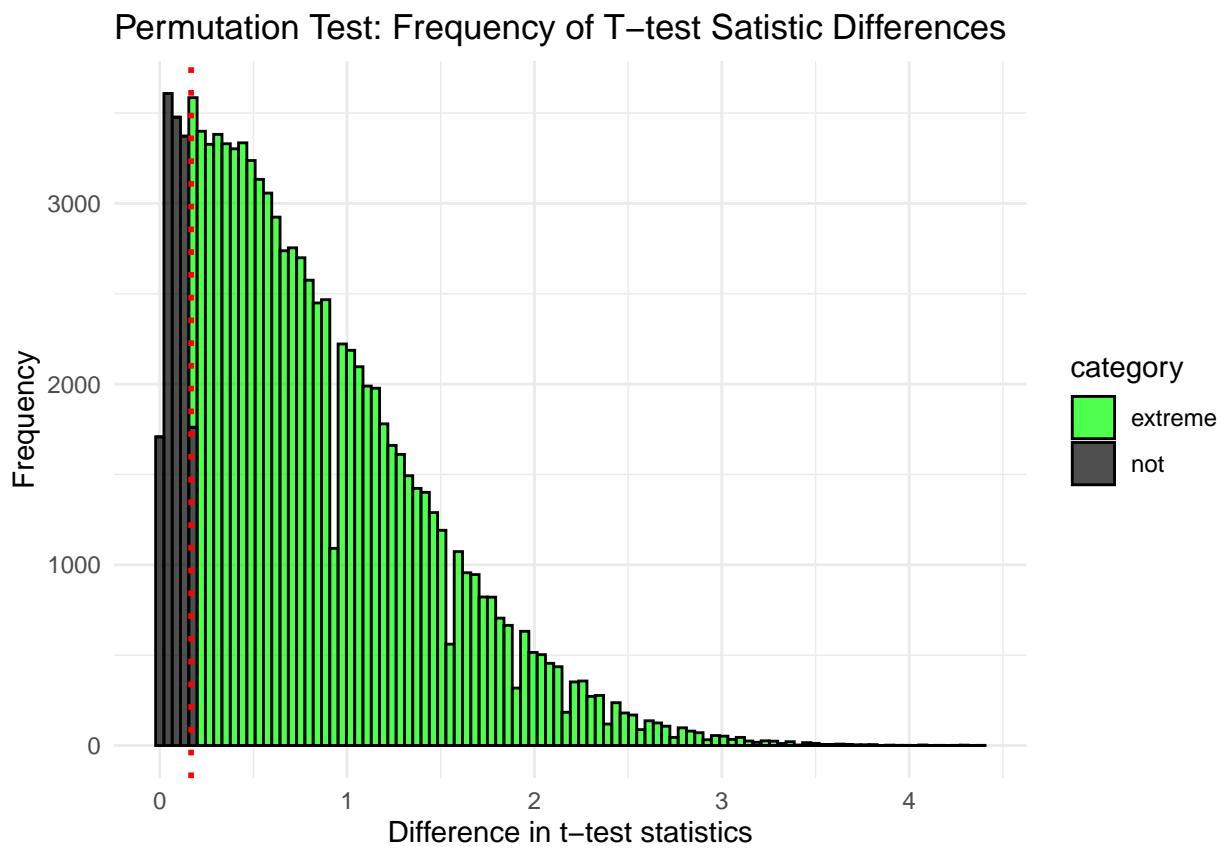
```
validation_accuracy_summary(filter(task_data, split == '95%'))
```

```
## # A tibble: 2 x 8
##   selection count na_cnt  min median  mean   max      IQR
##   <fct>     <int> <int> <dbl> <dbl> <dbl> <dbl> <dbl>
## 1 tournament     40     0 0.818 0.844 0.844 0.864 0.00942
## 2 lexicase       40     0 0.808 0.843 0.843 0.871 0.0144
```

The permutation test revealed that the results are:

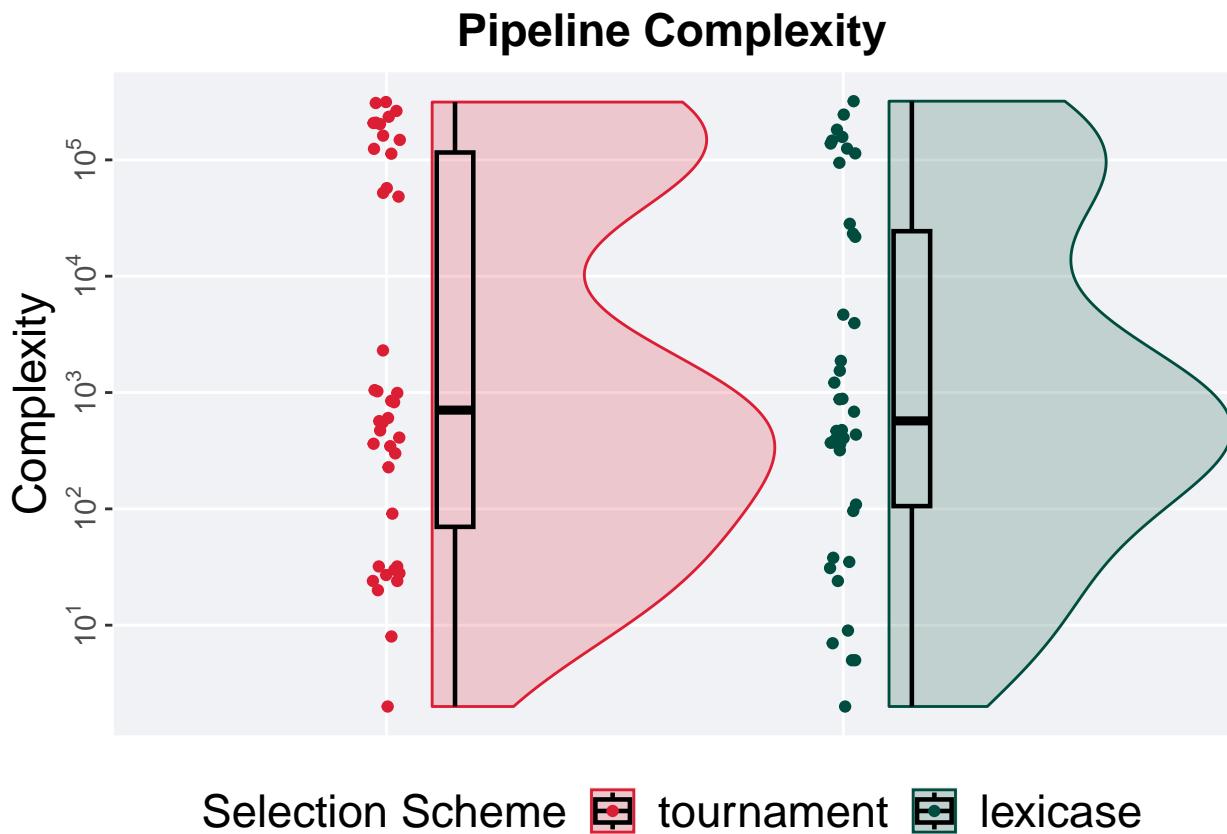
```
tournament_results <- filter(task_data, split == '95%' & selection == 'tournament')
lexicase_results <- filter(task_data, split == '95%' & selection == 'lexicase')
permutation_test(tournament_results$training_performance,
                 lexicase_results$training_performance,
                 seed = 70,
                 alternative = "t")
```

```
## [1] "observed_diff: 0.167199522669578"
## [1] "lower: -1.9996181495468"
## [1] "upper: 1.97561829637897"
## [1] "fail to reject null hypothesis"
## [1] "p-value: 0.86074"
```



8.5.3 Pipeline complexity

```
complexity_plot(filter(task_data, split == '95%'))
```



Summary statistics for the pipeline complexity of the selection schemes at the 5% selection set split:

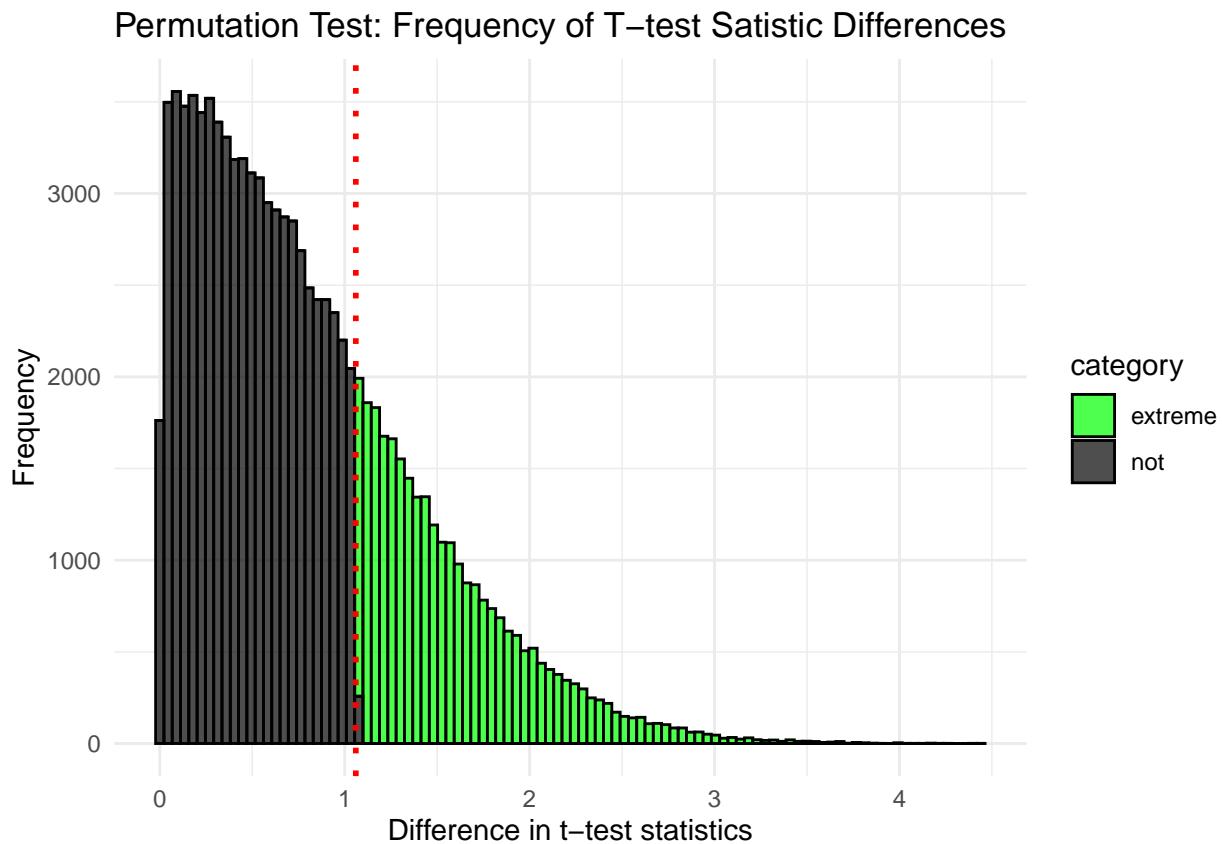
```
complexity_summary(filter(task_data, split == '95%'))
```

```
## # A tibble: 2 x 8
##   selection count na_cnt  min median   mean   max     IQR
##   <fct>     <int> <int> <int> <dbl> <dbl> <int> <dbl>
## 1 tournament     40     0     2    715 61419. 314201 115780.
## 2 lexicase       40     0     2    580 40371. 319591  24364
```

The permutation test revealed that the results are:

```
tournament_results <- filter(task_data, split == '95%' & selection == 'tournament')
lexicase_results <- filter(task_data, split == '95%' & selection == 'lexicase')
permutation_test(tournament_results$testing_complexity,
                 lexicase_results$testing_complexity,
                 seed = 224,
                 alternative = "t")
```

```
## [1] "observed_diff: 1.06010023604268"
## [1] "lower: -1.99701492048251"
## [1] "upper: 1.99935933922988"
## [1] "fail to reject null hypothesis"
## [1] "p-value: 0.29495"
```



Chapter 9

Task 359958

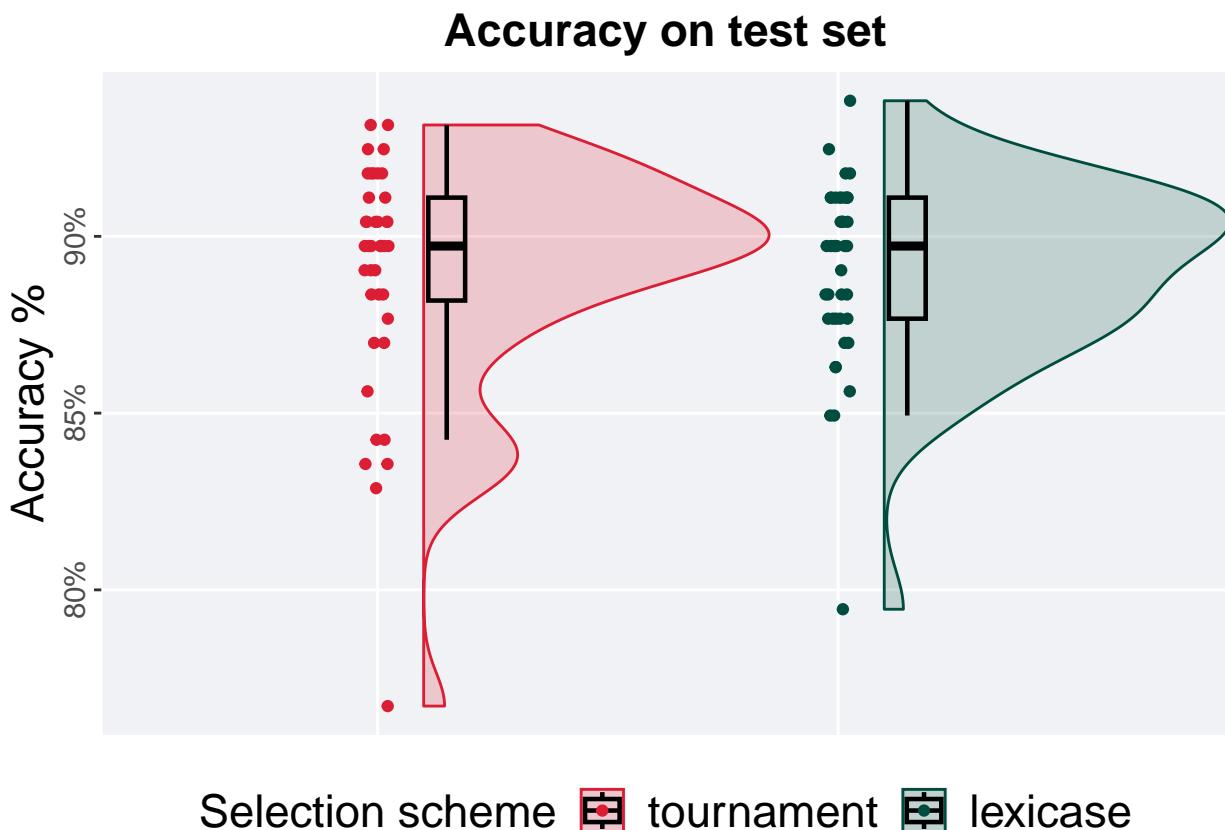
We present the results of our analysis of task 359958 with the different selection set splits used in our study.

```
task_data <- filter(results, task_id == 359958)
```

9.1 5%

9.1.1 Test accuracy

```
test_plot(filter(task_data, split == '5%'))
```



Summary statistics for the testing performance of the selection schemes at the 5% selection set split:

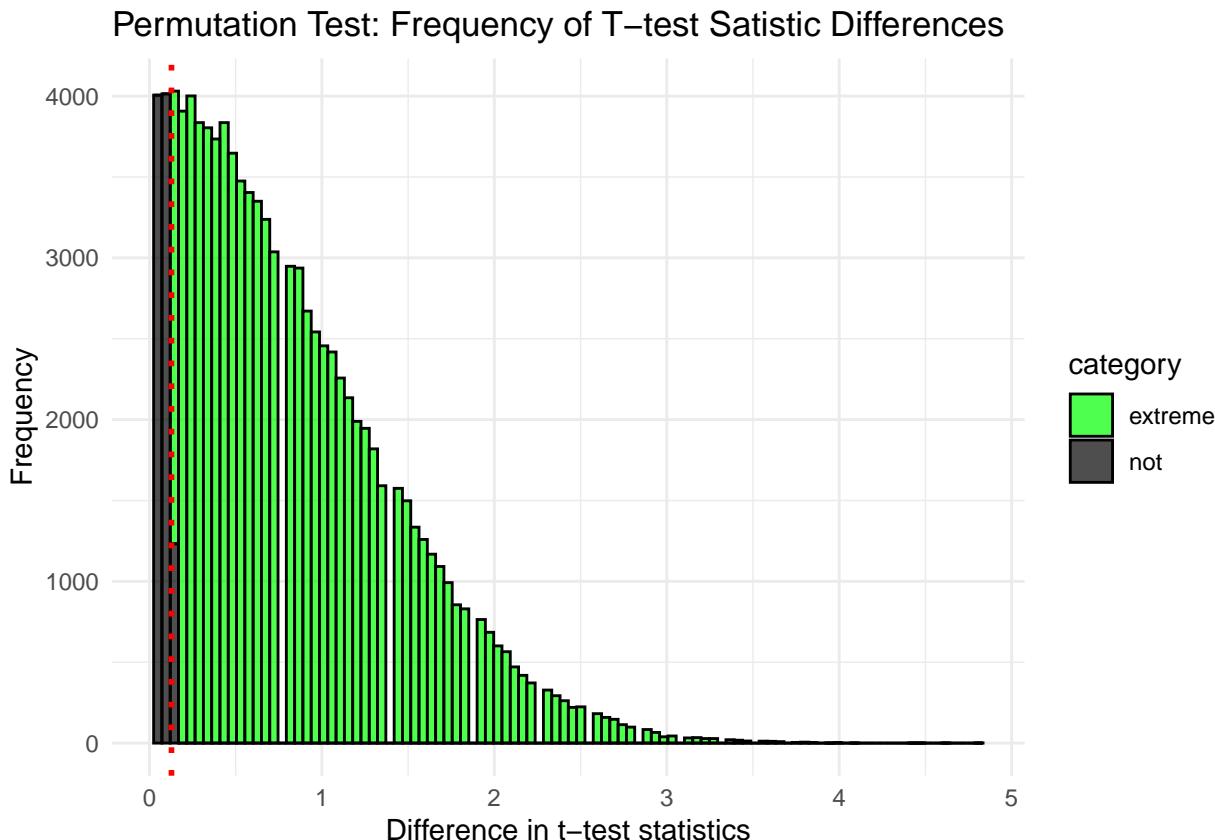
```
test_results_summary(filter(task_data, split == '5%'))
```

```
## # A tibble: 2 x 8
##   selection count na_cnt  min median  mean  max    IQR
##   <fct>     <int> <int> <dbl> <dbl> <dbl> <dbl> <dbl>
## 1 tournament     40     0 0.767 0.897 0.889 0.932 0.0291
## 2 lexicase       40     0 0.795 0.897 0.890 0.938 0.0342
```

The permutation test revealed that the results are:

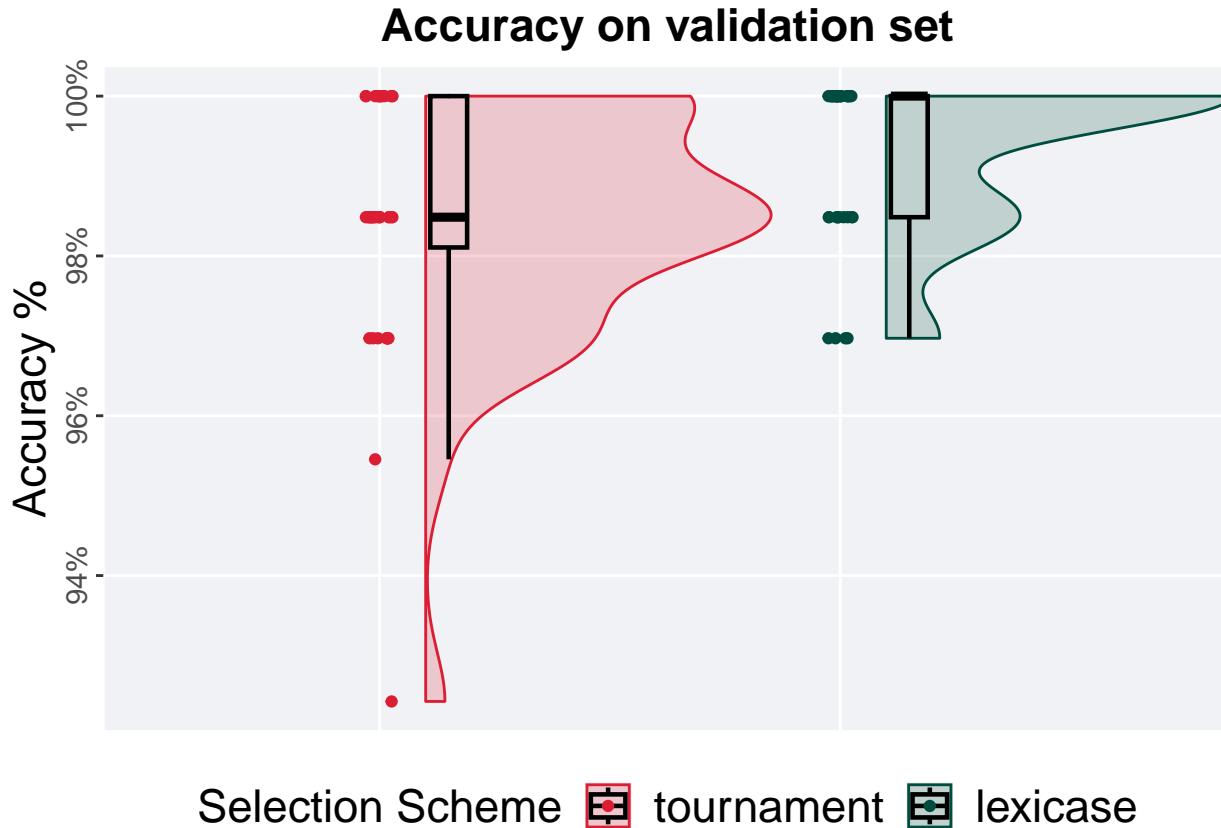
```
tournament_results <- filter(task_data, split == '5%' & selection == 'tournament')
lexicase_results <- filter(task_data, split == '5%' & selection == 'lexicase')
permutation_test(tournament_results$testing_performance,
                 lexicase_results$testing_performance,
                 seed = 71,
                 alternative = "t")
```

```
## [1] "observed_diff: -0.127556855492129"
## [1] "lower: -1.95969411811632"
## [1] "upper: 1.95969419817028"
## [1] "fail to reject null hypothesis"
## [1] "p-value: 0.90748"
```



9.1.2 Validation accuracy

```
validation_plot(filter(task_data, split == '5%'))
```



Summary statistics for the testing performance of the selection schemes at the 5% selection set split:

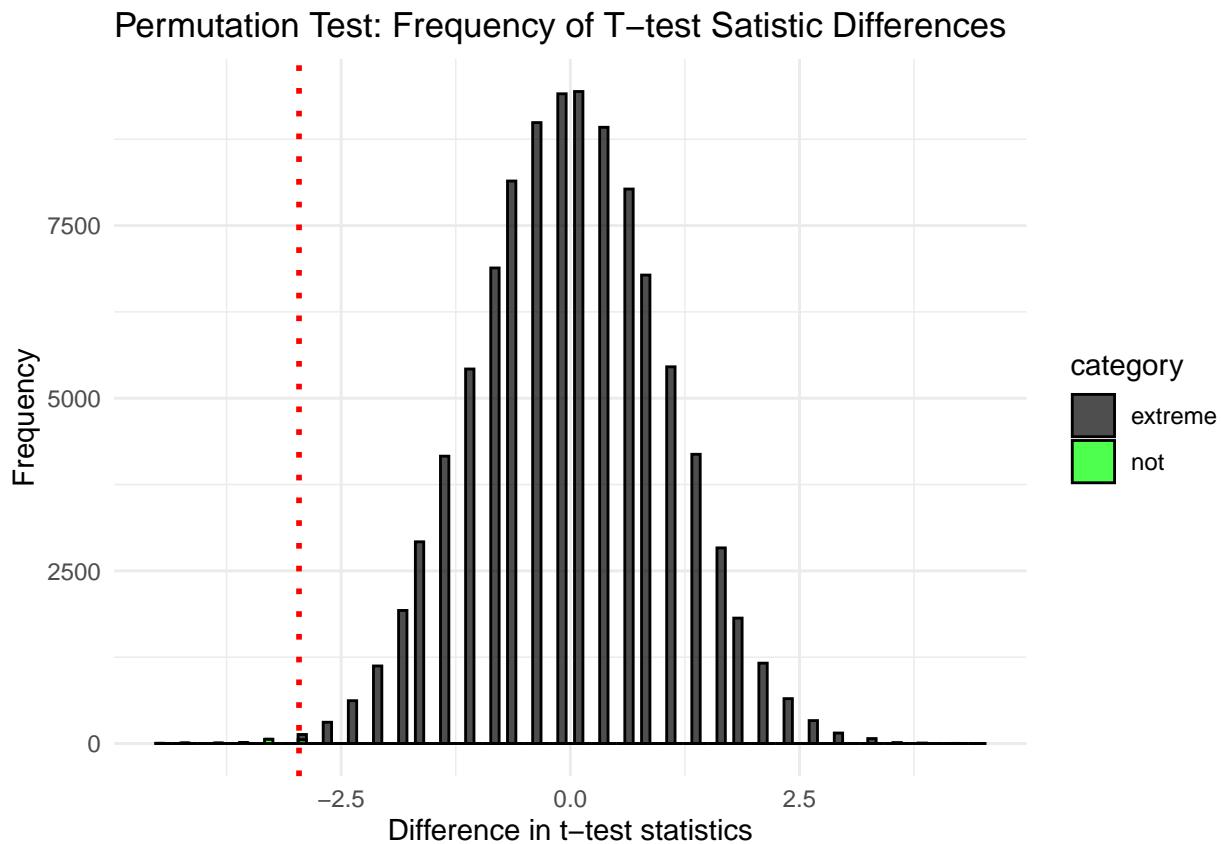
```
validation_accuracy_summary(filter(task_data, split == '5%'))
```

```
## # A tibble: 2 x 8
##   selection count na_cnt  min median  mean   max     IQR
##   <fct>     <int> <int> <dbl> <dbl> <dbl> <dbl> <dbl>
## 1 tournament     40     0 0.924  0.985  0.984     1  0.0189
## 2 lexicase       40     0 0.970   1      0.993     1  0.0152
```

The permutation test revealed that the results are:

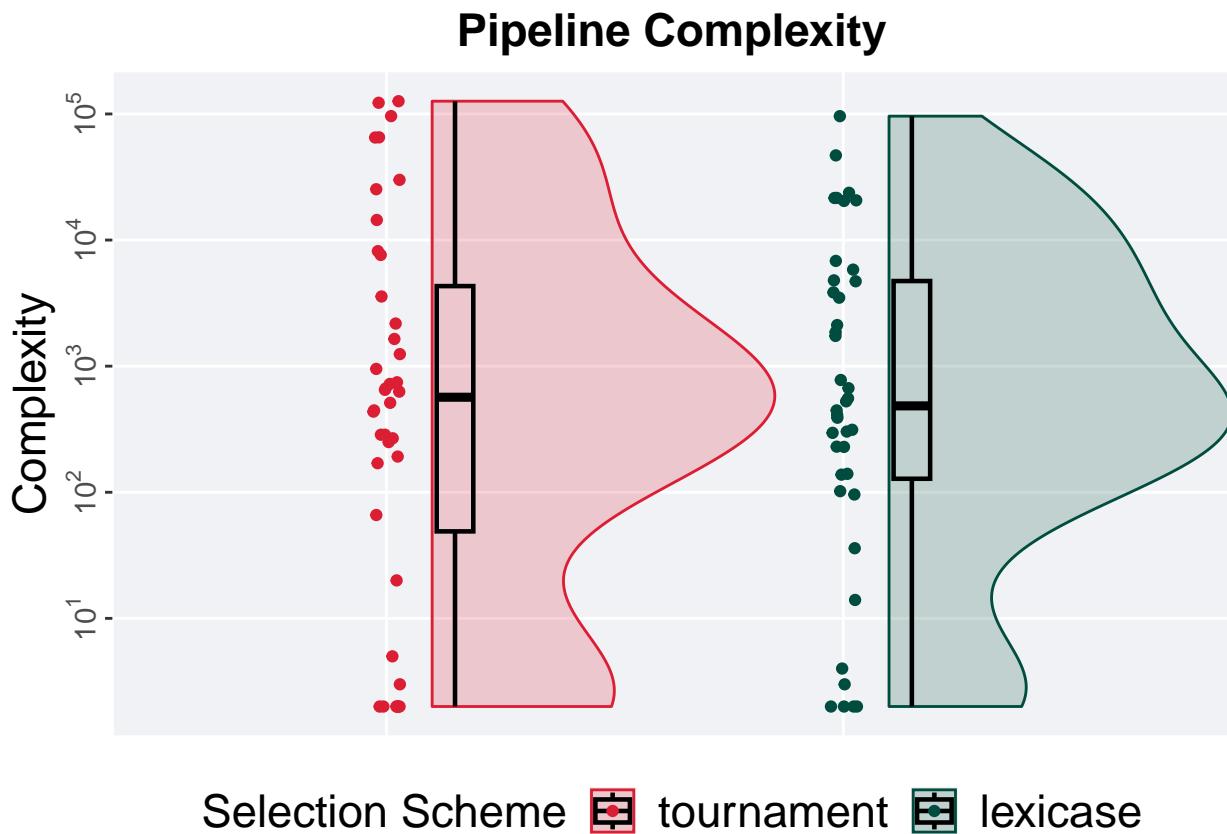
```
tournament_results <- filter(task_data, split == '5%' & selection == 'tournament')
lexicase_results <- filter(task_data, split == '5%' & selection == 'lexicase')
permutation_test(tournament_results$training_performance,
                 lexicase_results$training_performance,
                 seed = 72,
                 alternative = "1")
```

```
## [1] "observed_diff: -2.9598156320086"
## [1] "permutation_diffs[0.05 * n_permutations]: -1.61245190733767"
## [1] "reject null hypothesis"
## [1] "p-value: 0.00147"
```



9.1.3 Pipeline complexity

```
complexity_plot(filter(task_data, split == '5%'))
```



Summary statistics for the pipeline complexity of the selection schemes at the 5% selection set split:

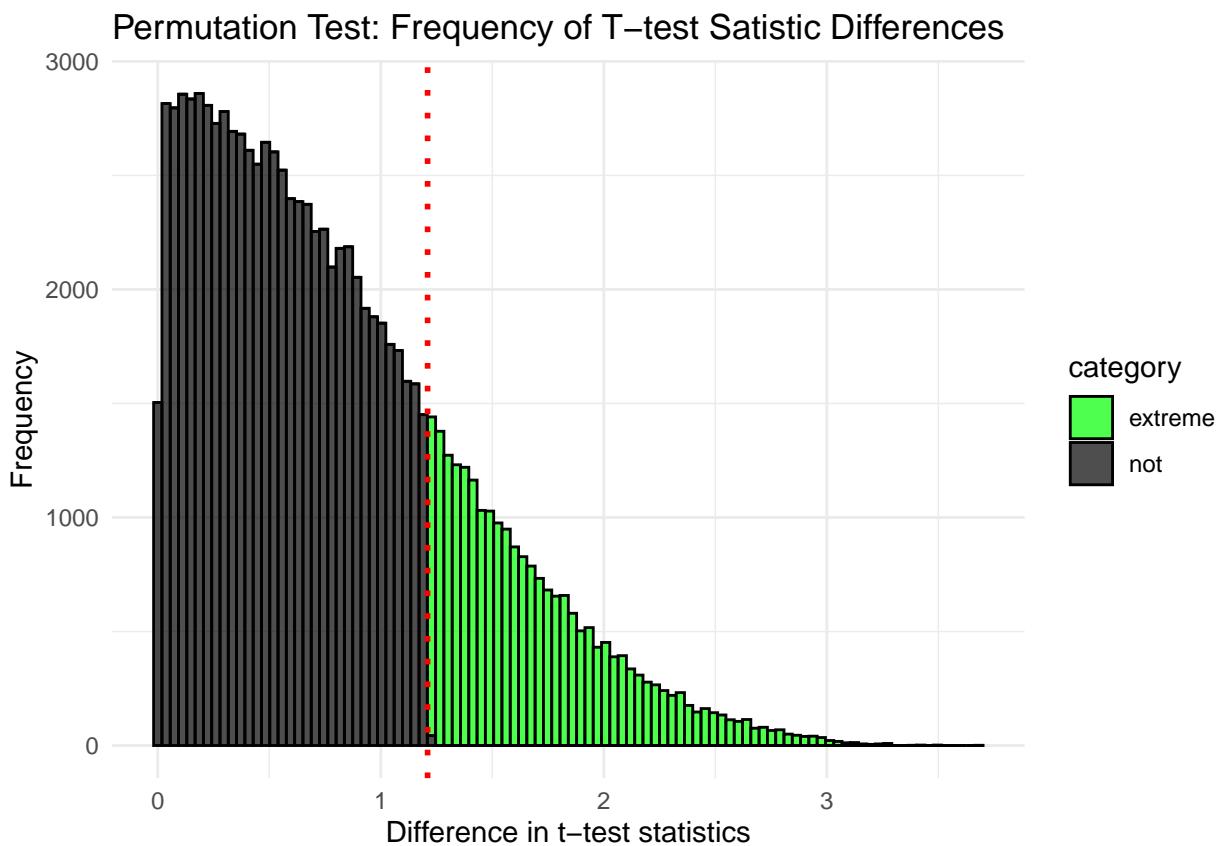
```
complexity_summary(filter(task_data, split == '5%'))
```

```
## # A tibble: 2 x 8
##   selection count na_cnt  min median   mean   max   IQR
##   <fct>     <int> <int> <dbl> <dbl> <dbl> <int> <dbl>
## 1 tournament     40     0     2    570 14420. 126232 4528.
## 2 lexicase       40     0     2    486  7295.  96062 4602
```

The permutation test revealed that the results are:

```
tournament_results <- filter(task_data, split == '5%' & selection == 'tournament')
lexicase_results <- filter(task_data, split == '5%' & selection == 'lexicase')
permutation_test(tournament_results$testing_complexity,
                 lexicase_results$testing_complexity,
                 seed = 223,
                 alternative = "t")
```

```
## [1] "observed_diff: 1.20985798363081"
## [1] "lower: -1.97788971368724"
## [1] "upper: 1.97026008720771"
## [1] "fail to reject null hypothesis"
## [1] "p-value: 0.23706"
```

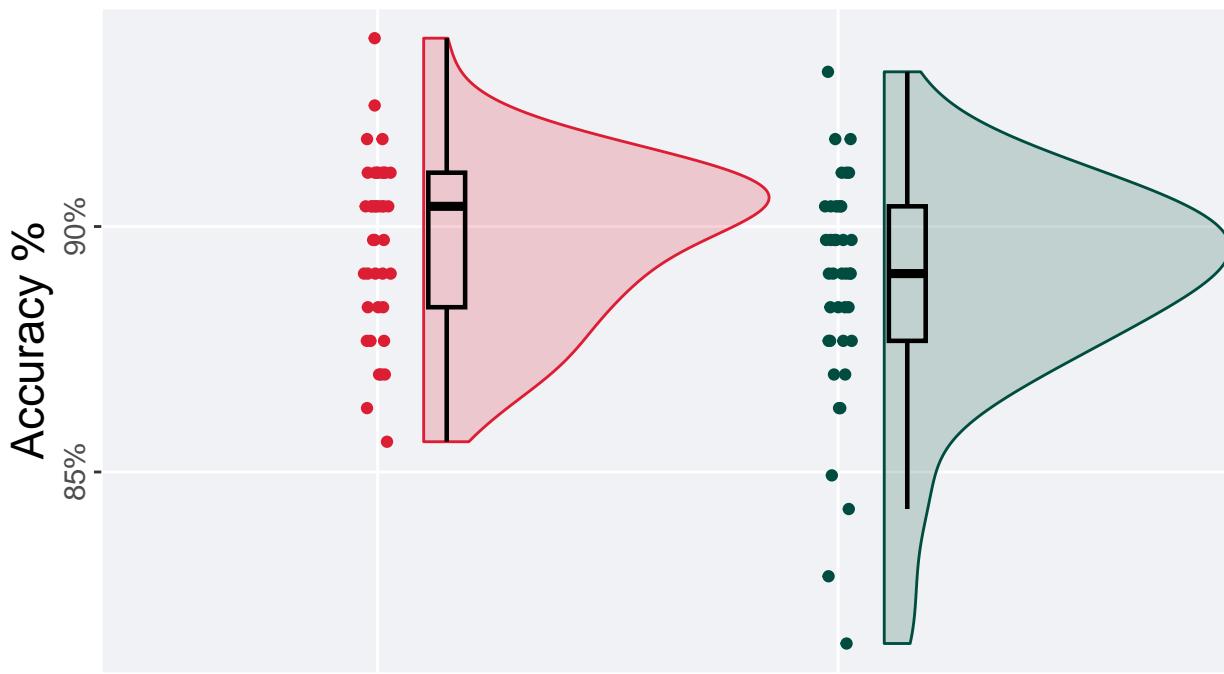


9.2 10%

9.2.1 Test accuracy

```
test_plot(filter(task_data, split == '10%'))
```

Accuracy on test set



Selection scheme  tournament  lexicase

Summary statistics for the testing performance of the selection schemes at the 5% selection set split:

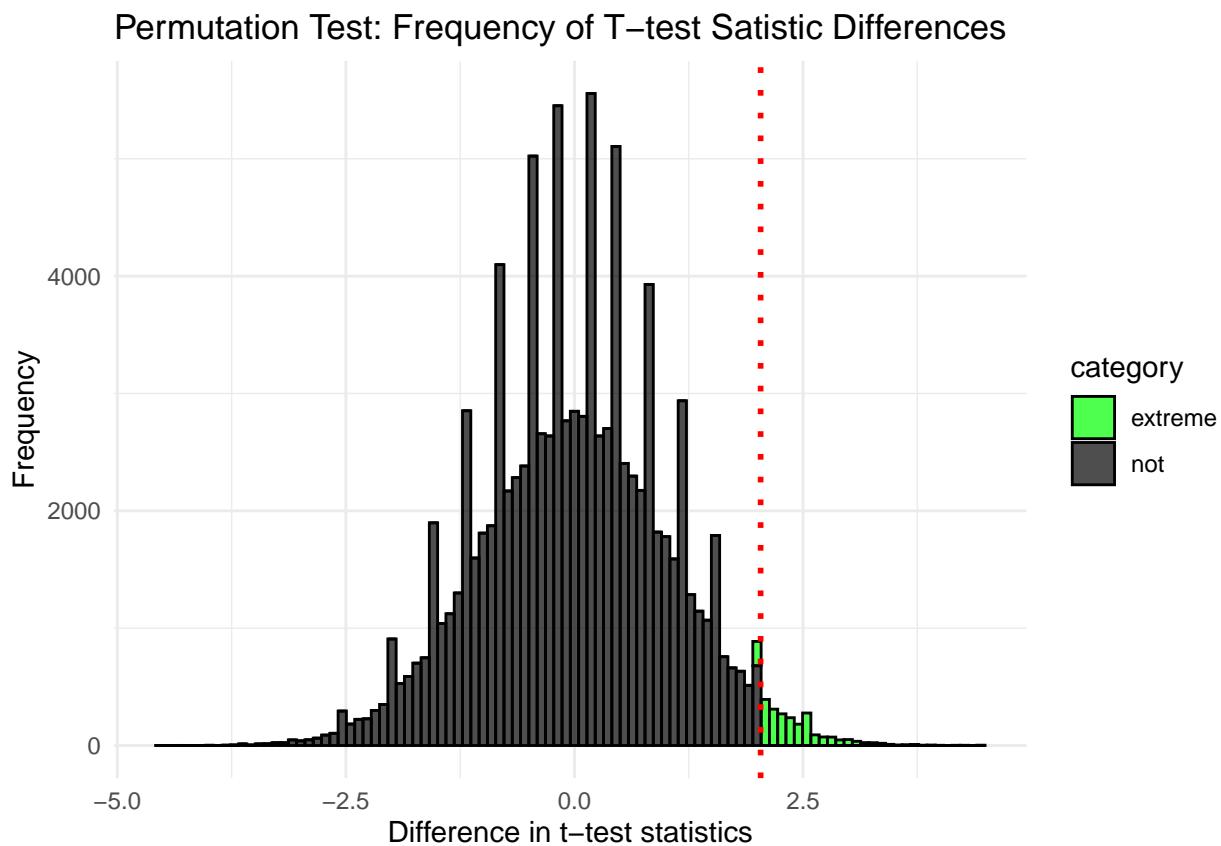
```
test_results_summary(filter(task_data, split == '10%'))
```

```
## # A tibble: 2 x 8
##   selection  count  na_cnt  min  median  mean  max    IQR
##   <fct>      <int>   <int> <dbl>  <dbl>  <dbl> <dbl>  <dbl>
## 1 tournament    40      0  0.856  0.904  0.897  0.938  0.0274
## 2 lexicase      40      0  0.815  0.890  0.887  0.932  0.0274
```

The permutation test revealed that the results are:

```
tournament_results <- filter(task_data, split == '10%' & selection == 'tournament')
lexicase_results <- filter(task_data, split == '10%' & selection == 'lexicase')
permutation_test(tournament_results$testing_performance,
                  lexicase_results$testing_performance,
                  seed = 73,
                  alternative = "g")
```

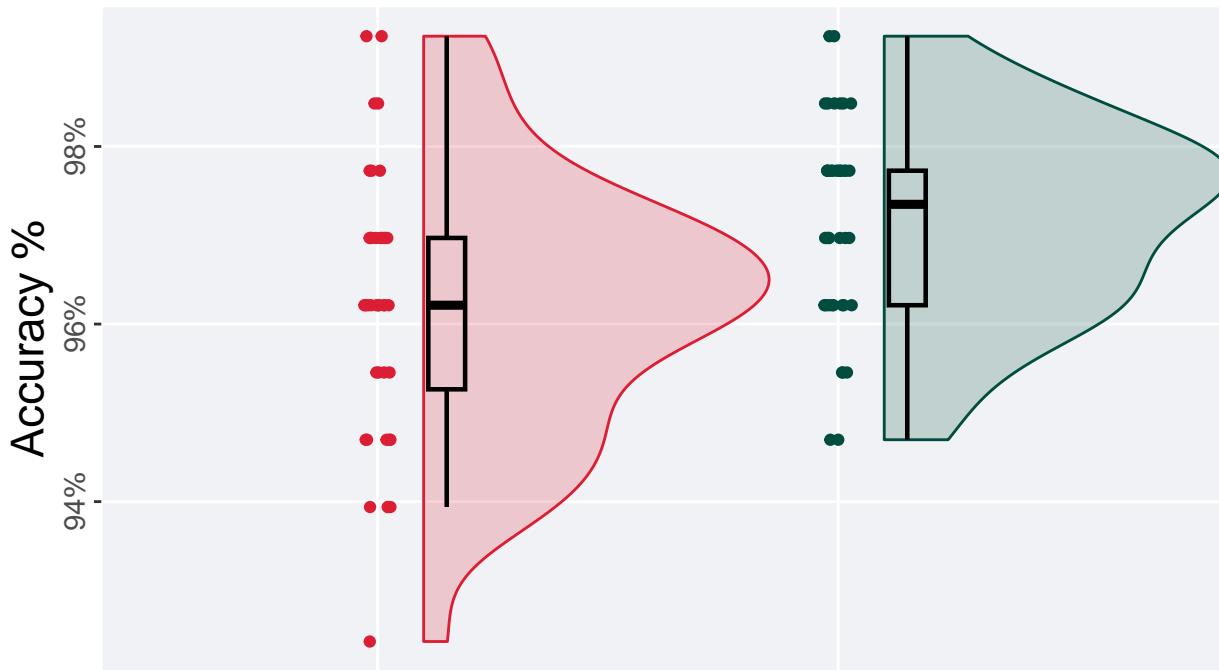
```
## [1] "observed_diff: 2.03573402533652"
## [1] "permutation_diffs[0.95 * n_permutations]: 1.6579467289571"
## [1] "reject null hypothesis"
## [1] "p-value: 0.0235"
```



9.2.2 Validation accuracy

```
validation_plot(filter(task_data, split == '10%'))
```

Accuracy on validation set



Summary statistics for the testing performance of the selection schemes at the 5% selection set split:

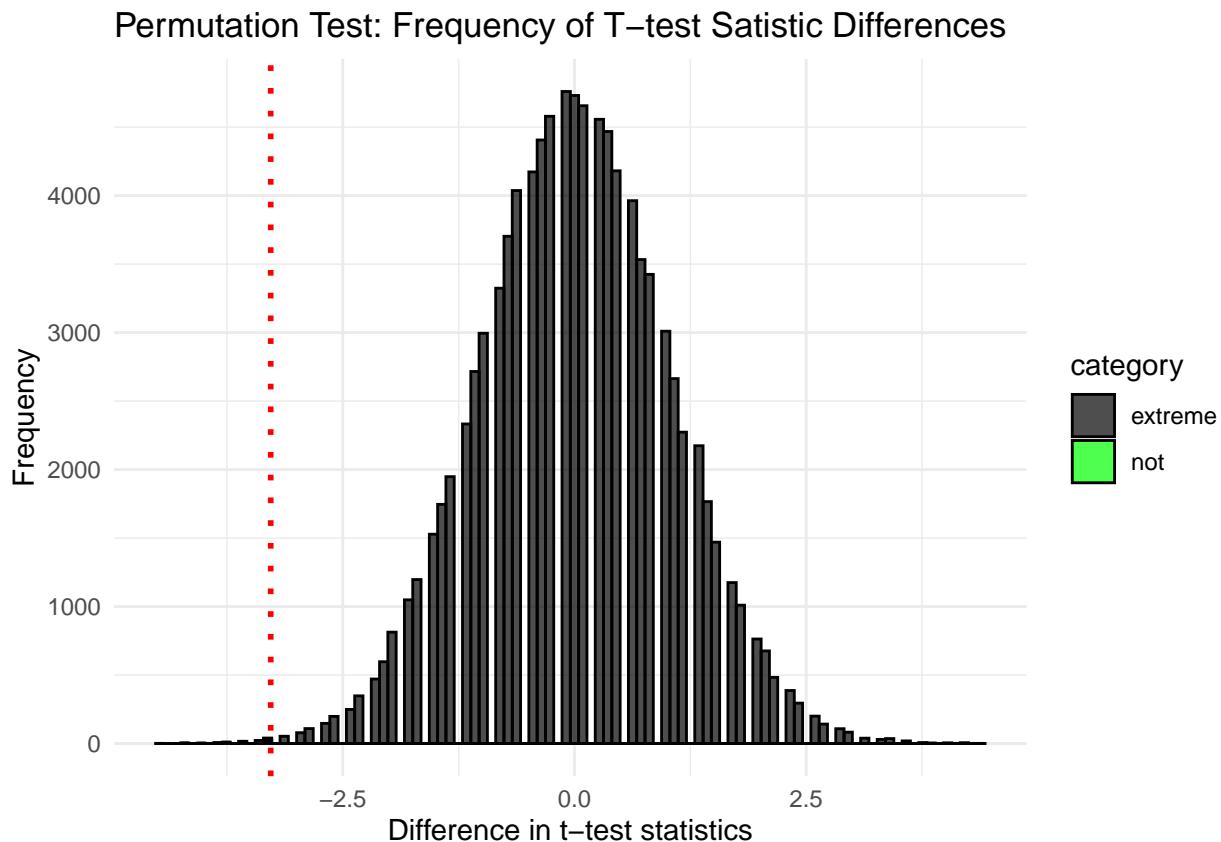
```
validation_accuracy_summary(filter(task_data, split == '10%'))
```

```
## # A tibble: 2 x 8
##   selection count na_cnt  min median  mean  max    IQR
##   <fct>     <int>  <int> <dbl>  <dbl> <dbl> <dbl> <dbl>
## 1 tournament     40      0 0.924  0.962  0.962 0.992 0.0170
## 2 lexicase       40      0 0.947  0.973  0.972 0.992 0.0152
```

The permutation test revealed that the results are:

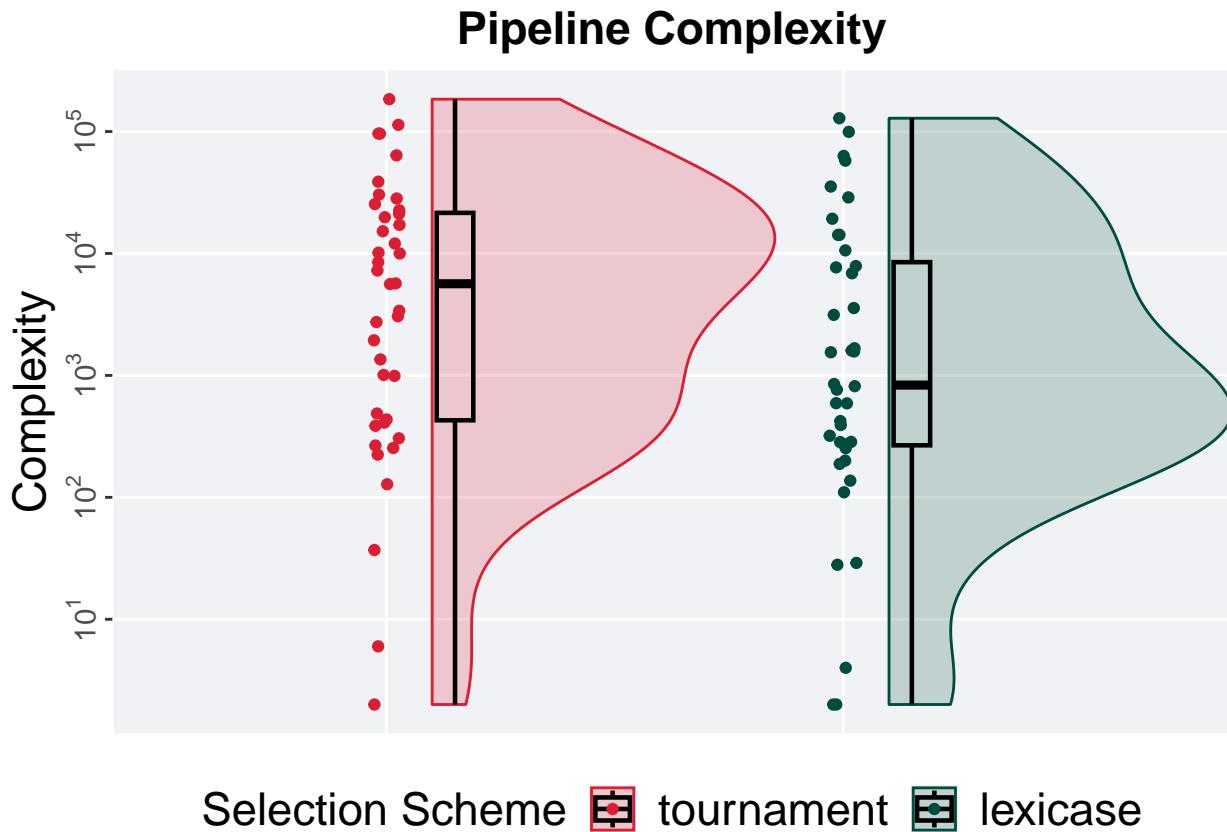
```
tournament_results <- filter(task_data, split == '10%' & selection == 'tournament')
lexicase_results <- filter(task_data, split == '10%' & selection == 'lexicase')
permutation_test(tournament_results$training_performance,
                 lexicase_results$training_performance,
                 seed = 74,
                 alternative = "1")
```

```
## [1] "observed_diff: -3.27769366217578"
## [1] "permutation_diffs[0.05 * n_permutations]: -1.68446274227723"
## [1] "reject null hypothesis"
## [1] "p-value: 0.00083"
```



9.2.3 Pipeline complexity

```
complexity_plot(filter(task_data, split == '10%'))
```



Summary statistics for the pipeline complexity of the selection schemes at the 5% selection set split:

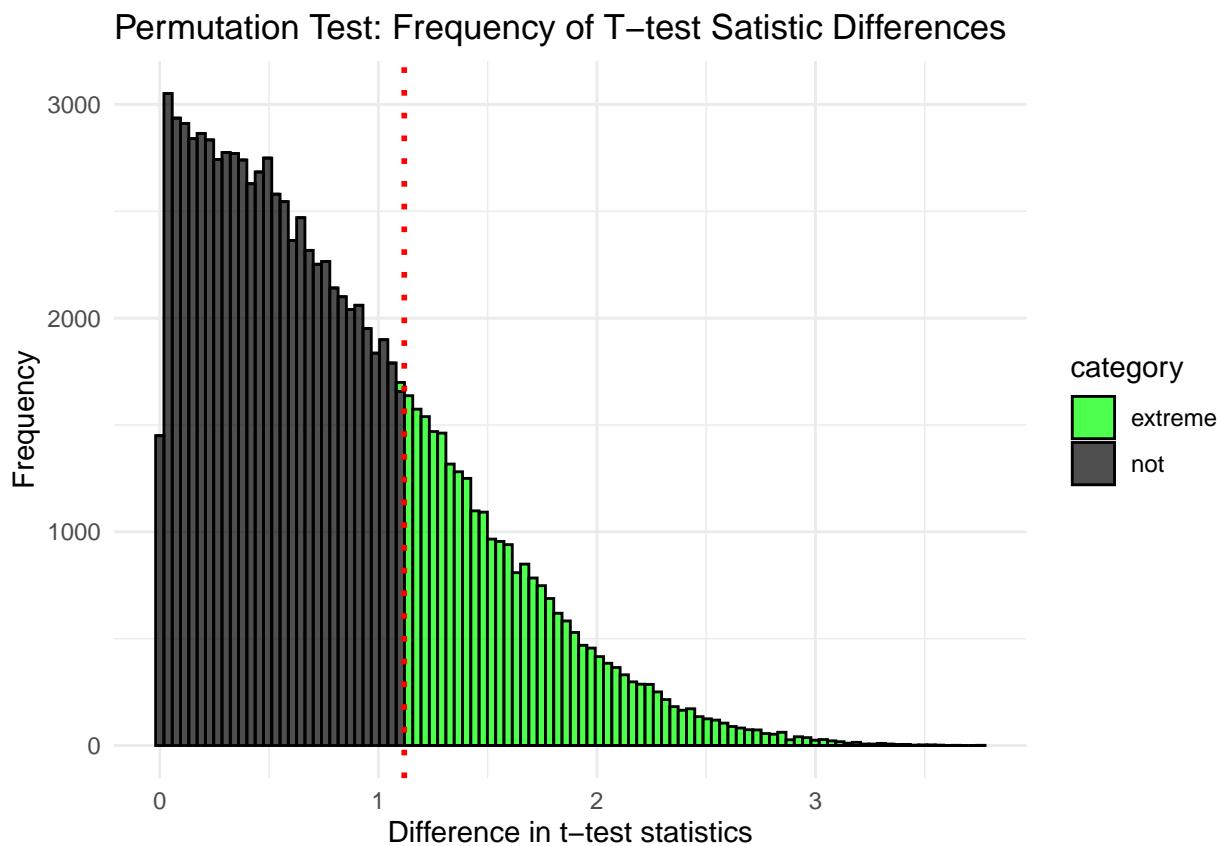
```
complexity_summary(filter(task_data, split == '10%'))
```

```
## # A tibble: 2 x 8
##   selection count na_cnt  min median   mean   max   IQR
##   <fct>     <int> <int> <int> <dbl> <dbl> <int> <dbl>
## 1 tournament     40     0     2    5646 21231. 184151 21105
## 2 lexicase       40     0     2     833 12831. 128541  8294.
```

The permutation test revealed that the results are:

```
tournament_results <- filter(task_data, split == '10%' & selection == 'tournament')
lexicase_results <- filter(task_data, split == '10%' & selection == 'lexicase')
permutation_test(tournament_results$testing_complexity,
                 lexicase_results$testing_complexity,
                 seed = 224,
                 alternative = "t")

## [1] "observed_diff: 1.11868422664352"
## [1] "lower: -1.96517418154003"
## [1] "upper: 1.95098196190349"
## [1] "fail to reject null hypothesis"
## [1] "p-value: 0.27759"
```

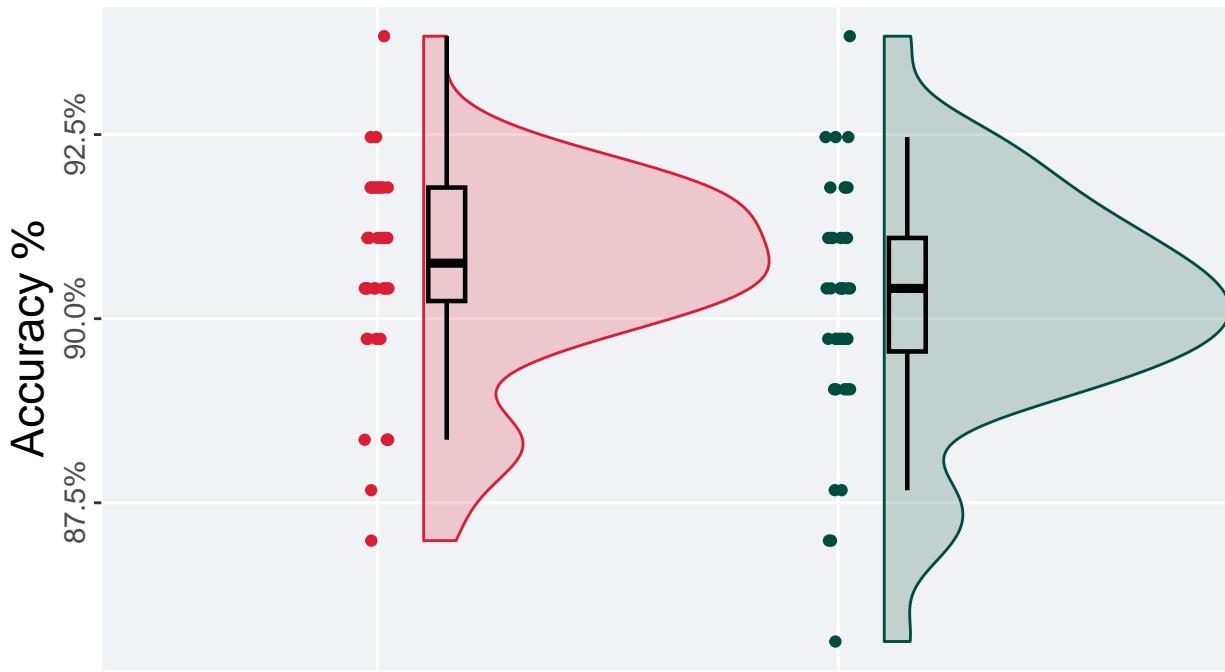


9.3 50%

9.3.1 Test accuracy

```
test_plot(filter(task_data, split == '50%'))
```

Accuracy on test set



Selection scheme tournament lexicase

Summary statistics for the testing performance of the selection schemes at the 5% selection set split:

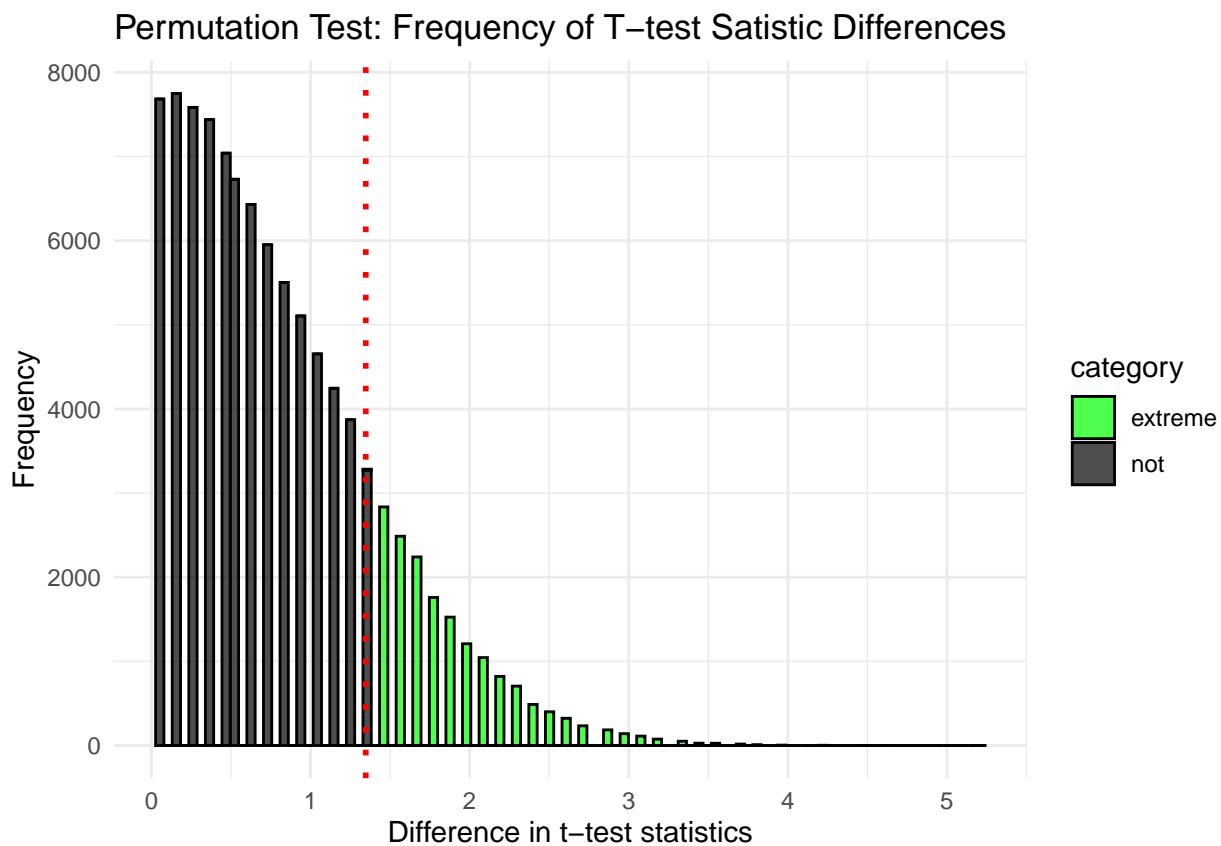
```
test_results_summary(filter(task_data, split == '50%'))
```

```
## # A tibble: 2 x 8
##   selection count na_cnt  min median  mean   max     IQR
##   <fct>     <int> <int> <dbl> <dbl> <dbl> <dbl> <dbl>
## 1 tournament     40     0 0.870 0.908 0.906 0.938 0.0154
## 2 lexicase       40     0 0.856 0.904 0.902 0.938 0.0154
```

The permutation test revealed that the results are:

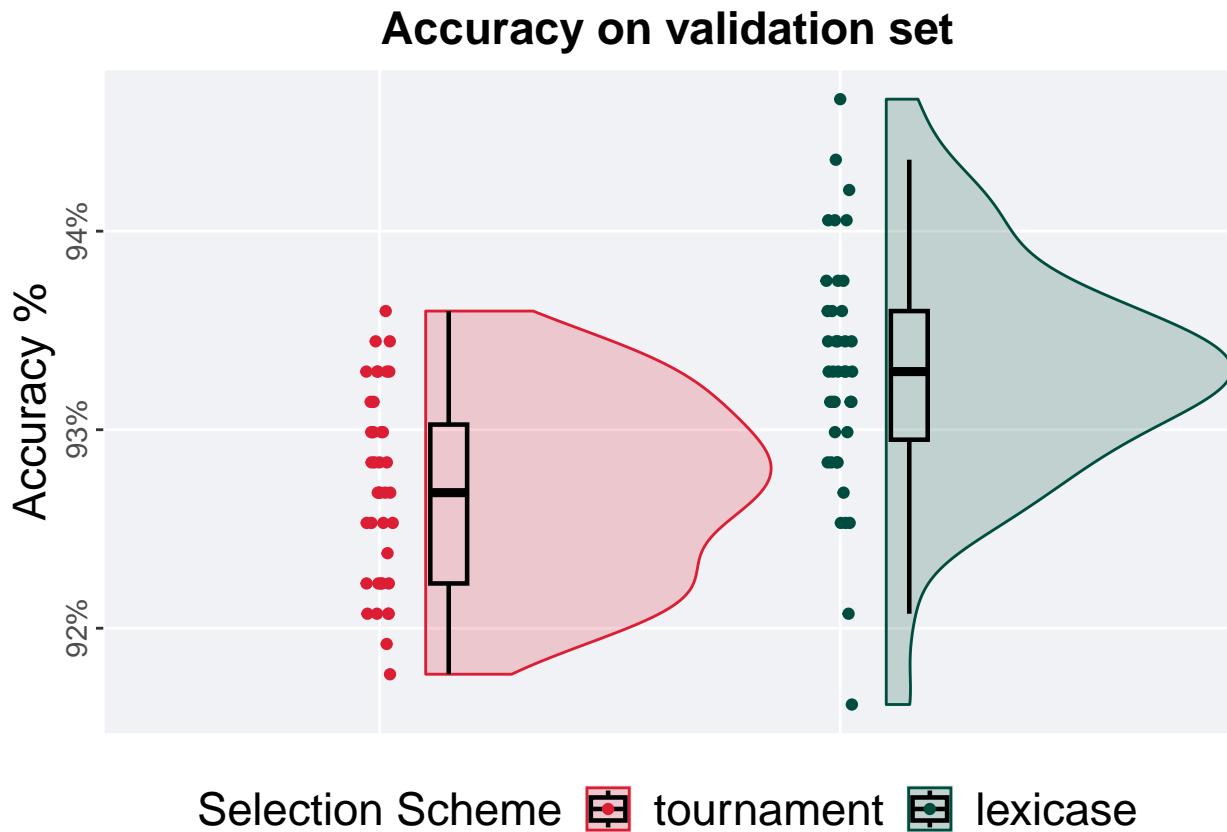
```
tournament_results <- filter(task_data, split == '50%' & selection == 'tournament')
lexicase_results <- filter(task_data, split == '50%' & selection == 'lexicase')
permutation_test(tournament_results$testing_performance,
                 lexicase_results$testing_performance,
                 seed = 75,
                 alternative = "t")
```

```
## [1] "observed_diff: 1.34694178441485"
## [1] "lower: -1.97063646149436"
## [1] "upper: 1.97063630660342"
## [1] "fail to reject null hypothesis"
## [1] "p-value: 0.16746"
```



9.3.2 Validation accuracy

```
validation_plot(filter(task_data, split == '50%'))
```



Summary statistics for the testing performance of the selection schemes at the 5% selection set split:

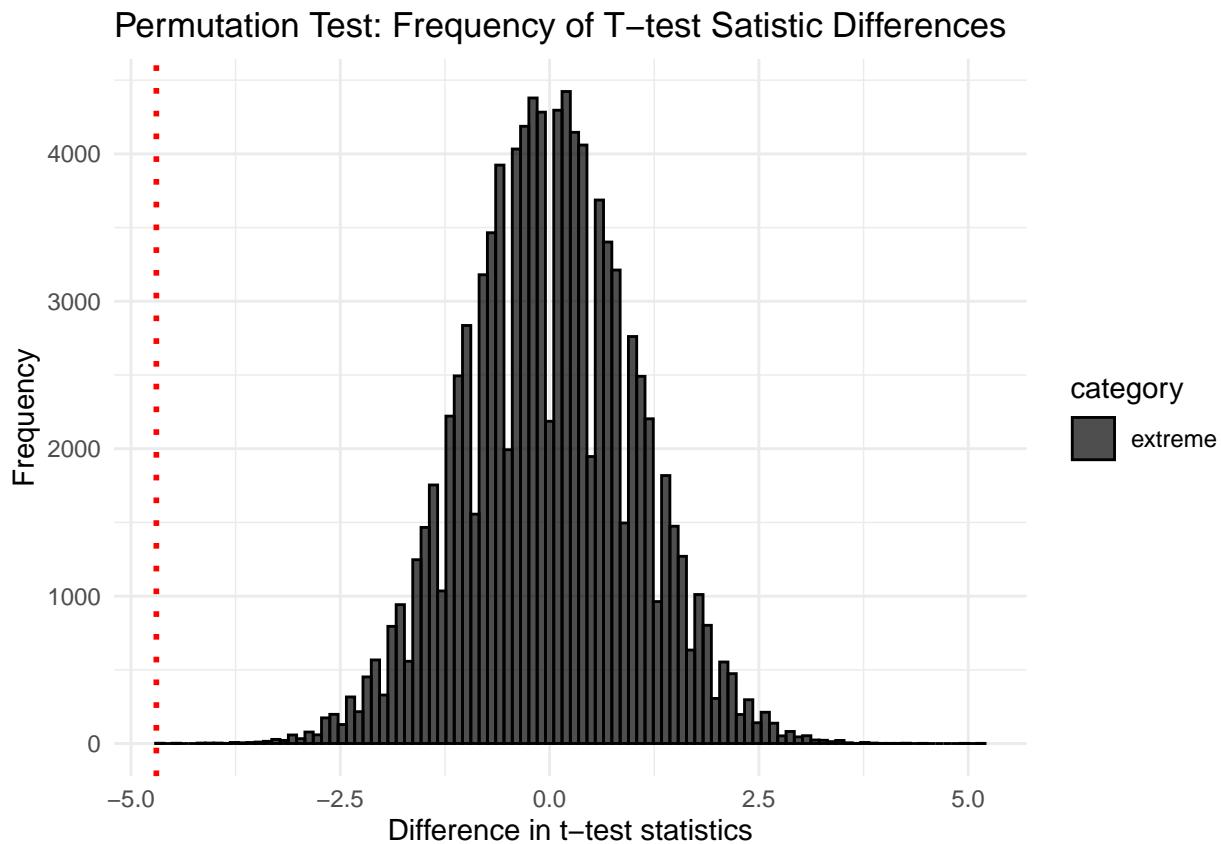
```
validation_accuracy_summary(filter(task_data, split == '50%'))
```

```
## # A tibble: 2 x 8
##   selection  count  na_cnt  min  median  mean   max      IQR
##   <fct>      <int>  <int>  <dbl>  <dbl>  <dbl>  <dbl>  <dbl>
## 1 tournament    40      0  0.918  0.927  0.927  0.936  0.00800
## 2 lexicase      40      0  0.916  0.933  0.933  0.947  0.00648
```

The permutation test revealed that the results are:

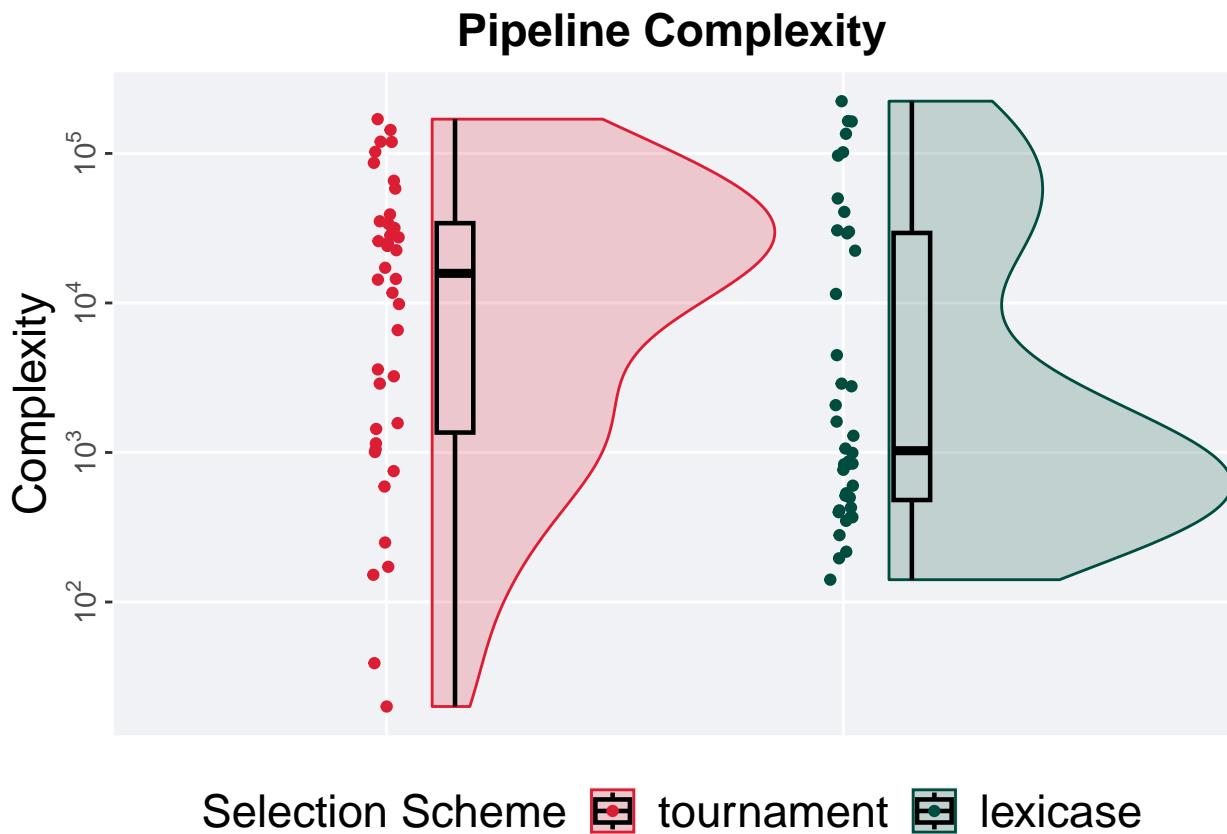
```
tournament_results <- filter(task_data, split == '50%' & selection == 'tournament')
lexicase_results <- filter(task_data, split == '50%' & selection == 'lexicase')
permutation_test(tournament_results$training_performance,
                 lexicase_results$training_performance,
                 seed = 76,
                 alternative = "1")
```

```
## [1] "observed_diff: -4.69715356107344"
## [1] "permutation_diffs[0.05 * n_permutations]: -1.6888955870353"
## [1] "reject null hypothesis"
## [1] "p-value: 1e-05"
```



9.3.3 Pipeline complexity

```
complexity_plot(filter(task_data, split == '50%'))
```



Summary statistics for the pipeline complexity of the selection schemes at the 5% selection set split:

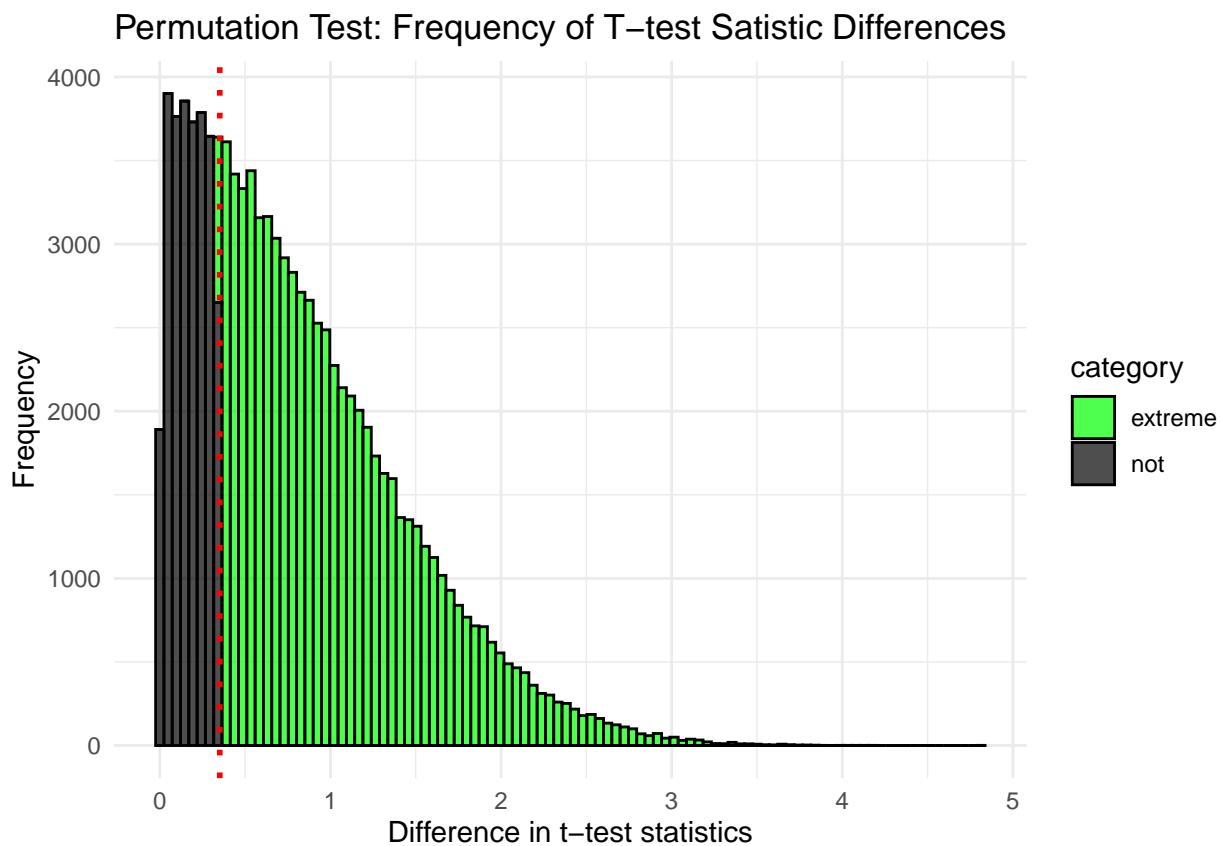
```
complexity_summary(filter(task_data, split == '50%'))
```

```
## # A tibble: 2 x 8
##   selection count na_cnt  min median   mean   max    IQR
##   <fct>     <int> <int> <dbl> <dbl> <dbl> <int> <dbl>
## 1 tournament     40     0  20 15836  32056. 169601 32812.
## 2 lexicase       40     0 141 1028. 28164. 223581 28912.
```

The permutation test revealed that the results are:

```
tournament_results <- filter(task_data, split == '50%' & selection == 'tournament')
lexicase_results <- filter(task_data, split == '50%' & selection == 'lexicase')
permutation_test(tournament_results$testing_complexity,
                 lexicase_results$testing_complexity,
                 seed = 225,
                 alternative = "t")
```

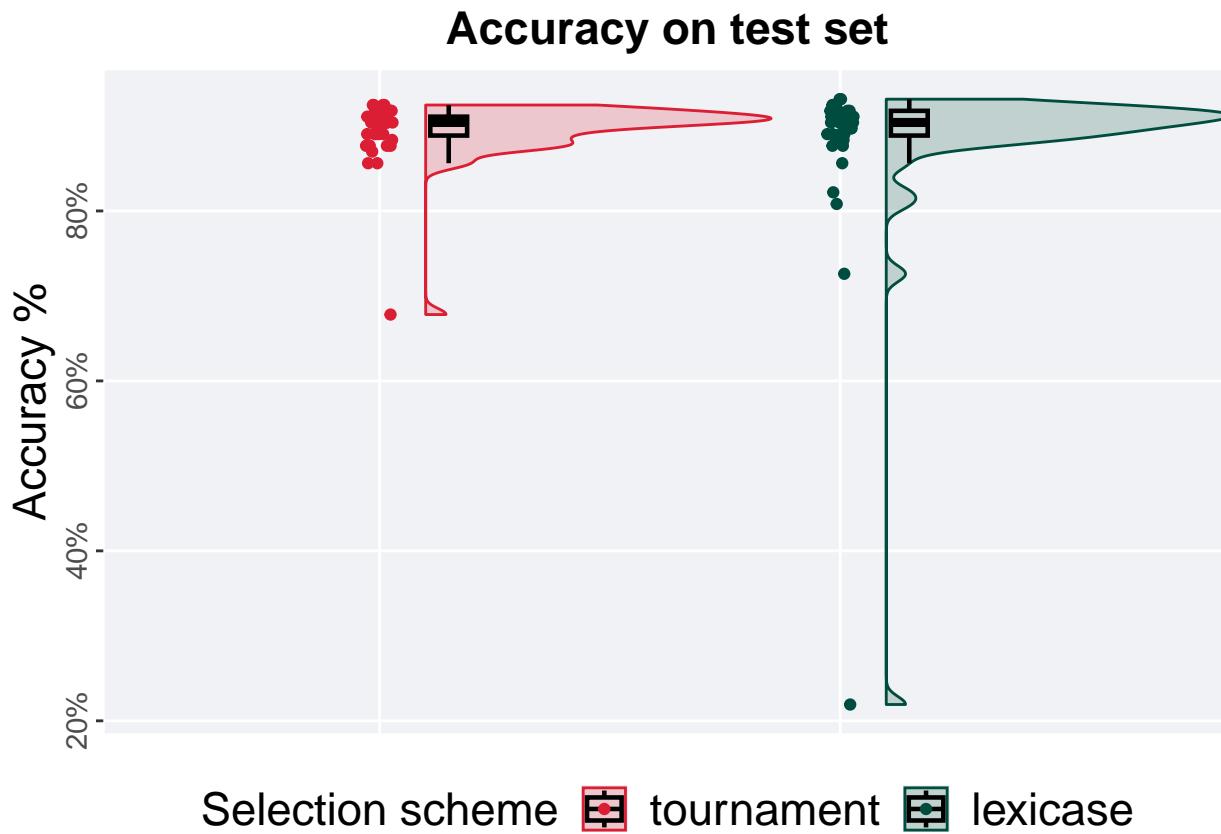
```
## [1] "observed_diff: 0.351583832380435"
## [1] "lower: -1.9874587796869"
## [1] "upper: 1.98005380767045"
## [1] "fail to reject null hypothesis"
## [1] "p-value: 0.72775"
```



9.4 90%

9.4.1 Test accuracy

```
test_plot(filter(task_data, split == '90%'))
```



Summary statistics for the testing performance of the selection schemes at the 5% selection set split:

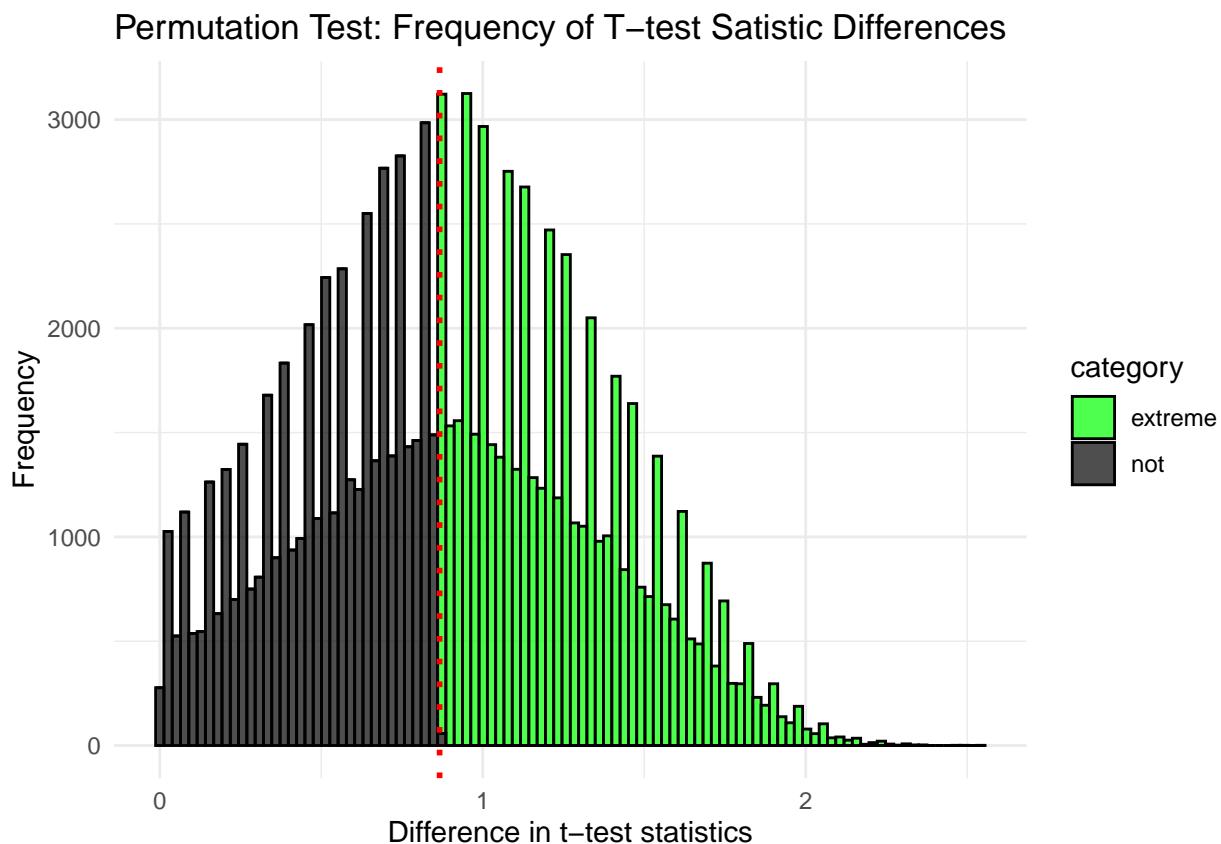
```
test_results_summary(filter(task_data, split == '90%'))
```

```
## # A tibble: 2 x 8
##   selection count na_cnt  min median  mean   max     IQR
##   <fct>     <int> <int> <dbl> <dbl> <dbl> <dbl> <dbl>
## 1 tournament     40     0 0.678  0.904  0.895  0.925  0.0223
## 2 lexicase       40     0 0.219  0.904  0.878  0.932  0.0291
```

The permutation test revealed that the results are:

```
tournament_results <- filter(task_data, split == '90%' & selection == 'tournament')
lexicase_results <- filter(task_data, split == '90%' & selection == 'lexicase')
permutation_test(tournament_results$testing_performance,
                 lexicase_results$testing_performance,
                 seed = 77,
                 alternative = "t")
```

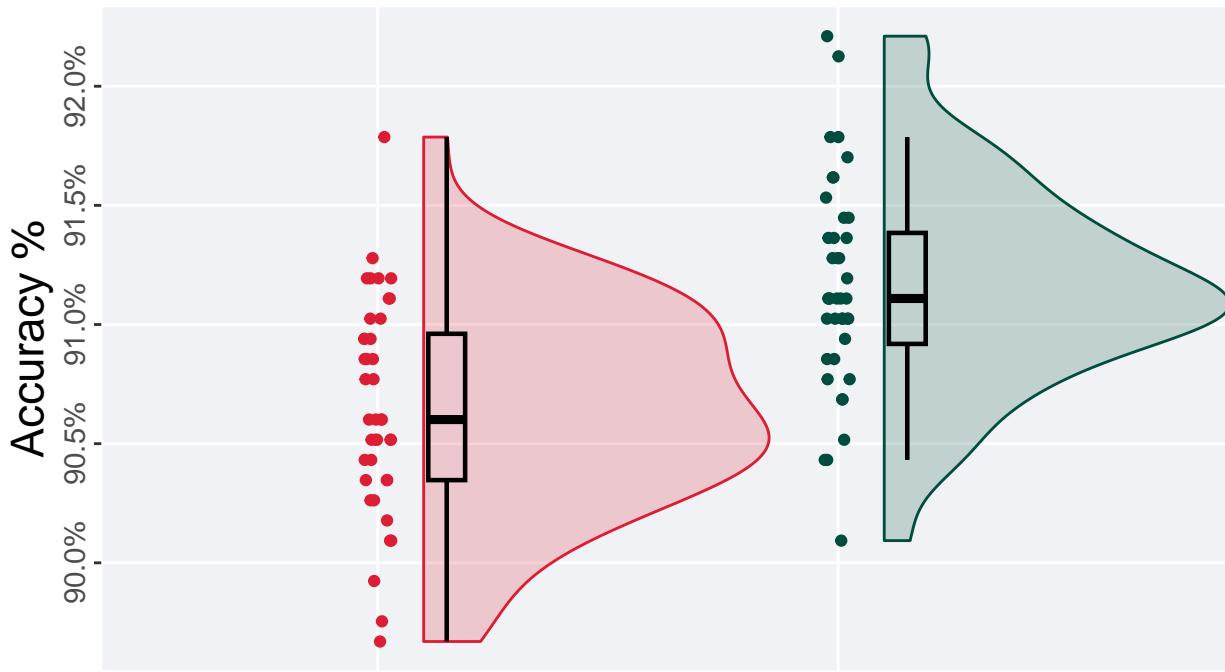
```
## [1] "observed_diff: 0.866475250333264"
## [1] "lower: -1.66359475306712"
## [1] "upper: 1.66359458724114"
## [1] "fail to reject null hypothesis"
## [1] "p-value: 0.53139"
```



9.4.2 Validation accuracy

```
validation_plot(filter(task_data, split == '90%'))
```

Accuracy on validation set



Selection Scheme  tournament  lexicase

Summary statistics for the testing performance of the selection schemes at the 5% selection set split:

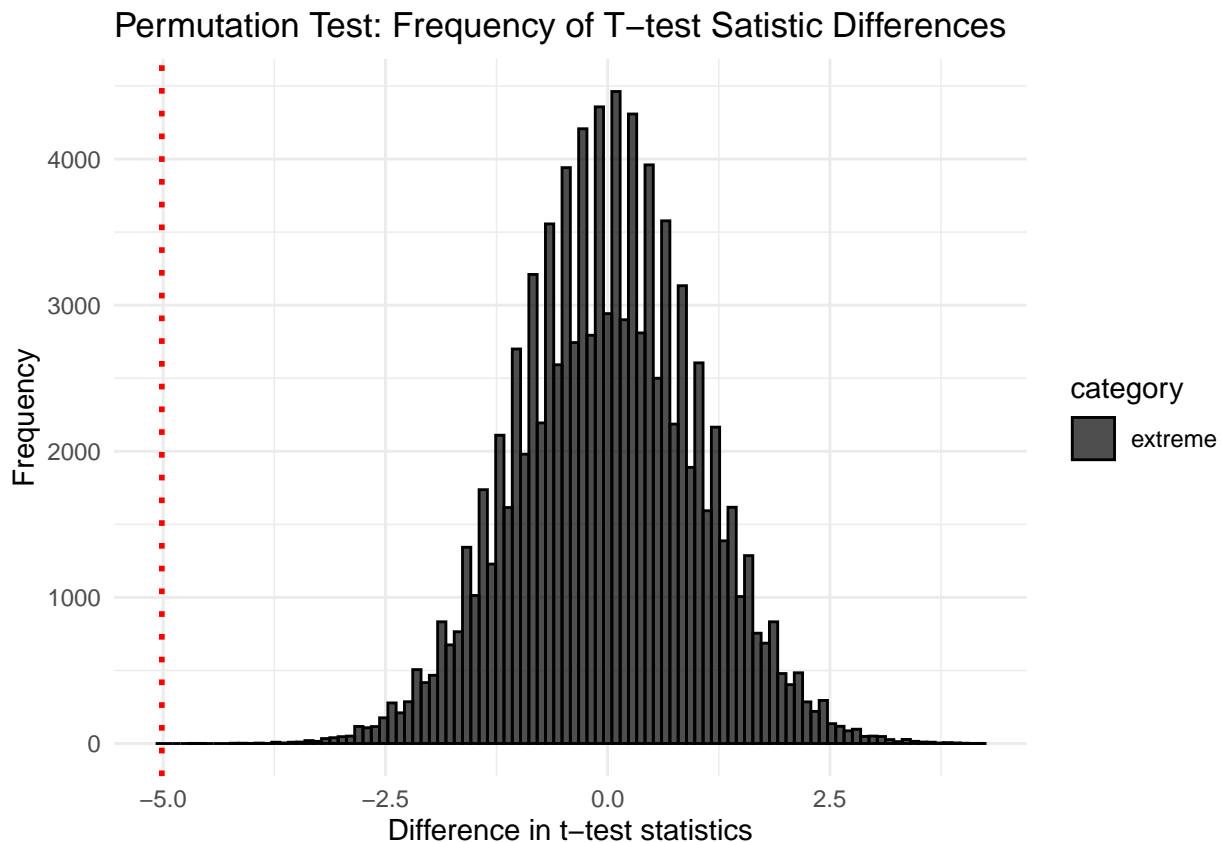
```
validation_accuracy_summary(filter(task_data, split == '90%'))
```

```
## # A tibble: 2 x 8
##   selection  count  na_cnt  min  median  mean  max      IQR
##   <fct>      <int>  <int>  <dbl>  <dbl>  <dbl>  <dbl>  <dbl>
## 1 tournament    40      0  0.897  0.906  0.907  0.918  0.00614
## 2 lexicase      40      0  0.901  0.911  0.912  0.922  0.00466
```

The permutation test revealed that the results are:

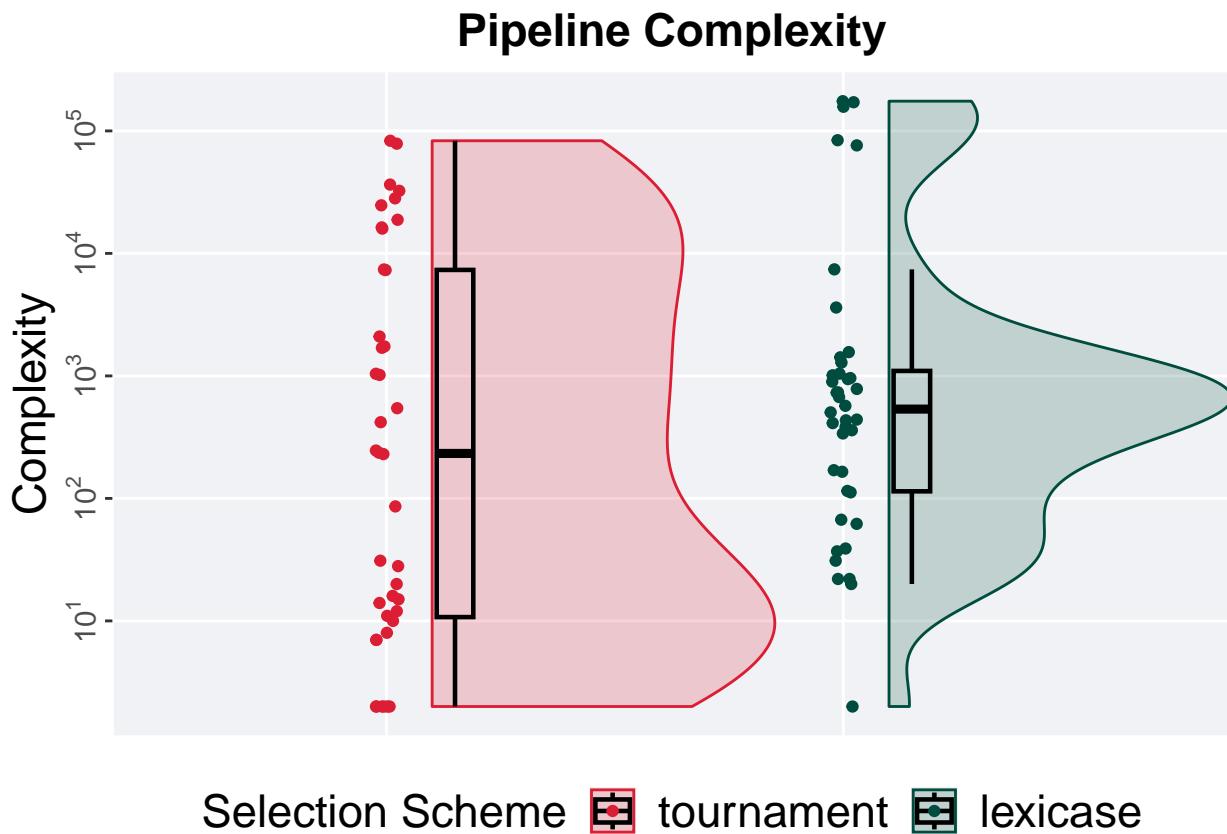
```
tournament_results <- filter(task_data, split == '90%' & selection == 'tournament')
lexicase_results <- filter(task_data, split == '90%' & selection == 'lexicase')
permutation_test(tournament_results$training_performance,
                 lexicase_results$training_performance,
                 seed = 78,
                 alternative = "1")
```

```
## [1] "observed_diff: -5.01526490705556"
## [1] "permutation_diffs[0.05 * n_permutations]: -1.66663584058073"
## [1] "reject null hypothesis"
## [1] "p-value: 1e-05"
```



9.4.3 Pipeline complexity

```
complexity_plot(filter(task_data, split == '90%'))
```



Summary statistics for the pipeline complexity of the selection schemes at the 5% selection set split:

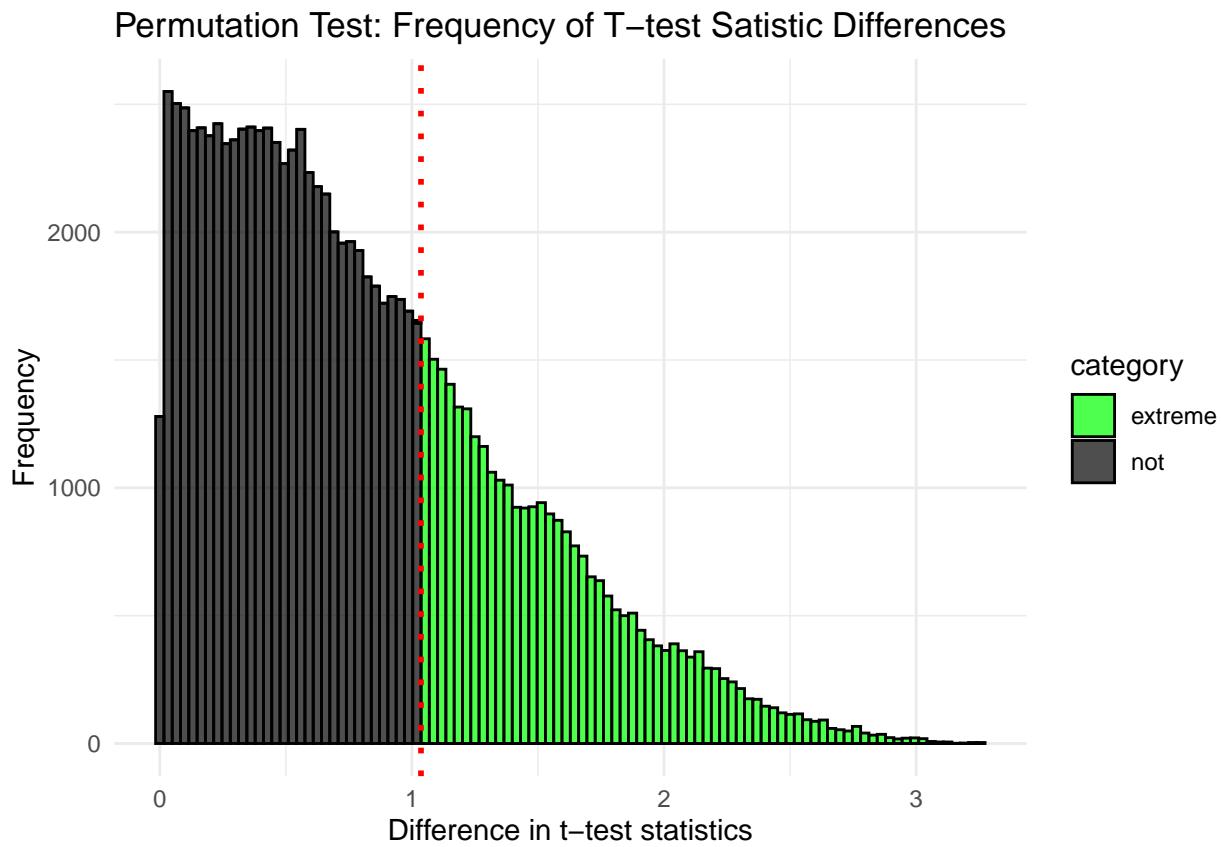
```
complexity_summary(filter(task_data, split == '90%'))
```

```
## # A tibble: 2 x 8
##   selection count na_cnt  min median   mean   max   IQR
##   <fct>     <int> <int> <dbl> <dbl> <dbl> <int> <dbl>
## 1 tournament     40     0     2    233   8972. 83121 7322.
## 2 lexicase       40     0     2    537 17285. 174871  989.
```

The permutation test revealed that the results are:

```
tournament_results <- filter(task_data, split == '90%' & selection == 'tournament')
lexicase_results <- filter(task_data, split == '90%' & selection == 'lexicase')
permutation_test(tournament_results$testing_complexity,
                 lexicase_results$testing_complexity,
                 seed = 226,
                 alternative = "t")

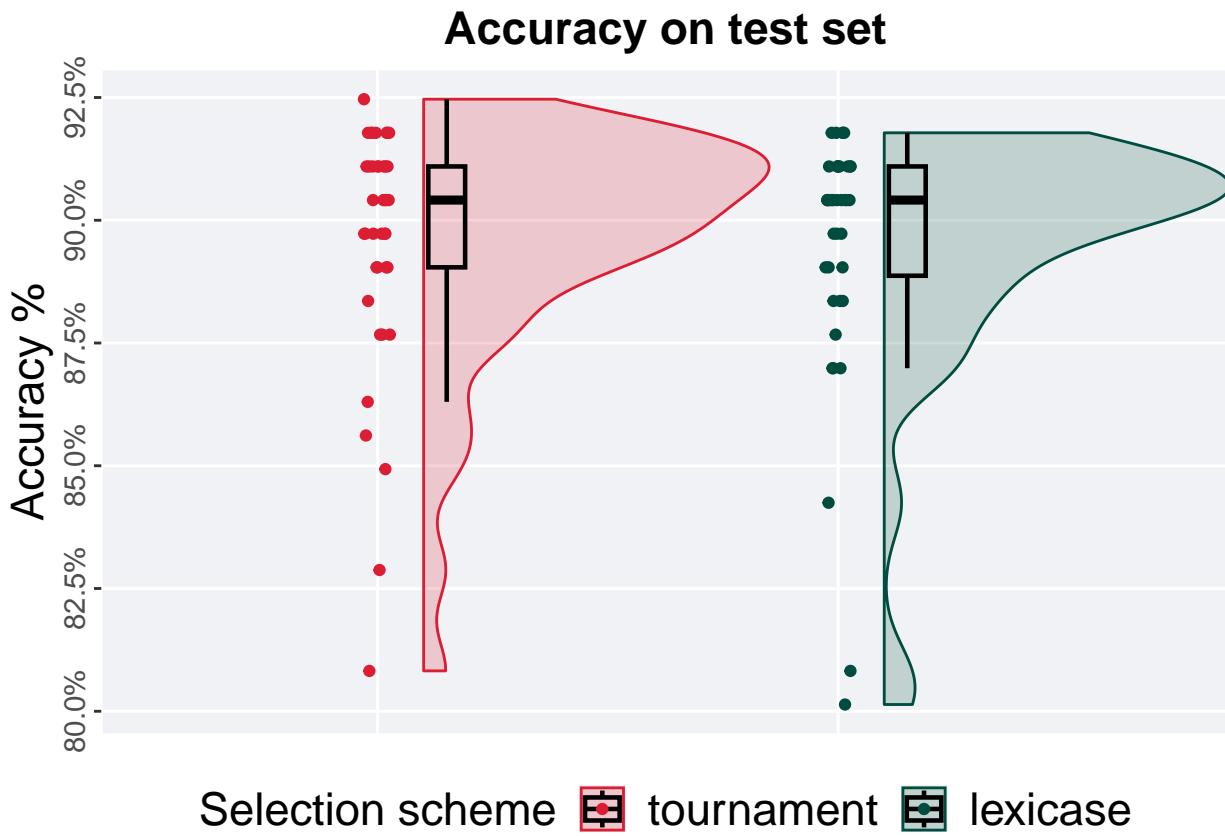
## [1] "observed_diff: -1.03603328023373"
## [1] "lower: -1.96725851592498"
## [1] "upper: 1.98568541196877"
## [1] "fail to reject null hypothesis"
## [1] "p-value: 0.31344"
```



9.5 95%

9.5.1 Test accuracy

```
test_plot(filter(task_data, split == '95%'))
```



Summary statistics for the testing performance of the selection schemes at the 5% selection set split:

```
test_results_summary(filter(task_data, split == '95%'))
```

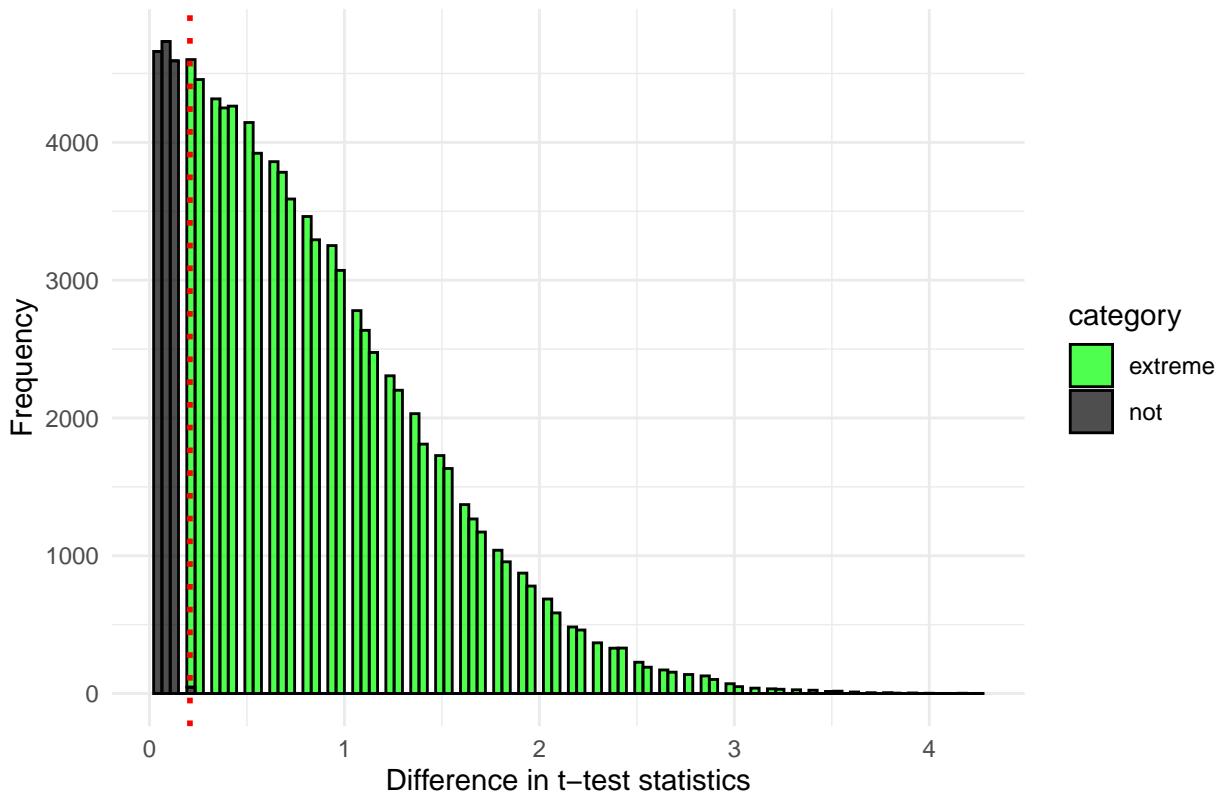
```
## # A tibble: 2 x 8
##   selection count na_cnt  min median  mean   max     IQR
##   <fct>     <int> <int> <dbl> <dbl> <dbl> <dbl> <dbl>
## 1 tournament     40     0 0.808 0.904 0.896 0.925 0.0205
## 2 lexicase       40     0 0.801 0.904 0.894 0.918 0.0223
```

The permutation test revealed that the results are:

```
tournament_results <- filter(task_data, split == '95%' & selection == 'tournament')
lexicase_results <- filter(task_data, split == '95%' & selection == 'lexicase')
permutation_test(tournament_results$testing_performance,
                 lexicase_results$testing_performance,
                 seed = 79,
                 alternative = "t")
```

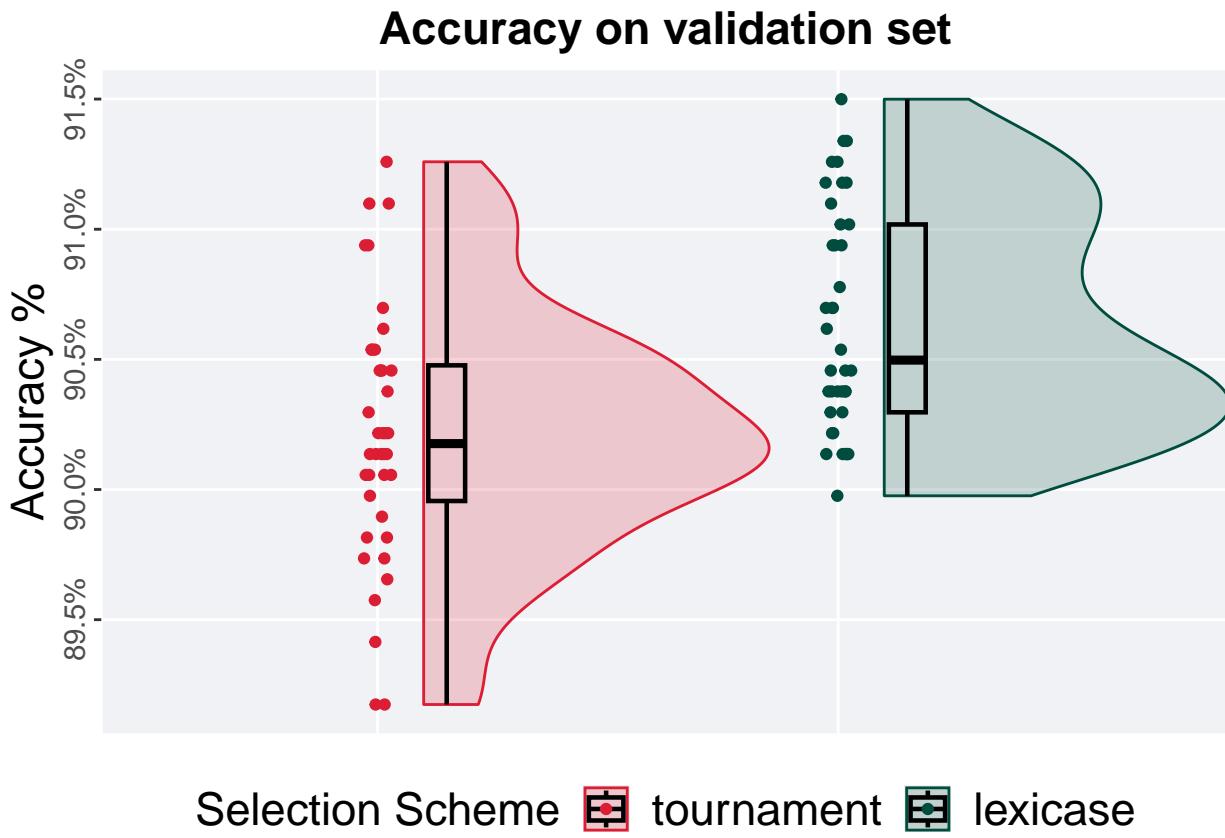
```
## [1] "observed_diff: 0.207528262200296"
## [1] "lower: -1.97407111043758"
## [1] "upper: 1.97407120354954"
## [1] "fail to reject null hypothesis"
## [1] "p-value: 0.8597"
```

Permutation Test: Frequency of T-test Statistic Differences



9.5.2 Validation accuracy

```
validation_plot(filter(task_data, split == '95%'))
```



Summary statistics for the testing performance of the selection schemes at the 5% selection set split:

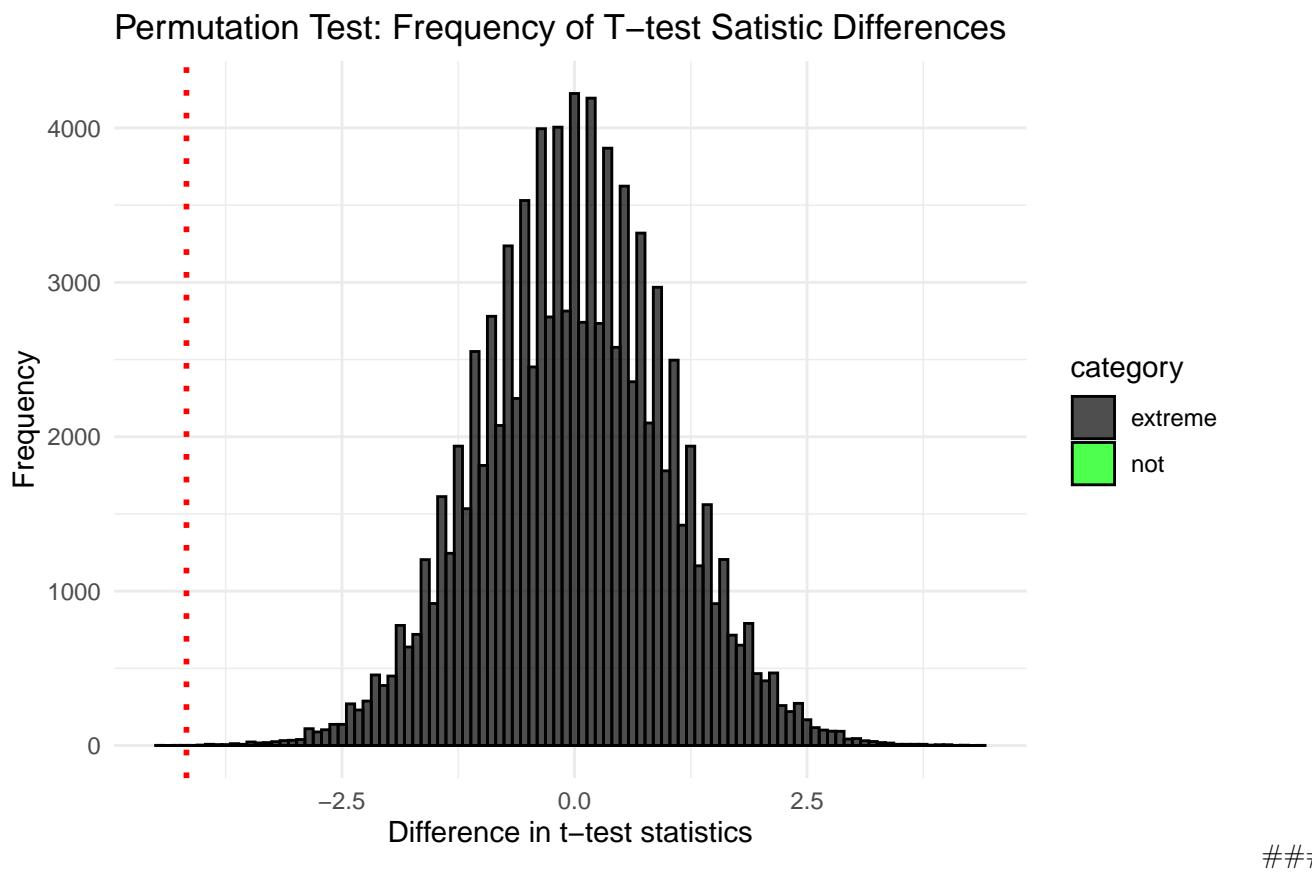
```
validation_accuracy_summary(filter(task_data, split == '95%'))
```

```
## # A tibble: 2 x 8
##   selection  count  na_cnt  min  median  mean   max      IQR
##   <fct>      <int>  <int>  <dbl>  <dbl>  <dbl>  <dbl>  <dbl>
## 1 tournament    40      0  0.892  0.902  0.902  0.913  0.00521
## 2 lexicase      40      0  0.900  0.905  0.906  0.915  0.00722
```

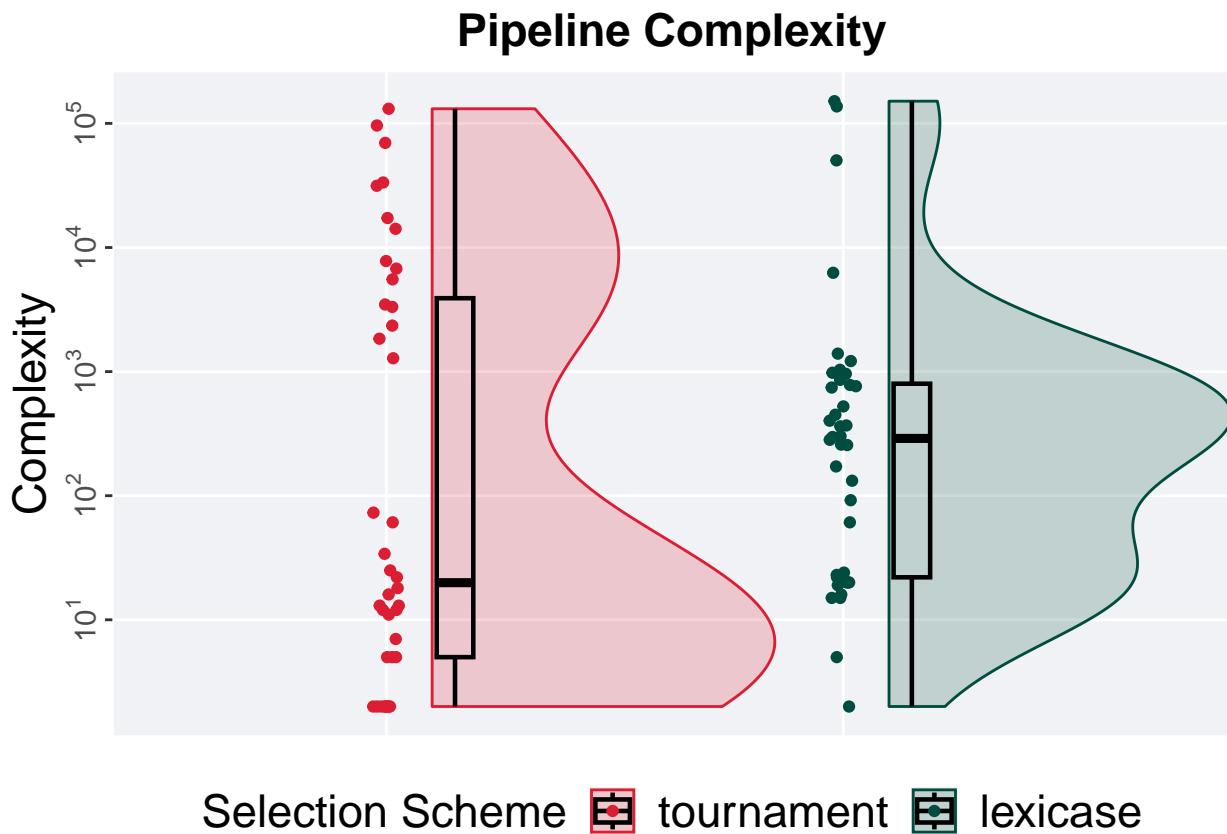
The permutation test revealed that the results are:

```
tournament_results <- filter(task_data, split == '95%' & selection == 'tournament')
lexicase_results <- filter(task_data, split == '95%' & selection == 'lexicase')
permutation_test(tournament_results$training_performance,
                 lexicase_results$training_performance,
                 seed = 80,
                 alternative = "1")
```

```
## [1] "observed_diff: -4.17120455552955"
## [1] "permutation_diffs[0.05 * n_permutations]: -1.68666533186232"
## [1] "reject null hypothesis"
## [1] "p-value: 3e-05"
```



```
complexity_plot(filter(task_data, split == '95%'))
```



Summary statistics for the pipeline complexity of the selection schemes at the 5% selection set split:

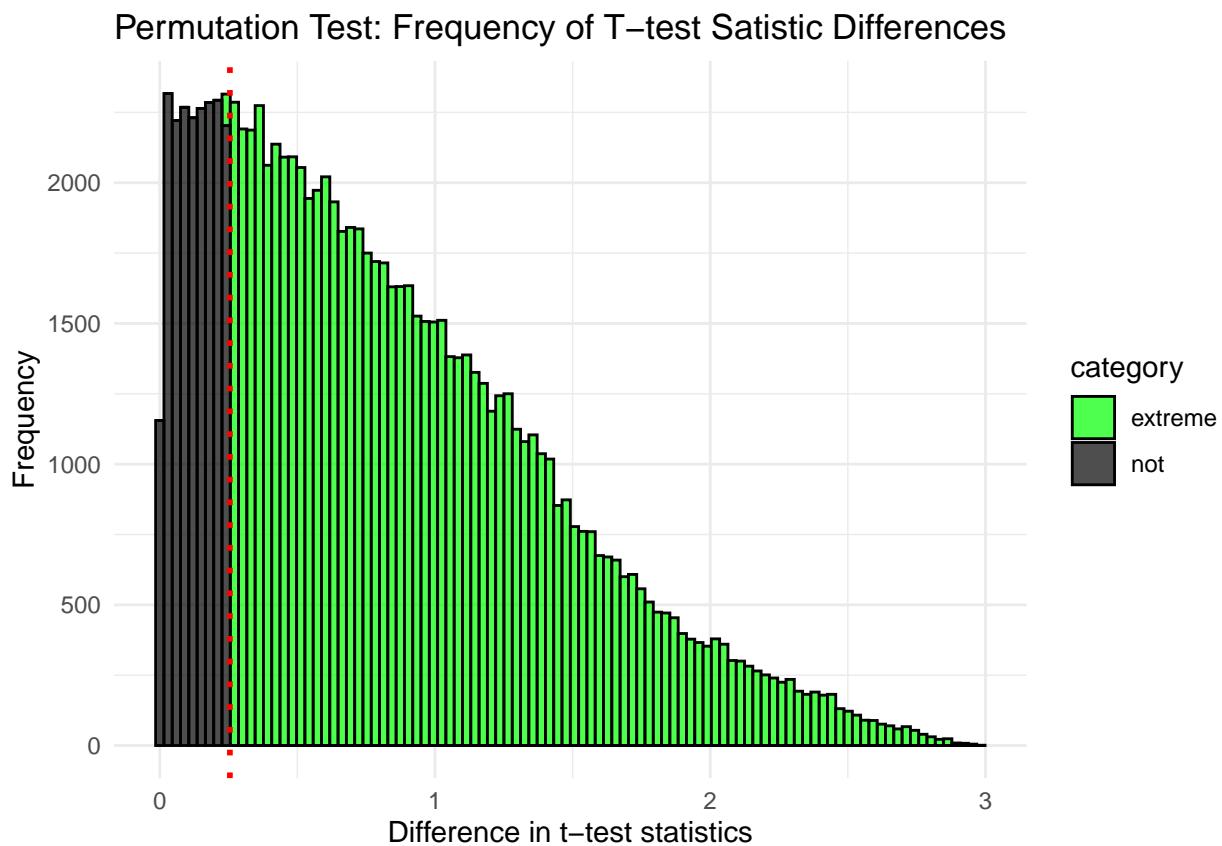
```
complexity_summary(filter(task_data, split == '95%'))
```

```
## # A tibble: 2 x 8
##   selection count na_cnt  min median   mean   max   IQR
##   <fct>     <int> <int> <dbl> <dbl> <dbl> <int> <dbl>
## 1 tournament     40     0     2    20 10652. 131231 3984
## 2 lexicase       40     0     2   290. 8939. 151011  778.
```

The permutation test revealed that the results are:

```
tournament_results <- filter(task_data, split == '95%' & selection == 'tournament')
lexicase_results <- filter(task_data, split == '95%' & selection == 'lexicase')
permutation_test(tournament_results$testing_complexity,
                 lexicase_results$testing_complexity,
                 seed = 227,
                 alternative = "t")

## [1] "observed_diff: 0.254787795716525"
## [1] "lower: -1.98593995917044"
## [1] "upper: 1.98260289829468"
## [1] "fail to reject null hypothesis"
## [1] "p-value: 0.80763"
```



Chapter 10

Task 359959

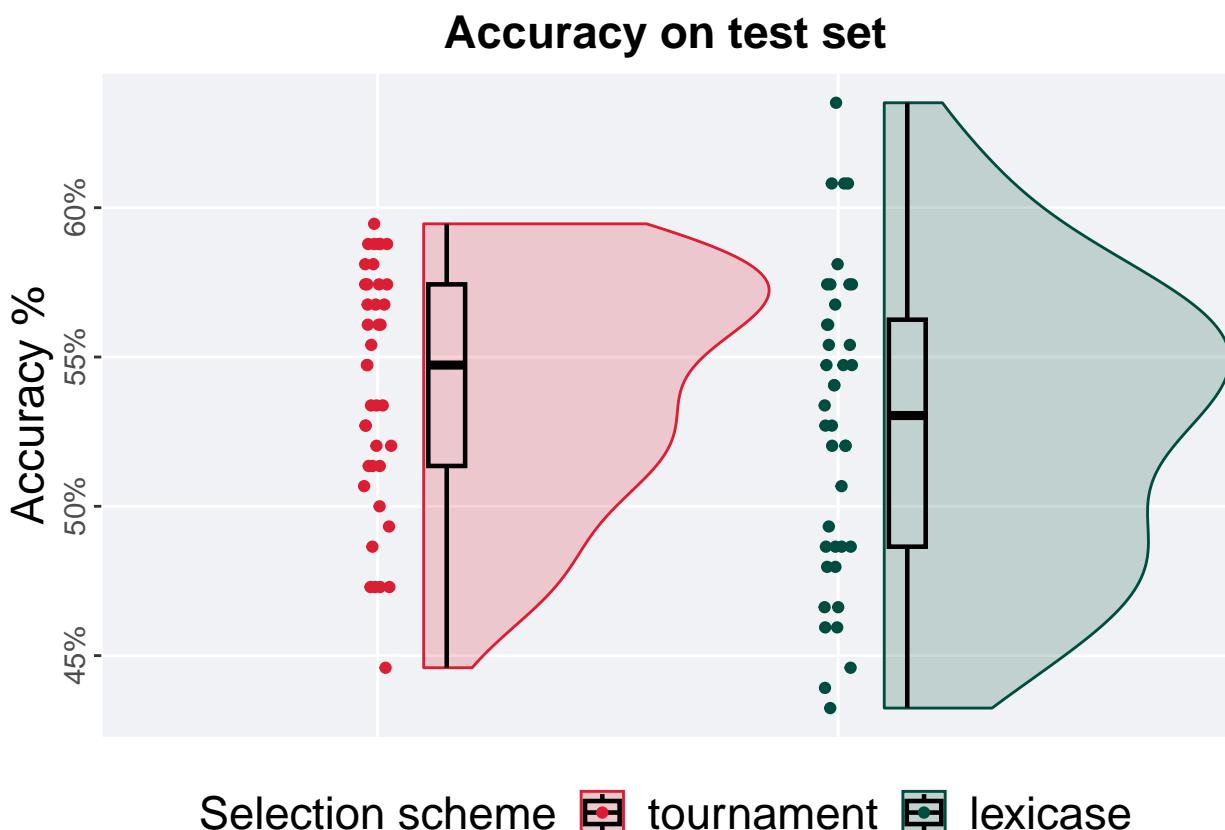
We present the results of our analysis of task 359959 with the different selection set splits used in our study.

```
task_data <- filter(results, task_id == 359959)
```

10.1 5%

10.1.1 Test accuracy

```
test_plot(filter(task_data, split == '5%'))
```



Summary statistics for the testing performance of the selection schemes at the 5% selection set split:

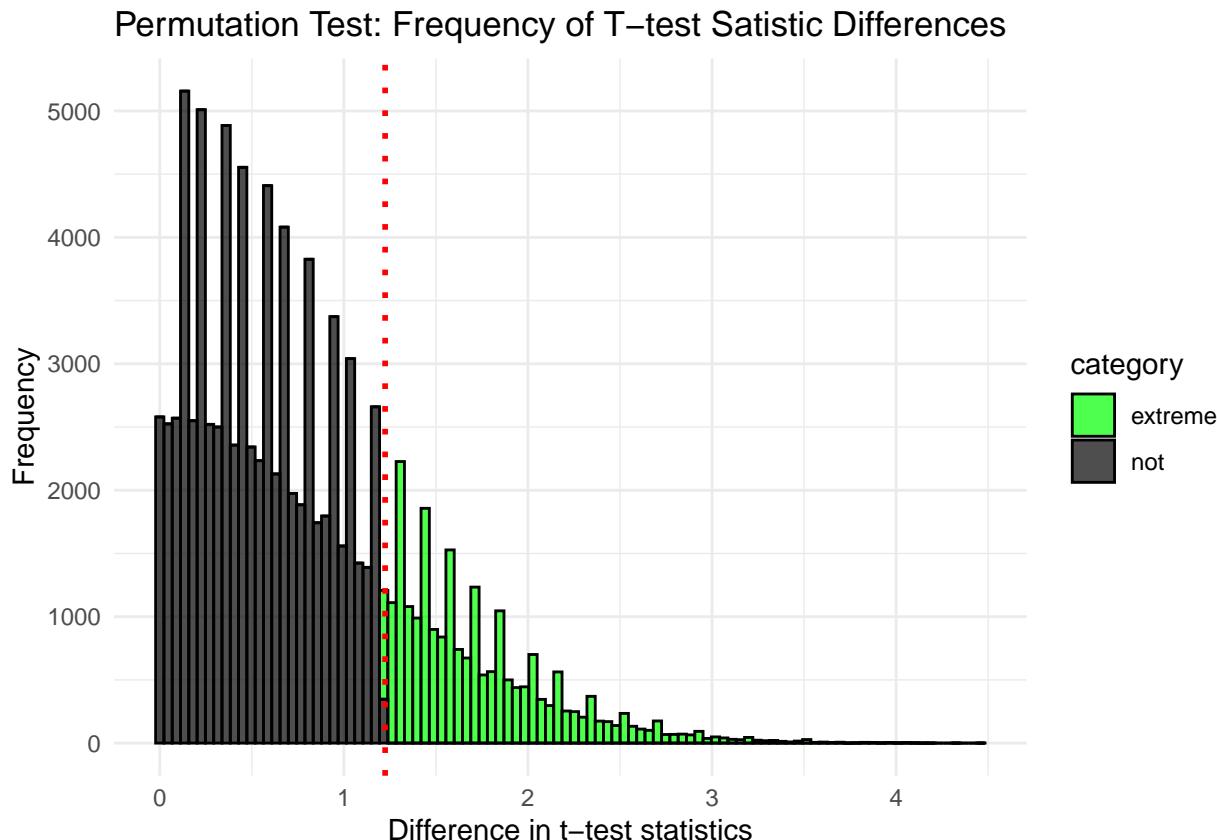
```
test_results_summary(filter(task_data, split == '5%'))
```

```
## # A tibble: 2 x 8
##   selection count na_cnt  min median  mean   max   IQR
##   <fct>     <int> <int> <dbl> <dbl> <dbl> <dbl> <dbl>
## 1 tournament     40     0 0.446  0.547 0.540 0.595 0.0608
## 2 lexicase       40     0 0.432  0.530 0.527 0.635 0.0760
```

The permutation test revealed that the results are:

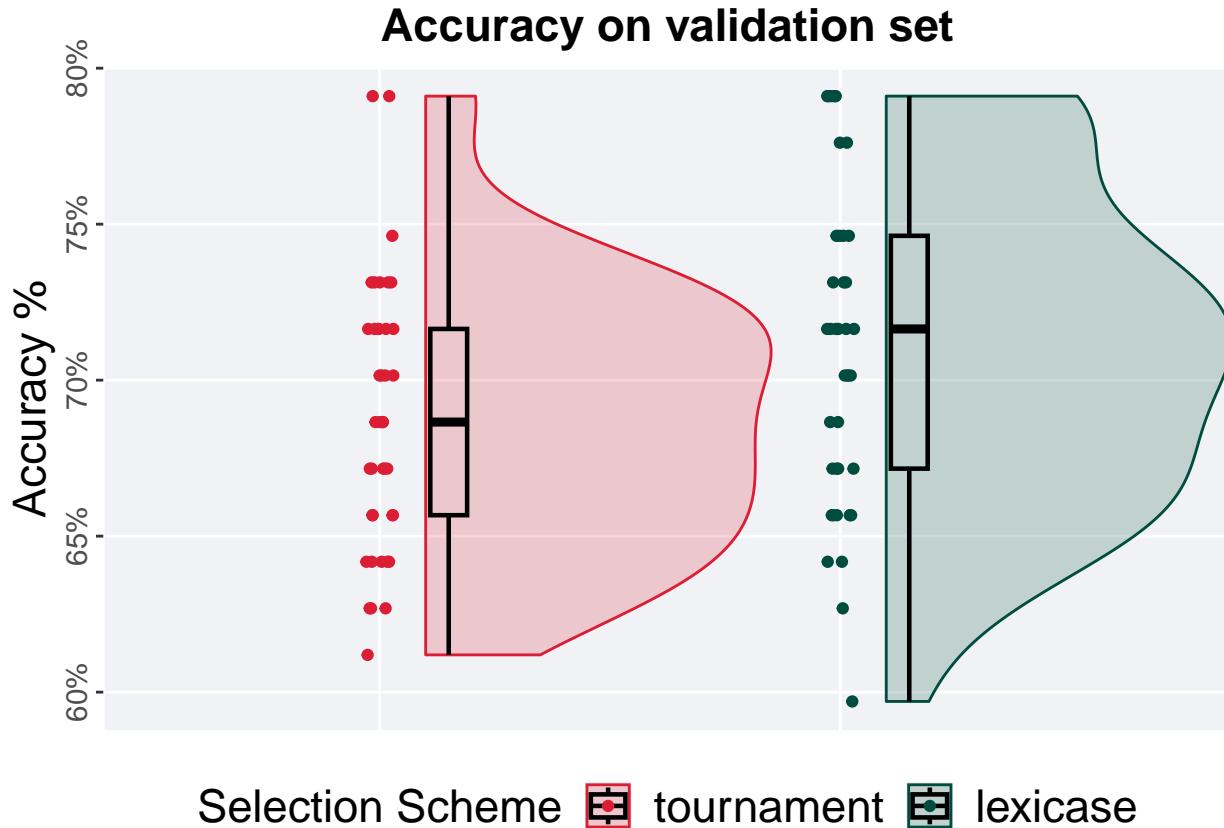
```
tournament_results <- filter(task_data, split == '5%' & selection == 'tournament')
lexicase_results <- filter(task_data, split == '5%' & selection == 'lexicase')
permutation_test(tournament_results$testing_performance,
                  lexicase_results$testing_performance,
                  seed = 81,
                  alternative = "t")
```

```
## [1] "observed_diff: 1.22405437285302"
## [1] "lower: -1.97110919644234"
## [1] "upper: 2.00592908399195"
## [1] "fail to reject null hypothesis"
## [1] "p-value: 0.22576"
```



10.1.2 Validation accuracy

```
validation_plot(filter(task_data, split == '5%'))
```



Summary statistics for the testing performance of the selection schemes at the 5% selection set split:

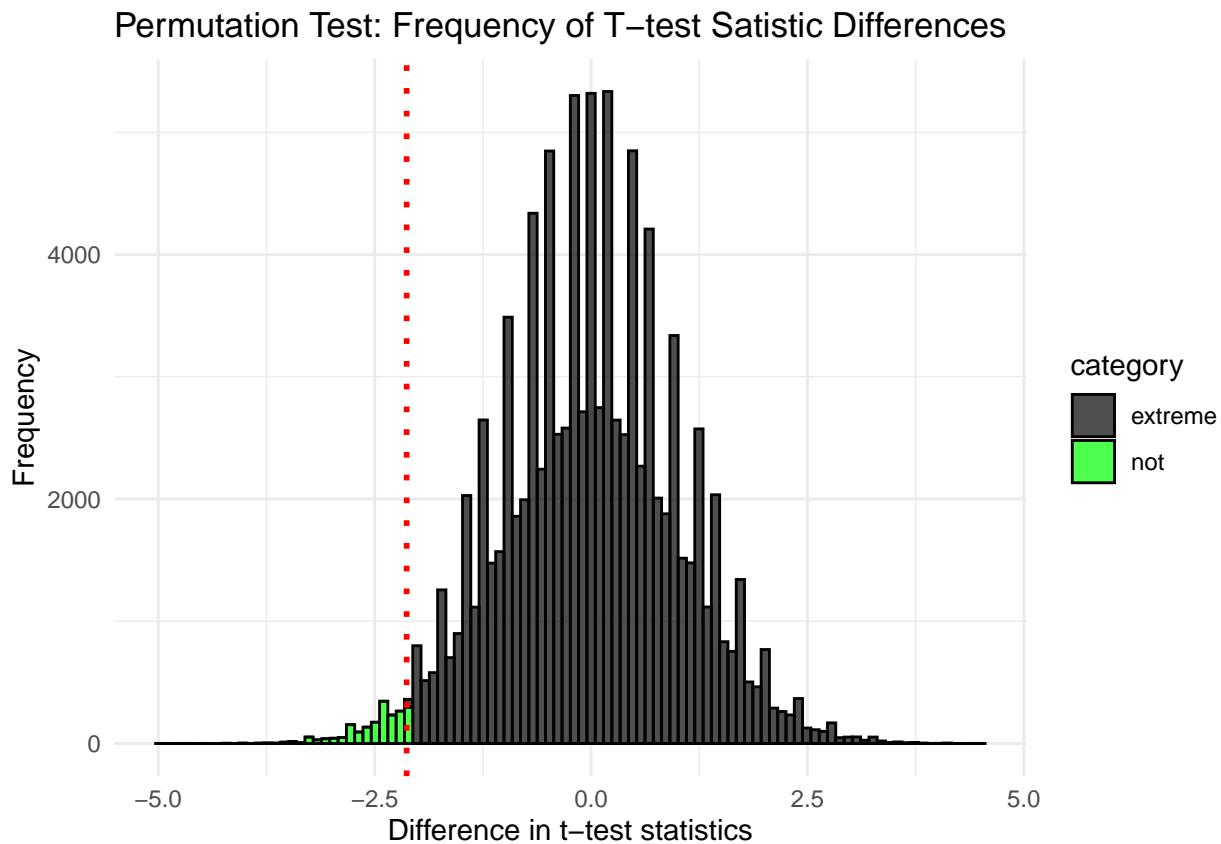
```
validation_accuracy_summary(filter(task_data, split == '5%'))
```

```
## # A tibble: 2 x 8
##   selection count na_cnt  min median  mean   max     IQR
##   <fct>     <int> <int> <dbl> <dbl> <dbl> <dbl> <dbl>
## 1 tournament     40     0 0.612  0.687  0.688  0.791  0.0597
## 2 lexicase       40     0 0.597  0.716  0.711  0.791  0.0746
```

The permutation test revealed that the results are:

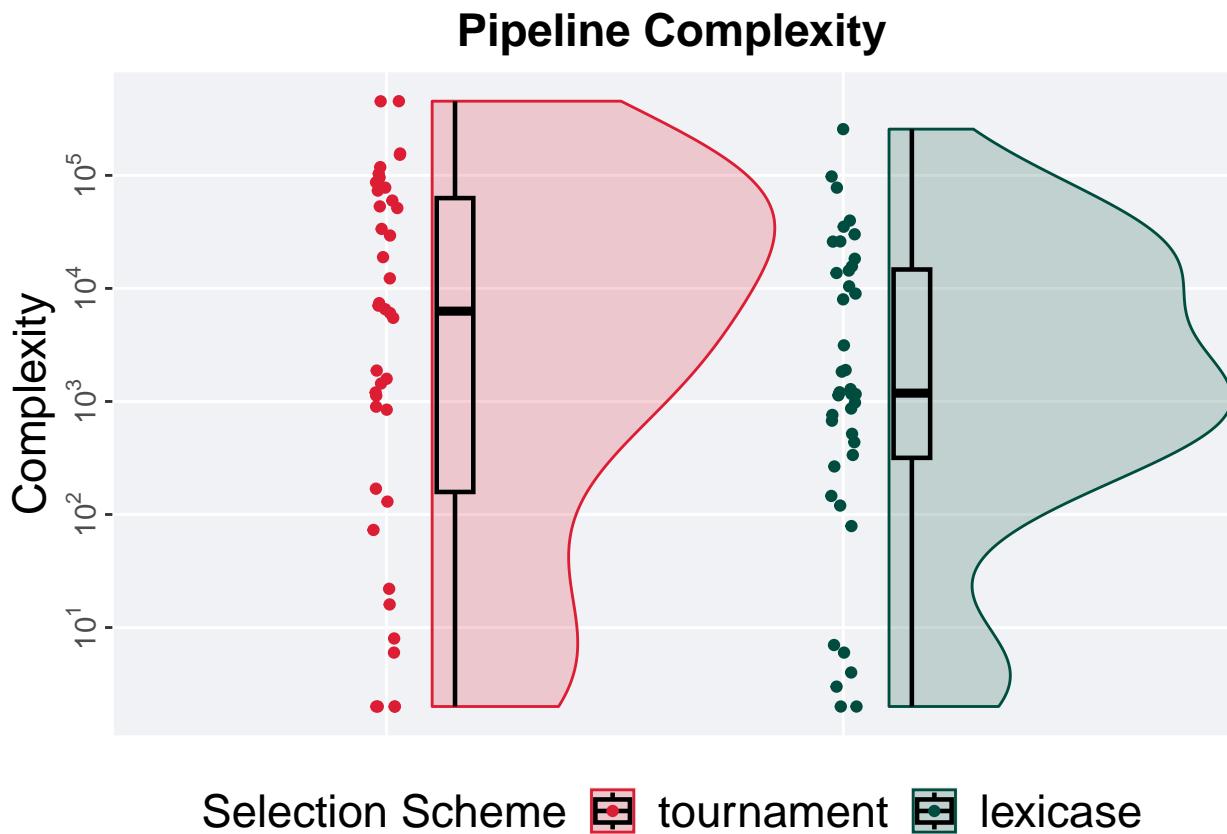
```
tournament_results <- filter(task_data, split == '5%' & selection == 'tournament')
lexicase_results <- filter(task_data, split == '5%' & selection == 'lexicase')
permutation_test(tournament_results$training_performance,
                 lexicase_results$training_performance,
                 seed = 82,
                 alternative = "1")
```

```
## [1] "observed_diff: -2.13091175211124"
## [1] "permutation_diffs[0.05 * n_permutations]: -1.69431049247206"
## [1] "reject null hypothesis"
## [1] "p-value: 0.01963"
```



10.1.3 Pipeline complexity

```
complexity_plot(filter(task_data, split == '5%'))
```



Summary statistics for the pipeline complexity of the selection schemes at the 5% selection set split:

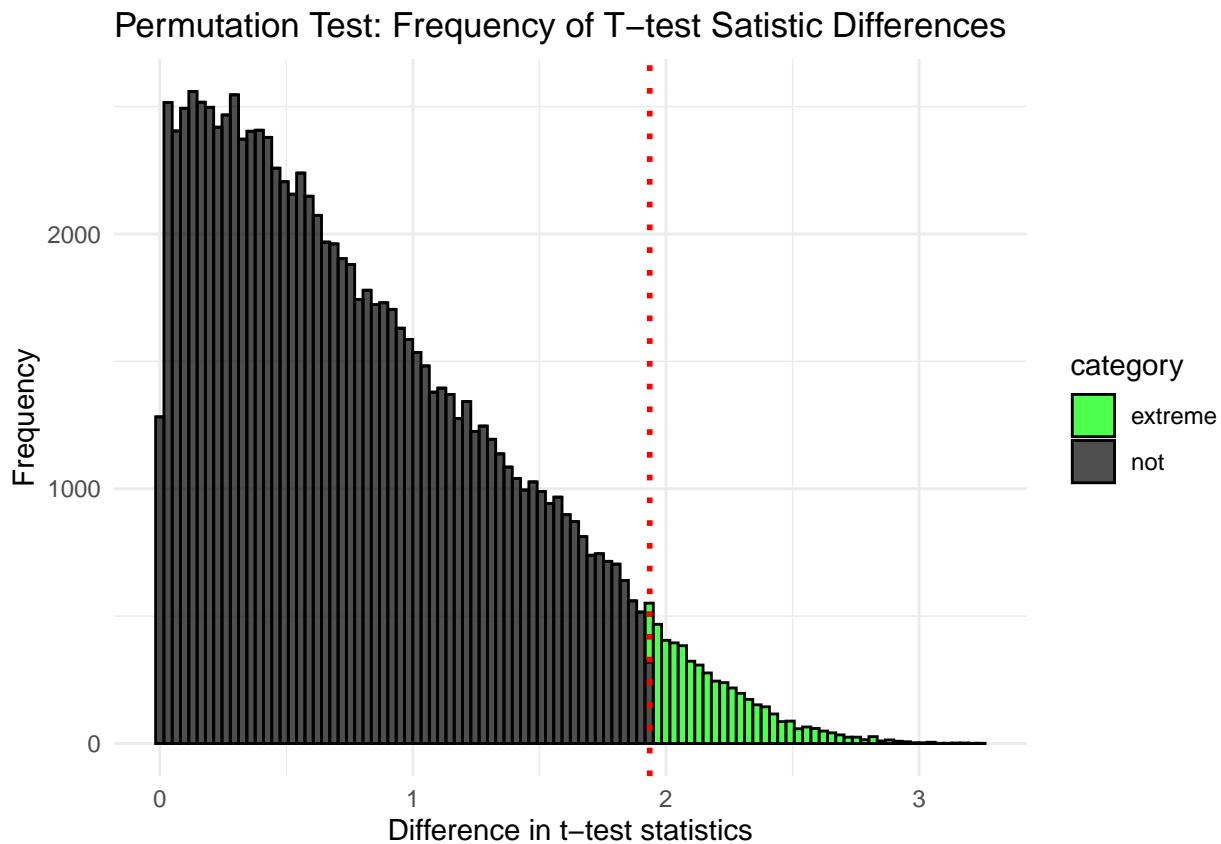
```
complexity_summary(filter(task_data, split == '5%'))
```

```
## # A tibble: 2 x 8
##   selection count na_cnt  min median  mean    max    IQR
##   <fct>     <int> <int> <dbl> <dbl> <dbl> <int> <dbl>
## 1 tournament     40     0     2  6301  51775. 453472 63067.
## 2 lexicase       40     0     2 1186. 17428. 256641 14401.
```

The permutation test revealed that the results are:

```
tournament_results <- filter(task_data, split == '5%' & selection == 'tournament')
lexicase_results <- filter(task_data, split == '5%' & selection == 'lexicase')
permutation_test(tournament_results$testing_complexity,
                 lexicase_results$testing_complexity,
                 seed = 228,
                 alternative = "t")
```

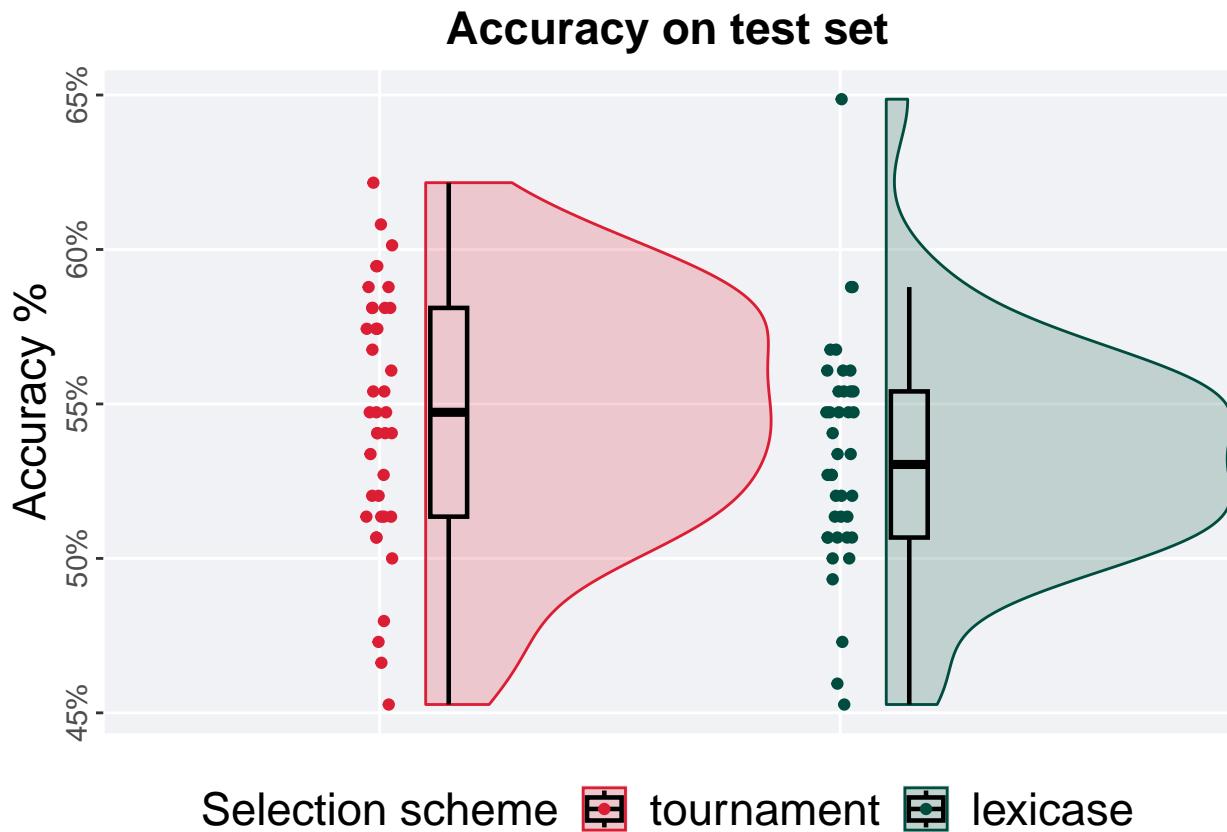
```
## [1] "observed_diff: 1.93562377930969"
## [1] "lower: -1.93105018066448"
## [1] "upper: 1.93074611405436"
## [1] "reject null hypothesis"
## [1] "p-value: 0.04907"
```



10.2 10%

10.2.1 Test accuracy

```
test_plot(filter(task_data, split == '10%'))
```



Summary statistics for the testing performance of the selection schemes at the 5% selection set split:

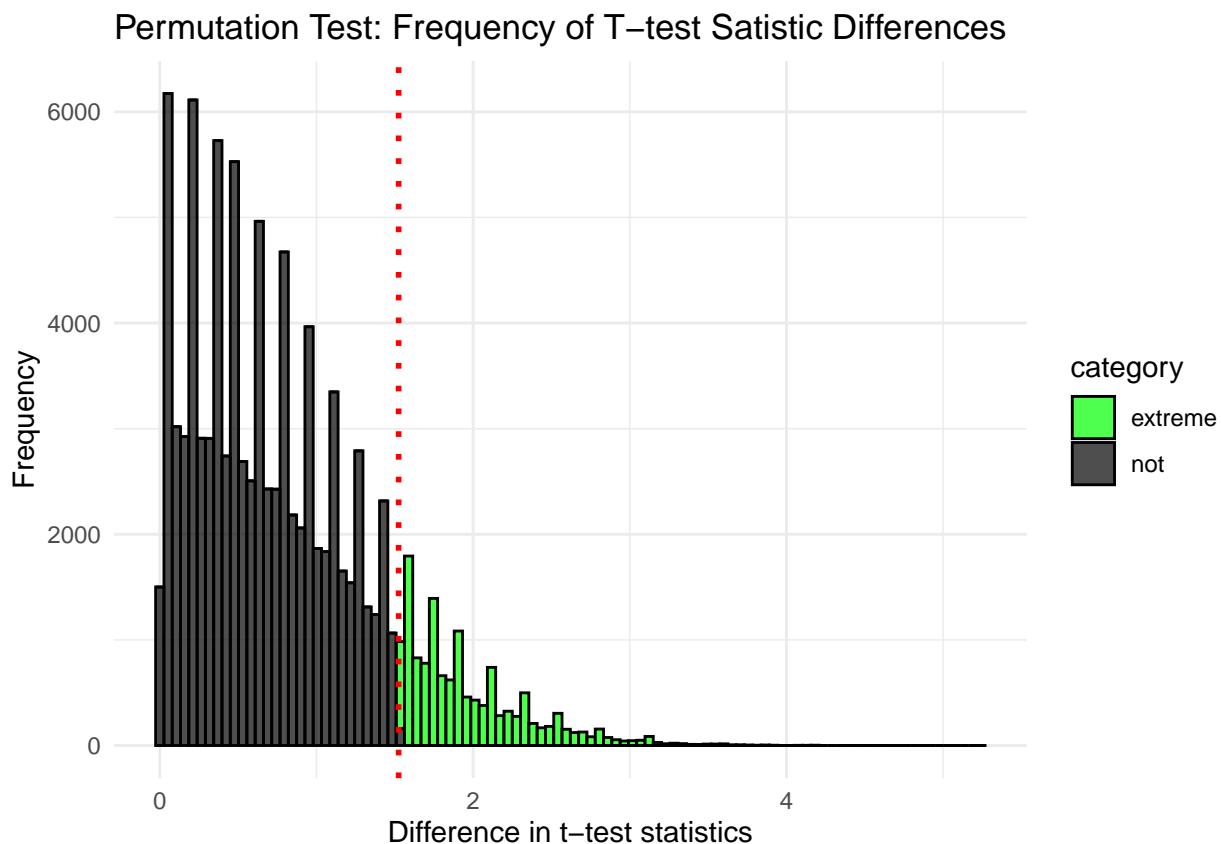
```
test_results_summary(filter(task_data, split == '10%'))
```

```
## # A tibble: 2 x 8
##   selection count na_cnt  min median  mean   max     IQR
##   <fct>     <int> <int> <dbl> <dbl> <dbl> <dbl> <dbl>
## 1 tournament     40     0 0.453 0.547 0.546 0.622 0.0676
## 2 lexicase       40     0 0.453 0.530 0.532 0.649 0.0473
```

The permutation test revealed that the results are:

```
tournament_results <- filter(task_data, split == '10%' & selection == 'tournament')
lexicase_results <- filter(task_data, split == '10%' & selection == 'lexicase')
permutation_test(tournament_results$testing_performance,
                 lexicase_results$testing_performance,
                 seed = 83,
                 alternative = "t")
```

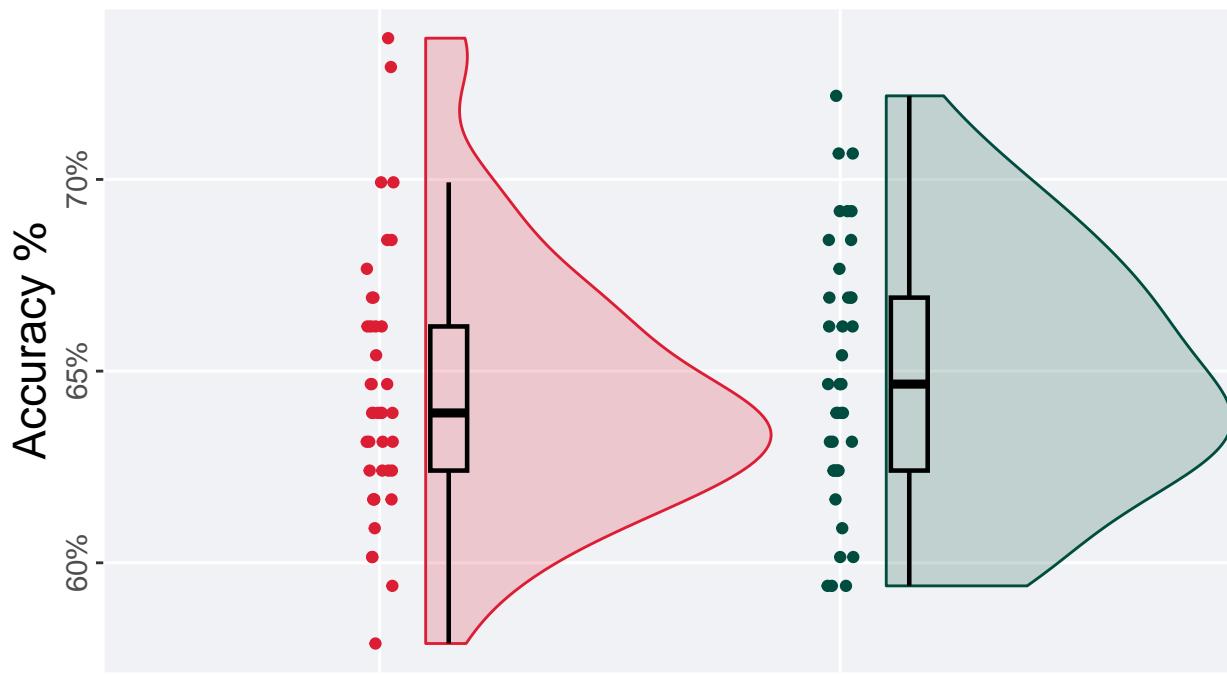
```
## [1] "observed_diff: 1.52463147615729"
## [1] "lower: -2.01518489005153"
## [1] "upper: 1.97367777918776"
## [1] "fail to reject null hypothesis"
## [1] "p-value: 0.13434"
```



10.2.2 Validation accuracy

```
validation_plot(filter(task_data, split == '10%'))
```

Accuracy on validation set



Selection Scheme tournament lexicase

Summary statistics for the testing performance of the selection schemes at the 5% selection set split:

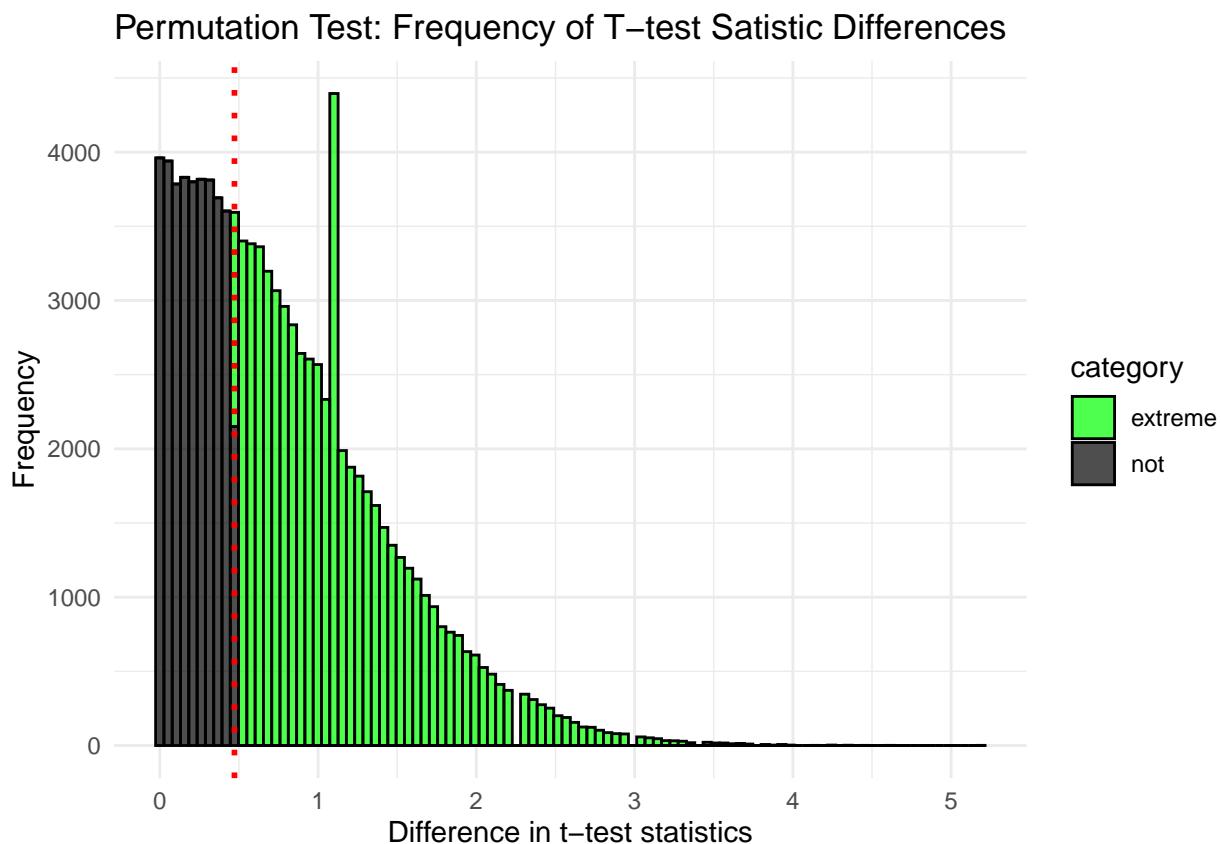
```
validation_accuracy_summary(filter(task_data, split == '10%'))
```

```
## # A tibble: 2 x 8
##   selection count na_cnt  min median  mean   max     IQR
##   <fct>     <int> <int> <dbl> <dbl> <dbl> <dbl> <dbl>
## 1 tournament     40     0 0.579 0.639 0.645 0.737 0.0376
## 2 lexicase       40     0 0.594 0.647 0.648 0.722 0.0451
```

The permutation test revealed that the results are:

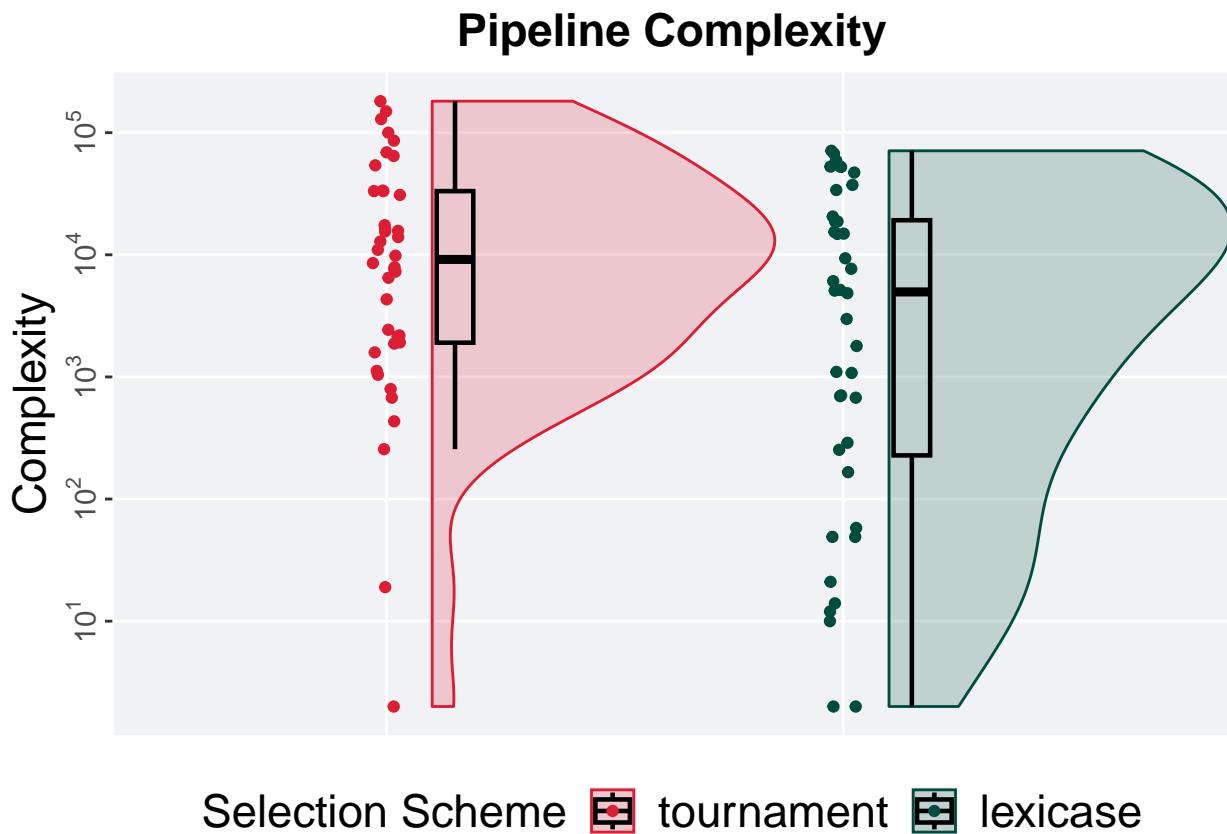
```
tournament_results <- filter(task_data, split == '10%' & selection == 'tournament')
lexicase_results <- filter(task_data, split == '10%' & selection == 'lexicase')
permutation_test(tournament_results$training_performance,
                 lexicase_results$training_performance,
                 seed = 84,
                 alternative = "t")
```

```
## [1] "observed_diff: -0.471913297591953"
## [1] "lower: -2.0094448325709"
## [1] "upper: 2.00944582884462"
## [1] "fail to reject null hypothesis"
## [1] "p-value: 0.63609"
```



10.2.3 Pipeline complexity

```
complexity_plot(filter(task_data, split == '10%'))
```



Summary statistics for the pipeline complexity of the selection schemes at the 5% selection set split:

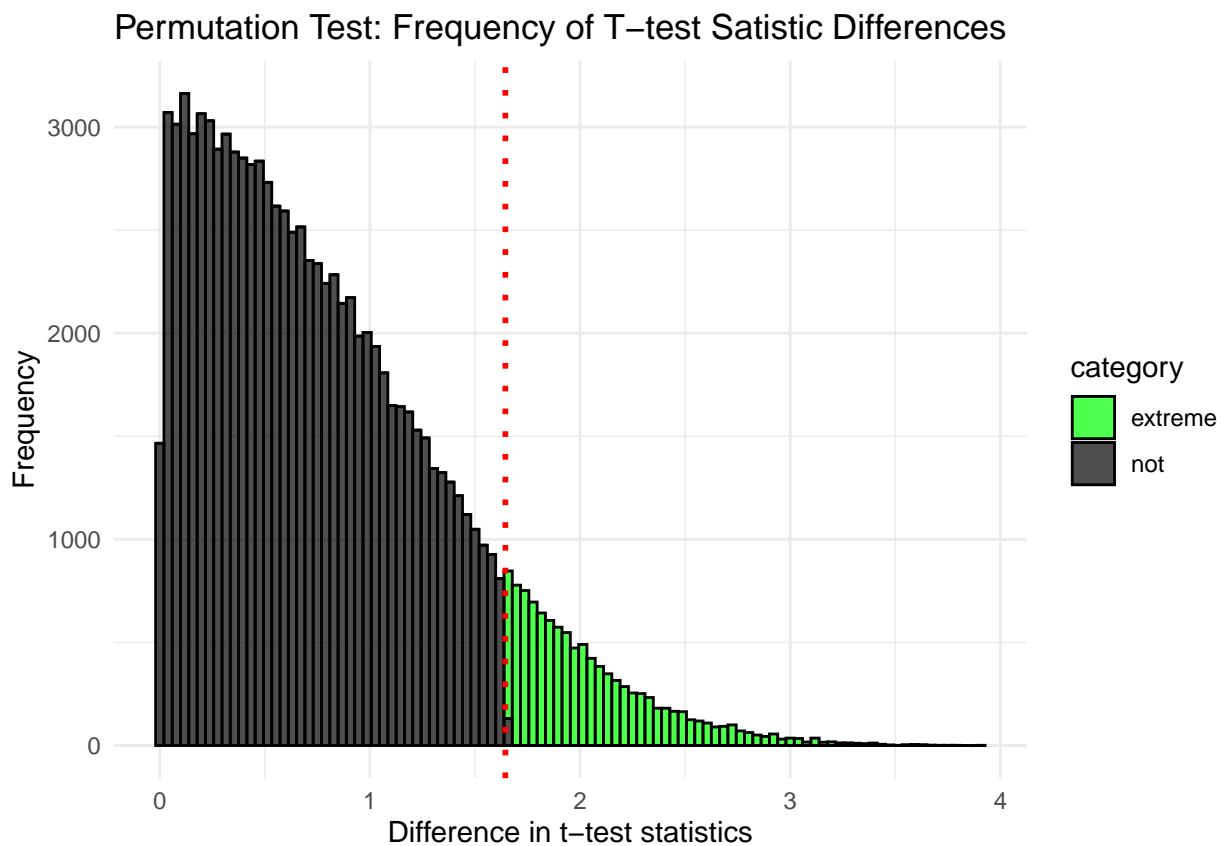
```
complexity_summary(filter(task_data, split == '10%'))
```

```
## # A tibble: 2 x 8
##   selection count na_cnt  min median   mean   max   IQR
##   <fct>     <int> <int> <dbl> <dbl> <dbl> <int> <dbl>
## 1 tournament     40     0     2  9191 28386. 181091 31333
## 2 lexicase       40     0     2  4969 15645.  71101 18977.
```

The permutation test revealed that the results are:

```
tournament_results <- filter(task_data, split == '10%' & selection == 'tournament')
lexicase_results <- filter(task_data, split == '10%' & selection == 'lexicase')
permutation_test(tournament_results$testing_complexity,
                 lexicase_results$testing_complexity,
                 seed = 229,
                 alternative = "t")
```

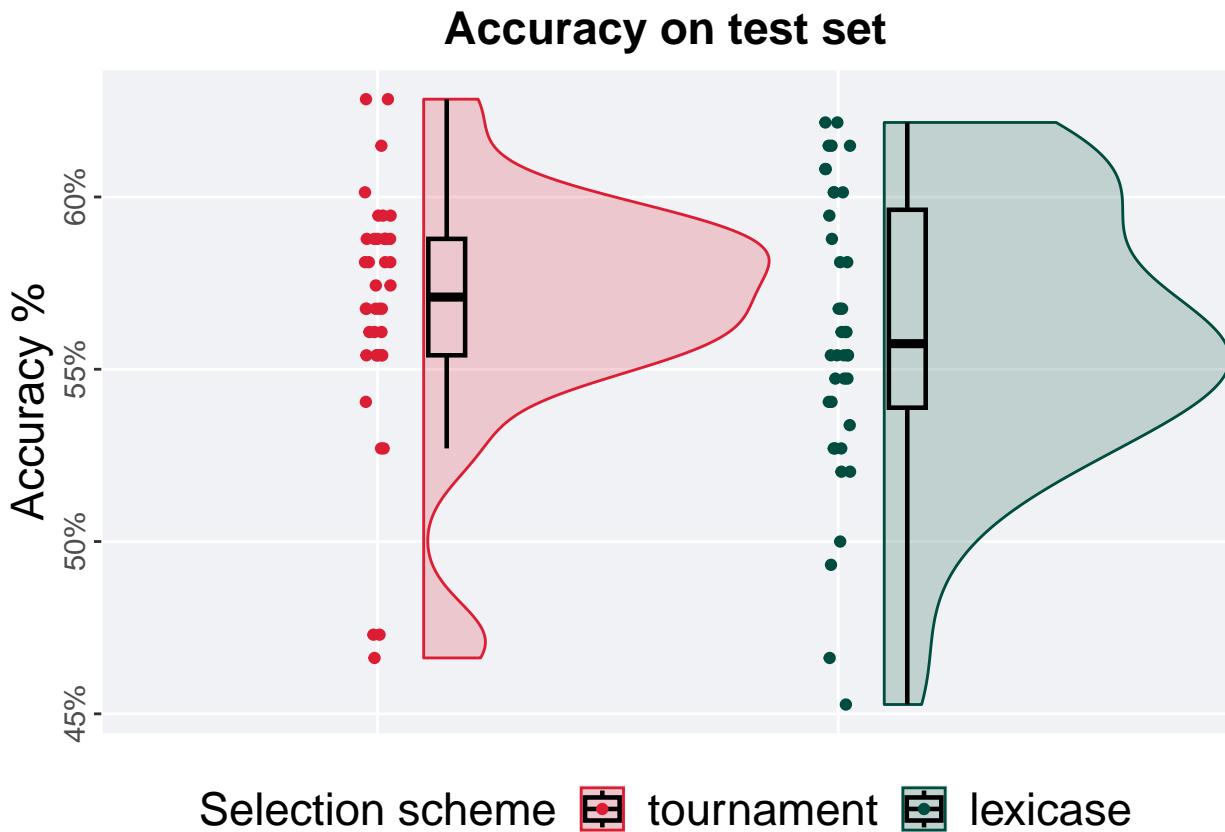
```
## [1] "observed_diff: 1.64447259548299"
## [1] "lower: -1.9785677519868"
## [1] "upper: 1.98485084066634"
## [1] "fail to reject null hypothesis"
## [1] "p-value: 0.10662"
```



10.3 50%

10.3.1 Test accuracy

```
test_plot(filter(task_data, split == '50%'))
```



Summary statistics for the testing performance of the selection schemes at the 5% selection set split:

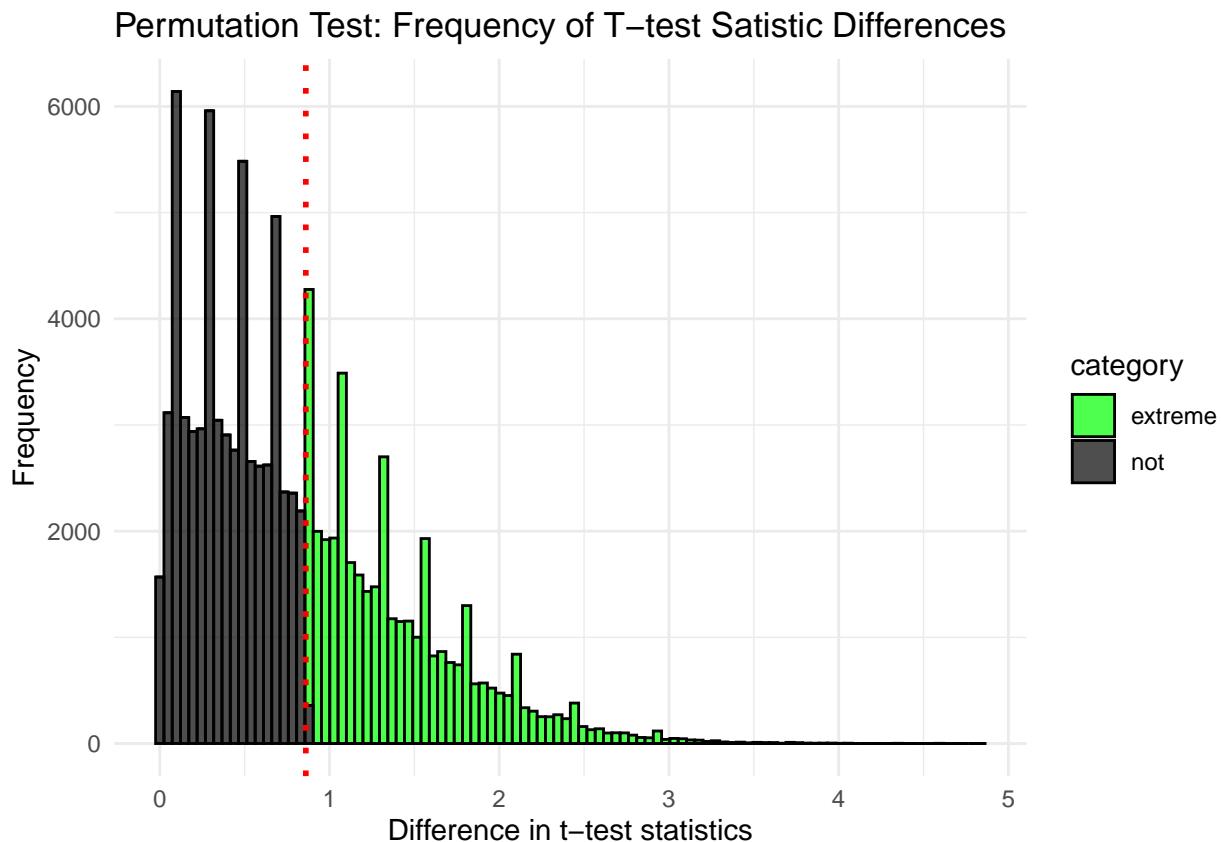
```
test_results_summary(filter(task_data, split == '50%'))
```

```
## # A tibble: 2 x 8
##   selection count na_cnt  min median  mean   max     IQR
##   <fct>     <int> <int> <dbl> <dbl> <dbl> <dbl> <dbl>
## 1 tournament     40     0 0.466 0.571 0.567 0.628 0.0338
## 2 lexicase       40     0 0.453 0.557 0.560 0.622 0.0574
```

The permutation test revealed that the results are:

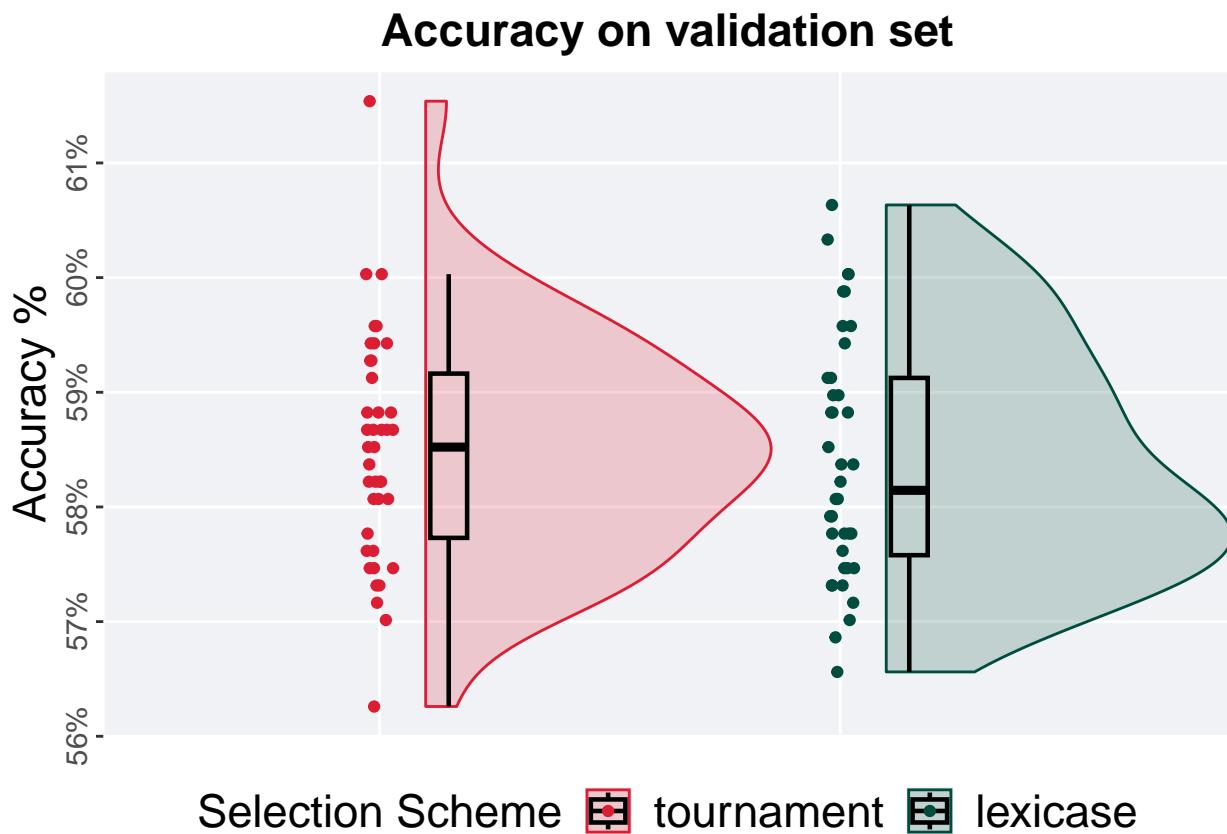
```
tournament_results <- filter(task_data, split == '50%' & selection == 'tournament')
lexicase_results <- filter(task_data, split == '50%' & selection == 'lexicase')
permutation_test(tournament_results$testing_performance,
                 lexicase_results$testing_performance,
                 seed = 85,
                 alternative = "t")
```

```
## [1] "observed_diff: 0.86047817755061"
## [1] "lower: -1.99548160987446"
## [1] "upper: 1.99548229277239"
## [1] "fail to reject null hypothesis"
## [1] "p-value: 0.39919"
```



10.3.2 Validation accuracy

```
validation_plot(filter(task_data, split == '50%'))
```



Summary statistics for the testing performance of the selection schemes at the 5% selection set split:

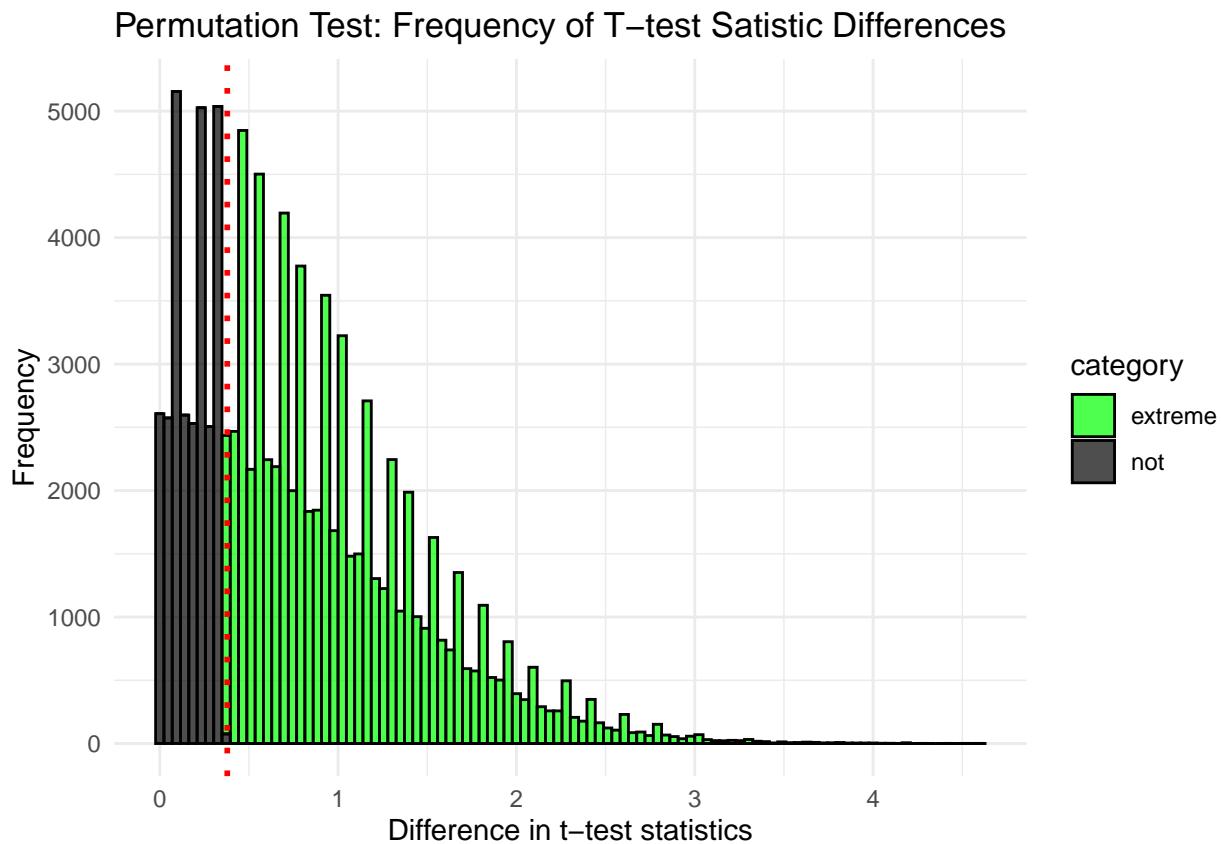
```
validation_accuracy_summary(filter(task_data, split == '50%'))
```

```
## # A tibble: 2 x 8
##   selection count na_cnt  min median  mean   max     IQR
##   <fct>     <int> <int> <dbl> <dbl> <dbl> <dbl> <dbl>
## 1 tournament     40     0 0.563 0.585 0.585 0.615 0.0143
## 2 lexicase       40     0 0.566 0.581 0.584 0.606 0.0155
```

The permutation test revealed that the results are:

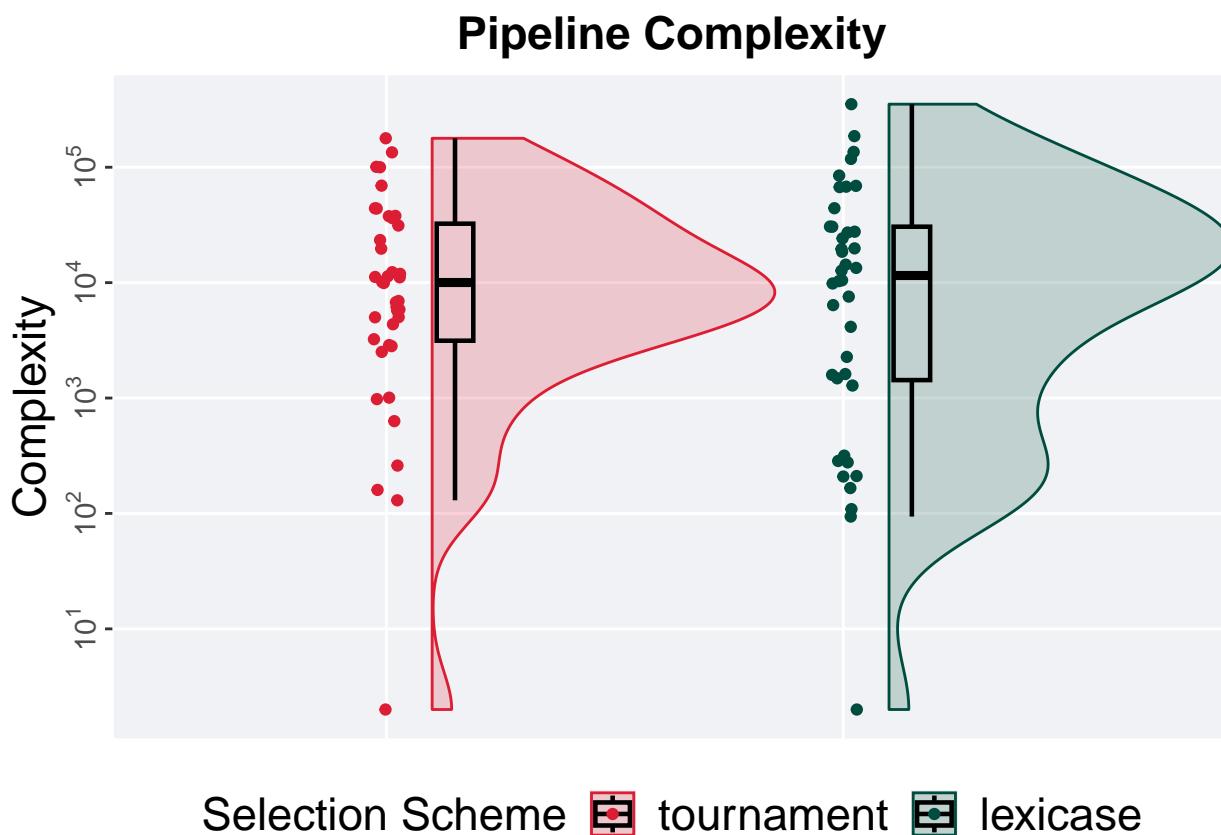
```
tournament_results <- filter(task_data, split == '50%' & selection == 'tournament')
lexicase_results <- filter(task_data, split == '50%' & selection == 'lexicase')
permutation_test(tournament_results$training_performance,
                 lexicase_results$training_performance,
                 seed = 86,
                 alternative = "t")

## [1] "observed_diff: 0.378541953307579"
## [1] "lower: -2.00662736312658"
## [1] "upper: 1.97121031816678"
## [1] "fail to reject null hypothesis"
## [1] "p-value: 0.7189"
```



10.3.3 Pipeline complexity

```
complexity_plot(filter(task_data, split == '50%'))
```



Summary statistics for the pipeline complexity of the selection schemes at the 5% selection set split:

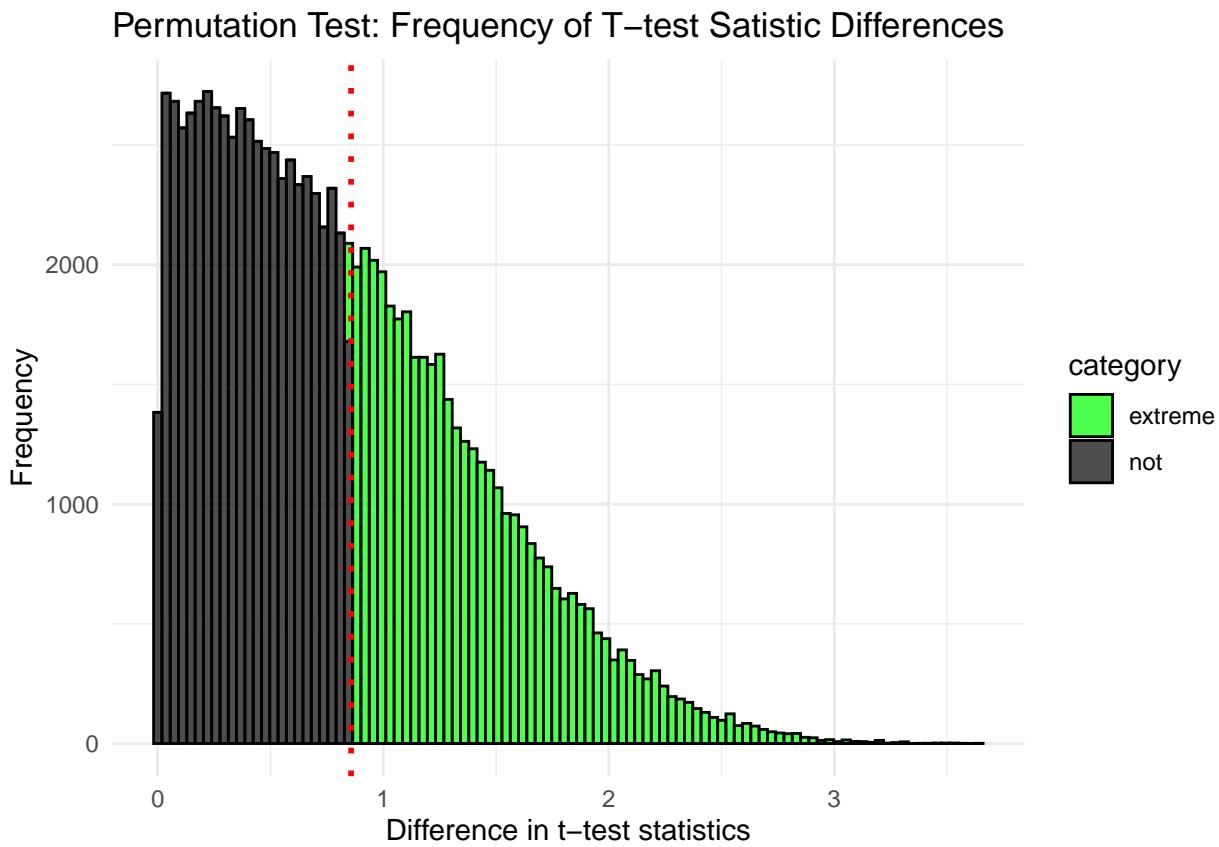
```
complexity_summary(filter(task_data, split == '50%'))
```

```
## # A tibble: 2 x 8
##   selection count na_cnt  min median   mean   max   IQR
##   <fct>     <int> <int> <dbl> <dbl> <dbl> <int> <dbl>
## 1 tournament     40     0     2 10024 25136. 178221 29345.
## 2 lexicase       40     0     2 11586 35546. 351831 29105.
```

The permutation test revealed that the results are:

```
tournament_results <- filter(task_data, split == '50%' & selection == 'tournament')
lexicase_results <- filter(task_data, split == '50%' & selection == 'lexicase')
permutation_test(tournament_results$testing_complexity,
                 lexicase_results$testing_complexity,
                 seed = 230,
                 alternative = "t")
```

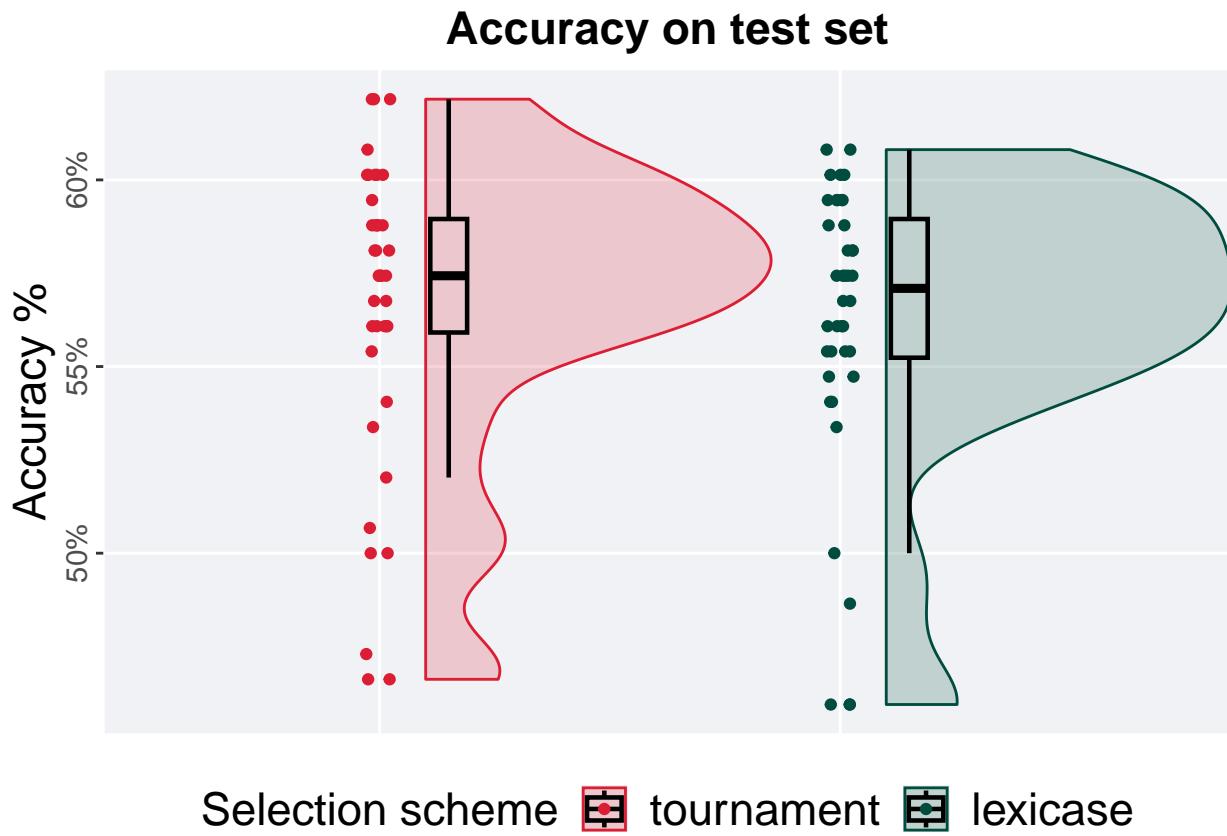
```
## [1] "observed_diff: -0.85713044327636"
## [1] "lower: -1.92108196651047"
## [1] "upper: 1.92449922073779"
## [1] "fail to reject null hypothesis"
## [1] "p-value: 0.41994"
```



10.4 90%

10.4.1 Test accuracy

```
test_plot(filter(task_data, split == '90%'))
```



Summary statistics for the testing performance of the selection schemes at the 5% selection set split:

```
test_results_summary(filter(task_data, split == '90%'))
```

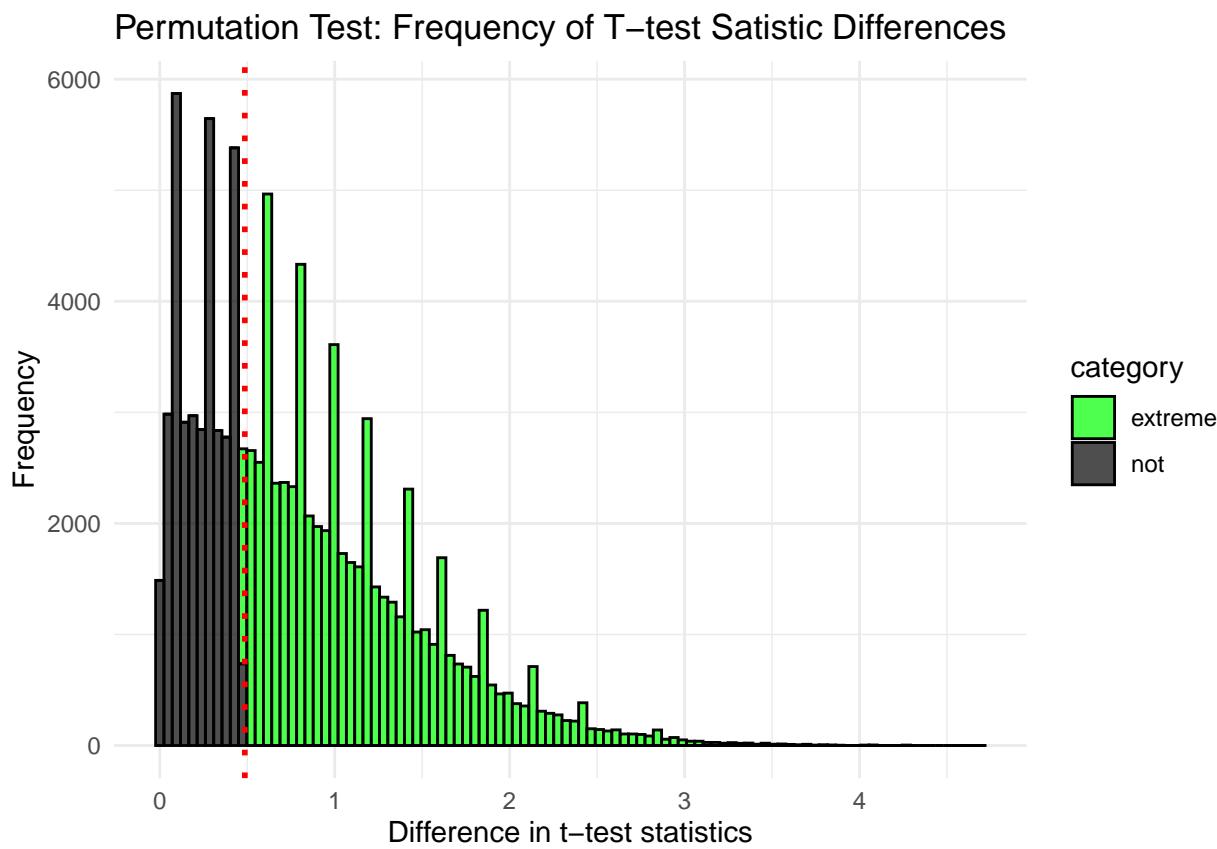
```
## # A tibble: 2 x 8
##   selection count na_cnt  min median  mean  max    IQR
##   <fct>     <int> <int> <dbl> <dbl> <dbl> <dbl> <dbl>
## 1 tournament     40     0 0.466 0.574 0.566 0.622 0.0304
## 2 lexicase       40     0 0.459 0.571 0.561 0.608 0.0372
```

The permutation test revealed that the results are:

```
tournament_results <- filter(task_data, split == '90%' & selection == 'tournament')
lexicase_results <- filter(task_data, split == '90%' & selection == 'lexicase')
```

```
permutation_test(tournament_results$testing_performance,
                  lexicase_results$testing_performance,
                  seed = 87,
                  alternative = "t")
```

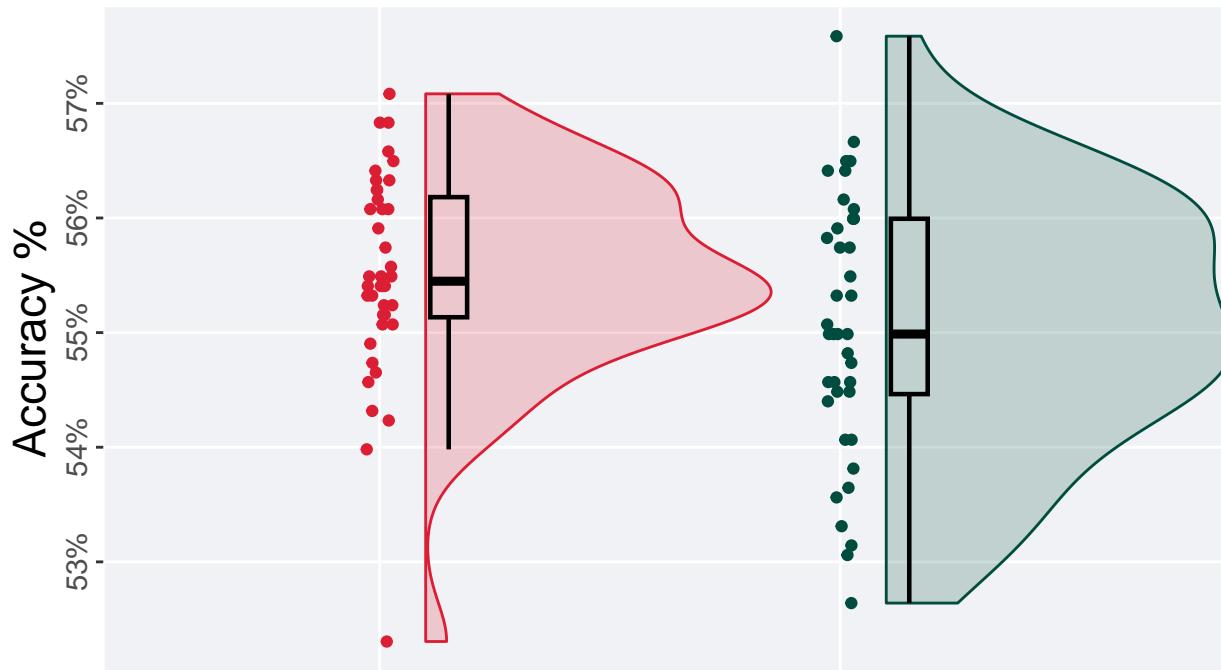
```
## [1] "observed_diff: 0.486150157327831"
## [1] "lower: -1.99036008856116"
## [1] "upper: 1.99036008856116"
## [1] "fail to reject null hypothesis"
## [1] "p-value: 0.63556"
```



10.4.2 Validation accuracy

```
validation_plot(filter(task_data, split == '90%'))
```

Accuracy on validation set



Selection Scheme  tournament  lexicase

Summary statistics for the testing performance of the selection schemes at the 5% selection set split:

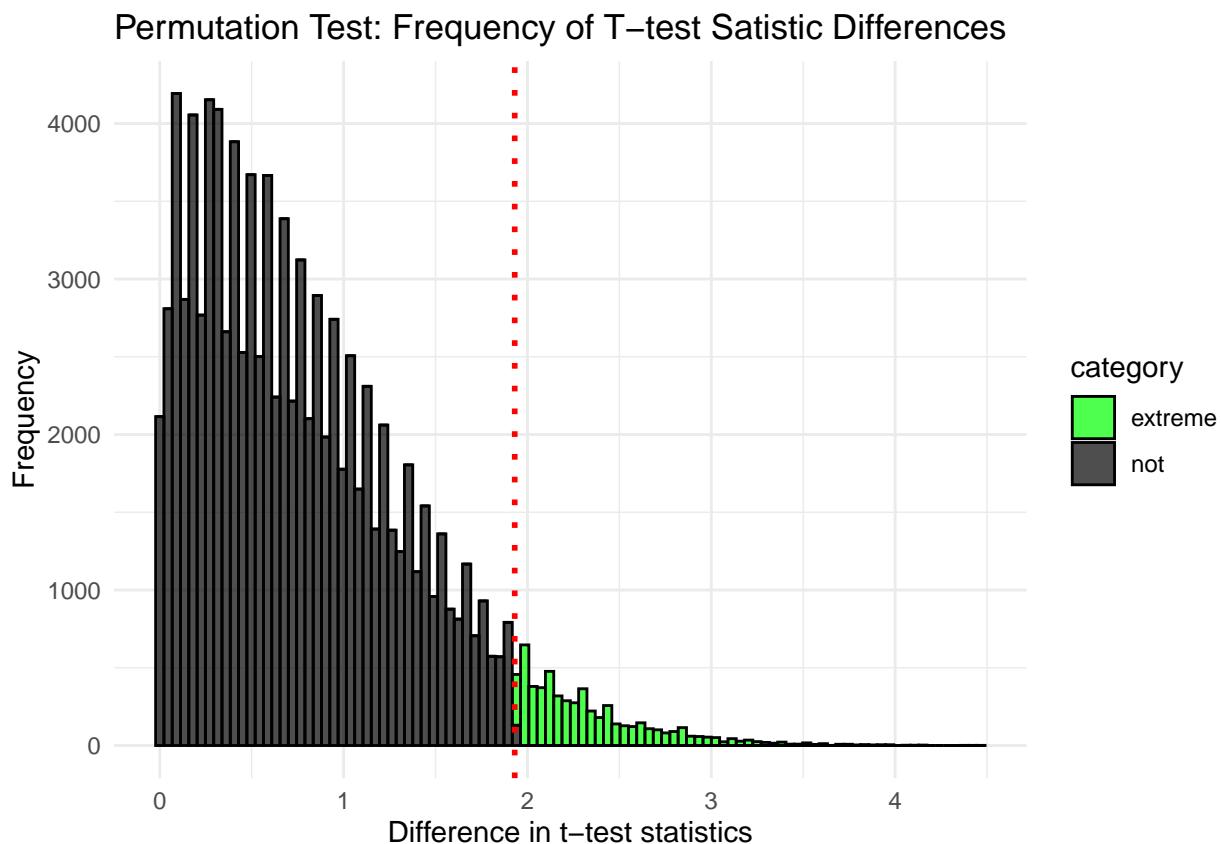
```
validation_accuracy_summary(filter(task_data, split == '90%'))
```

```
## # A tibble: 2 x 8
##   selection count na_cnt  min median  mean   max     IQR
##   <fct>     <int> <int> <dbl> <dbl> <dbl> <dbl> <dbl>
## 1 tournament     40     0 0.523 0.554 0.555 0.571 0.0105
## 2 lexicase       40     0 0.526 0.550 0.551 0.576 0.0153
```

The permutation test revealed that the results are:

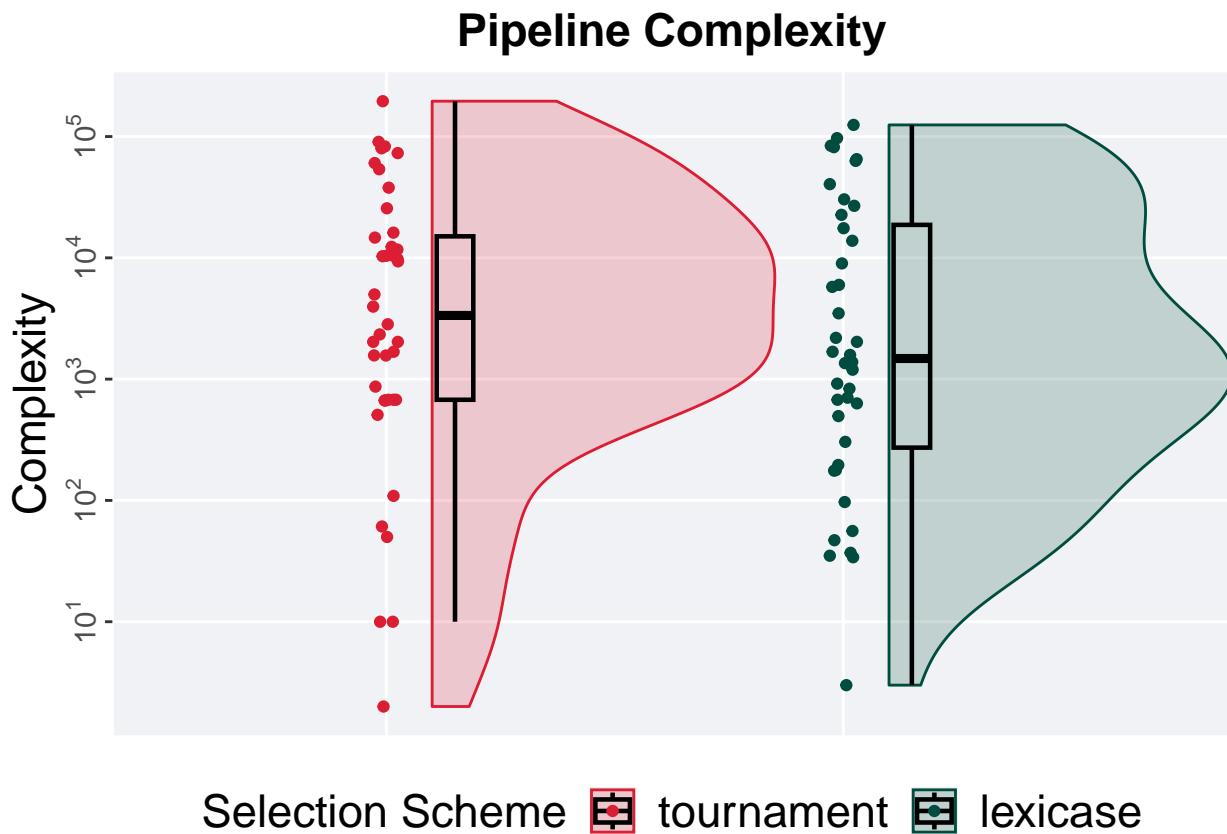
```
tournament_results <- filter(task_data, split == '90%' & selection == 'tournament')
lexicase_results <- filter(task_data, split == '90%' & selection == 'lexicase')
permutation_test(tournament_results$training_performance,
                 lexicase_results$training_performance,
                 seed = 88,
                 alternative = "t")
```

```
## [1] "observed_diff: 1.93069243094997"
## [1] "lower: -2.00723673836918"
## [1] "upper: 1.98806145362816"
## [1] "fail to reject null hypothesis"
## [1] "p-value: 0.05679"
```



10.4.3 Pipeline complexity

```
complexity_plot(filter(task_data, split == '90%'))
```



Summary statistics for the pipeline complexity of the selection schemes at the 5% selection set split:

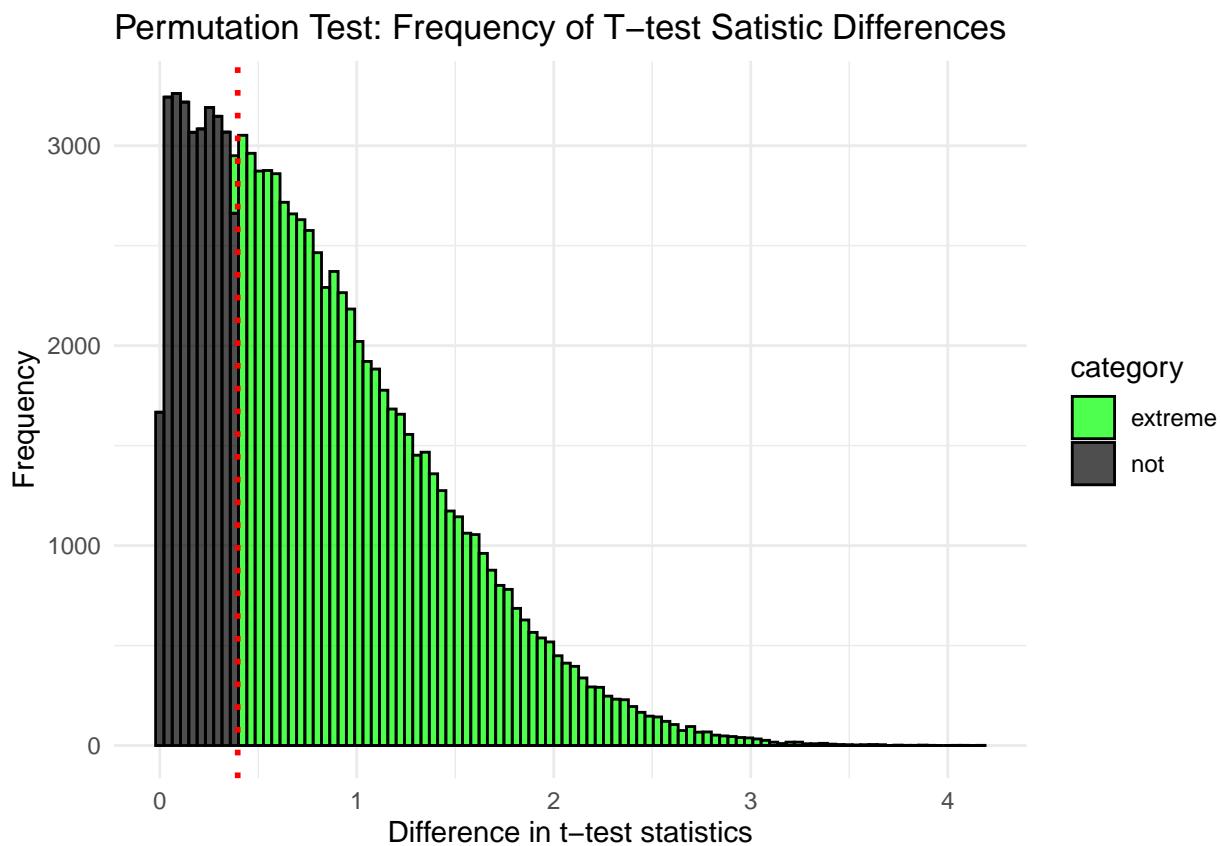
```
complexity_summary(filter(task_data, split == '90%'))
```

```
## # A tibble: 2 x 8
##   selection count na_cnt  min median   mean   max   IQR
##   <fct>     <int> <int> <dbl> <dbl> <dbl> <int> <dbl>
## 1 tournament     40     0     2   3397 20818. 195601 14376
## 2 lexicase       40     0     3   1483 17705. 124691 18538.
```

The permutation test revealed that the results are:

```
tournament_results <- filter(task_data, split == '90%' & selection == 'tournament')
lexicase_results <- filter(task_data, split == '90%' & selection == 'lexicase')
permutation_test(tournament_results$testing_complexity,
                 lexicase_results$testing_complexity,
                 seed = 231,
                 alternative = "t")

## [1] "observed_diff: 0.395735033956951"
## [1] "lower: -1.942667374373"
## [1] "upper: 1.97190877127037"
## [1] "fail to reject null hypothesis"
## [1] "p-value: 0.70392"
```

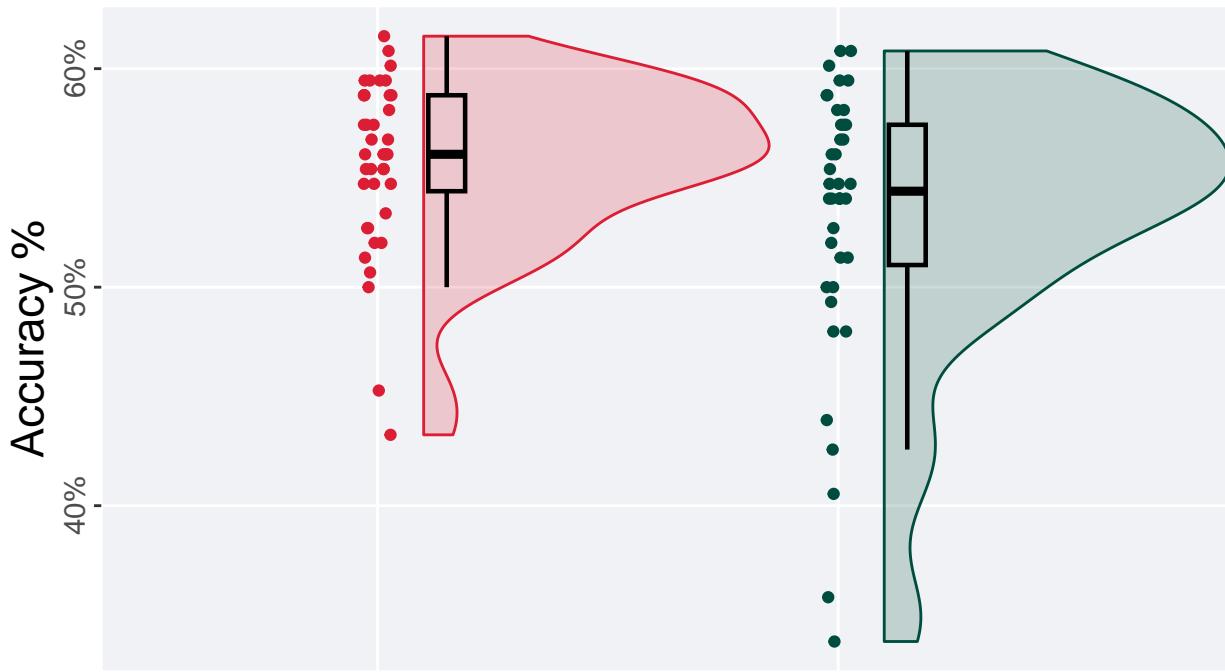


10.5 95%

10.5.1 Test accuracy

```
test_plot(filter(task_data, split == '95%'))
```

Accuracy on test set



Selection scheme tournament lexicase

Summary statistics for the testing performance of the selection schemes at the 5% selection set split:

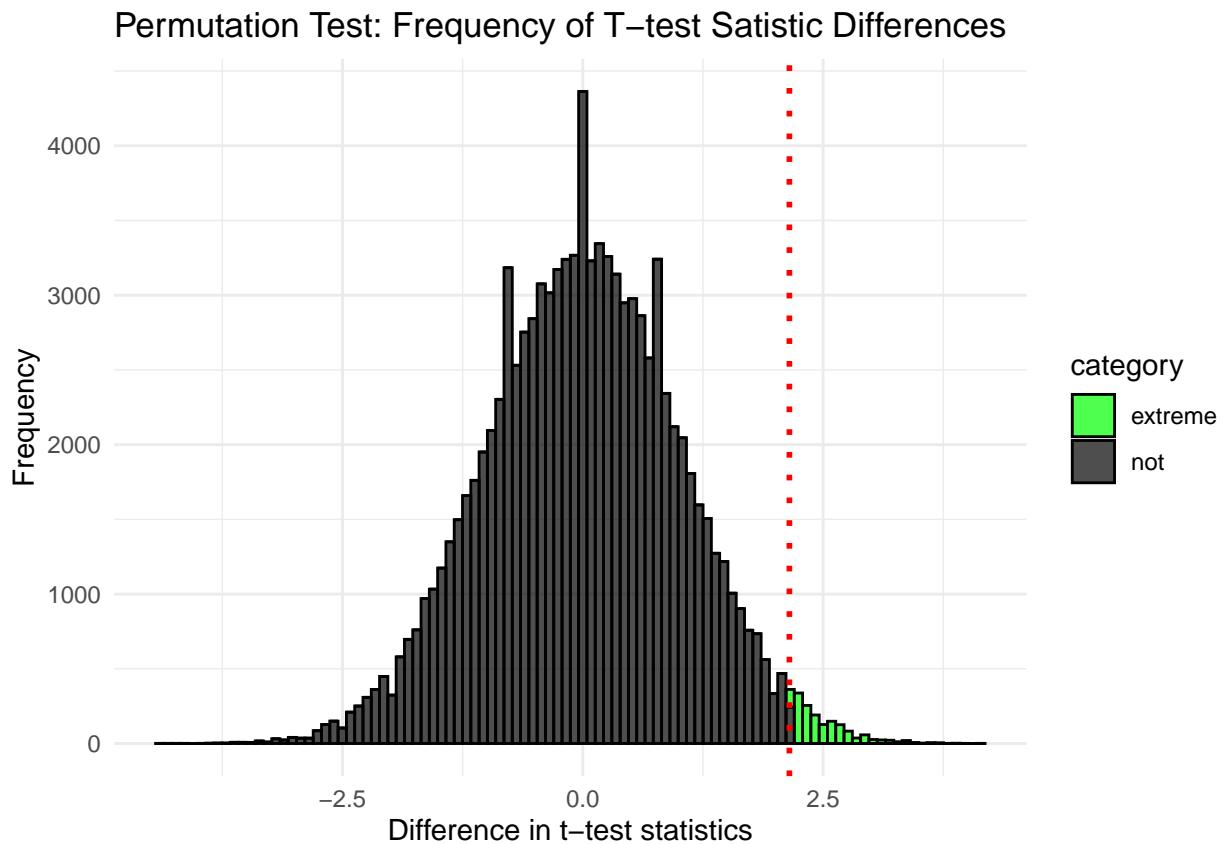
```
test_results_summary(filter(task_data, split == '95%'))
```

```
## # A tibble: 2 x 8
##   selection  count  na_cnt  min  median  mean  max    IQR
##   <fct>      <int>   <int> <dbl>  <dbl> <dbl> <dbl> <dbl>
## 1 tournament     40      0 0.432  0.561 0.557 0.615 0.0439
## 2 lexicase       40      0 0.338  0.544 0.531 0.608 0.0642
```

The permutation test revealed that the results are:

```
tournament_results <- filter(task_data, split == '95%' & selection == 'tournament')
lexicase_results <- filter(task_data, split == '95%' & selection == 'lexicase')
permutation_test(tournament_results$testing_performance,
                 lexicase_results$testing_performance,
                 seed = 89,
                 alternative = "g")
```

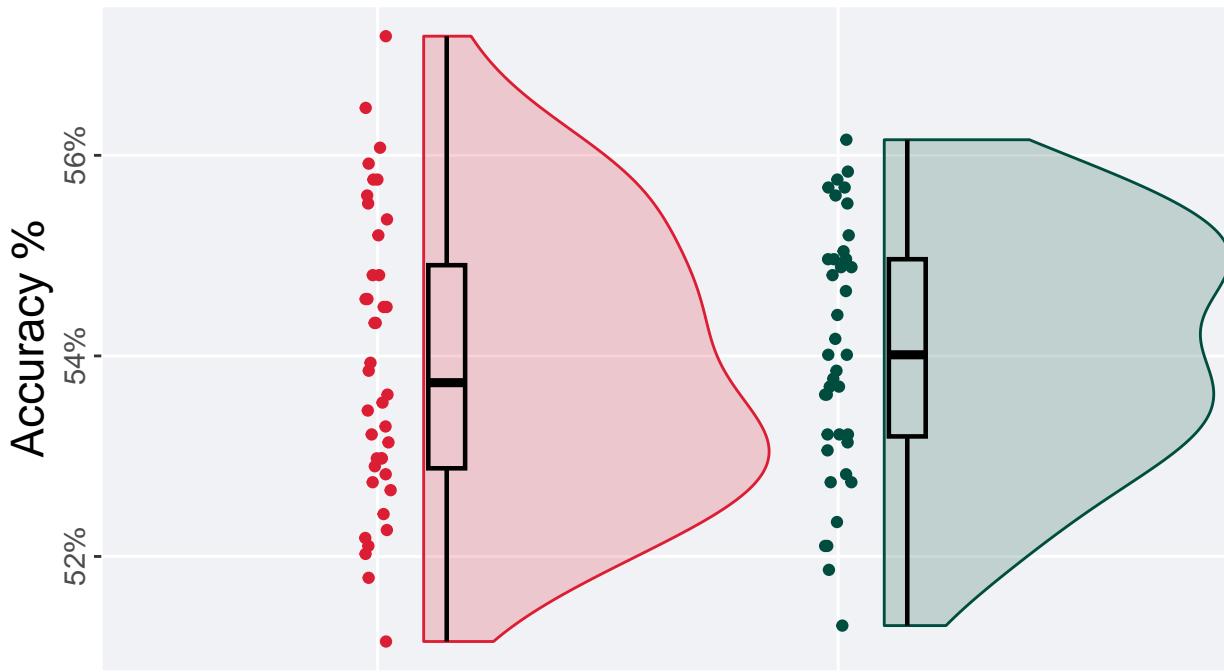
```
## [1] "observed_diff: 2.14920138405129"
## [1] "permutation_diffs[0.95 * n_permutations]: 1.64591413063659"
## [1] "reject null hypothesis"
## [1] "p-value: 0.01608"
```



10.5.2 Validation accuracy

```
validation_plot(filter(task_data, split == '95%'))
```

Accuracy on validation set



Selection Scheme  tournament  lexicase

Summary statistics for the testing performance of the selection schemes at the 5% selection set split:

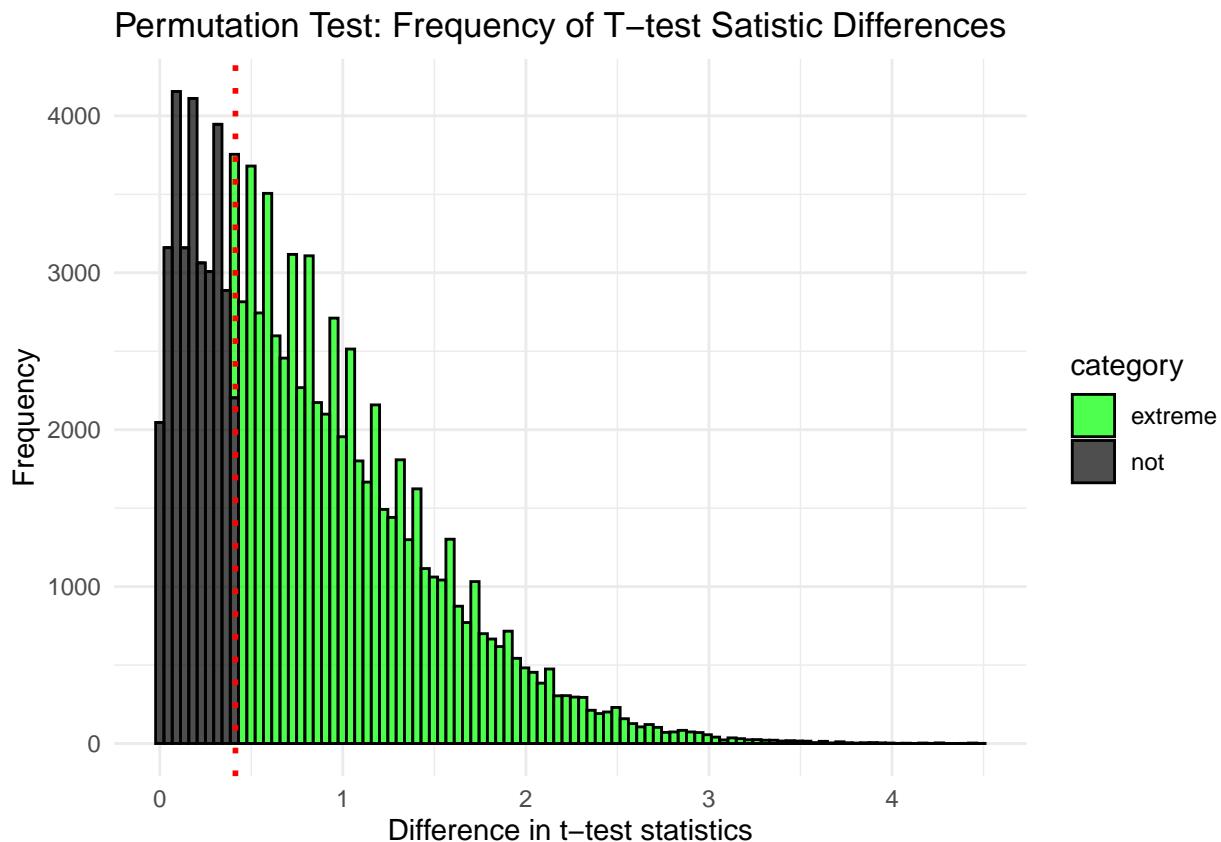
```
validation_accuracy_summary(filter(task_data, split == '95%'))
```

```
## # A tibble: 2 x 8
##   selection count na_cnt  min median  mean   max     IQR
##   <fct>     <int> <int> <dbl> <dbl> <dbl> <dbl> <dbl>
## 1 tournament     40     0 0.512 0.537 0.539 0.572 0.0203
## 2 lexicase       40     0 0.513 0.540 0.541 0.562 0.0177
```

The permutation test revealed that the results are:

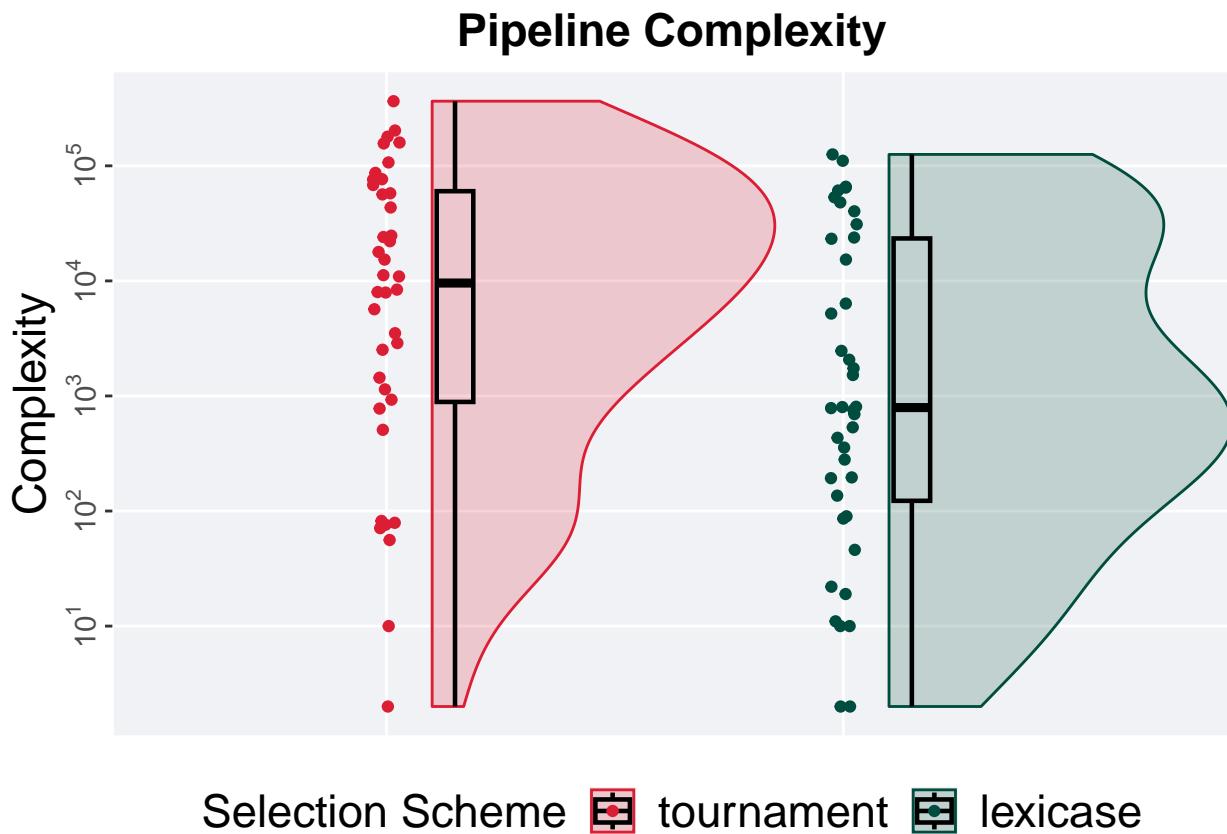
```
tournament_results <- filter(task_data, split == '95%' & selection == 'tournament')
lexicase_results <- filter(task_data, split == '95%' & selection == 'lexicase')
permutation_test(tournament_results$training_performance,
                 lexicase_results$training_performance,
                 seed = 90,
                 alternative = "t")
```

```
## [1] "observed_diff: -0.413305888924775"
## [1] "lower: -1.9953719420889"
## [1] "upper: 1.99537354233663"
## [1] "fail to reject null hypothesis"
## [1] "p-value: 0.68262"
```



10.5.3 Pipeline complexity

```
complexity_plot(filter(task_data, split == '95%'))
```



Summary statistics for the pipeline complexity of the selection schemes at the 5% selection set split:

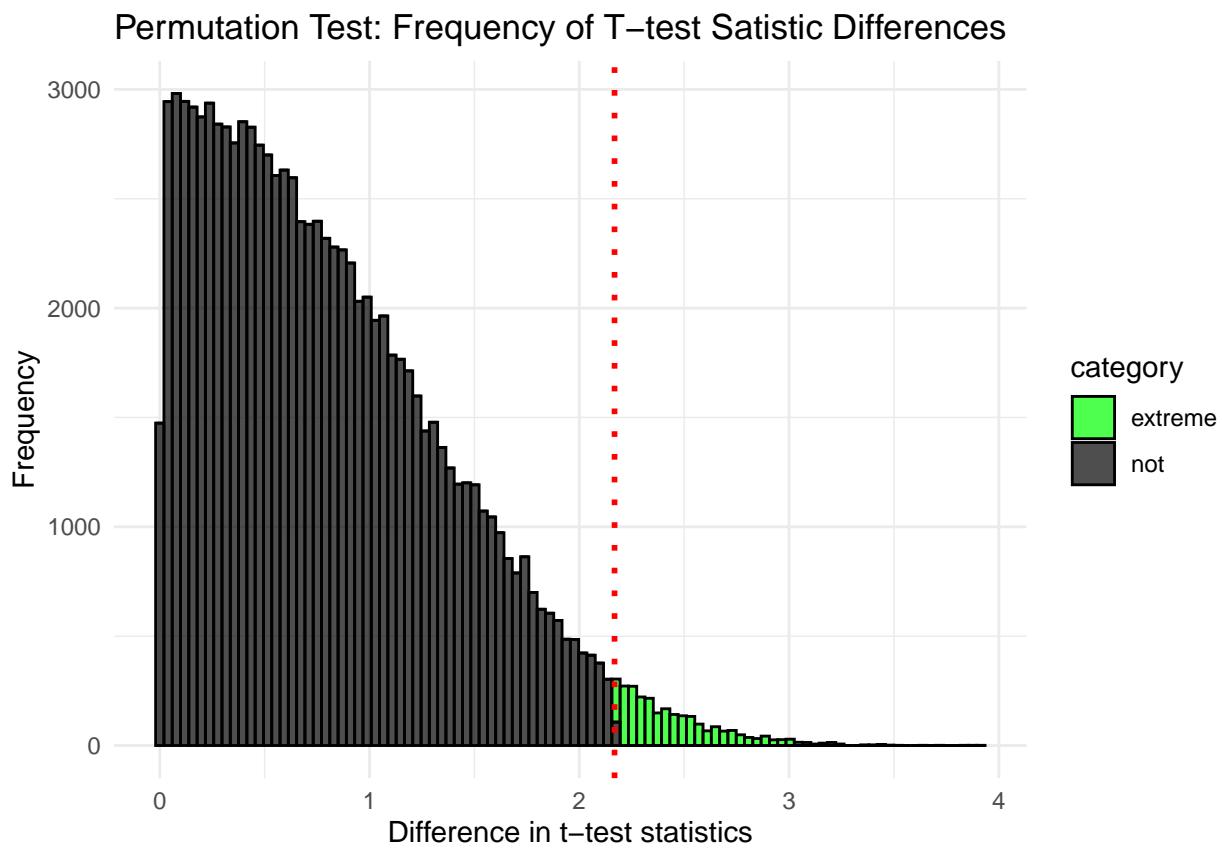
```
complexity_summary(filter(task_data, split == '95%'))
```

```
## # A tibble: 2 x 8
##   selection count na_cnt  min median   mean   max     IQR
##   <fct>     <int> <int> <dbl> <dbl> <dbl> <int> <dbl>
## 1 tournament     40     0     2  9664  45133. 364571 59463.
## 2 lexicase       40     0     2   792. 17215. 125661 23259.
```

The permutation test revealed that the results are:

```
tournament_results <- filter(task_data, split == '95%' & selection == 'tournament')
lexicase_results <- filter(task_data, split == '95%' & selection == 'lexicase')
permutation_test(tournament_results$testing_complexity,
                 lexicase_results$testing_complexity,
                 seed = 232,
                 alternative = "t")

## [1] "observed_diff: 2.16806162145202"
## [1] "lower: -1.94522382648133"
## [1] "upper: 1.92669292636808"
## [1] "reject null hypothesis"
## [1] "p-value: 0.02622"
```



Chapter 11

Task 2073

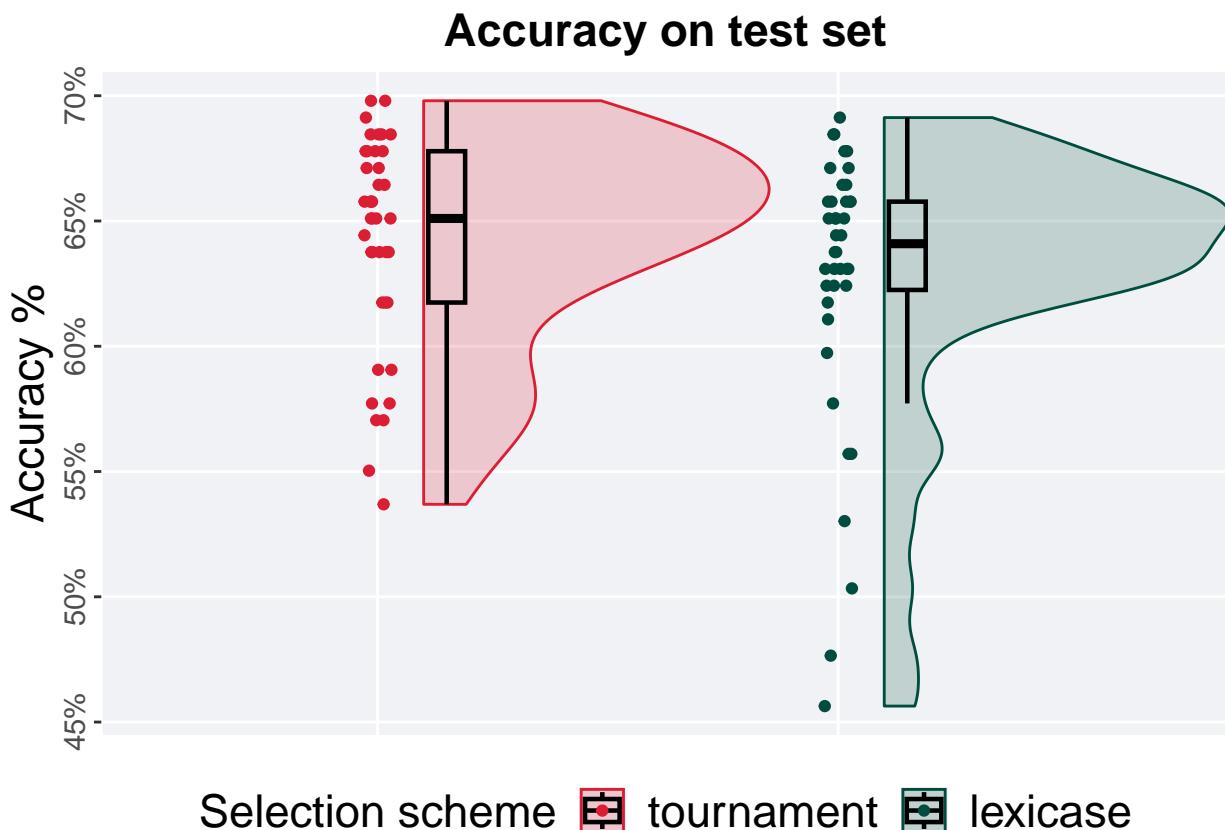
We present the results of our analysis of task 2073 with the different selection set splits used in our study.

```
task_data <- filter(results, task_id == 2073)
```

11.1 5%

11.1.1 Test accuracy

```
test_plot(filter(task_data, split == '5%'))
```



Summary statistics for the testing performance of the selection schemes at the 5% selection set split:

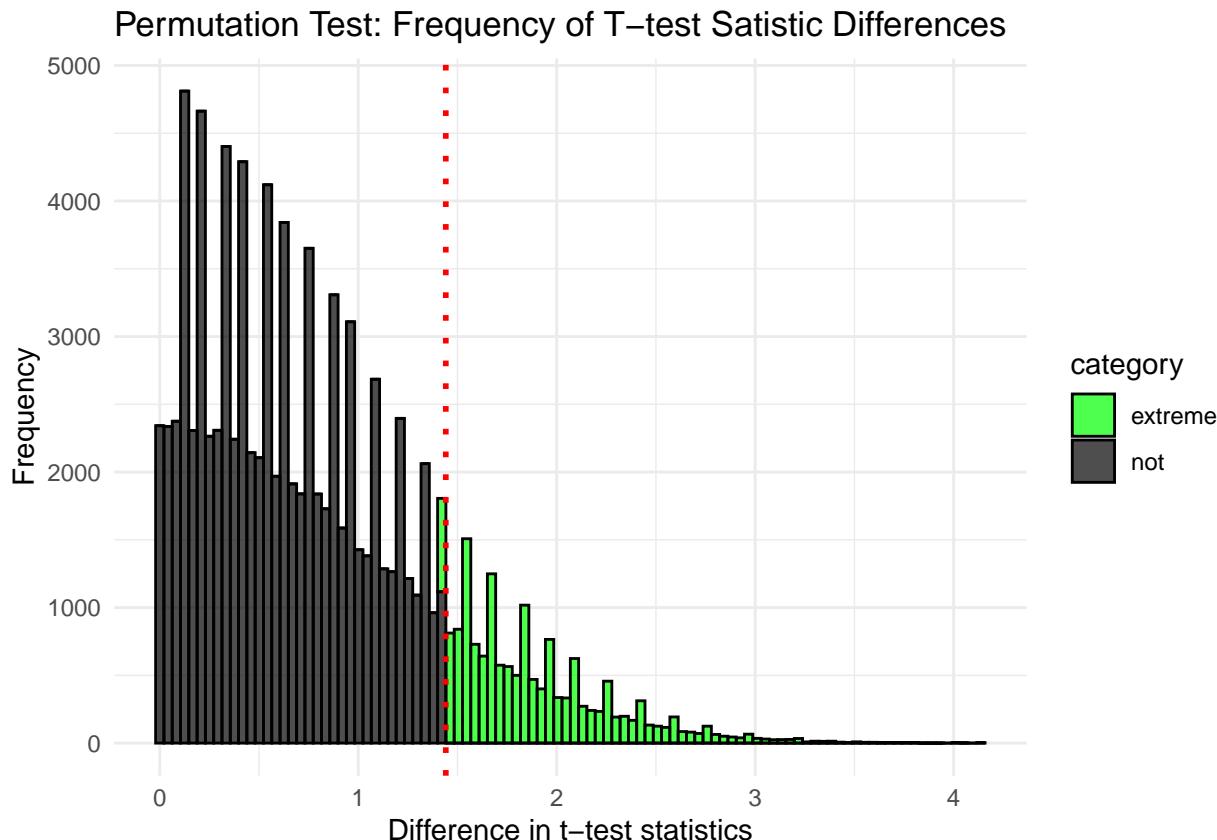
```
test_results_summary(filter(task_data, split == '5%'))
```

```
## # A tibble: 2 x 8
##   selection count na_cnt  min median  mean   max    IQR
##   <fct>     <int> <int> <dbl> <dbl> <dbl> <dbl> <dbl>
## 1 tournament     40     0 0.537  0.651  0.642  0.698  0.0604
## 2 lexicase       40     0 0.456  0.641  0.626  0.691  0.0352
```

The permutation test revealed that the results are:

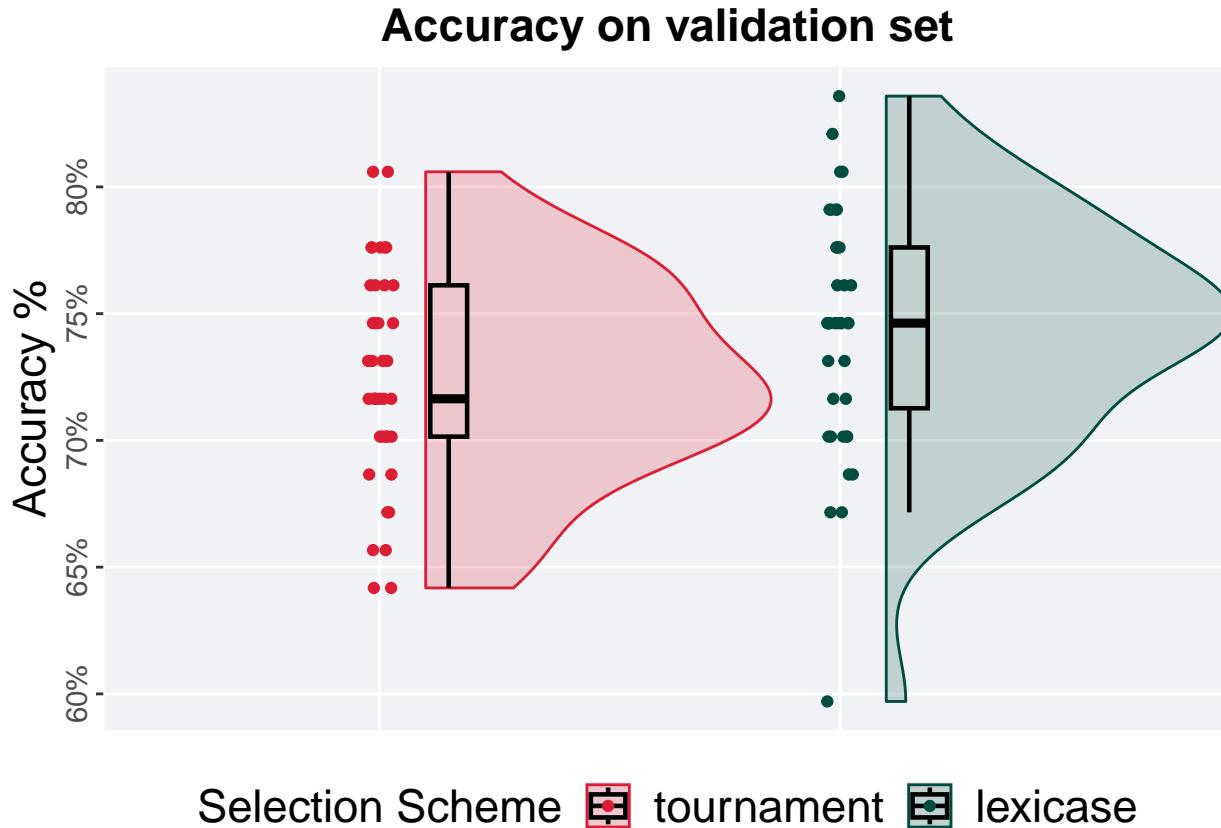
```
tournament_results <- filter(task_data, split == '5%' & selection == 'tournament')
lexicase_results <- filter(task_data, split == '5%' & selection == 'lexicase')
permutation_test(tournament_results$testing_performance,
                  lexicase_results$testing_performance,
                  seed = 91,
                  alternative = "t")
```

```
## [1] "observed_diff: 1.44107841467357"
## [1] "lower: -1.97919349953109"
## [1] "upper: 1.97919397970146"
## [1] "fail to reject null hypothesis"
## [1] "p-value: 0.15617"
```



11.1.2 Validation accuracy

```
validation_plot(filter(task_data, split == '5%'))
```



Summary statistics for the testing performance of the selection schemes at the 5% selection set split:

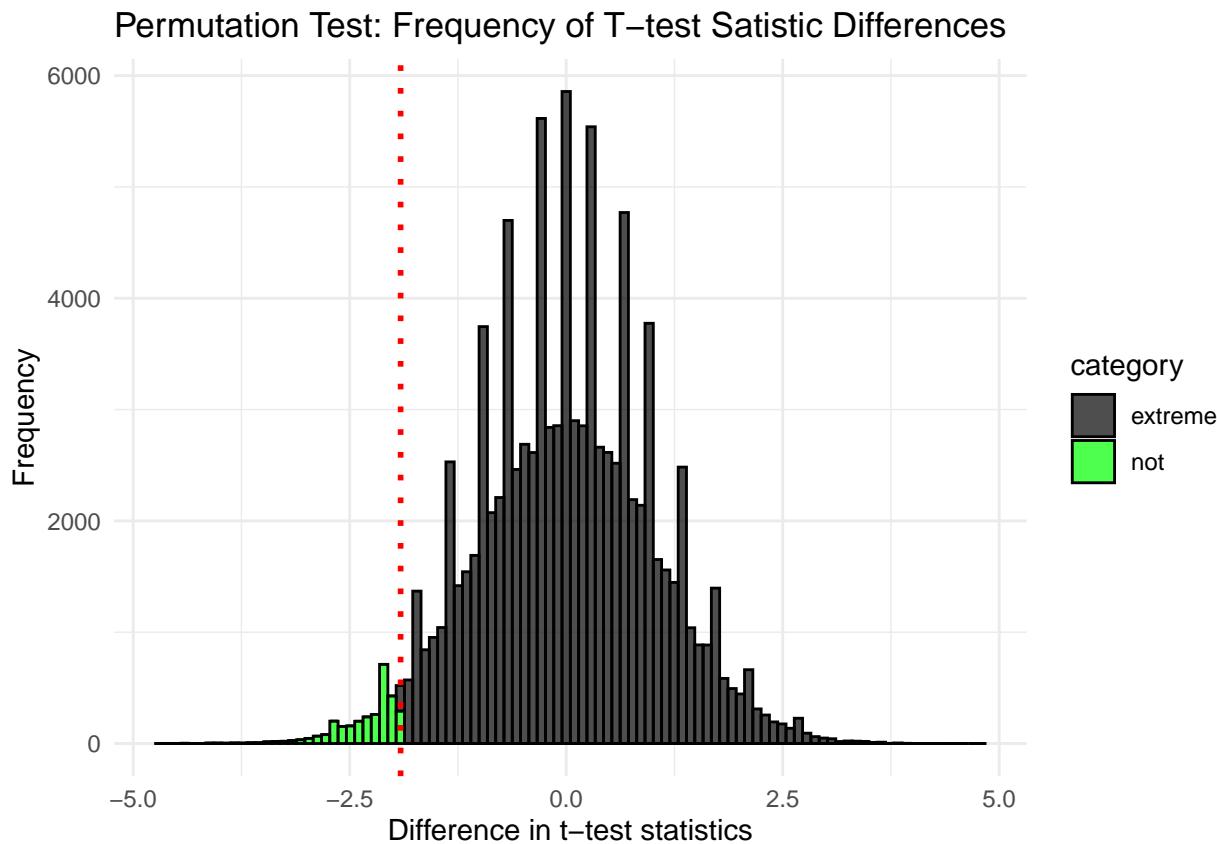
```
validation_accuracy_summary(filter(task_data, split == '5%'))
```

```
## # A tibble: 2 x 8
##   selection count na_cnt  min median  mean   max     IQR
##   <fct>     <int> <int> <dbl> <dbl> <dbl> <dbl> <dbl>
## 1 tournament     40     0 0.642  0.716  0.725  0.806  0.0597
## 2 lexicase       40     0 0.597  0.746  0.744  0.836  0.0634
```

The permutation test revealed that the results are:

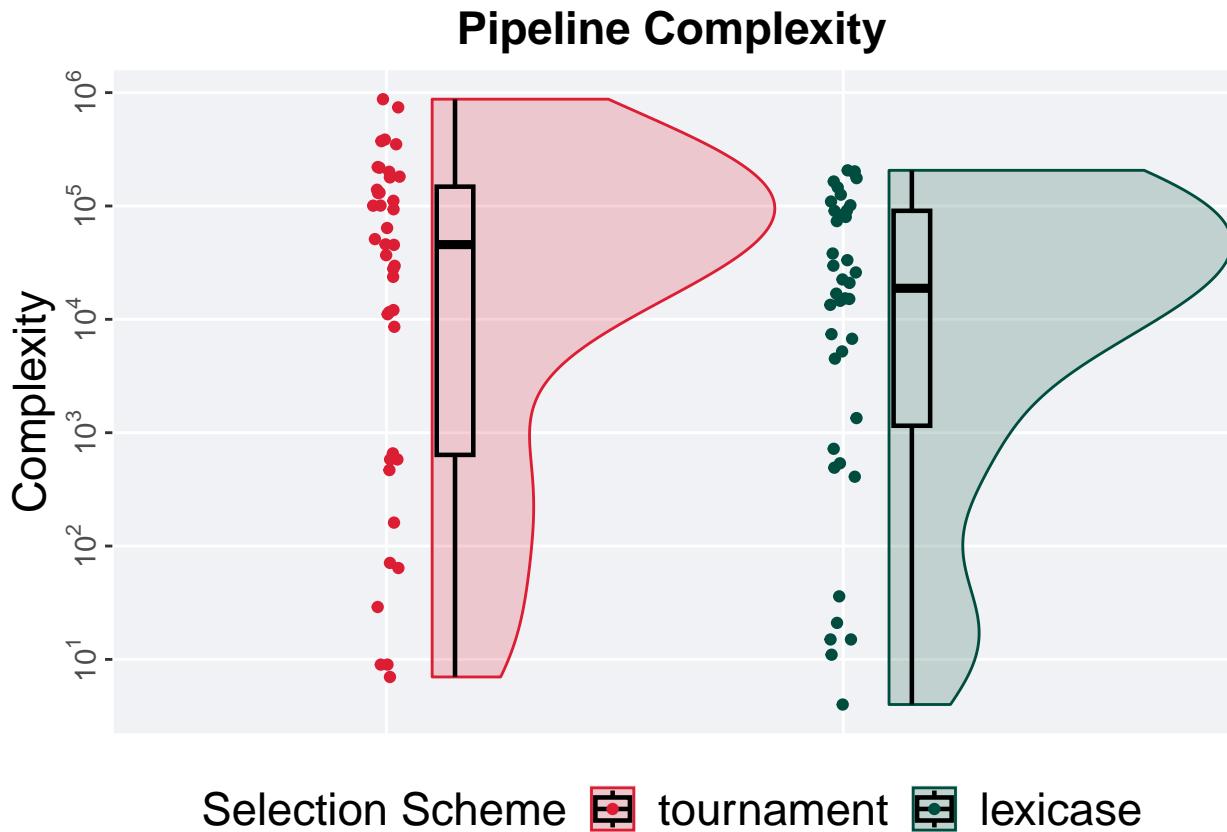
```
tournament_results <- filter(task_data, split == '5%' & selection == 'tournament')
lexicase_results <- filter(task_data, split == '5%' & selection == 'lexicase')
permutation_test(tournament_results$training_performance,
                 lexicase_results$training_performance,
                 seed = 92,
                 alternative = "1")
```

```
## [1] "observed_diff: -1.9120288209559"
## [1] "permutation_diffs[0.05 * n_permutations]: -1.67839605493495"
## [1] "reject null hypothesis"
## [1] "p-value: 0.02986"
```



11.1.3 Pipeline complexity

```
complexity_plot(filter(task_data, split == '5%'))
```



Summary statistics for the pipeline complexity of the selection schemes at the 5% selection set split:

```
complexity_summary(filter(task_data, split == '5%'))
```

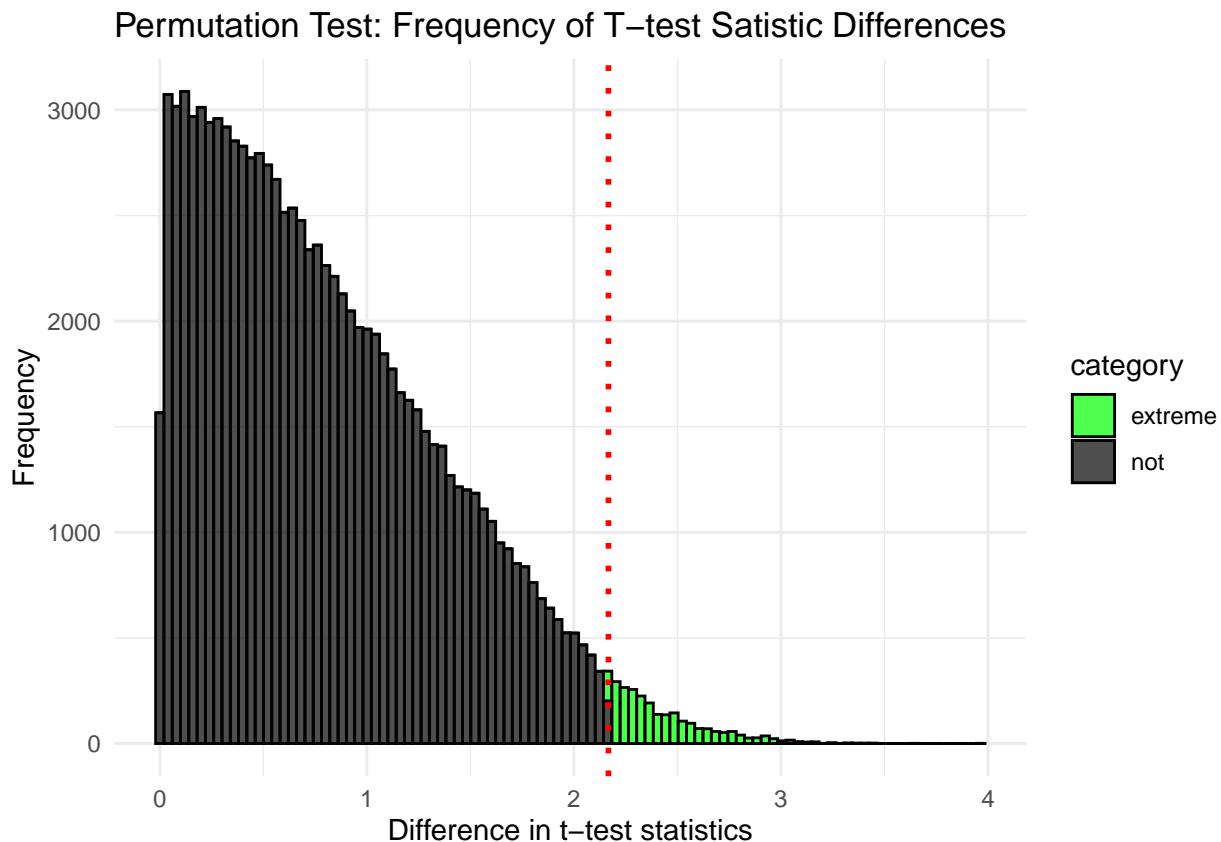
```
## # A tibble: 2 x 8
##   selection count na_cnt  min median    mean    max    IQR
##   <fct>     <int> <int> <dbl> <dbl> <dbl> <dbl>
## 1 tournament     40     0     7 45636. 122524. 875161 148323
## 2 lexicase       40     0     4 18905   53014. 206271 89465
```

The permutation test revealed that the results are:

```
tournament_results <- filter(task_data, split == '5%' & selection == 'tournament')
lexicase_results <- filter(task_data, split == '5%' & selection == 'lexicase')
```

```
permutation_test(tournament_results$testing_complexity,
                  lexicase_results$testing_complexity,
                  seed = 233,
                  alternative = "t")
```

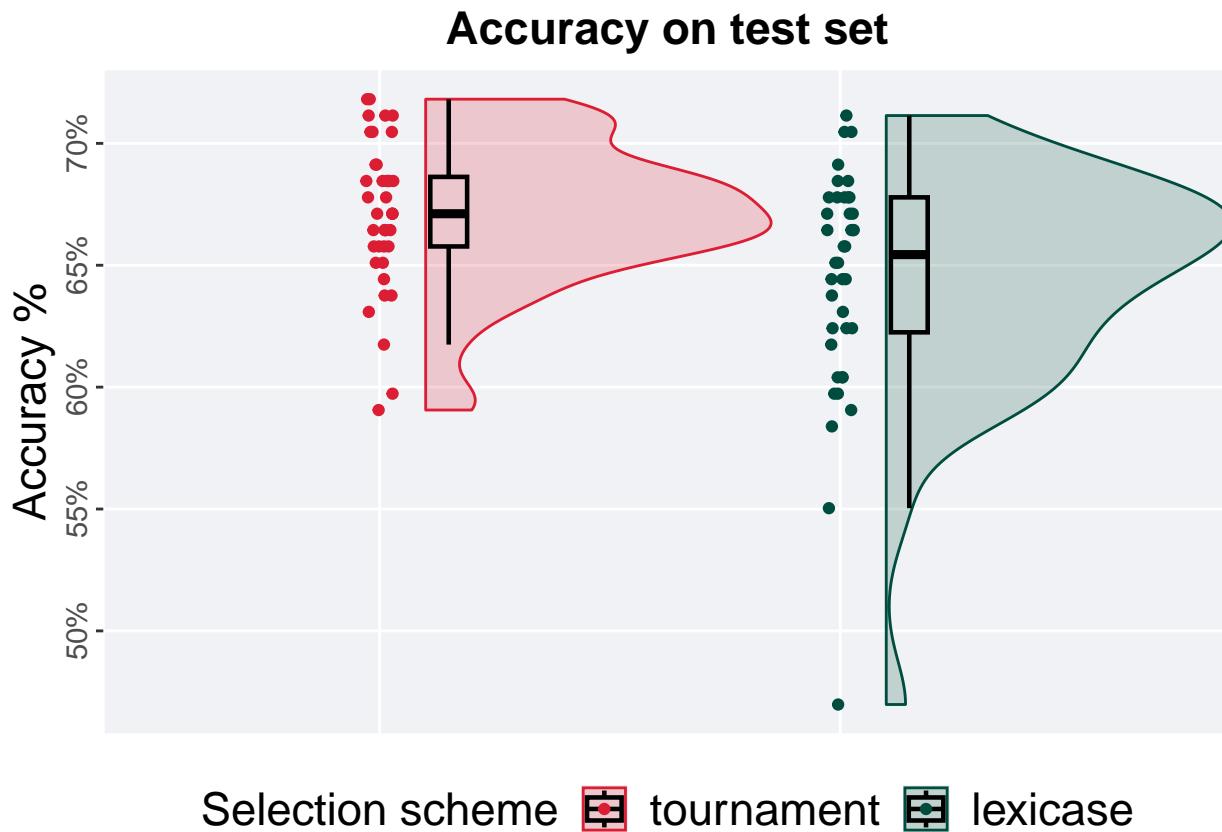
```
## [1] "observed_diff: 2.16683233457714"
## [1] "lower: -1.93601485009513"
## [1] "upper: 1.94751209893453"
## [1] "reject null hypothesis"
## [1] "p-value: 0.02523"
```



11.2 10%

11.2.1 Test accuracy

```
test_plot(filter(task_data, split == '10%'))
```



Summary statistics for the testing performance of the selection schemes at the 5% selection set split:

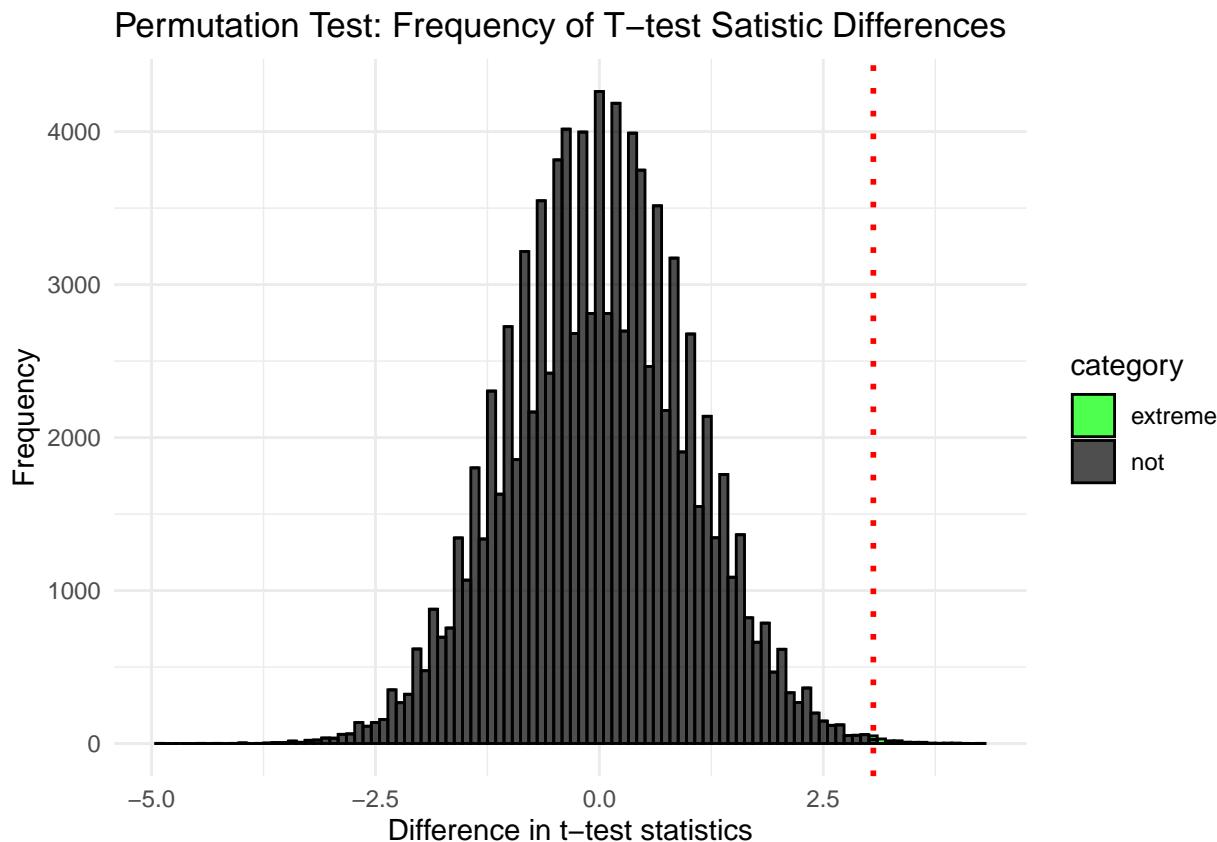
```
test_results_summary(filter(task_data, split == '10%'))
```

```
## # A tibble: 2 x 8
##   selection  count  na_cnt  min  median  mean  max  IQR
##   <fct>      <int>   <int> <dbl>  <dbl> <dbl> <dbl> <dbl>
## 1 tournament    40      0  0.591  0.671  0.671  0.718  0.0285
## 2 lexicase      40      0  0.470  0.654  0.644  0.711  0.0554
```

The permutation test revealed that the results are:

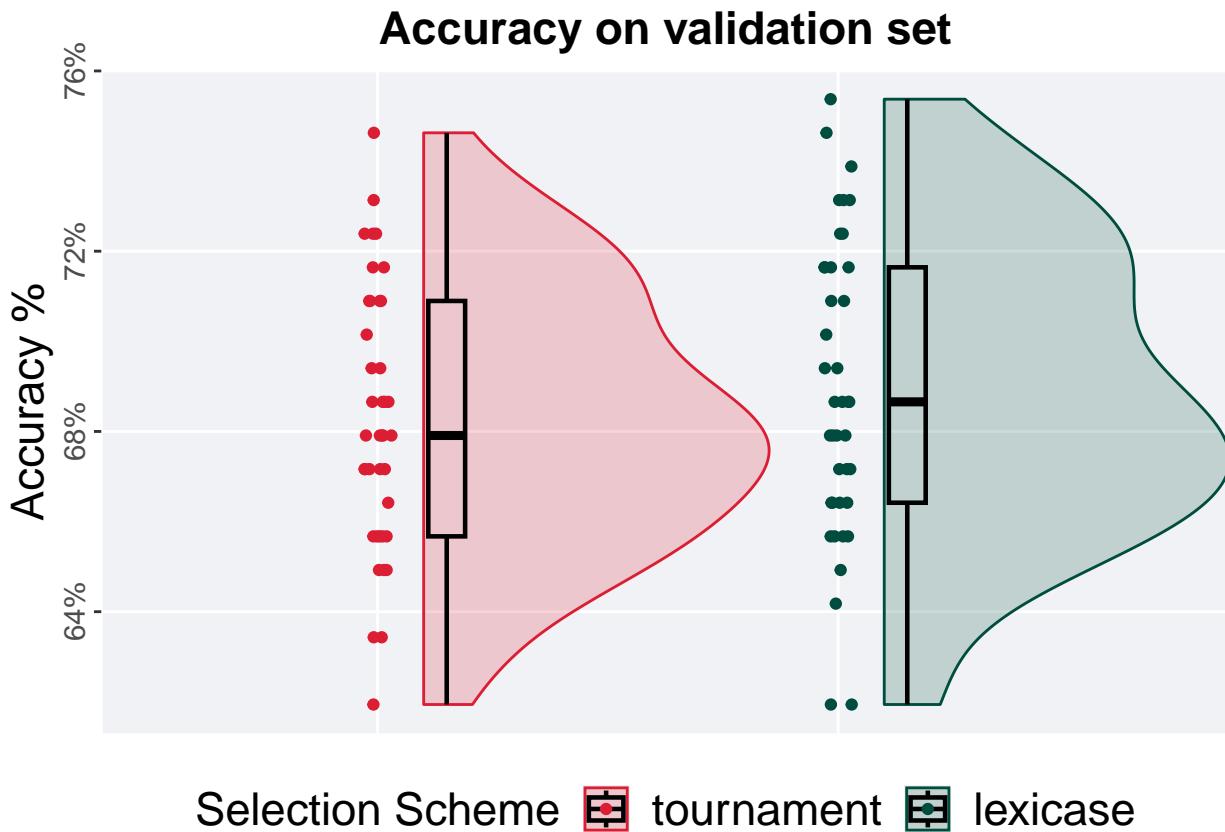
```
tournament_results <- filter(task_data, split == '10%' & selection == 'tournament')
lexicase_results <- filter(task_data, split == '10%' & selection == 'lexicase')
permutation_test(tournament_results$testing_performance,
                 lexicase_results$testing_performance,
                 seed = 93,
                 alternative = "g")
```

```
## [1] "observed_diff: 3.05890528357161"
## [1] "permutation_diffs[0.95 * n_permutations]: 1.6541469407571"
## [1] "reject null hypothesis"
## [1] "p-value: 0.00128"
```



11.2.2 Validation accuracy

```
validation_plot(filter(task_data, split == '10%'))
```



Summary statistics for the testing performance of the selection schemes at the 5% selection set split:

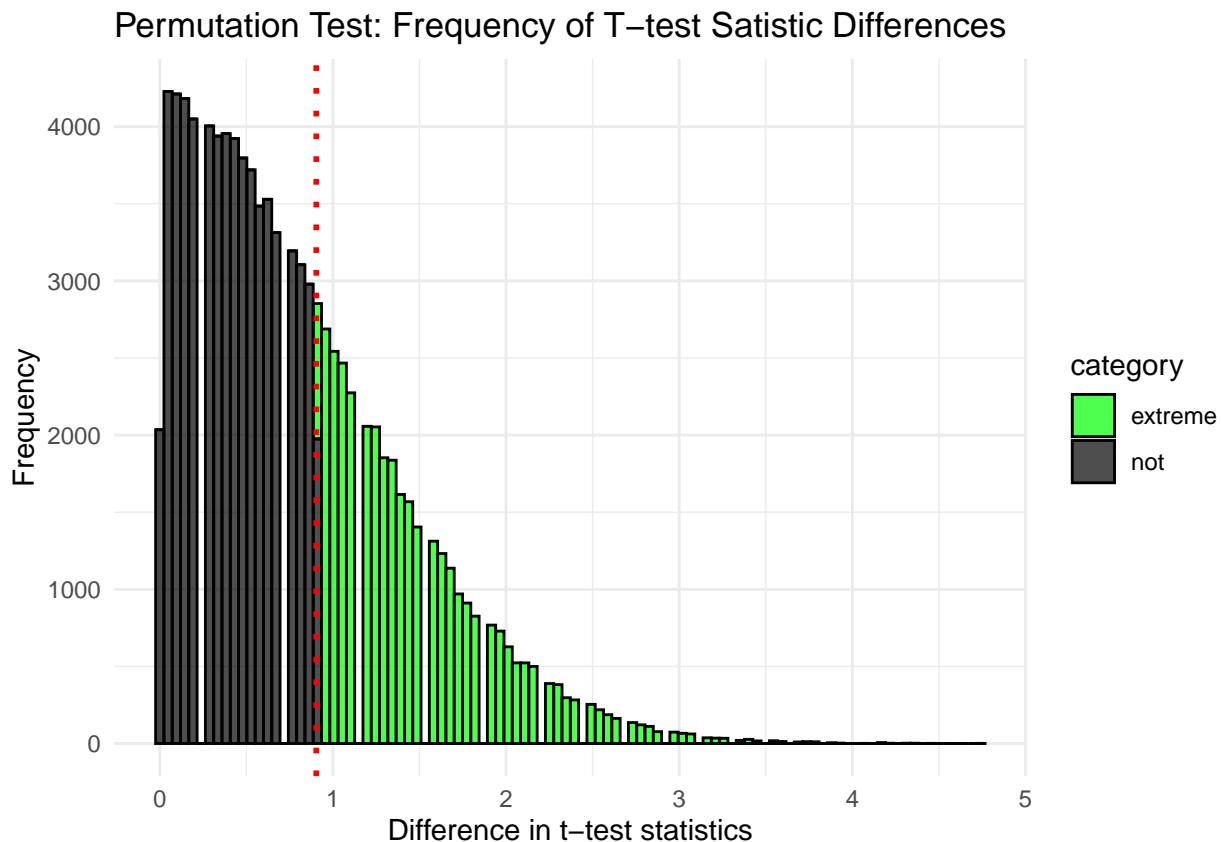
```
validation_accuracy_summary(filter(task_data, split == '10%'))
```

```
## # A tibble: 2 x 8
##   selection count na_cnt  min median  mean   max     IQR
##   <fct>     <int> <int> <dbl> <dbl> <dbl> <dbl> <dbl>
## 1 tournament     40     0 0.619 0.679 0.682 0.746 0.0522
## 2 lexicase       40     0 0.619 0.687 0.689 0.754 0.0522
```

The permutation test revealed that the results are:

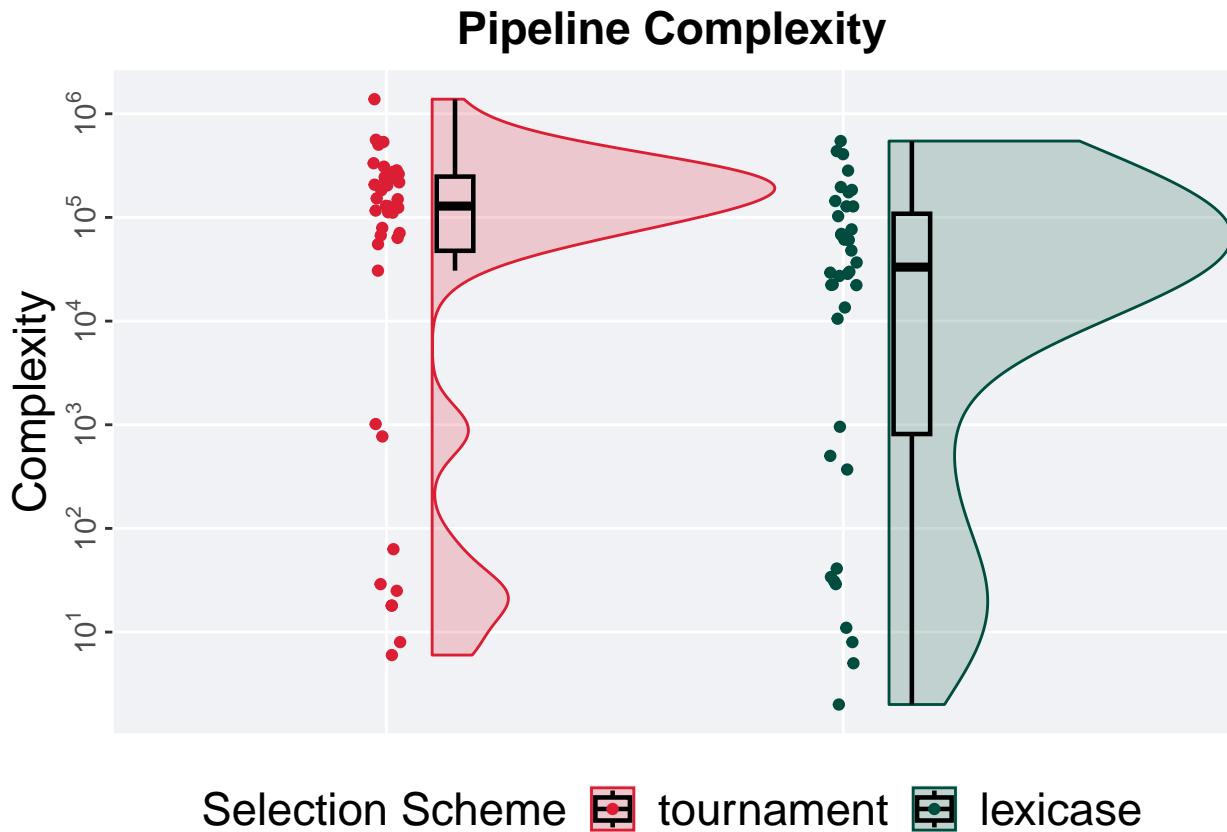
```
tournament_results <- filter(task_data, split == '10%' & selection == 'tournament')
lexicase_results <- filter(task_data, split == '10%' & selection == 'lexicase')
permutation_test(tournament_results$training_performance,
                 lexicase_results$training_performance,
                 seed = 94,
                 alternative = "t")
```

```
## [1] "observed_diff: -0.903818413638582"
## [1] "lower: -2.00679568647391"
## [1] "upper: 2.00679576290495"
## [1] "fail to reject null hypothesis"
## [1] "p-value: 0.36373"
```



11.2.3 Pipeline complexity

```
complexity_plot(filter(task_data, split == '10%'))
```



Summary statistics for the pipeline complexity of the selection schemes at the 5% selection set split:

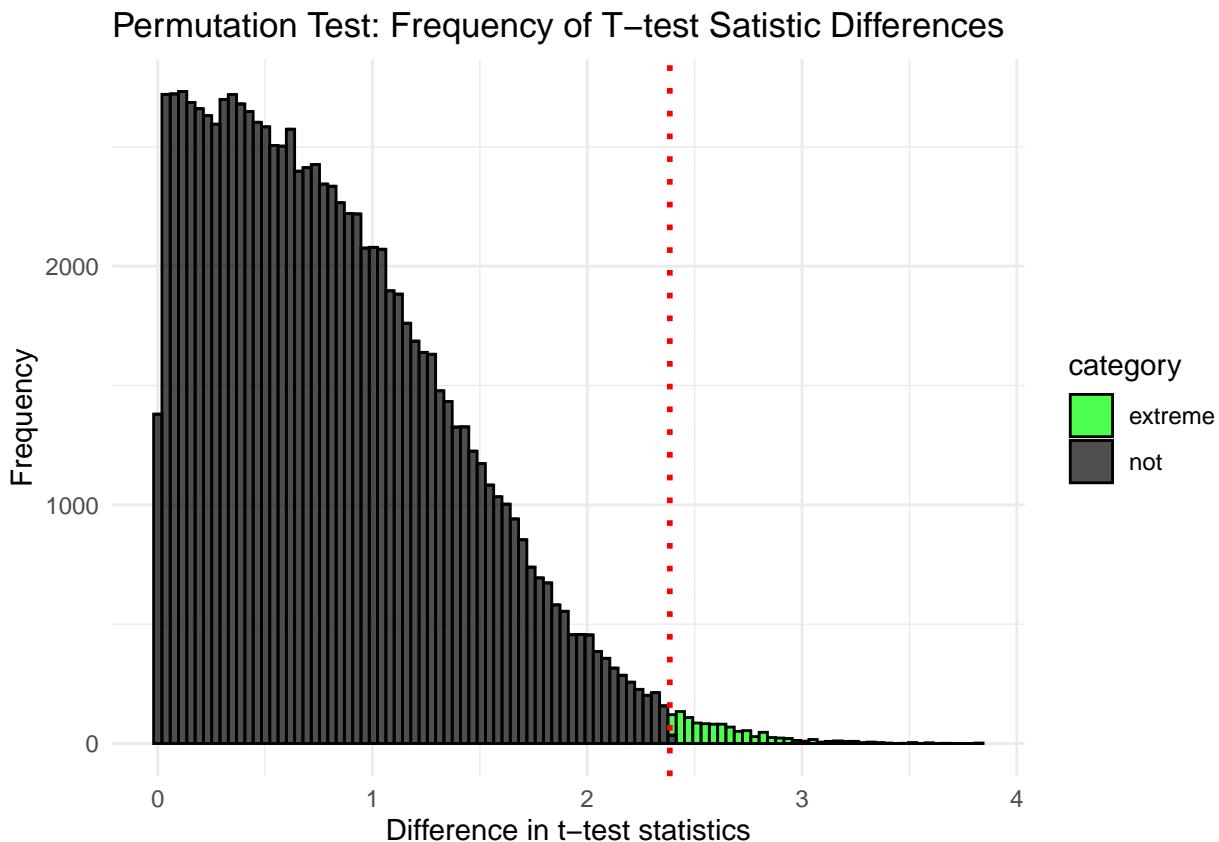
```
complexity_summary(filter(task_data, split == '10%'))
```

```
## # A tibble: 2 x 8
##   selection count na_cnt   min   median     mean     max     IQR
##   <fct>     <int> <int> <dbl>   <dbl>   <dbl>   <dbl>
## 1 tournament     40     0     6 128628. 190577. 1382462 198578
## 2 lexicase       40     0     2  33348.  87058.  545471 108184.
```

The permutation test revealed that the results are:

```
tournament_results <- filter(task_data, split == '10%' & selection == 'tournament')
lexicase_results <- filter(task_data, split == '10%' & selection == 'lexicase')
permutation_test(tournament_results$testing_complexity,
                 lexicase_results$testing_complexity,
                 seed = 234,
                 alternative = "t")
```

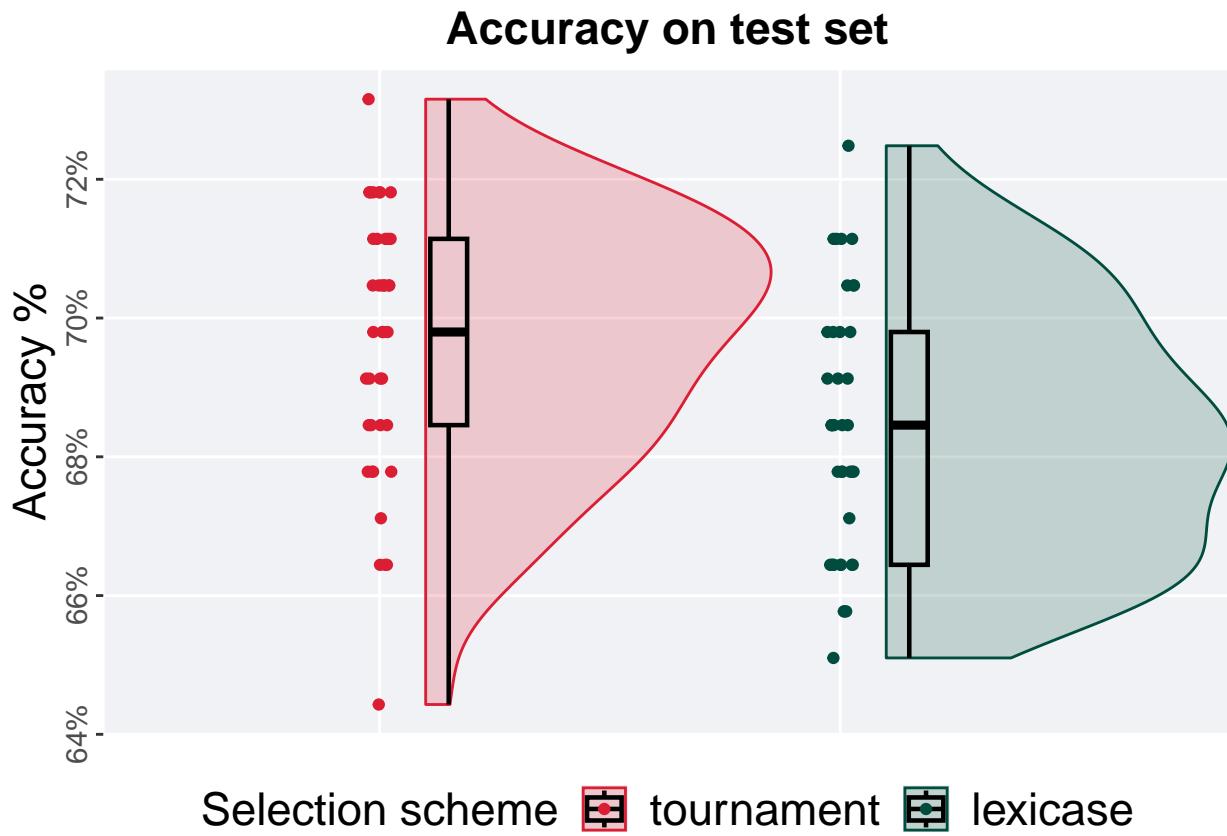
```
## [1] "observed_diff: 2.38443094230566"
## [1] "lower: -1.90173819000153"
## [1] "upper: 1.90756609550569"
## [1] "reject null hypothesis"
## [1] "p-value: 0.0109"
```



11.3 50%

11.3.1 Test accuracy

```
test_plot(filter(task_data, split == '50%'))
```



Summary statistics for the testing performance of the selection schemes at the 5% selection set split:

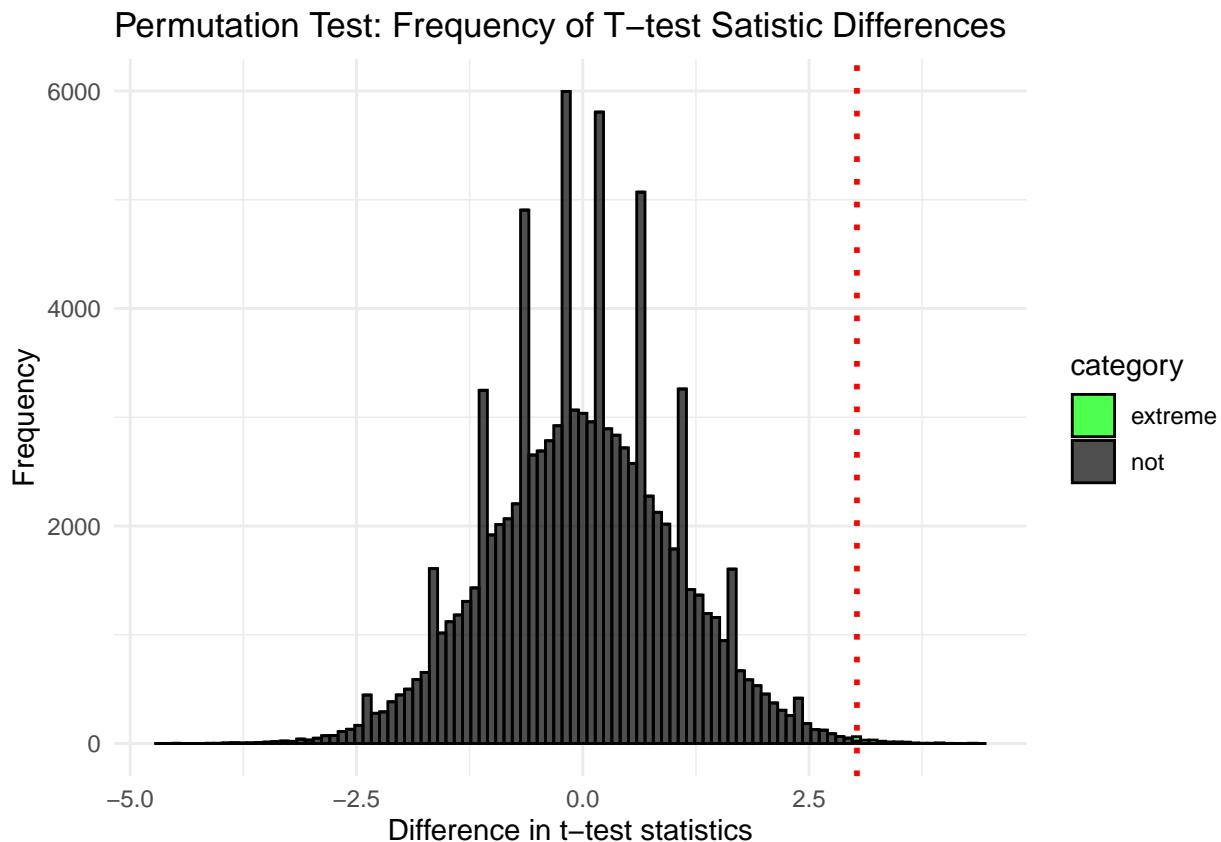
```
test_results_summary(filter(task_data, split == '50%'))
```

```
## # A tibble: 2 x 8
##   selection  count  na_cnt  min  median  mean  max    IQR
##   <fct>      <int>   <int> <dbl>   <dbl> <dbl> <dbl> <dbl>
## 1 tournament    40      0  0.644  0.698  0.696  0.732  0.0268
## 2 lexicase      40      0  0.651  0.685  0.683  0.725  0.0336
```

The permutation test revealed that the results are:

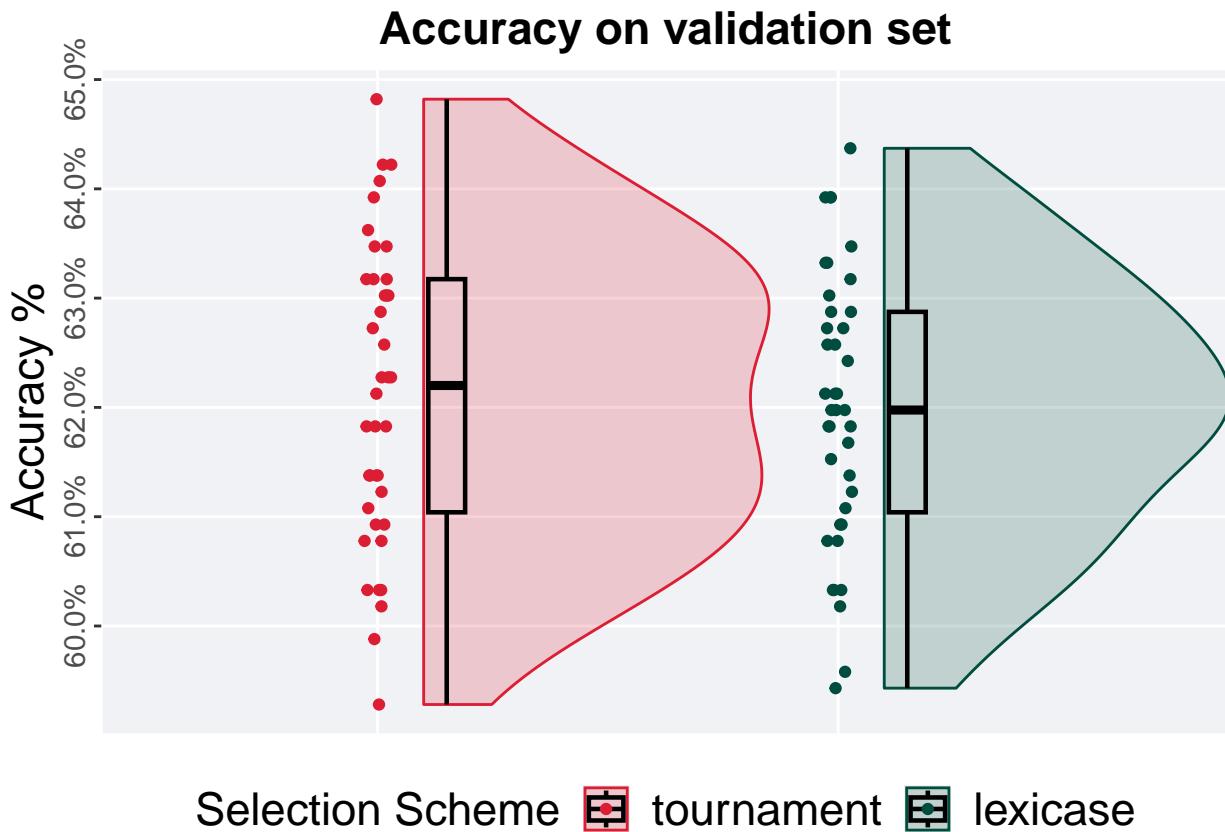
```
tournament_results <- filter(task_data, split == '50%' & selection == 'tournament')
lexicase_results <- filter(task_data, split == '50%' & selection == 'lexicase')
permutation_test(tournament_results$testing_performance,
                 lexicase_results$testing_performance,
                 seed = 95,
                 alternative = "g")
```

```
## [1] "observed_diff: 3.02914521008793"
## [1] "permutation_diffs[0.95 * n_permutations]: 1.68891527019981"
## [1] "reject null hypothesis"
## [1] "p-value: 0.0019"
```



11.3.2 Validation accuracy

```
validation_plot(filter(task_data, split == '50%'))
```



Summary statistics for the testing performance of the selection schemes at the 5% selection set split:

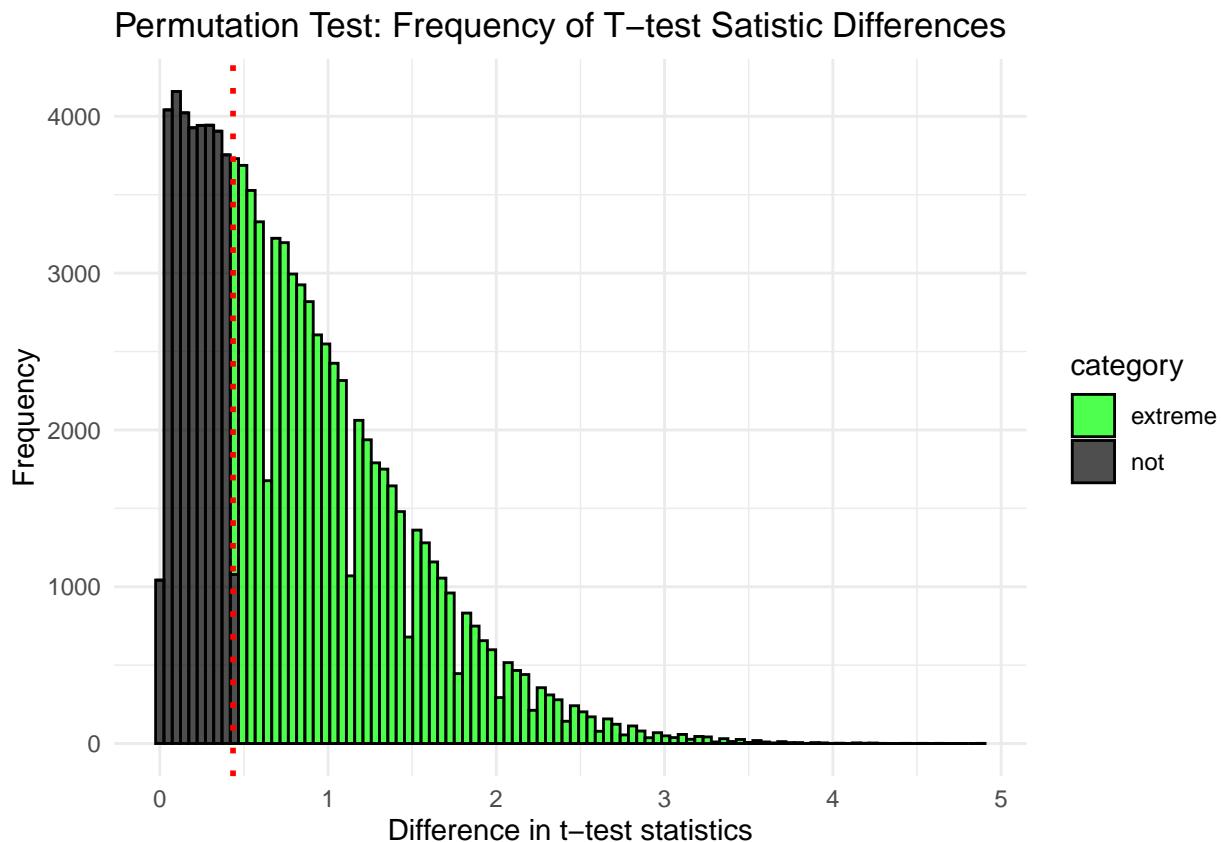
```
validation_accuracy_summary(filter(task_data, split == '50%'))
```

```
## # A tibble: 2 x 8
##   selection count na_cnt  min median  mean   max     IQR
##   <fct>     <int> <int> <dbl> <dbl> <dbl> <dbl> <dbl>
## 1 tournament     40     0 0.593  0.622  0.621  0.648  0.0213
## 2 lexicase       40     0 0.594  0.620  0.620  0.644  0.0183
```

The permutation test revealed that the results are:

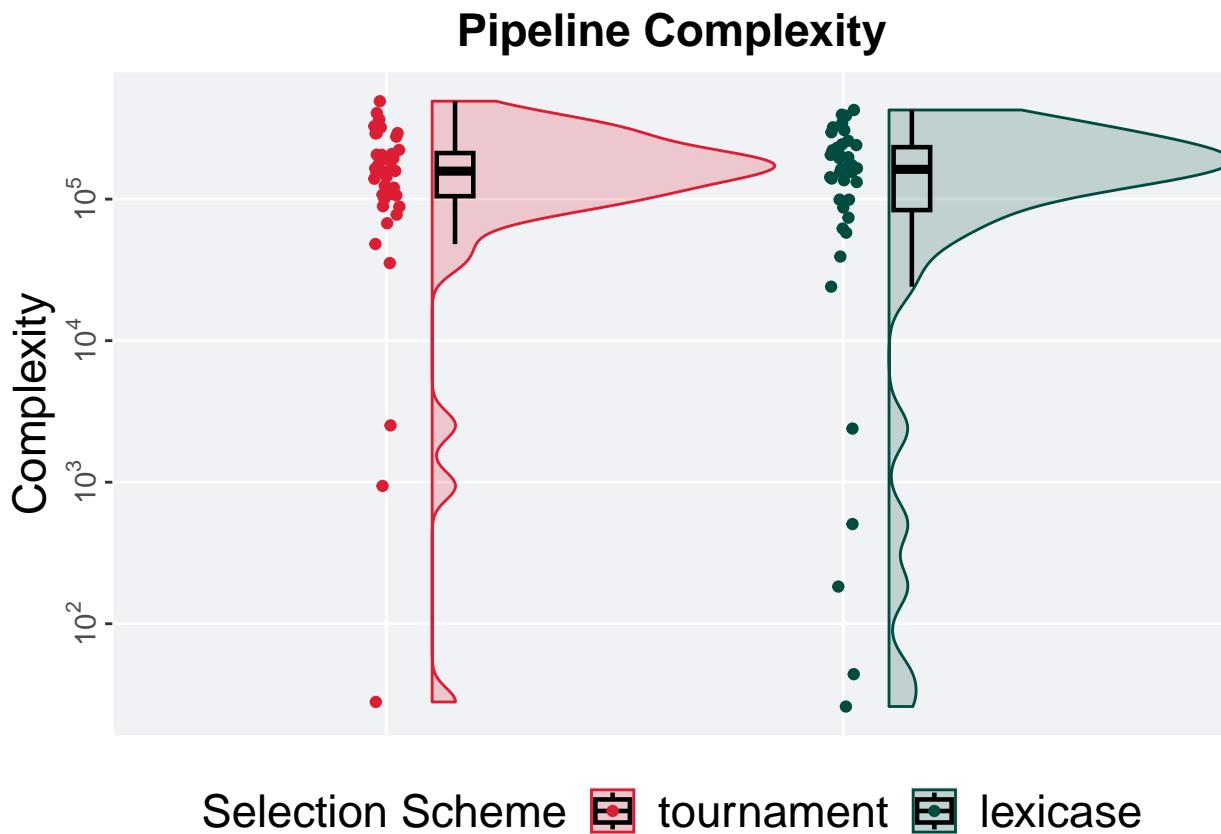
```
tournament_results <- filter(task_data, split == '50%' & selection == 'tournament')
lexicase_results <- filter(task_data, split == '50%' & selection == 'lexicase')
permutation_test(tournament_results$training_performance,
                 lexicase_results$training_performance,
                 seed = 96,
                 alternative = "t")
```

```
## [1] "observed_diff: 0.435323861862444"
## [1] "lower: -1.99265142509362"
## [1] "upper: 1.99265308163027"
## [1] "fail to reject null hypothesis"
## [1] "p-value: 0.66188"
```



11.3.3 Pipeline complexity

```
complexity_plot(filter(task_data, split == '50%'))
```



Summary statistics for the pipeline complexity of the selection schemes at the 5% selection set split:

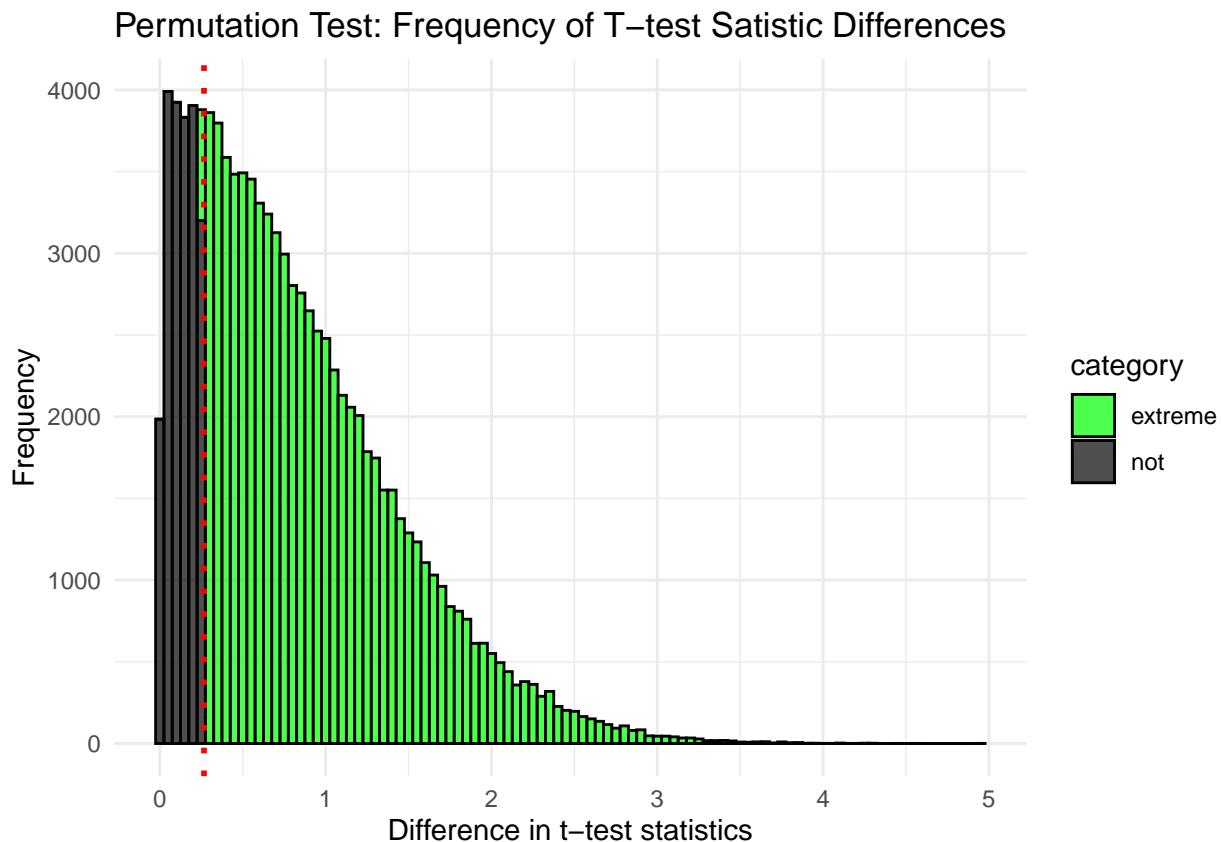
```
complexity_summary(filter(task_data, split == '50'))
```

```
## # A tibble: 2 x 8
##   selection count na_cnt   min   median     mean     max     IQR
##   <fct>     <int> <int> <dbl>   <dbl>   <dbl>   <dbl>
## 1 tournament     40     0  28 157736  174928. 492141 106853
## 2 lexicase       40     0  26 162372. 168173. 426472 149046.
```

The permutation test revealed that the results are:

```
tournament_results <- filter(task_data, split == '50' & selection == 'tournament')
lexicase_results <- filter(task_data, split == '50' & selection == 'lexicase')
permutation_test(tournament_results$testing_complexity,
                 lexicase_results$testing_complexity,
                 seed = 235,
                 alternative = "t")

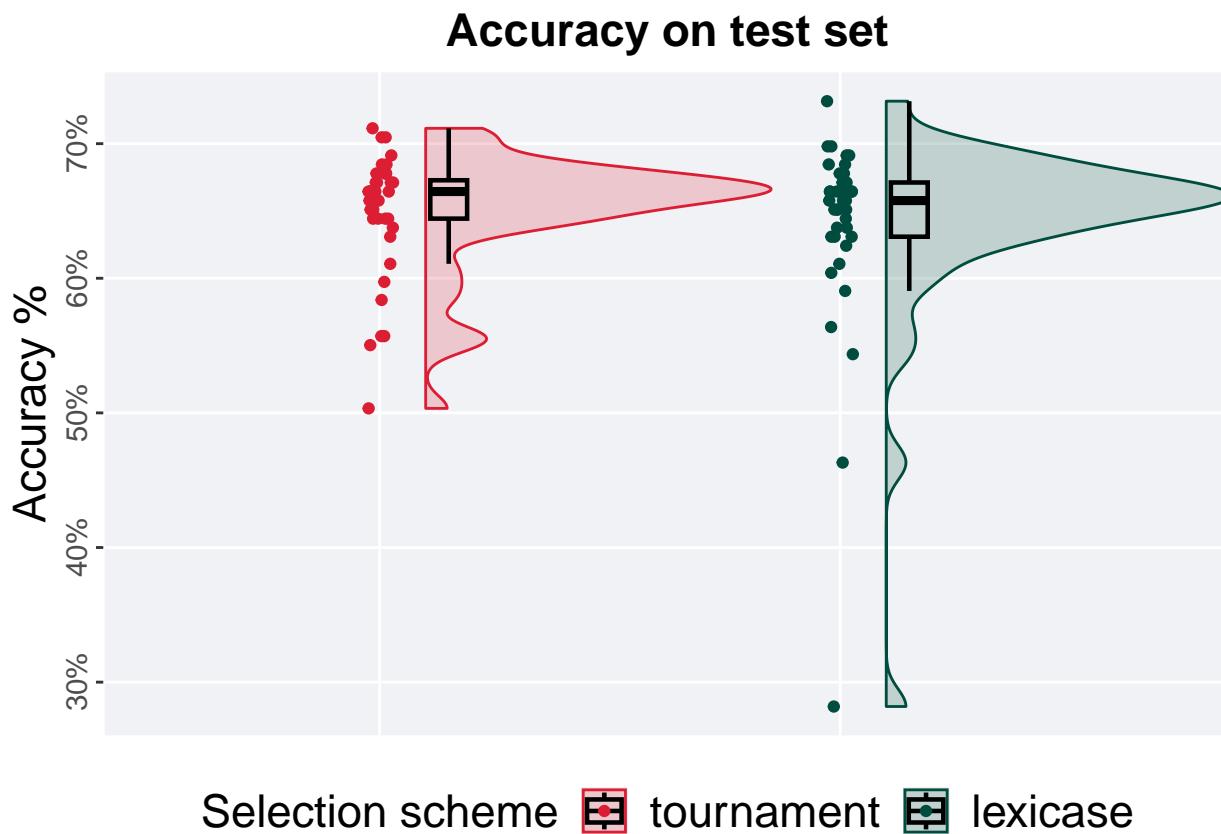
## [1] "observed_diff: 0.266610276707756"
## [1] "lower: -1.97872951337238"
## [1] "upper: 2.00460877927521"
## [1] "fail to reject null hypothesis"
## [1] "p-value: 0.79164"
```



11.4 90%

11.4.1 Test accuracy

```
test_plot(filter(task_data, split == '90%'))
```



Summary statistics for the testing performance of the selection schemes at the 5% selection set split:

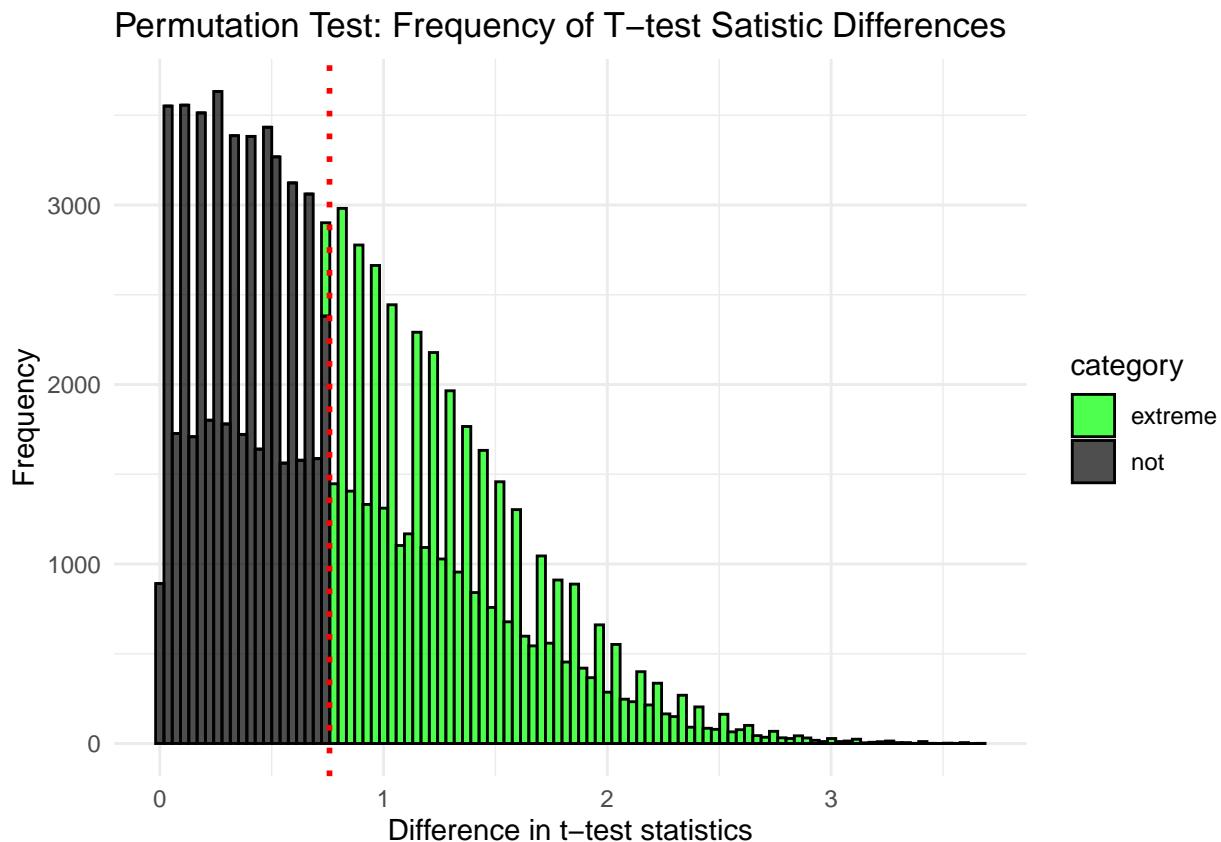
```
test_results_summary(filter(task_data, split == '90%'))
```

```
## # A tibble: 2 x 8
##   selection count na_cnt  min median  mean  max    IQR
##   <fct>     <int> <int> <dbl> <dbl> <dbl> <dbl> <dbl>
## 1 tournament     40     0 0.503 0.664 0.649 0.711 0.0285
## 2 lexicase       40     0 0.282 0.658 0.639 0.732 0.0403
```

The permutation test revealed that the results are:

```
tournament_results <- filter(task_data, split == '90%' & selection == 'tournament')
lexicase_results <- filter(task_data, split == '90%' & selection == 'lexicase')
permutation_test(tournament_results$testing_performance,
                 lexicase_results$testing_performance,
                 seed = 97,
                 alternative = "t")
```

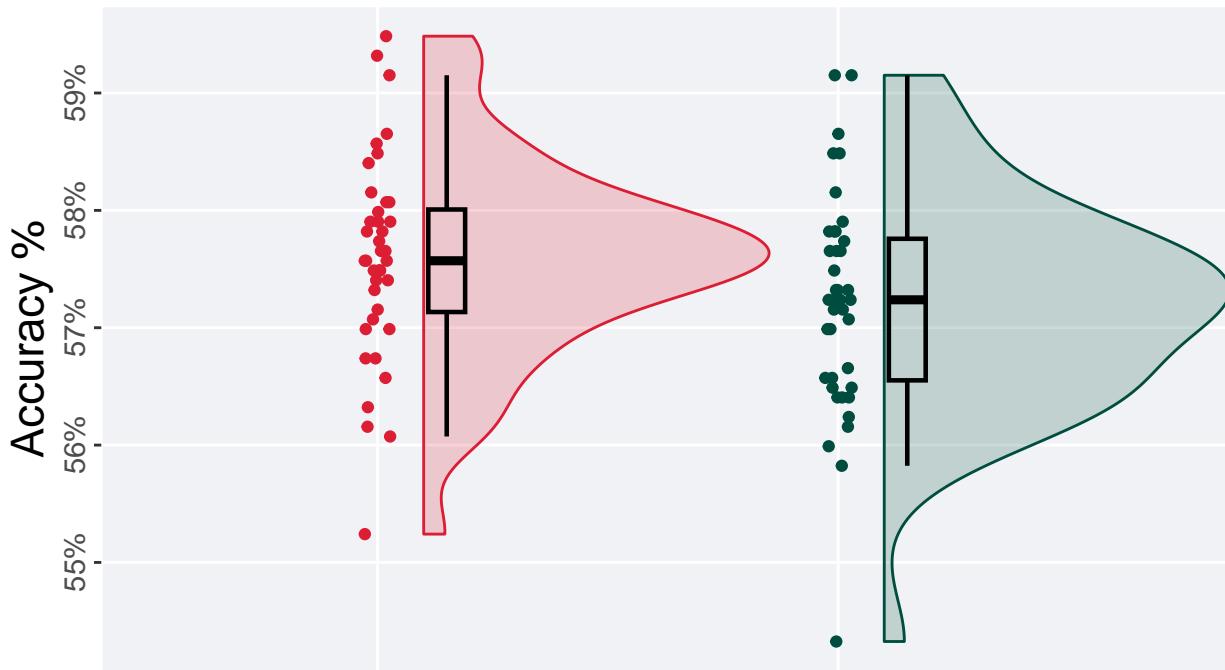
```
## [1] "observed_diff: 0.758531310895758"
## [1] "lower: -1.92107872814561"
## [1] "upper: 1.92107872814561"
## [1] "fail to reject null hypothesis"
## [1] "p-value: 0.47717"
```



11.4.2 Validation accuracy

```
validation_plot(filter(task_data, split == '90%'))
```

Accuracy on validation set



Selection Scheme  tournament  lexicase

Summary statistics for the testing performance of the selection schemes at the 5% selection set split:

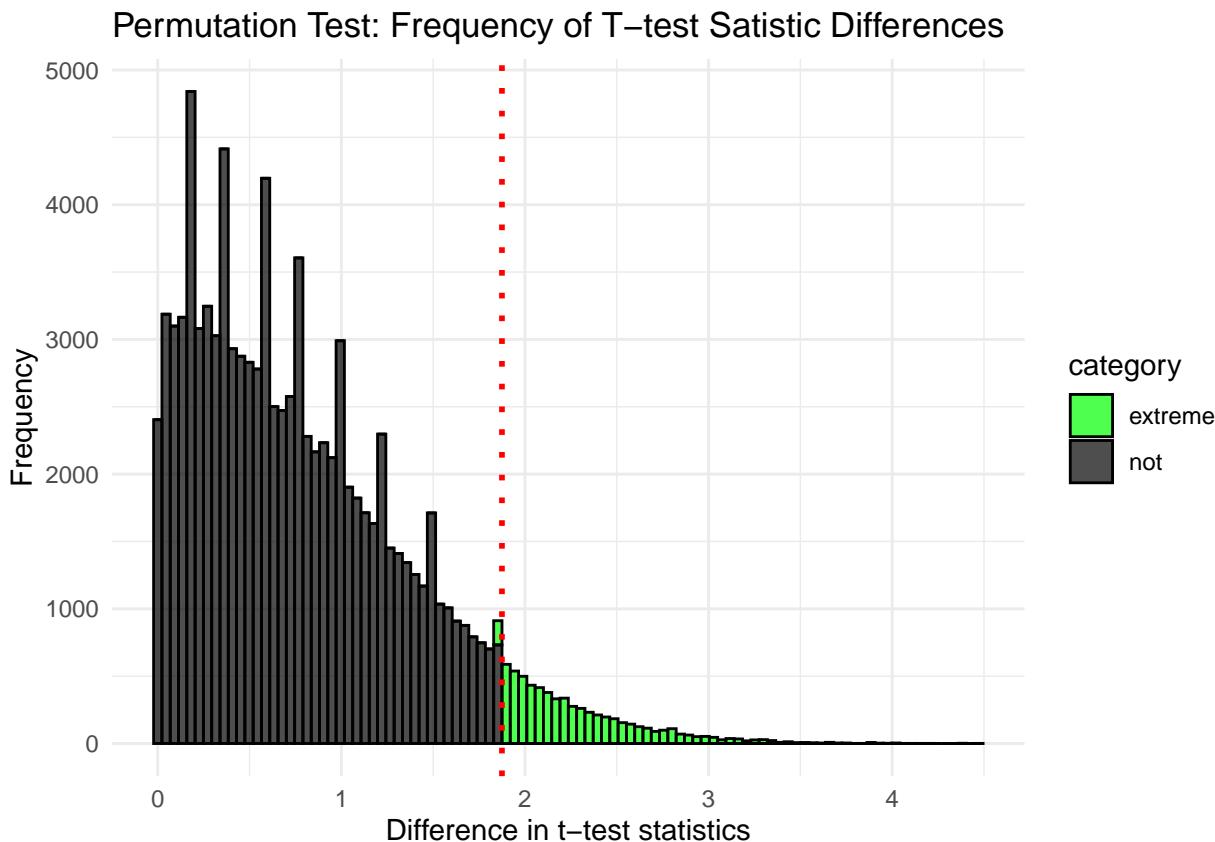
```
validation_accuracy_summary(filter(task_data, split == '90%'))
```

```
## # A tibble: 2 x 8
##   selection count na_cnt  min median  mean   max      IQR
##   <fct>     <int> <int> <dbl> <dbl> <dbl> <dbl> <dbl>
## 1 tournament     40     0 0.552 0.576 0.576 0.595 0.00874
## 2 lexicase       40     0 0.543 0.572 0.572 0.592 0.0121
```

The permutation test revealed that the results are:

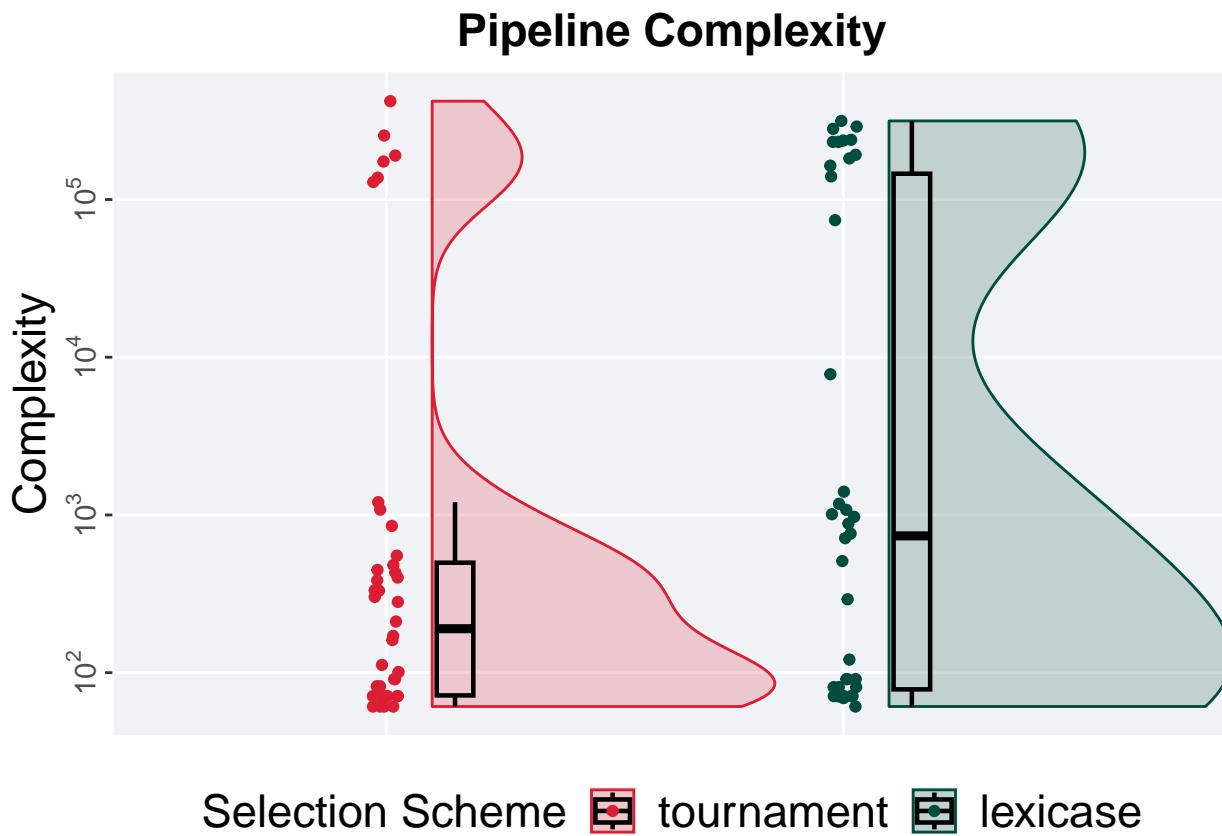
```
tournament_results <- filter(task_data, split == '90%' & selection == 'tournament')
lexicase_results <- filter(task_data, split == '90%' & selection == 'lexicase')
permutation_test(tournament_results$training_performance,
                 lexicase_results$training_performance,
                 seed = 98,
                 alternative = "t")
```

```
## [1] "observed_diff: 1.87465601726122"
## [1] "lower: -1.98271127158185"
## [1] "upper: 1.98270996776461"
## [1] "fail to reject null hypothesis"
## [1] "p-value: 0.0647"
```



11.4.3 Pipeline complexity

```
complexity_plot(filter(task_data, split == '90%'))
```



Summary statistics for the pipeline complexity of the selection schemes at the 5% selection set split:

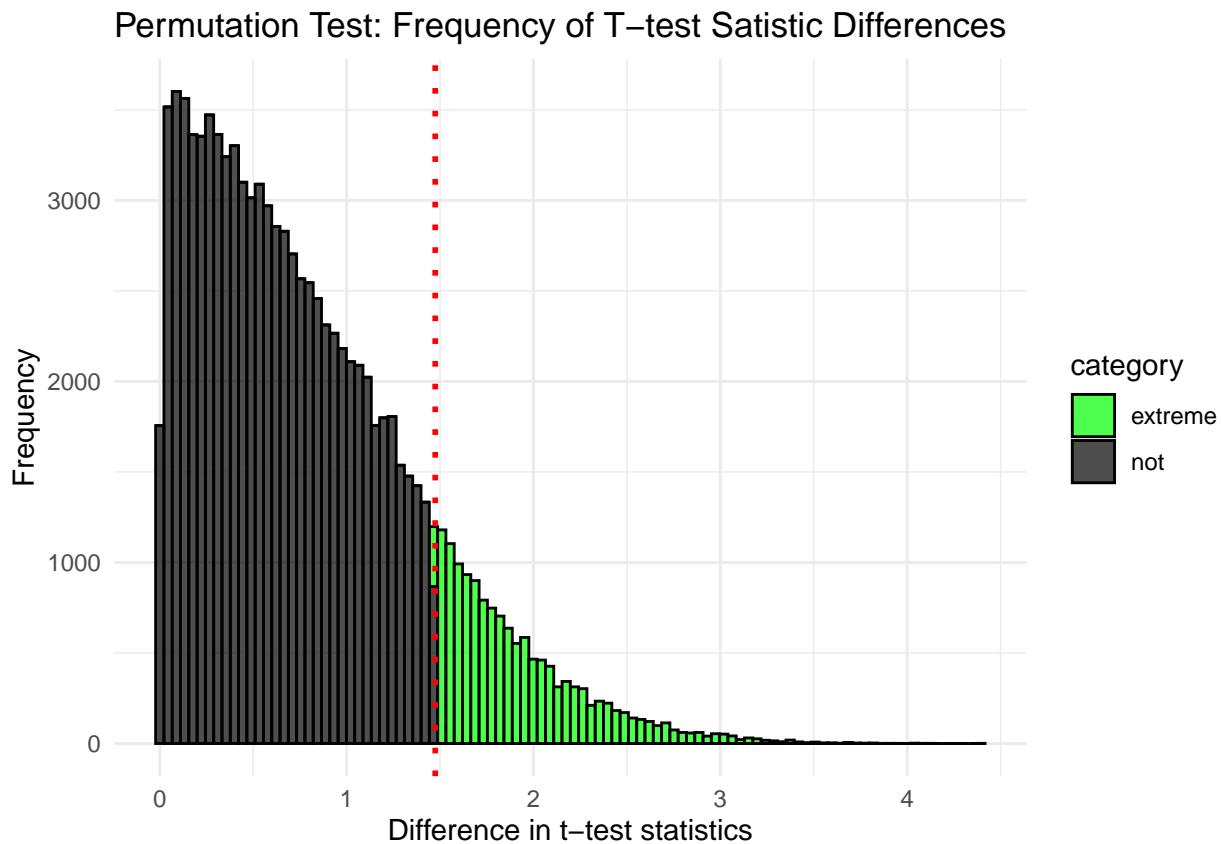
```
complexity_summary(filter(task_data, split == '90%'))
```

```
## # A tibble: 2 x 8
##   selection count na_cnt  min median   mean   max   IQR
##   <fct>     <int>  <int> <dbl> <dbl> <dbl> <int> <dbl>
## 1 tournament     40      0    61   191  32855. 420152    427
## 2 lexicase       40      0    61   737  64889. 314982 146007
```

The permutation test revealed that the results are:

```
tournament_results <- filter(task_data, split == '90%' & selection == 'tournament')
lexicase_results <- filter(task_data, split == '90%' & selection == 'lexicase')
permutation_test(tournament_results$testing_complexity,
                 lexicase_results$testing_complexity,
                 seed = 236,
                 alternative = "t")

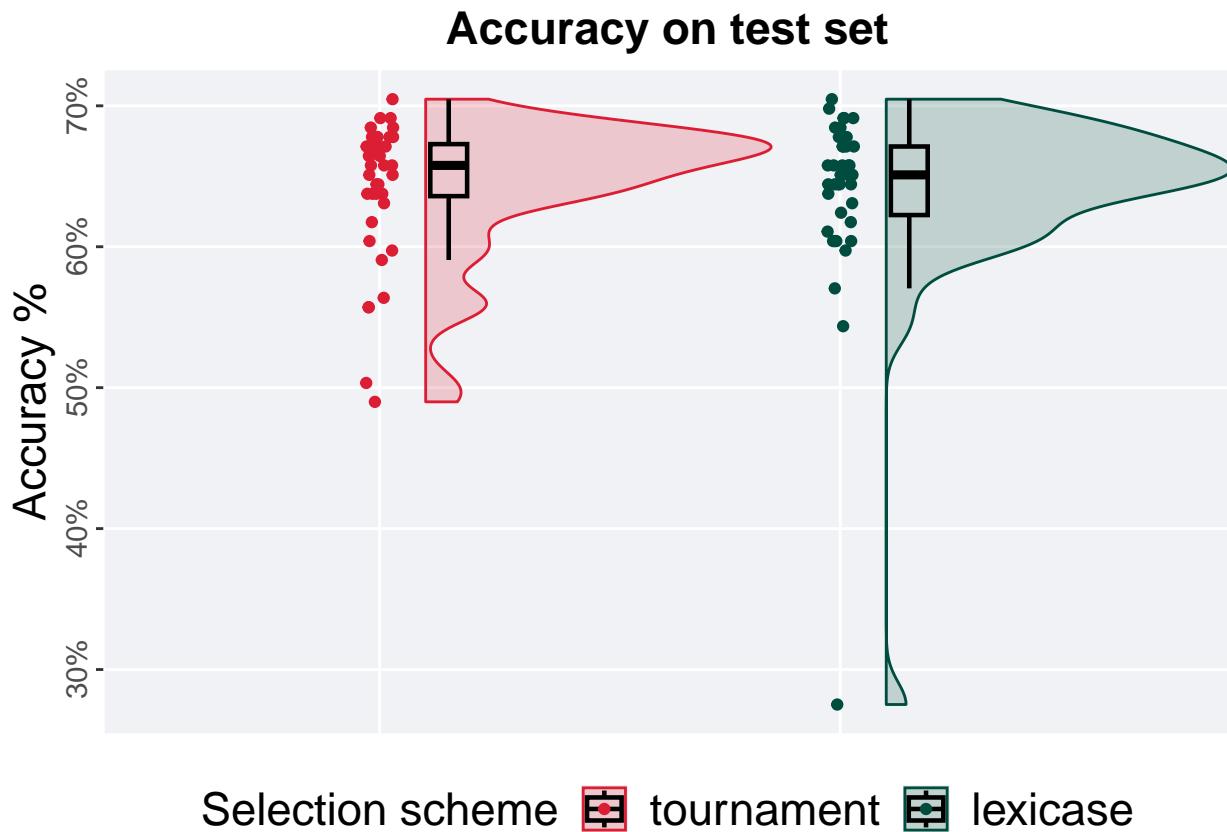
## [1] "observed_diff: -1.47418178221288"
## [1] "lower: -1.96253111190165"
## [1] "upper: 1.96959096899538"
## [1] "fail to reject null hypothesis"
## [1] "p-value: 0.14355"
```



11.5 95%

11.5.1 Test accuracy

```
test_plot(filter(task_data, split == '95%'))
```



Summary statistics for the testing performance of the selection schemes at the 5% selection set split:

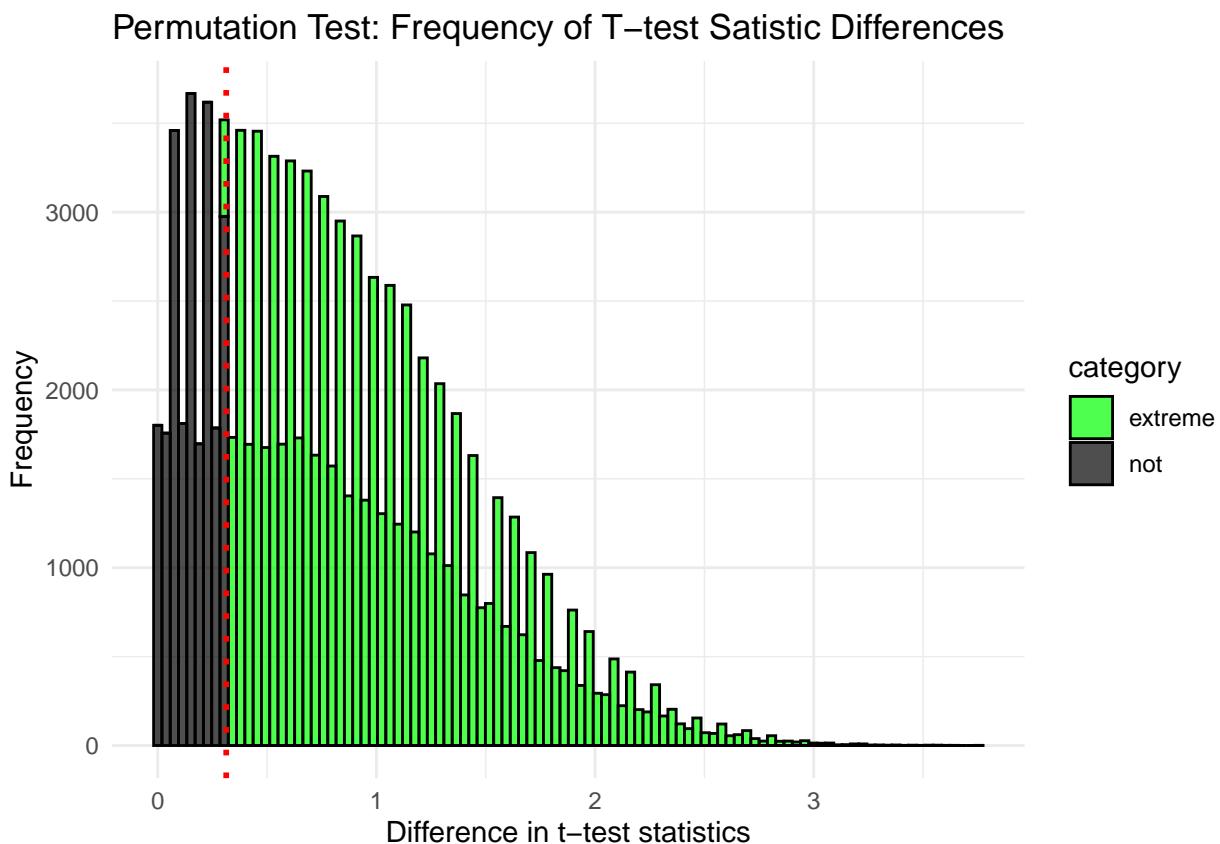
```
test_results_summary(filter(task_data, split == '95%'))
```

```
## # A tibble: 2 x 8
##   selection count na_cnt  min median  mean   max     IQR
##   <fct>     <int> <int> <dbl> <dbl> <dbl> <dbl> <dbl>
## 1 tournament     40     0 0.490 0.658 0.642 0.705 0.0369
## 2 lexicase       40     0 0.275 0.651 0.638 0.705 0.0487
```

The permutation test revealed that the results are:

```
tournament_results <- filter(task_data, split == '95%' & selection == 'tournament')
lexicase_results <- filter(task_data, split == '95%' & selection == 'lexicase')
permutation_test(tournament_results$testing_performance,
                 lexicase_results$testing_performance,
                 seed = 99,
                 alternative = "t")
```

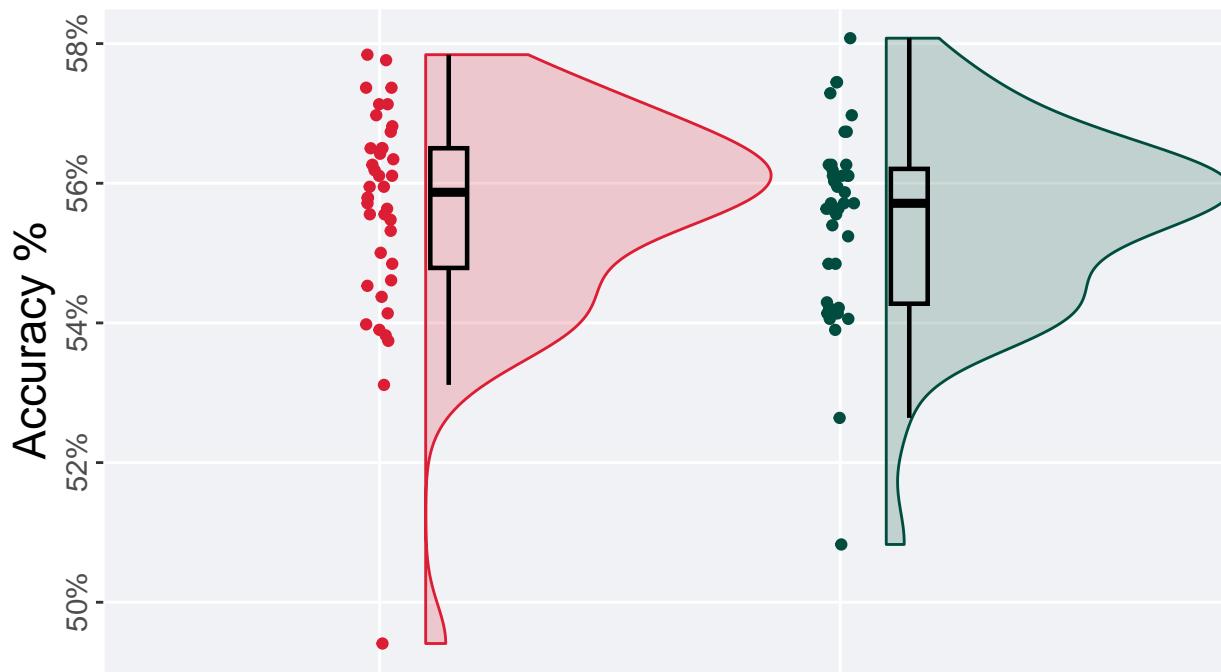
```
## [1] "observed_diff: 0.312972589294503"
## [1] "lower: -1.90711257326836"
## [1] "upper: 1.90711273295152"
## [1] "fail to reject null hypothesis"
## [1] "p-value: 0.77428"
```



11.5.2 Validation accuracy

```
validation_plot(filter(task_data, split == '95%'))
```

Accuracy on validation set



Selection Scheme tournament lexicase

Summary statistics for the testing performance of the selection schemes at the 5% selection set split:

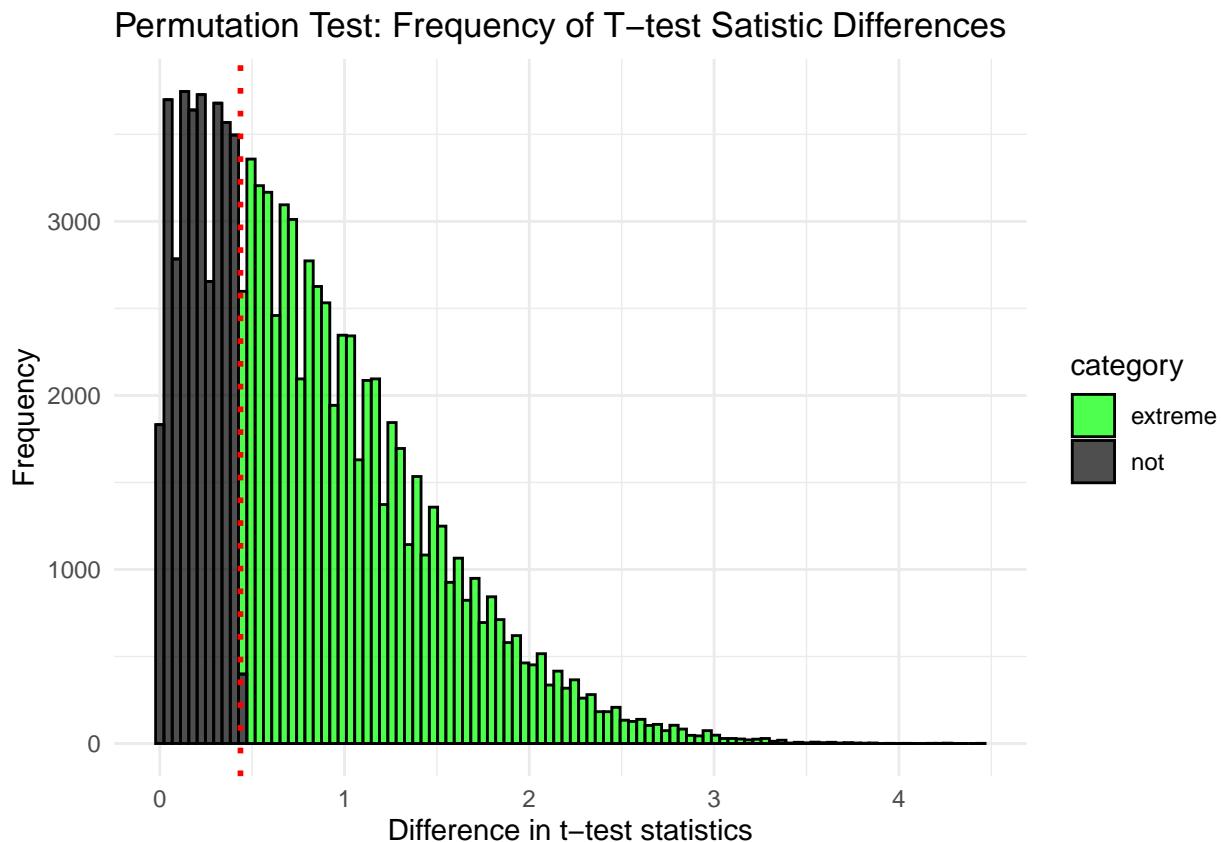
```
validation_accuracy_summary(filter(task_data, split == '95%'))
```

```
## # A tibble: 2 x 8
##   selection count na_cnt  min median  mean   max     IQR
##   <fct>     <int> <int> <dbl>  <dbl> <dbl> <dbl> <dbl>
## 1 tournament     40     0 0.494  0.559 0.556 0.578 0.0171
## 2 lexicase       40     0 0.508  0.557 0.555 0.581 0.0193
```

The permutation test revealed that the results are:

```
tournament_results <- filter(task_data, split == '95%' & selection == 'tournament')
lexicase_results <- filter(task_data, split == '95%' & selection == 'lexicase')
permutation_test(tournament_results$training_performance,
                 lexicase_results$training_performance,
                 seed = 100,
                 alternative = "t")
```

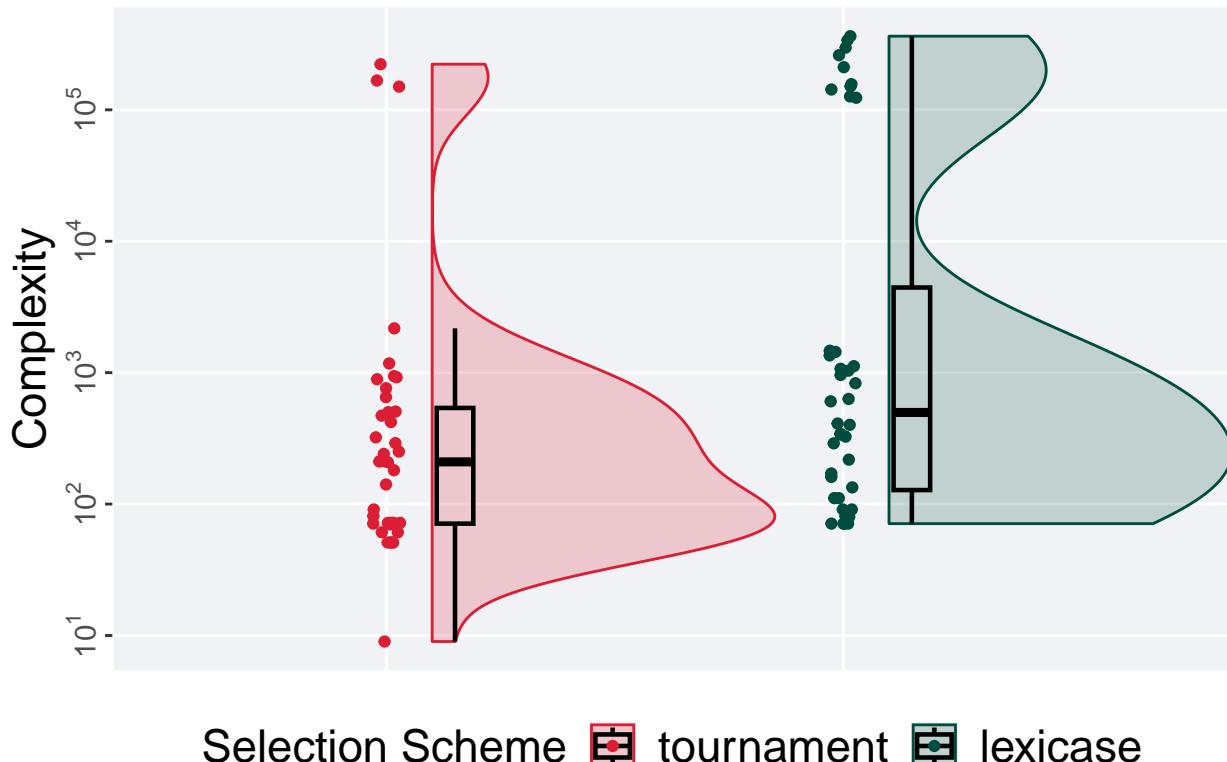
```
## [1] "observed_diff: 0.436779653664196"
## [1] "lower: -1.97806377537585"
## [1] "upper: 1.9909305869247"
## [1] "fail to reject null hypothesis"
## [1] "p-value: 0.66774"
```



11.5.3 Pipeline complexity

```
complexity_plot(filter(task_data, split == '95%'))
```

Pipeline Complexity



Summary statistics for the pipeline complexity of the selection schemes at the 5% selection set split:

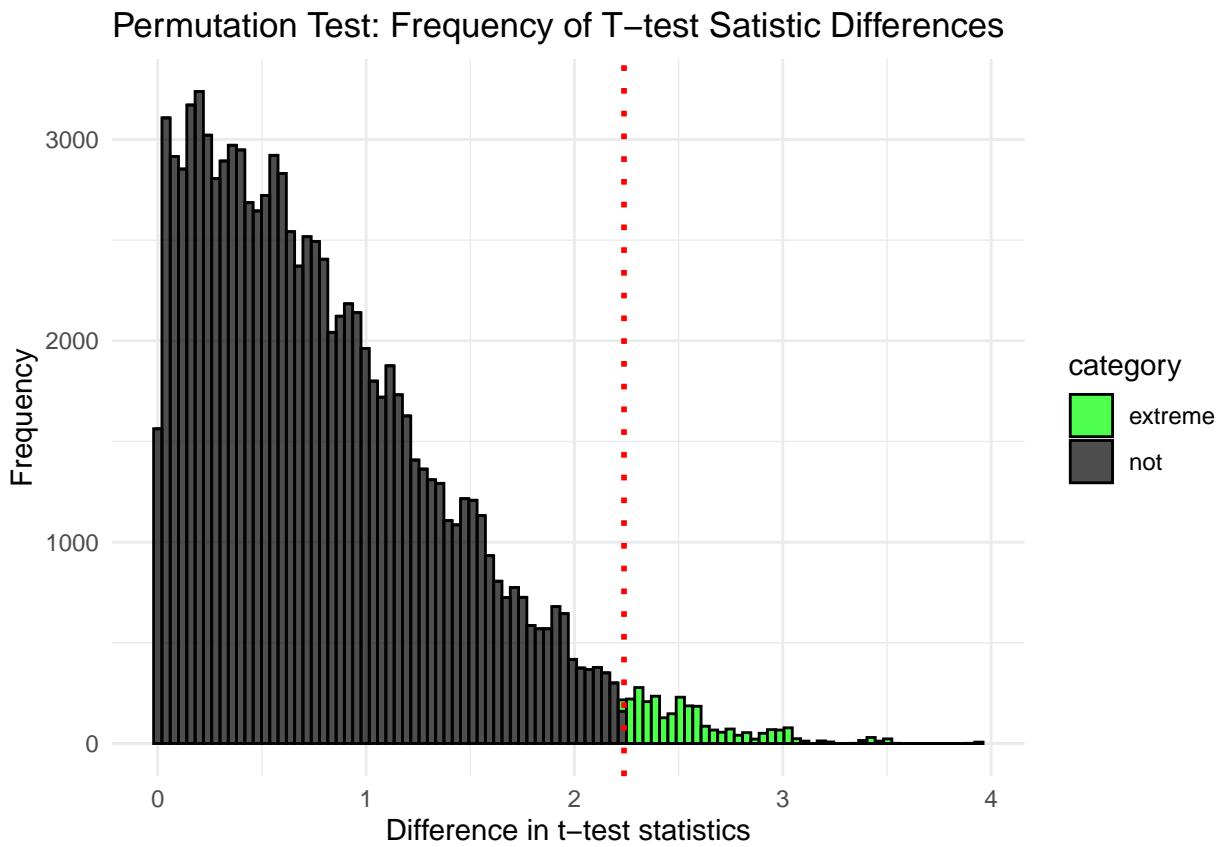
```
complexity_summary(filter(task_data, split == '95%'))
```

```
## # A tibble: 2 x 8
##   selection count na_cnt  min median   mean   max   IQR
##   <fct>     <int> <int> <dbl> <dbl> <dbl> <int> <dbl>
## 1 tournament     40     0     9  210. 13803. 222361  471.
## 2 lexicase       40     0    71  508. 54632. 363261 31856.
```

The permutation test revealed that the results are:

```
tournament_results <- filter(task_data, split == '95%' & selection == 'tournament')
lexicase_results <- filter(task_data, split == '95%' & selection == 'lexicase')
permutation_test(tournament_results$testing_complexity,
                 lexicase_results$testing_complexity,
                 seed = 237,
                 alternative = "t")
```

```
## [1] "observed_diff: -2.23790477967569"
## [1] "lower: -1.96389591237019"
## [1] "upper: 1.98625160778109"
## [1] "reject null hypothesis"
## [1] "p-value: 0.02687"
```



Chapter 12

Task 359960

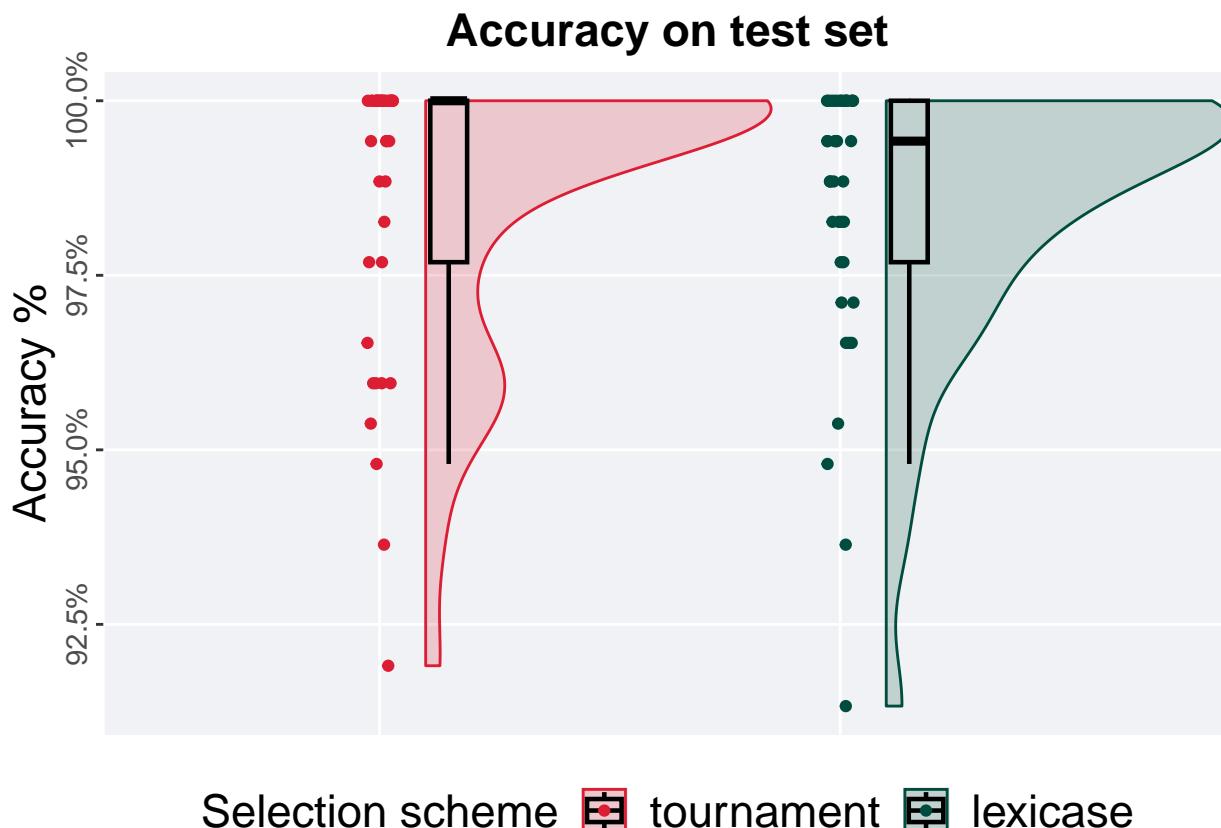
We present the results of our analysis of task 359960 with the different selection set splits used in our study.

```
task_data <- filter(results, task_id == 359960)
```

12.1 5%

12.1.1 Test accuracy

```
test_plot(filter(task_data, split == '5%'))
```



Summary statistics for the testing performance of the selection schemes at the 5% selection set split:

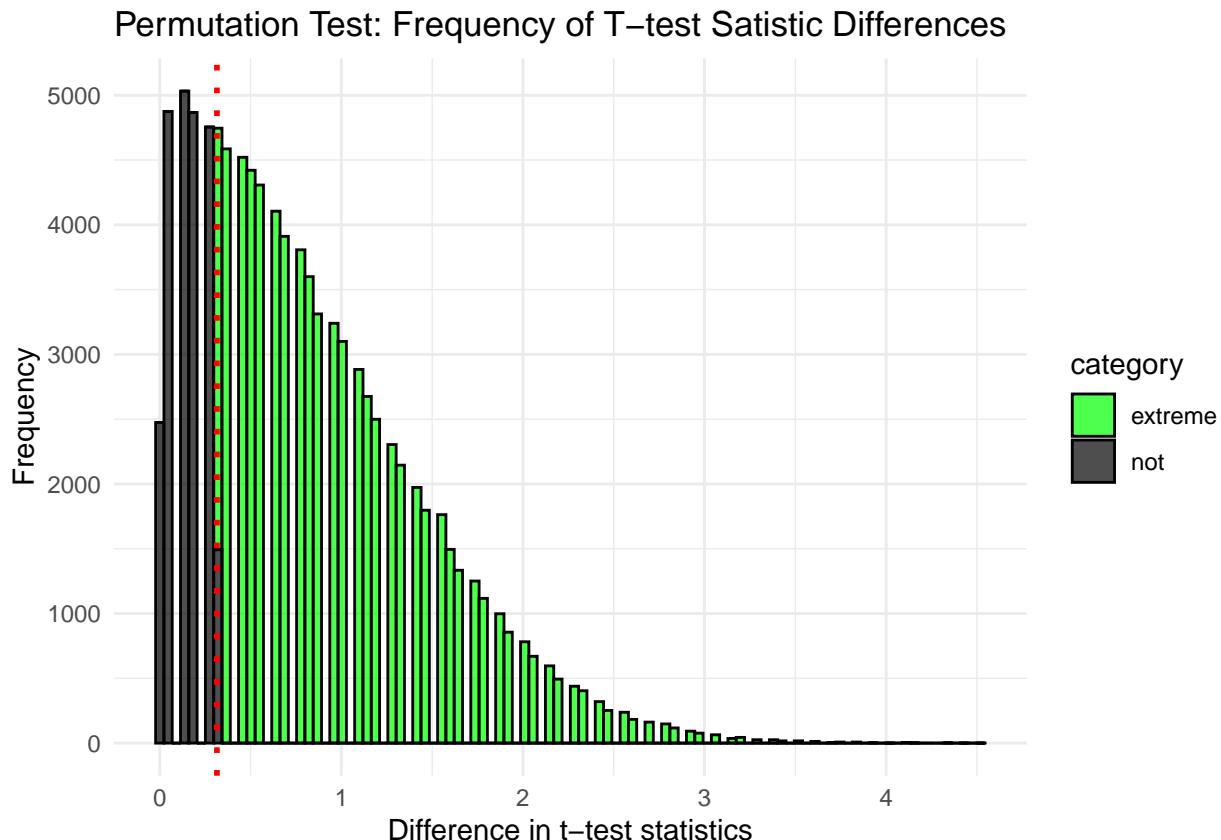
```
test_results_summary(filter(task_data, split == '5%'))
```

```
## # A tibble: 2 x 8
##   selection count na_cnt  min median  mean   max   IQR
##   <fct>     <int> <int> <dbl> <dbl> <dbl> <dbl> <dbl>
## 1 tournament     40     0 0.919  1     0.986     1 0.0231
## 2 lexicase       40     0 0.913  0.994  0.985     1 0.0231
```

The permutation test revealed that the results are:

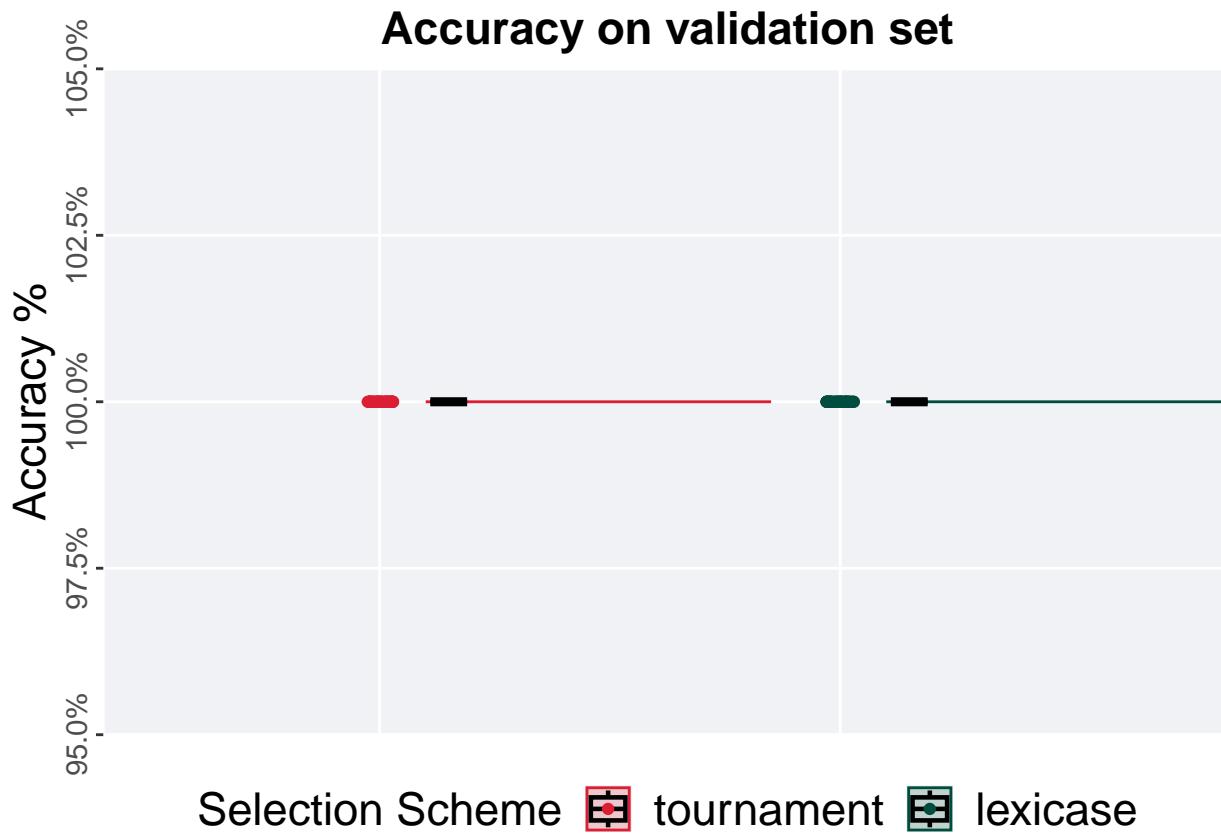
```
tournament_results <- filter(task_data, split == '5%' & selection == 'tournament')
lexicase_results <- filter(task_data, split == '5%' & selection == 'lexicase')
permutation_test(tournament_results$testing_performance,
                 lexicase_results$testing_performance,
                 seed = 101,
                 alternative = "t")
```

```
## [1] "observed_diff: 0.314658049512107"
## [1] "lower: -1.99895989583997"
## [1] "upper: 1.99895977857028"
## [1] "fail to reject null hypothesis"
## [1] "p-value: 0.76504"
```



12.1.2 Validation accuracy

```
validation_plot(filter(task_data, split == '5%'))
```



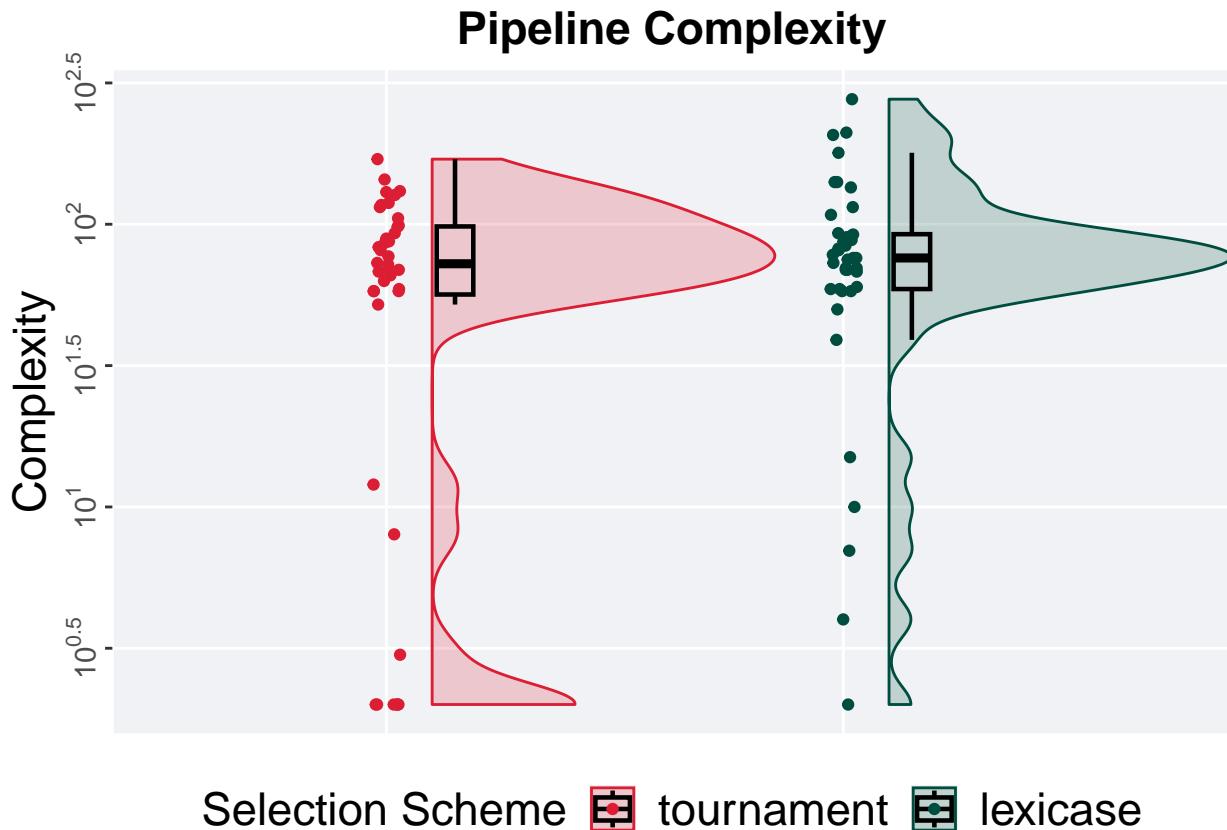
Summary statistics for the testing performance of the selection schemes at the 5% selection set split:

```
validation_accuracy_summary(filter(task_data, split == '5%'))
```

```
## # A tibble: 2 x 8
##   selection count na_cnt  min median  mean  max  IQR
##   <fct>     <int> <int> <dbl> <dbl> <dbl> <dbl> <dbl>
## 1 tournament     40     0     1     1     1     1     0
## 2 lexicase       40     0     1     1     1     1     0
```

12.1.3 Pipeline complexity

```
complexity_plot(filter(task_data, split == '5%'))
```



Summary statistics for the pipeline complexity of the selection schemes at the 5% selection set split:

```
complexity_summary(filter(task_data, split == '5%'))
```

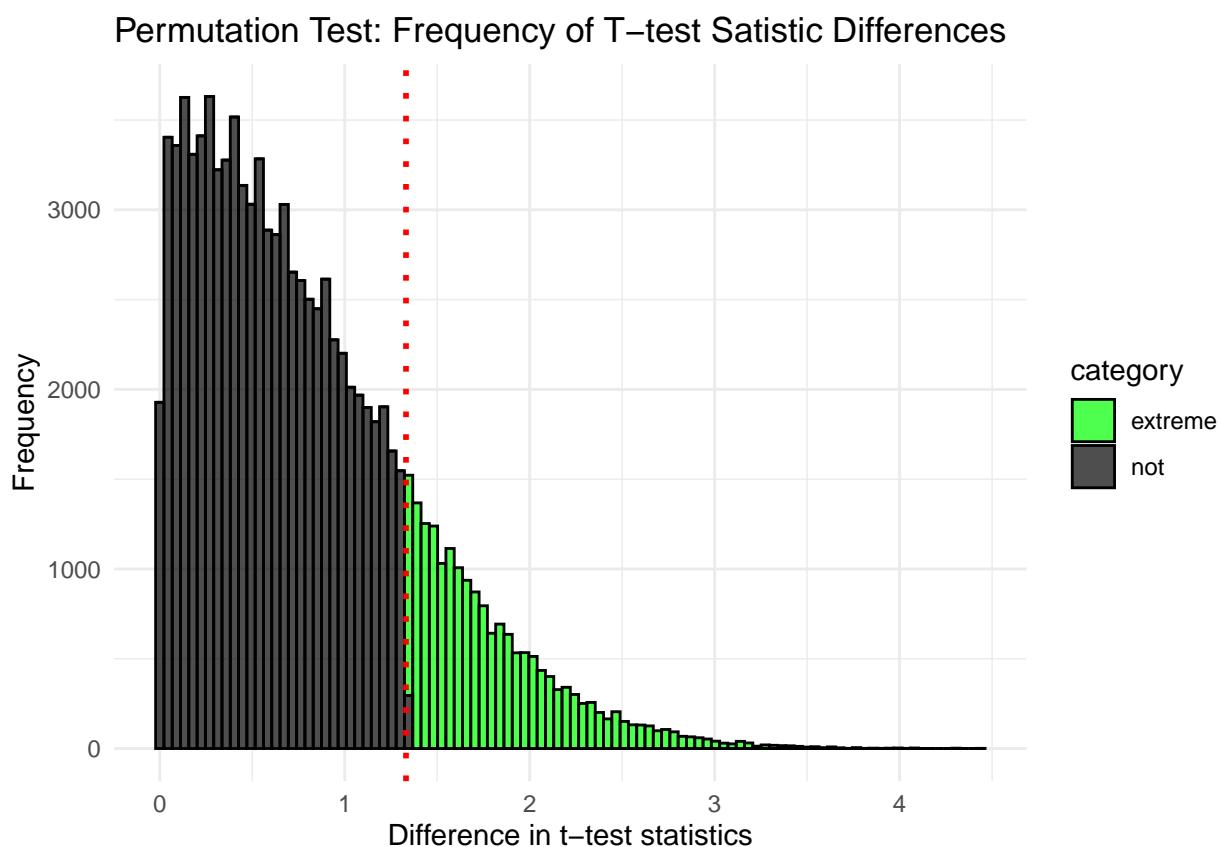
```
## # A tibble: 2 x 8
##   selection count na_cnt  min median  mean    max    IQR
##   <fct>     <int> <int> <dbl> <dbl> <dbl> <int> <dbl>
## 1 tournament     40     0     2   72.5   71.0   170   41.8
## 2 lexicase       40     0     2    76     86.2   277   33.2
```

The permutation test revealed that the results are:

```
tournament_results <- filter(task_data, split == '5%' & selection == 'tournament')
lexicase_results <- filter(task_data, split == '5%' & selection == 'lexicase')
```

```
permutation_test(tournament_results$testing_complexity,
                  lexicase_results$testing_complexity,
                  seed = 238,
                  alternative = "t")
```

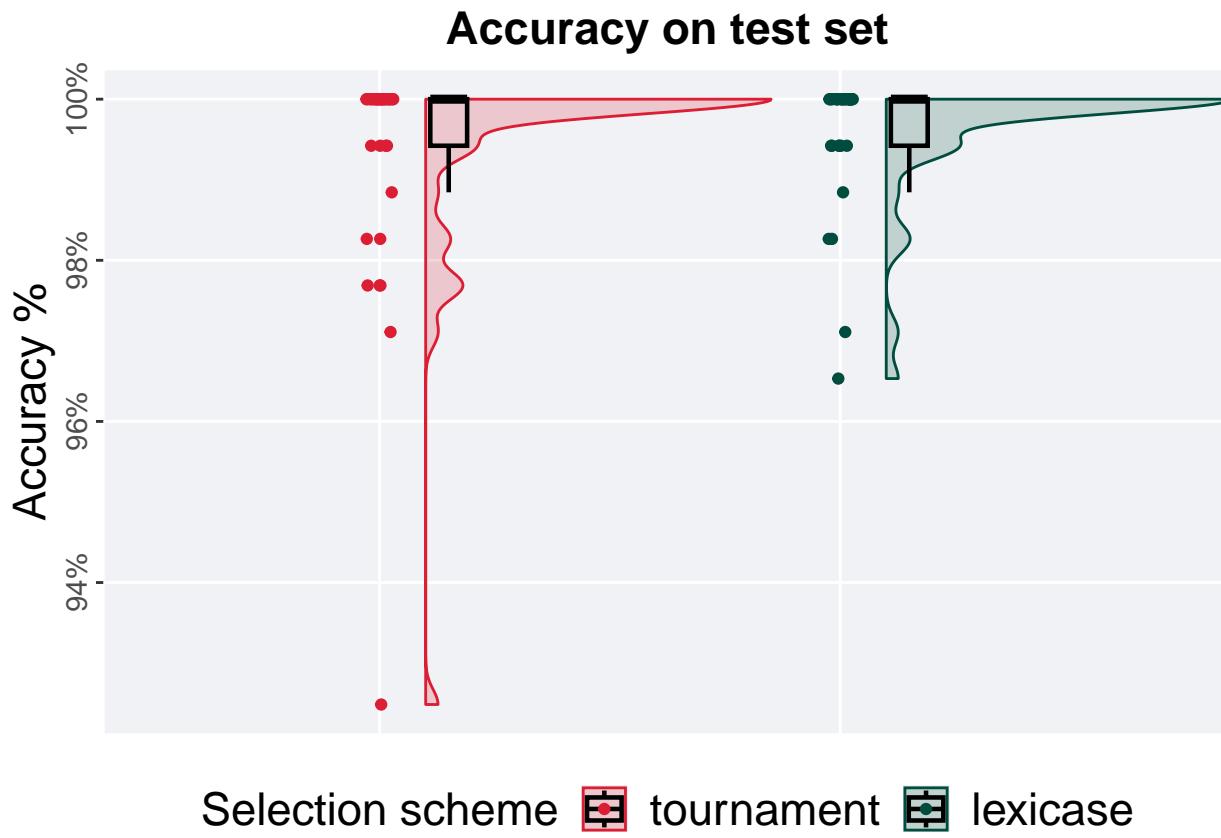
```
## [1] "observed_diff: -1.33131511321038"
## [1] "lower: -1.97926586935899"
## [1] "upper: 1.97926586935899"
## [1] "fail to reject null hypothesis"
## [1] "p-value: 0.18683"
```



12.2 10%

12.2.1 Test accuracy

```
test_plot(filter(task_data, split == '10%'))
```



Summary statistics for the testing performance of the selection schemes at the 5% selection set split:

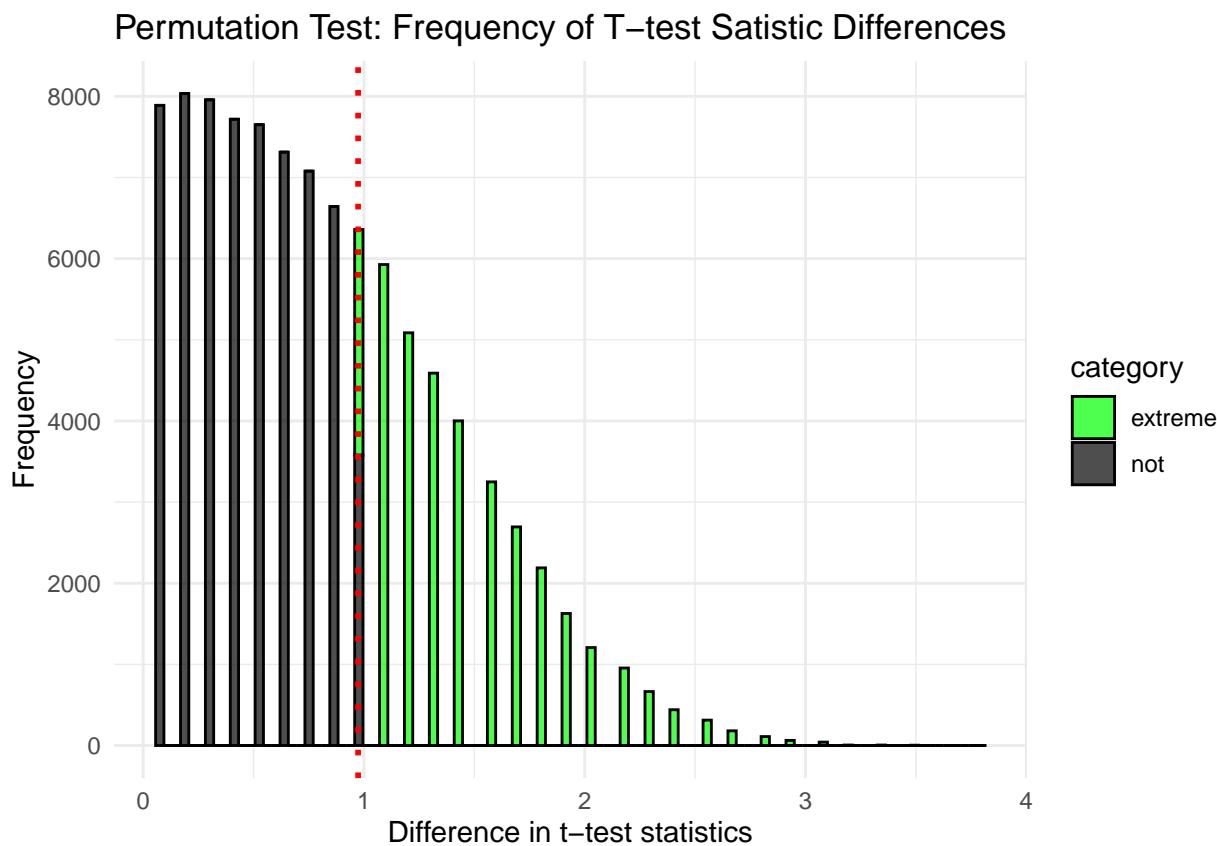
```
test_results_summary(filter(task_data, split == '10%'))
```

```
## # A tibble: 2 x 8
##   selection count na_cnt  min median  mean   max     IQR
##   <fct>     <int> <int> <dbl> <dbl> <dbl> <dbl> <dbl>
## 1 tournament     40     0 0.925    1 0.994    1 0.00578
## 2 lexicase       40     0 0.965    1 0.996    1 0.00578
```

The permutation test revealed that the results are:

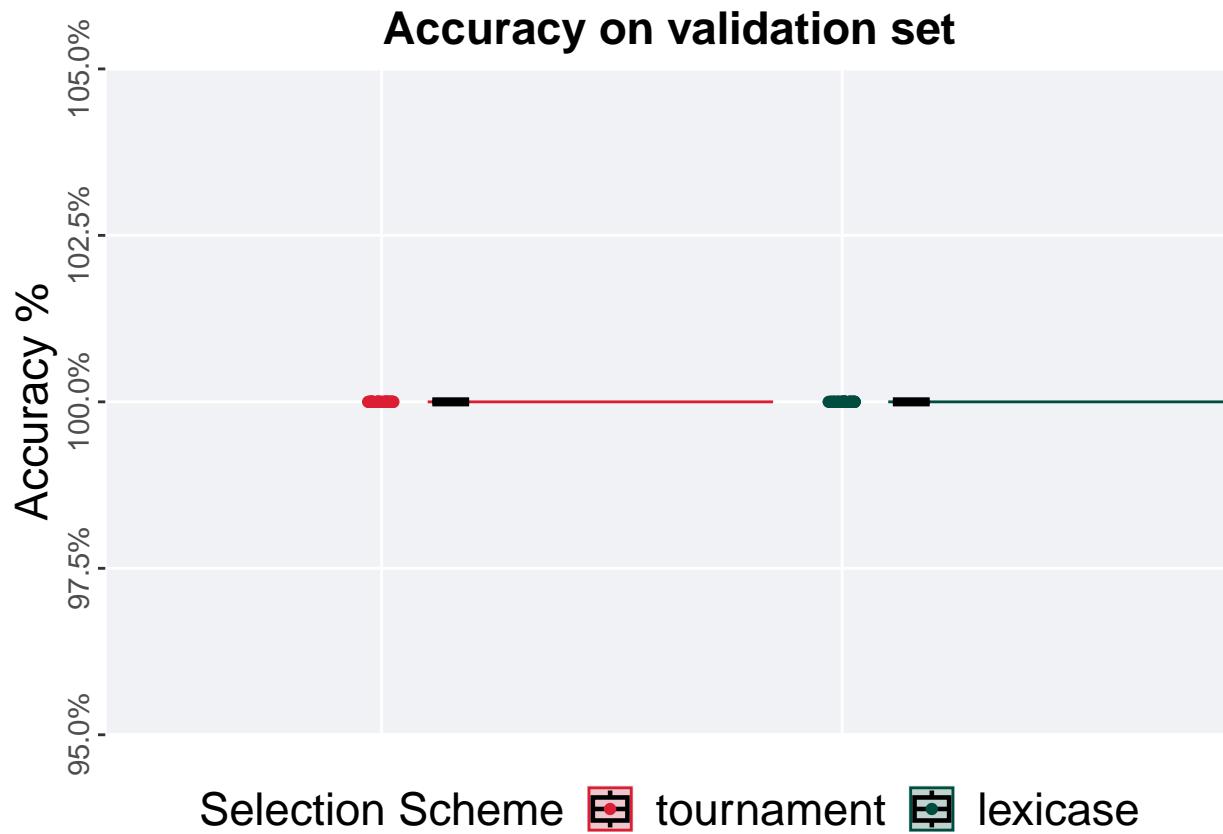
```
tournament_results <- filter(task_data, split == '10%' & selection == 'tournament')
lexicase_results <- filter(task_data, split == '10%' & selection == 'lexicase')
permutation_test(tournament_results$testing_performance,
                 lexicase_results$testing_performance,
                 seed = 103,
                 alternative = "t")
```

```
## [1] "observed_diff: -0.97358112870935"
## [1] "lower: -1.92250389346255"
## [1] "upper: 1.92250389346255"
## [1] "fail to reject null hypothesis"
## [1] "p-value: 0.36137"
```



12.2.2 Validation accuracy

```
validation_plot(filter(task_data, split == '10%'))
```



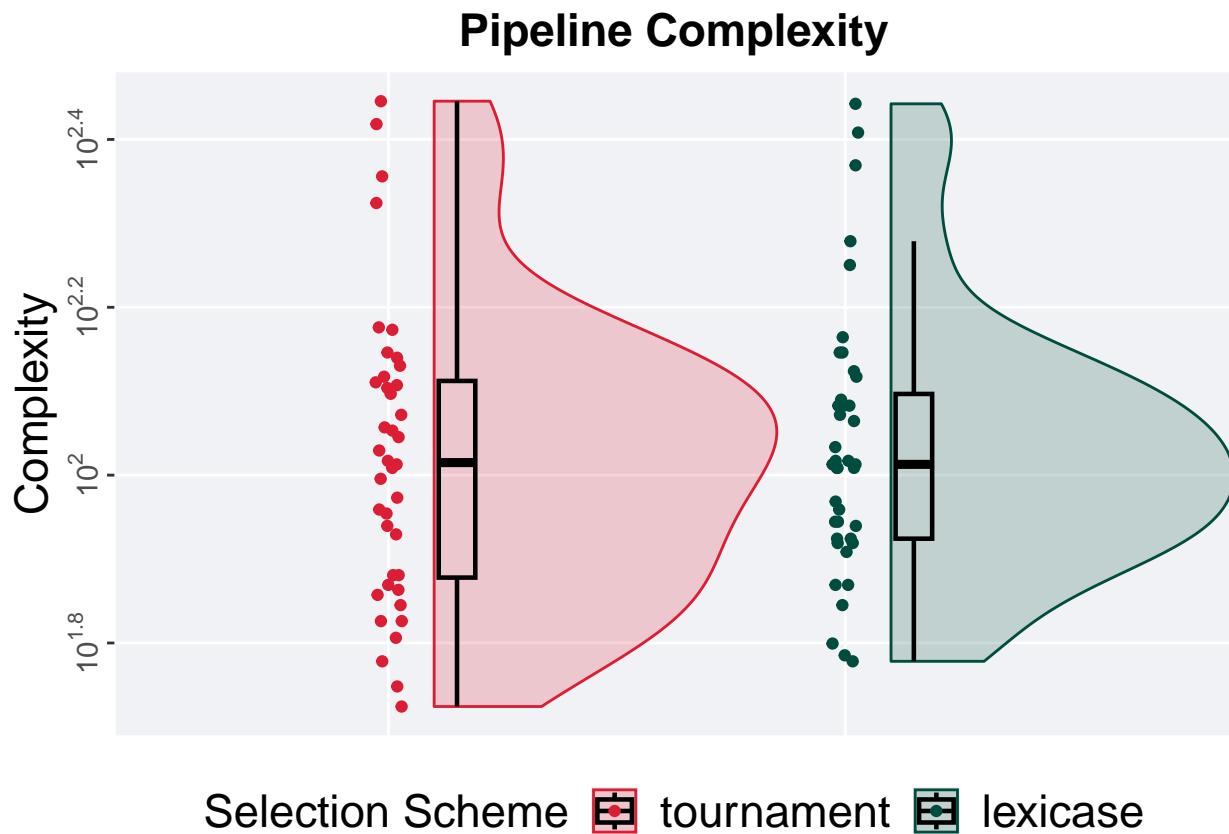
Summary statistics for the testing performance of the selection schemes at the 5% selection set split:

```
validation_accuracy_summary(filter(task_data, split == '10%'))
```

```
## # A tibble: 2 x 8
##   selection  count na_cnt   min median   mean   max   IQR
##   <fct>     <int> <int> <dbl> <dbl> <dbl> <dbl> <dbl>
## 1 tournament     40     0     1     1     1     1     0
## 2 lexicase       40     0     1     1     1     1     0
```

12.2.3 Pipeline complexity

```
complexity_plot(filter(task_data, split == '10%'))
```



Summary statistics for the pipeline complexity of the selection schemes at the 5% selection set split:

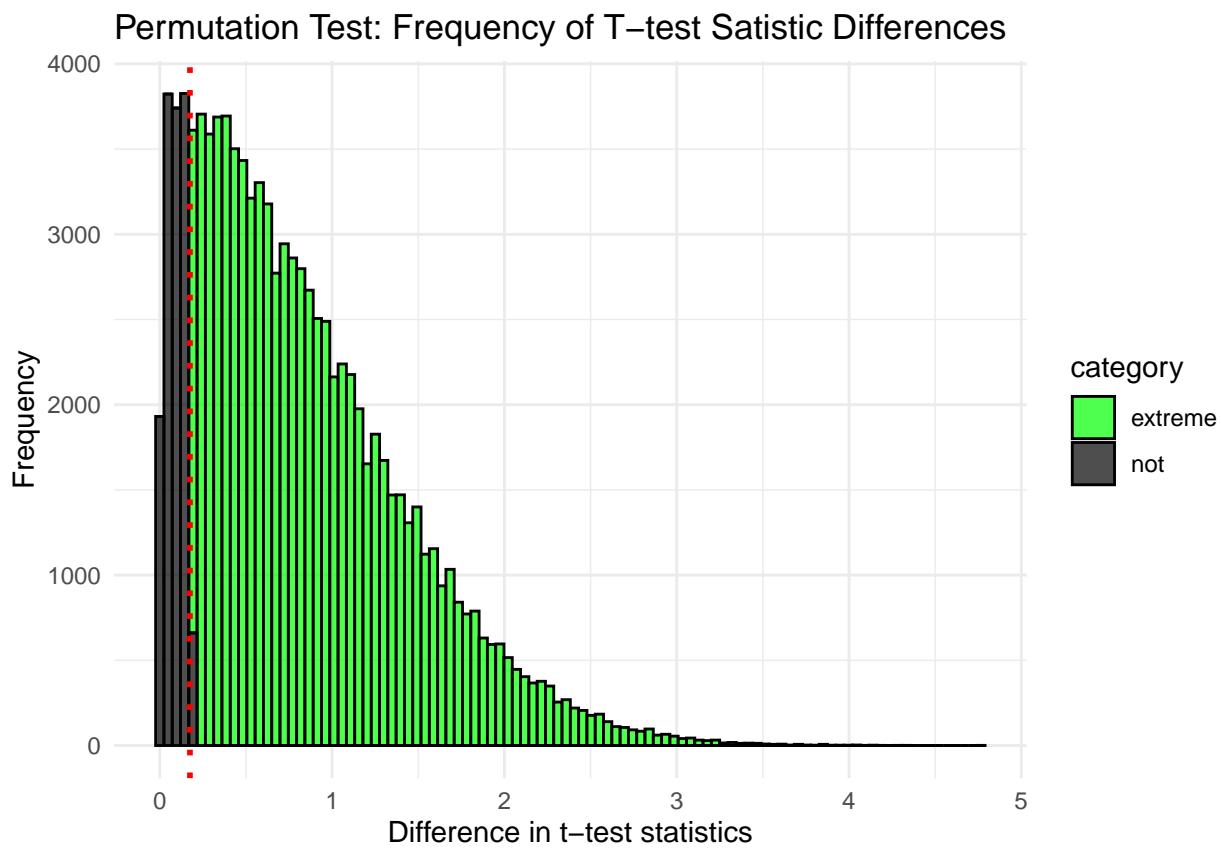
```
complexity_summary(filter(task_data, split == '10%'))
```

```
## # A tibble: 2 x 8
##   selection count na_cnt  min median  mean   max   IQR
##   <fct>     <int> <int> <dbl> <dbl> <dbl> <int> <dbl>
## 1 tournament     40     0    53  104.  114.   279    54
## 2 lexicase       40     0    60  103.  116.   277    41
```

The permutation test revealed that the results are:

```
tournament_results <- filter(task_data, split == '10%' & selection == 'tournament')
lexicase_results <- filter(task_data, split == '10%' & selection == 'lexicase')
permutation_test(tournament_results$testing_complexity,
                 lexicase_results$testing_complexity,
                 seed = 239,
                 alternative = "t")

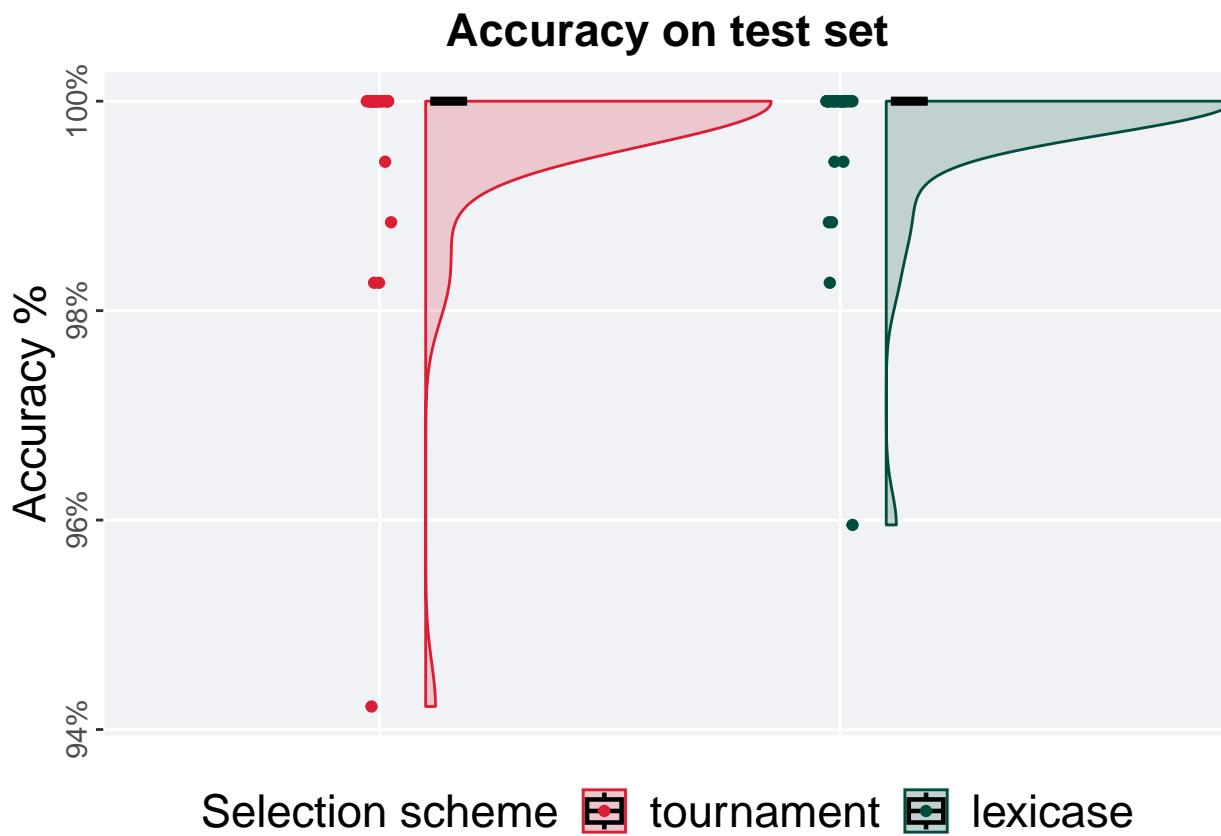
## [1] "observed_diff: -0.17514685977786"
## [1] "lower: -1.9973228693994"
## [1] "upper: 1.97846414300996"
## [1] "fail to reject null hypothesis"
## [1] "p-value: 0.86018"
```



12.3 50%

12.3.1 Test accuracy

```
test_plot(filter(task_data, split == '50%'))
```



Summary statistics for the testing performance of the selection schemes at the 5% selection set split:

```
test_results_summary(filter(task_data, split == '50%'))
```

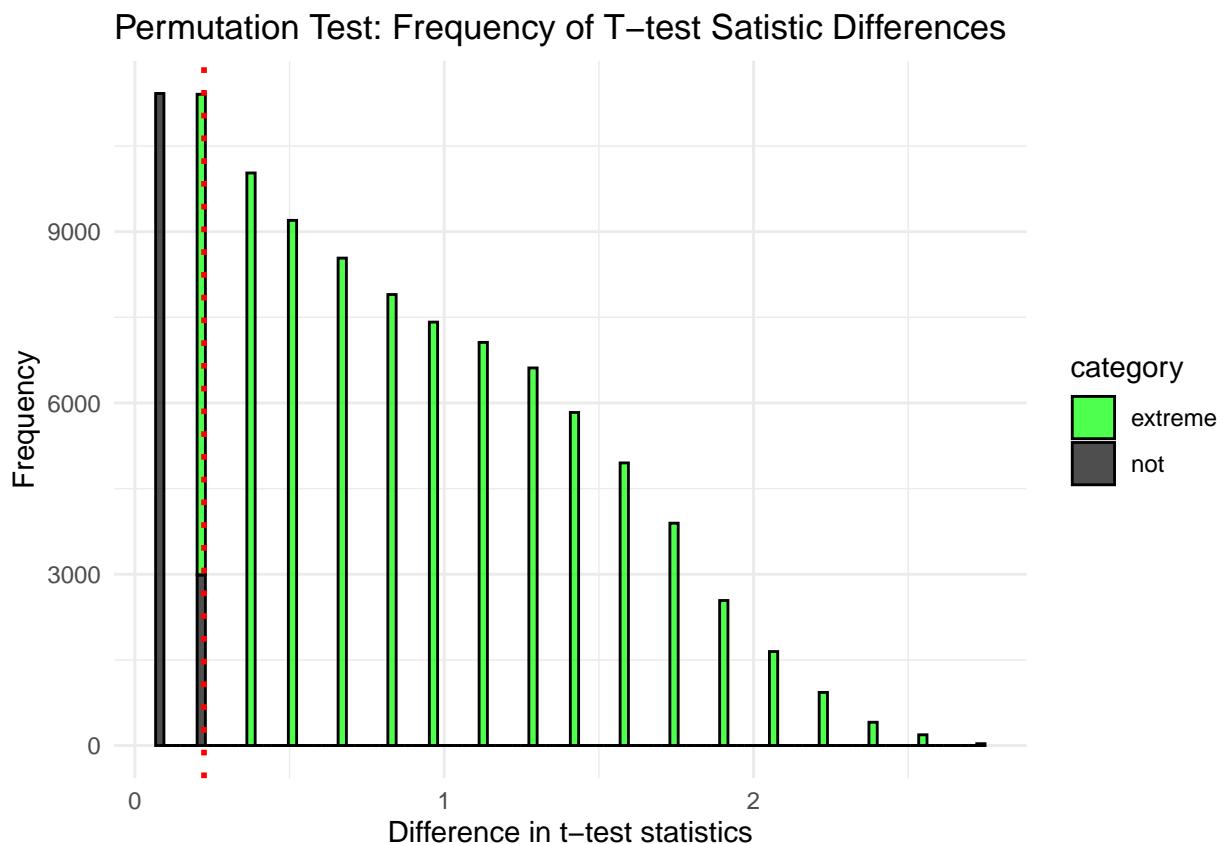
```
## # A tibble: 2 x 8
##   selection count na_cnt  min median  mean   max   IQR
##   <fct>     <int> <int> <dbl> <dbl> <dbl> <dbl> <dbl>
## 1 tournament     40     0 0.942    1 0.997    1     0
## 2 lexicase       40     0 0.960    1 0.998    1     0
```

The permutation test revealed that the results are:

```
tournament_results <- filter(task_data, split == '50%' & selection == 'tournament')
lexicase_results <- filter(task_data, split == '50%' & selection == 'lexicase')
```

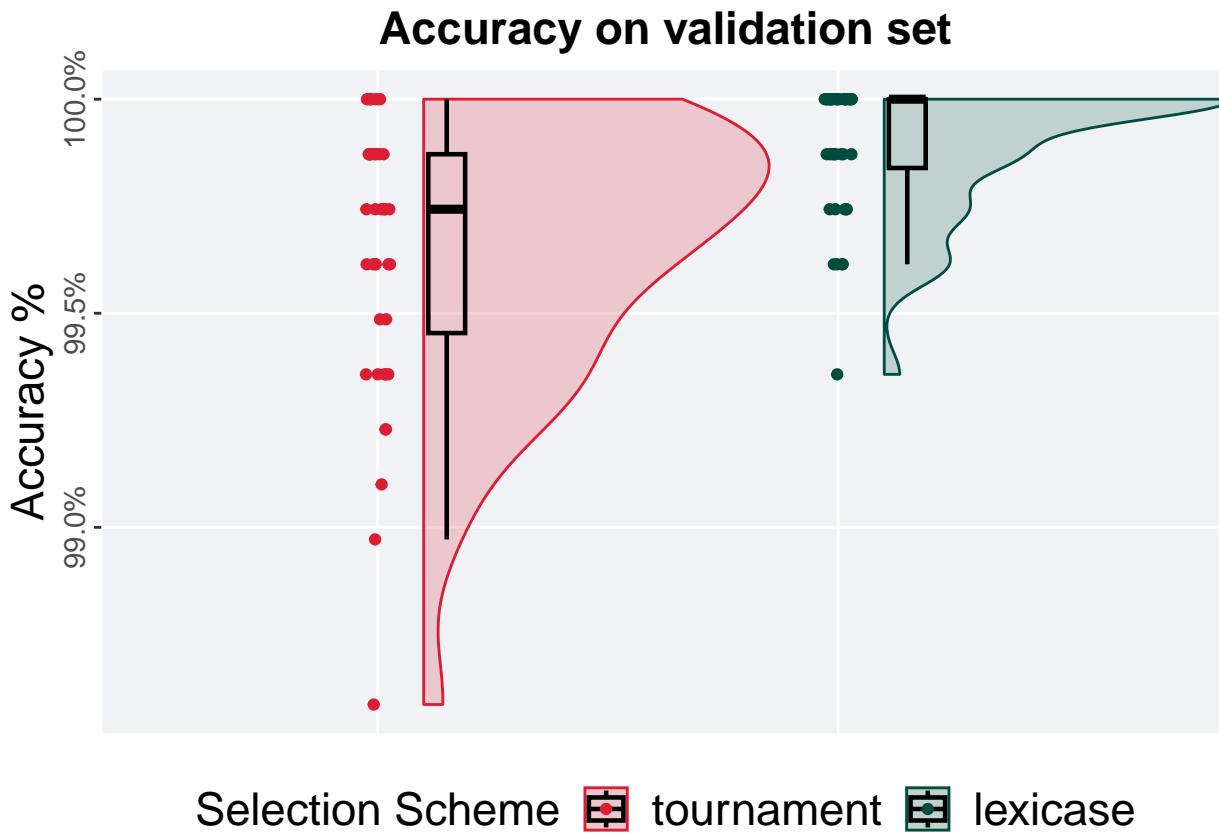
```
permutation_test(tournament_results$testing_performance,
                  lexicase_results$testing_performance,
                  seed = 105,
                  alternative = "t")
```

```
## [1] "observed_diff: -0.223558353436898"
## [1] "lower: -1.90523711235765"
## [1] "upper: 1.90523711235765"
## [1] "fail to reject null hypothesis"
## [1] "p-value: 0.85593"
```



12.3.2 Validation accuracy

```
validation_plot(filter(task_data, split == '50%'))
```



Summary statistics for the testing performance of the selection schemes at the 5% selection set split:

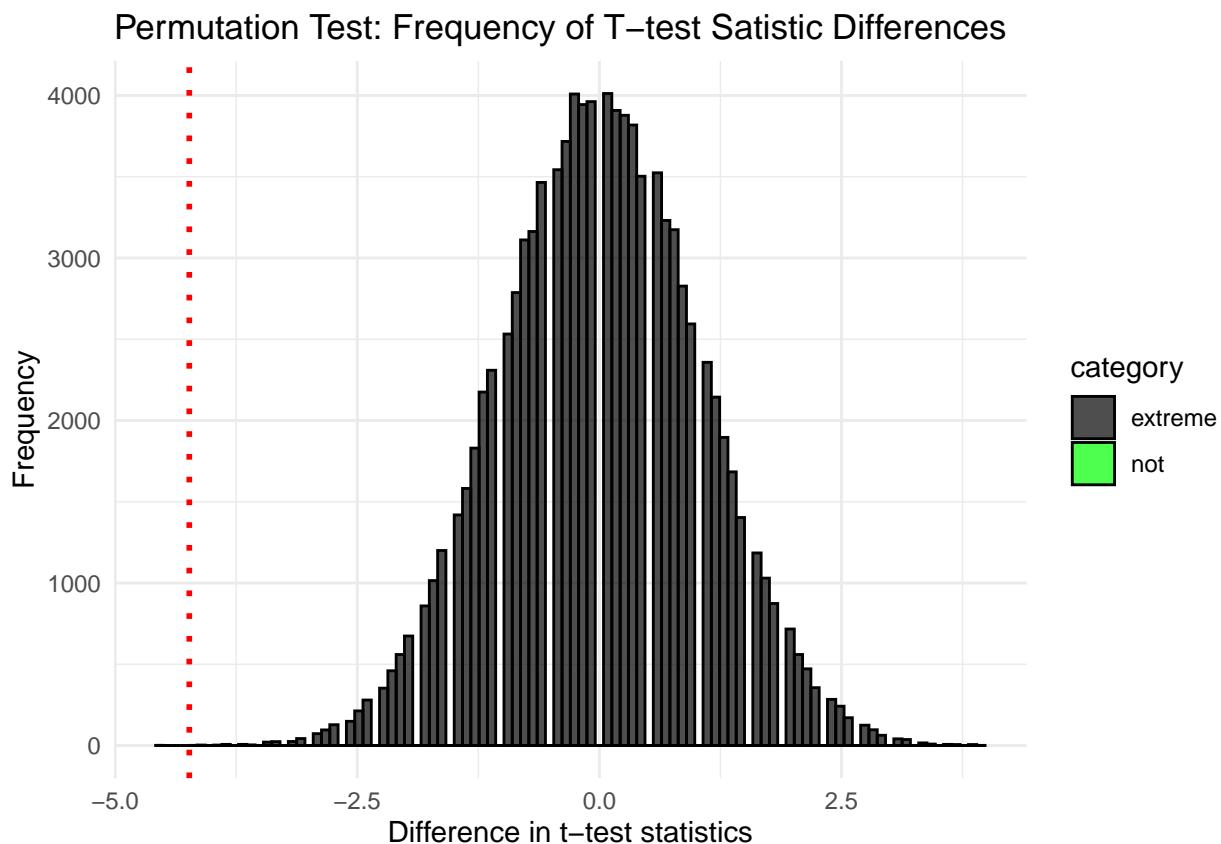
```
validation_accuracy_summary(filter(task_data, split == '50%'))
```

```
## # A tibble: 2 x 8
##   selection  count  na_cnt  min  median  mean  max      IQR
##   <fct>      <int>   <int>  <dbl>   <dbl>  <dbl> <dbl>   <dbl>
## 1 tournament     40       0  0.986  0.997  0.996     1  0.00418
## 2 lexicase       40       0  0.994   1       0.999     1  0.00161
```

The permutation test revealed that the results are:

```
tournament_results <- filter(task_data, split == '50%' & selection == 'tournament')
lexicase_results <- filter(task_data, split == '50%' & selection == 'lexicase')
permutation_test(tournament_results$training_performance,
                 lexicase_results$training_performance,
                 seed = 106,
                 alternative = "1")
```

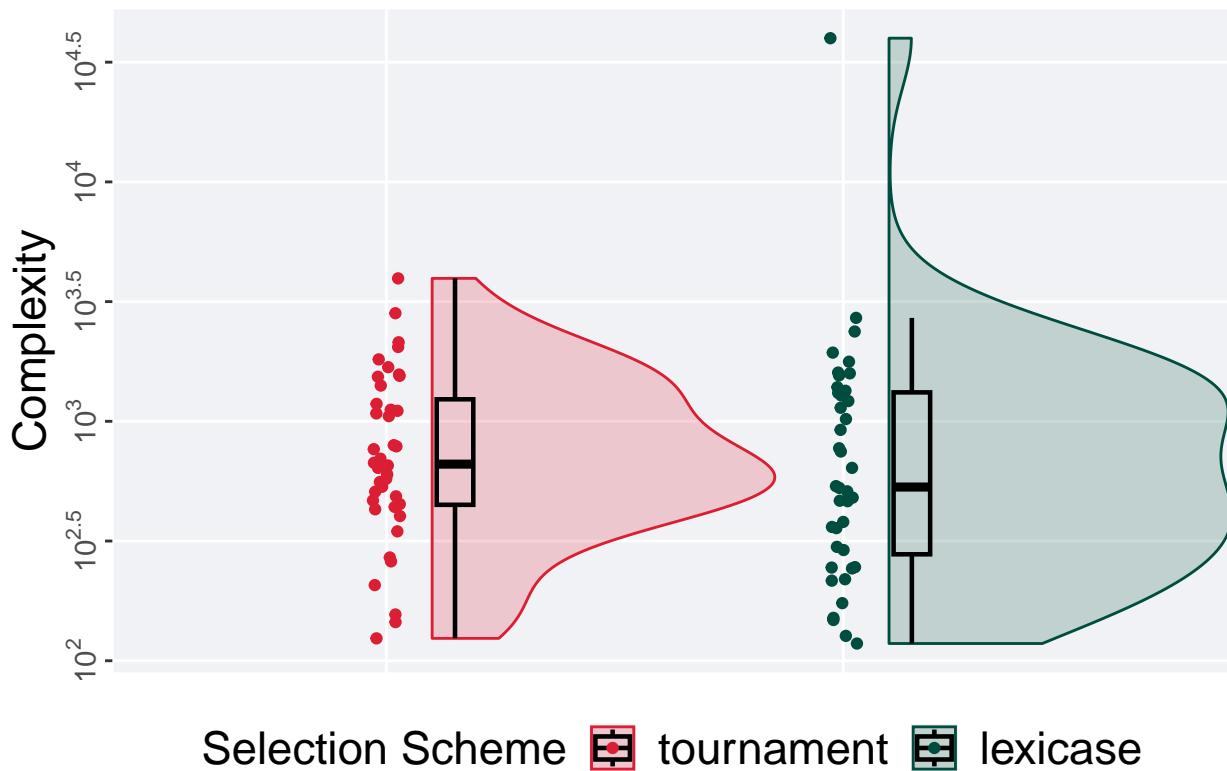
```
## [1] "observed_diff: -4.23440936732469"
## [1] "permutation_diffs[0.05 * n_permutations]: -1.60404248357023"
## [1] "reject null hypothesis"
## [1] "p-value: 1e-05"
```



12.3.3 Pipeline complexity

```
complexity_plot(filter(task_data, split == '50%'))
```

Pipeline Complexity



Summary statistics for the pipeline complexity of the selection schemes at the 5% selection set split:

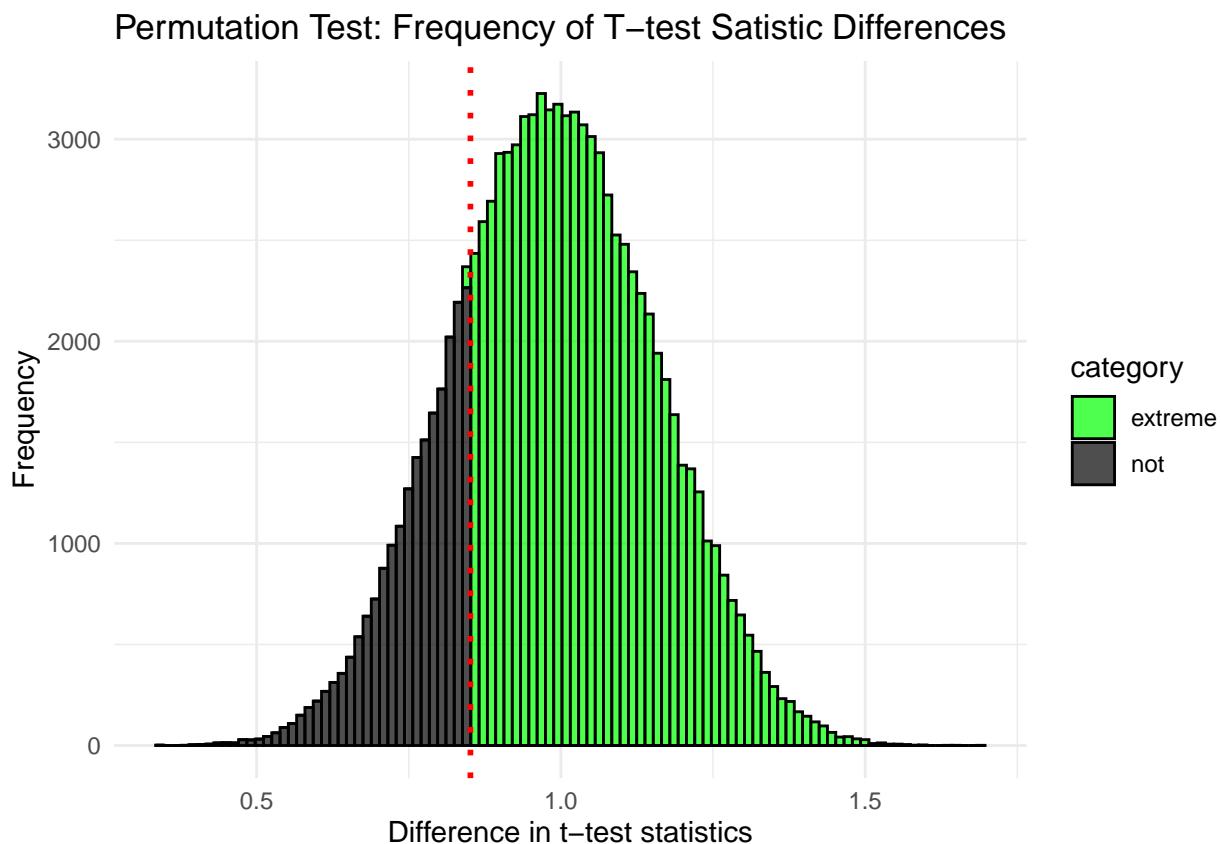
```
complexity_summary(filter(task_data, split == '50%'))
```

```
## # A tibble: 2 x 8
##   selection count na_cnt  min median  mean   max   IQR
##   <fct>     <int> <int> <dbl> <dbl> <dbl> <int> <dbl>
## 1 tournament     40     0  124   662.  951. 3954  792.
## 2 lexicase       40     0  118   532. 1792. 39831 1042
```

The permutation test revealed that the results are:

```
tournament_results <- filter(task_data, split == '50%' & selection == 'tournament')
lexicase_results <- filter(task_data, split == '50%' & selection == 'lexicase')
permutation_test(tournament_results$testing_complexity,
                  lexicase_results$testing_complexity,
                  seed = 240,
                  alternative = "t")
```

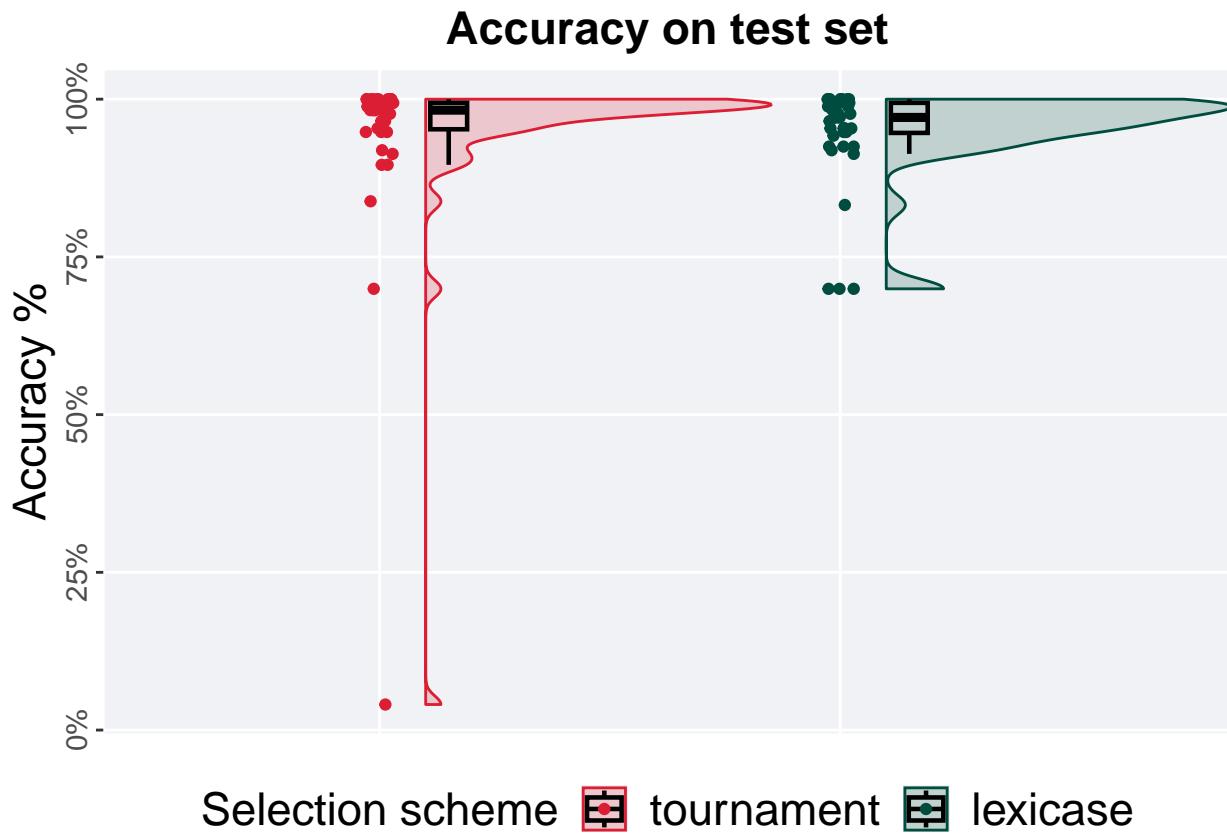
```
## [1] "observed_diff: -0.851290499309084"
## [1] "lower: -1.26104577566952"
## [1] "upper: 1.26343220206979"
## [1] "fail to reject null hypothesis"
## [1] "p-value: 0.78663"
```



12.4 90%

12.4.1 Test accuracy

```
test_plot(filter(task_data, split == '90%'))
```



Summary statistics for the testing performance of the selection schemes at the 5% selection set split:

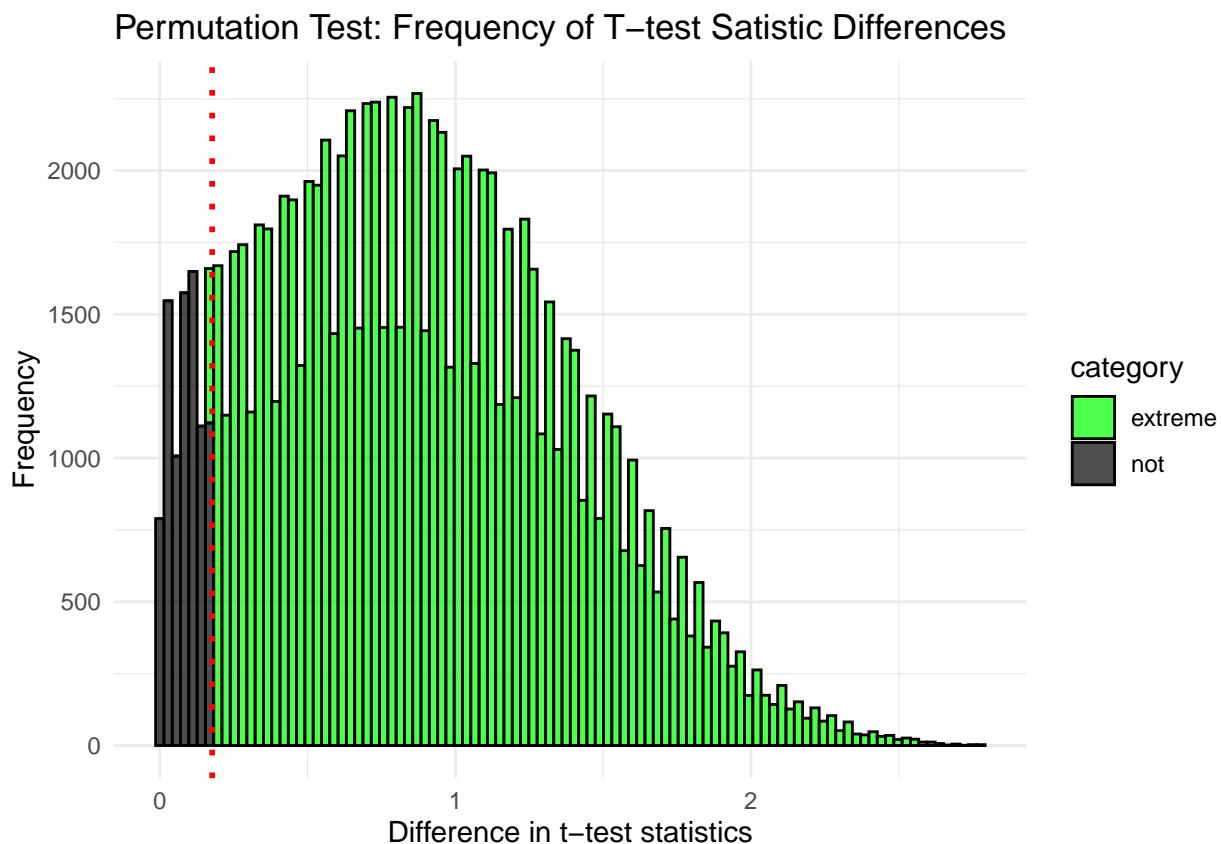
```
test_results_summary(filter(task_data, split == '90%'))
```

```
## # A tibble: 2 x 8
##   selection count na_cnt   min median   mean   max   IQR
##   <fct>     <int> <int> <dbl> <dbl> <dbl> <dbl> <dbl>
## 1 tournament     40     0 0.0405  0.983  0.942     1 0.0419
## 2 lexicase       40     0 0.699   0.971  0.947     1 0.0477
```

The permutation test revealed that the results are:

```
tournament_results <- filter(task_data, split == '90%' & selection == 'tournament')
lexicase_results <- filter(task_data, split == '90%' & selection == 'lexicase')
permutation_test(tournament_results$testing_performance,
                 lexicase_results$testing_performance,
                 seed = 107,
                 alternative = "t")
```

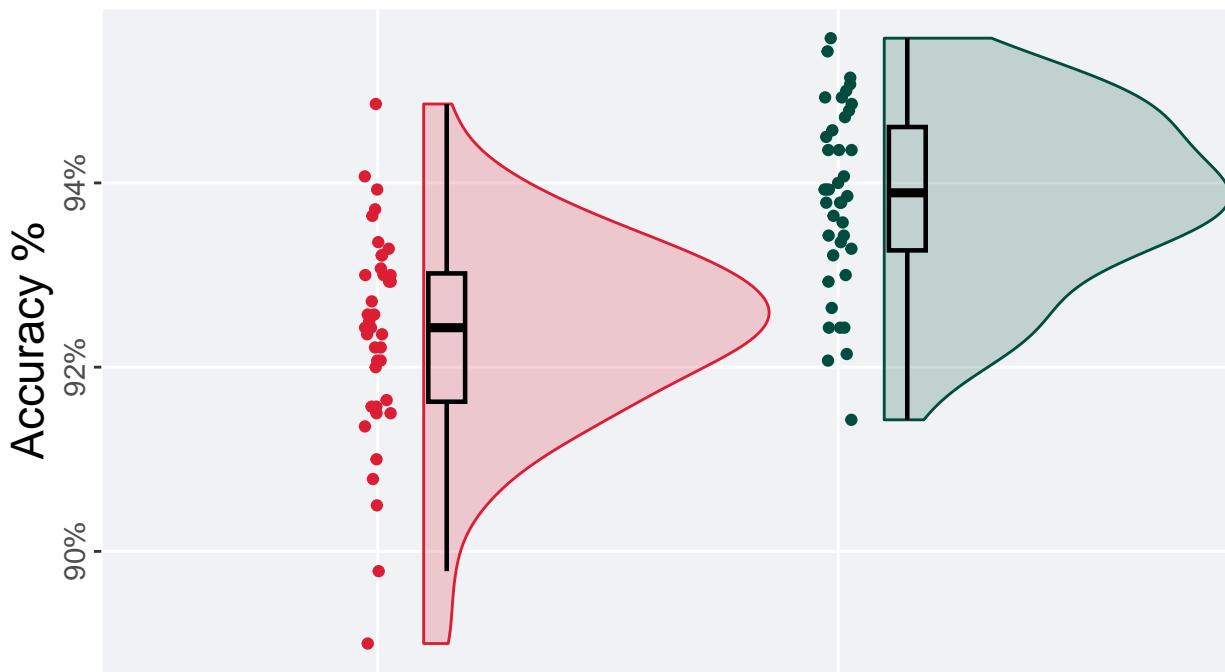
```
## [1] "observed_diff: -0.177387484050444"
## [1] "lower: -1.77714489478587"
## [1] "upper: 1.77714495225199"
## [1] "fail to reject null hypothesis"
## [1] "p-value: 0.912"
```



12.4.2 Validation accuracy

```
validation_plot(filter(task_data, split == '90%'))
```

Accuracy on validation set



Selection Scheme  tournament  lexicase

Summary statistics for the testing performance of the selection schemes at the 5% selection set split:

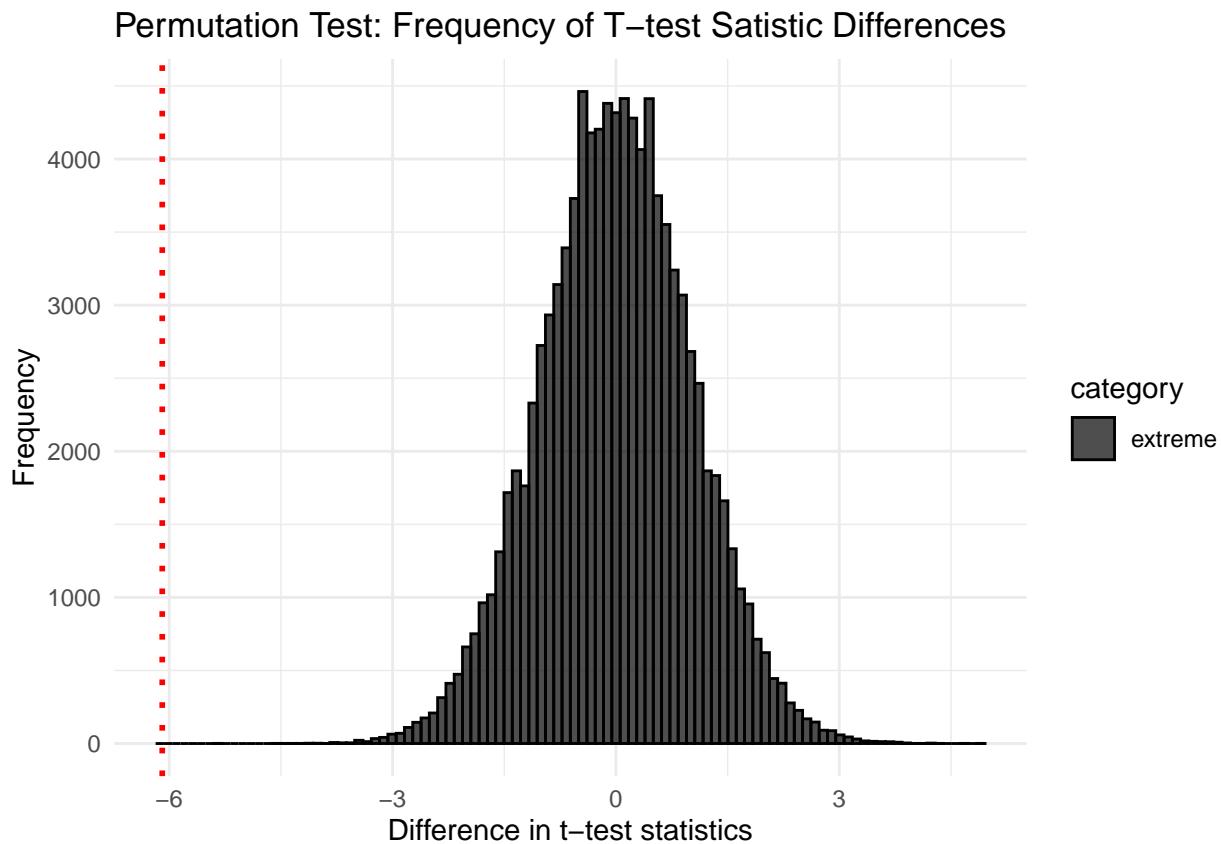
```
validation_accuracy_summary(filter(task_data, split == '90%'))
```

```
## # A tibble: 2 x 8
##   selection  count  na_cnt  min  median  mean  max    IQR
##   <fct>      <int>   <int>  <dbl>   <dbl>  <dbl> <dbl>  <dbl>
## 1 tournament    40      0  0.89   0.924  0.924  0.949  0.0139
## 2 lexicase      40      0  0.914  0.939  0.938  0.956  0.0134
```

The permutation test revealed that the results are:

```
tournament_results <- filter(task_data, split == '90%' & selection == 'tournament')
lexicase_results <- filter(task_data, split == '90%' & selection == 'lexicase')
permutation_test(tournament_results$training_performance,
                 lexicase_results$training_performance,
                 seed = 108,
                 alternative = "1")
```

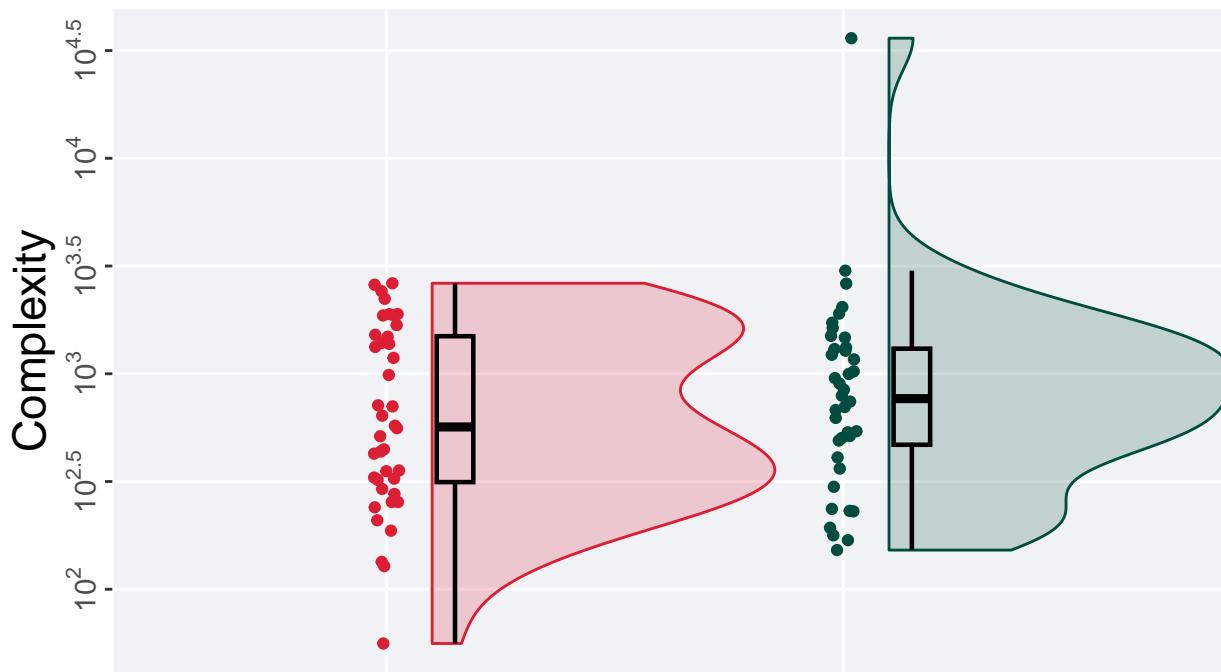
```
## [1] "observed_diff: -6.0941418442466"
## [1] "permutation_diffs[0.05 * n_permutations]: -1.66413613579426"
## [1] "reject null hypothesis"
## [1] "p-value: 1e-05"
```



12.4.3 Pipeline complexity

```
complexity_plot(filter(task_data, split == '90%'))
```

Pipeline Complexity



Selection Scheme tournament lexicase

Summary statistics for the pipeline complexity of the selection schemes at the 5% selection set split:

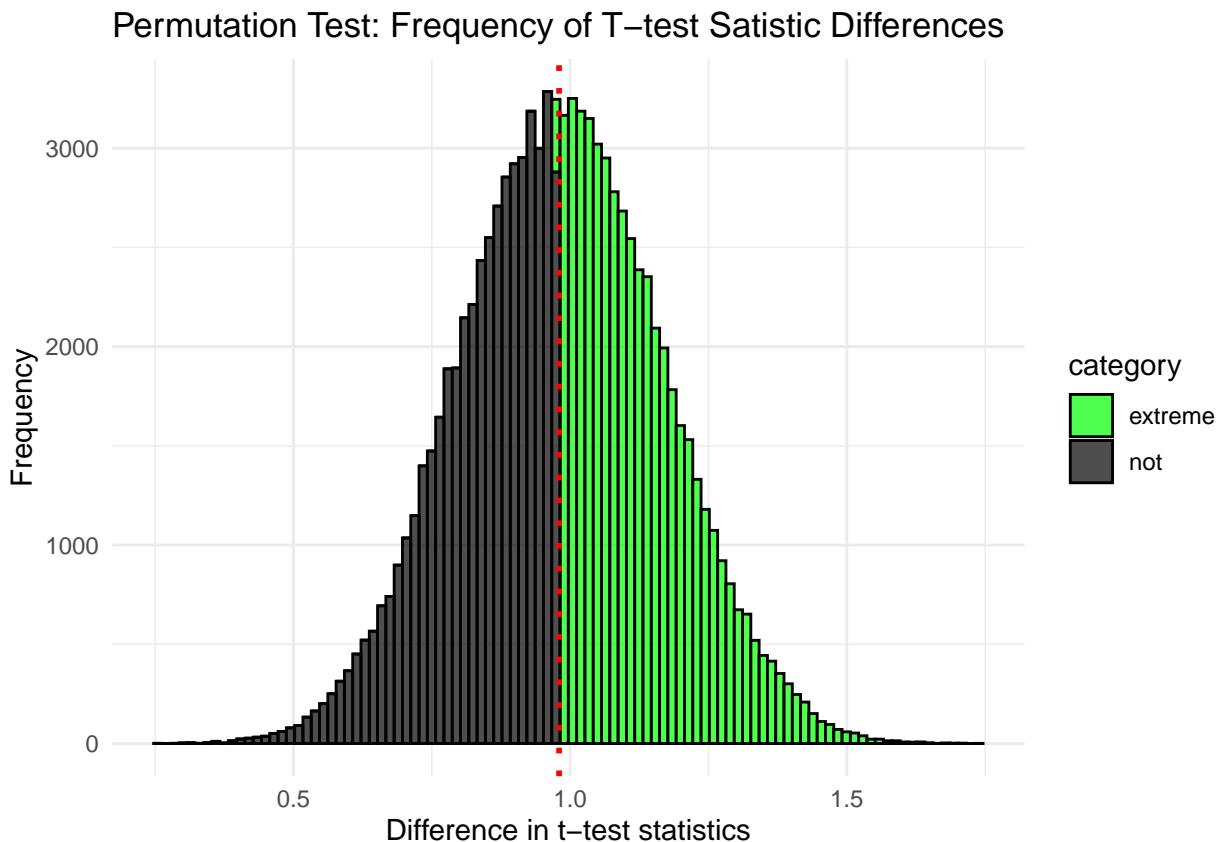
```
complexity_summary(filter(task_data, split == '90%'))
```

```
## # A tibble: 2 x 8
##   selection count na_cnt  min median  mean   max   IQR
##   <fct>     <int> <int> <dbl> <dbl> <dbl> <int> <dbl>
## 1 tournament     40     0    56   566.  927.  2631 1179.
## 2 lexicase       40     0   152   767  1803  36091  838
```

The permutation test revealed that the results are:

```
tournament_results <- filter(task_data, split == '90%' & selection == 'tournament')
lexicase_results <- filter(task_data, split == '90%' & selection == 'lexicase')
permutation_test(tournament_results$testing_complexity,
                 lexicase_results$testing_complexity,
                 seed = 241,
                 alternative = "t")
```

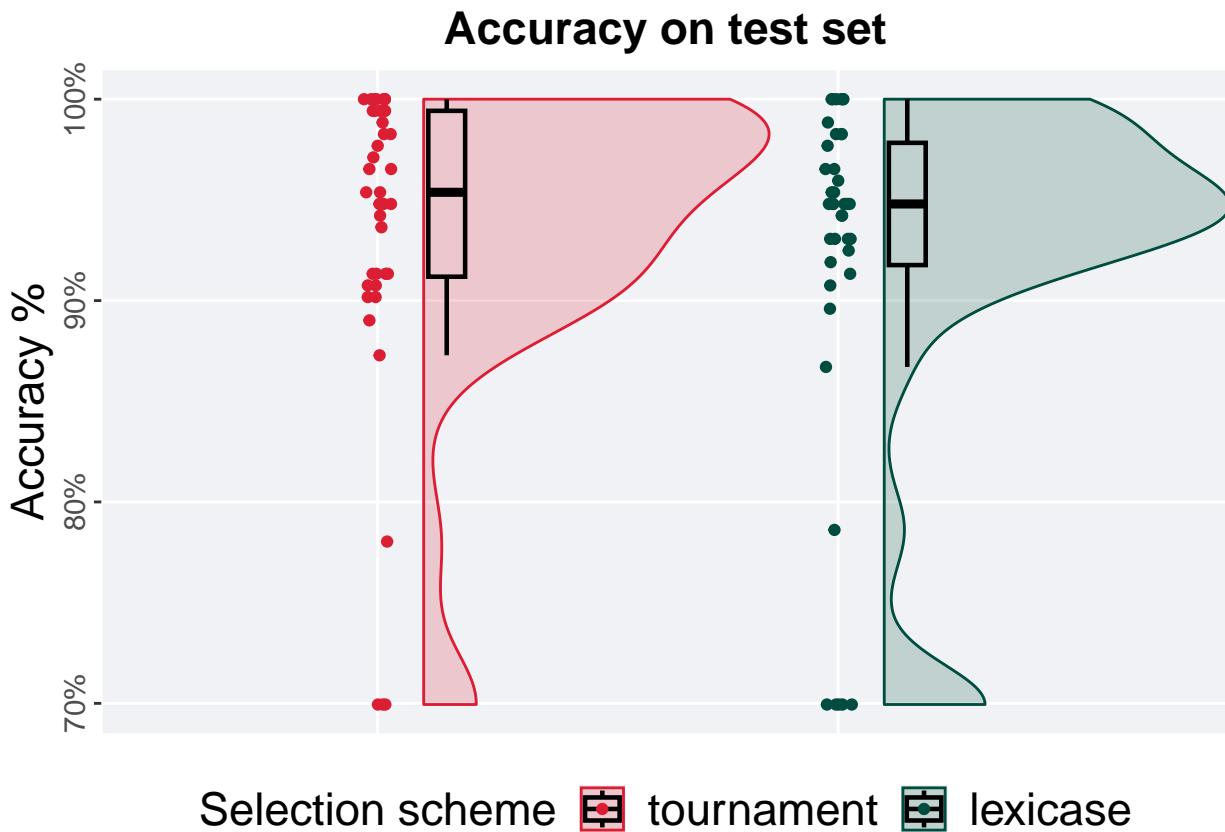
```
## [1] "observed_diff: -0.979988423917489"
## [1] "lower: -1.28648273891054"
## [1] "upper: 1.28791682709379"
## [1] "fail to reject null hypothesis"
## [1] "p-value: 0.50649"
```



12.5 95%

12.5.1 Test accuracy

```
test_plot(filter(task_data, split == '95%'))
```



Summary statistics for the testing performance of the selection schemes at the 5% selection set split:

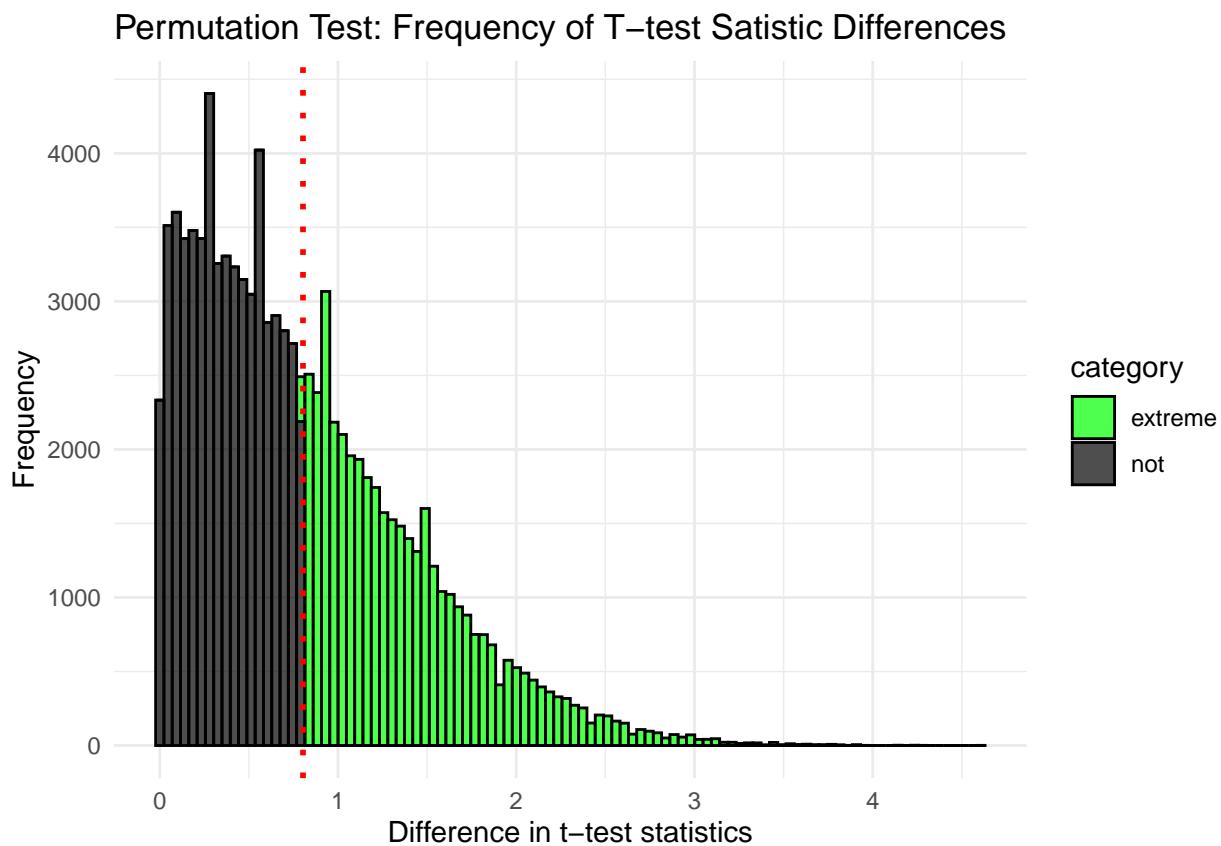
```
test_results_summary(filter(task_data, split == '95%'))
```

```
## # A tibble: 2 x 8
##   selection count na_cnt  min median  mean   max     IQR
##   <fct>     <int> <int> <dbl> <dbl> <dbl> <dbl> <dbl>
## 1 tournament     40     0 0.699 0.954 0.934     1 0.0824
## 2 lexicase       40     0 0.699 0.948 0.918     1 0.0607
```

The permutation test revealed that the results are:

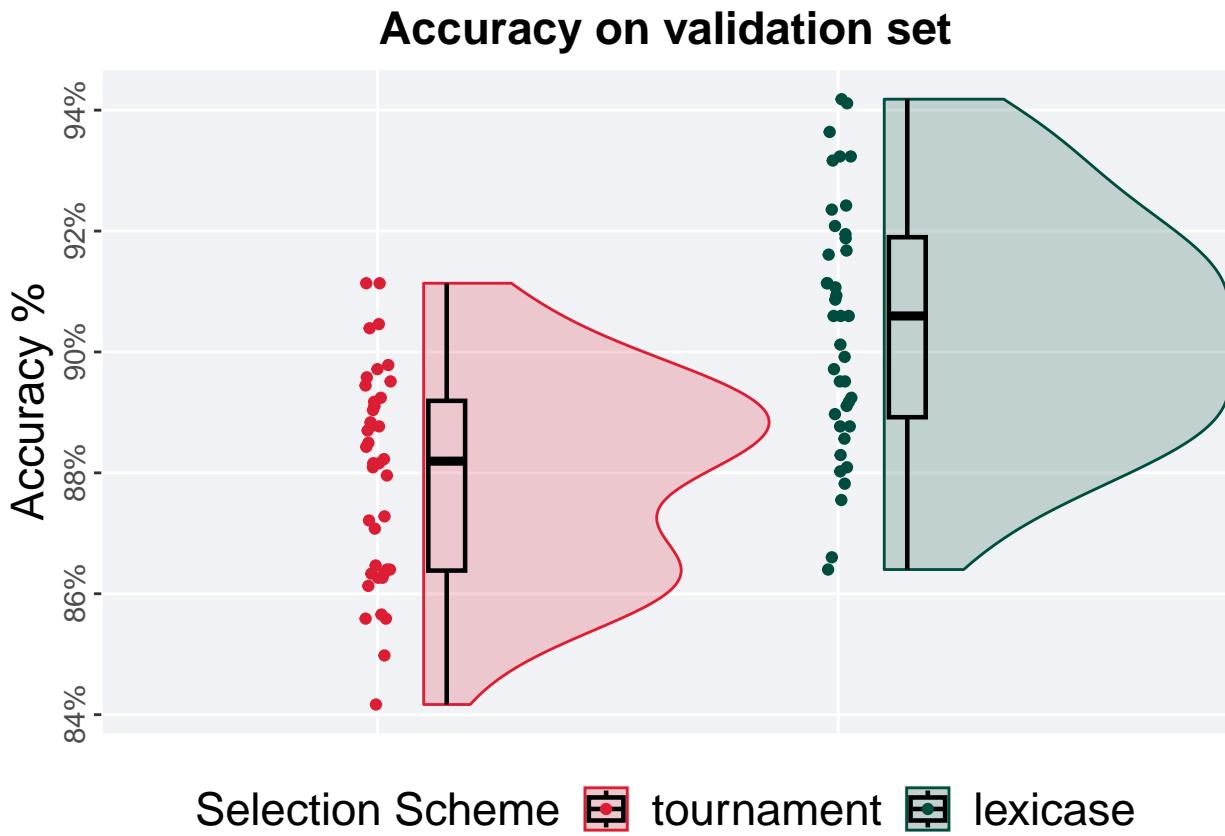
```
tournament_results <- filter(task_data, split == '95%' & selection == 'tournament')
lexicase_results <- filter(task_data, split == '95%' & selection == 'lexicase')
permutation_test(tournament_results$testing_performance,
                 lexicase_results$testing_performance,
                 seed = 109,
                 alternative = "t")
```

```
## [1] "observed_diff: 0.803639914004292"
## [1] "lower: -1.99477816242174"
## [1] "upper: 1.99477818979544"
## [1] "fail to reject null hypothesis"
## [1] "p-value: 0.4234"
```



12.5.2 Validation accuracy

```
validation_plot(filter(task_data, split == '95%'))
```



Summary statistics for the testing performance of the selection schemes at the 5% selection set split:

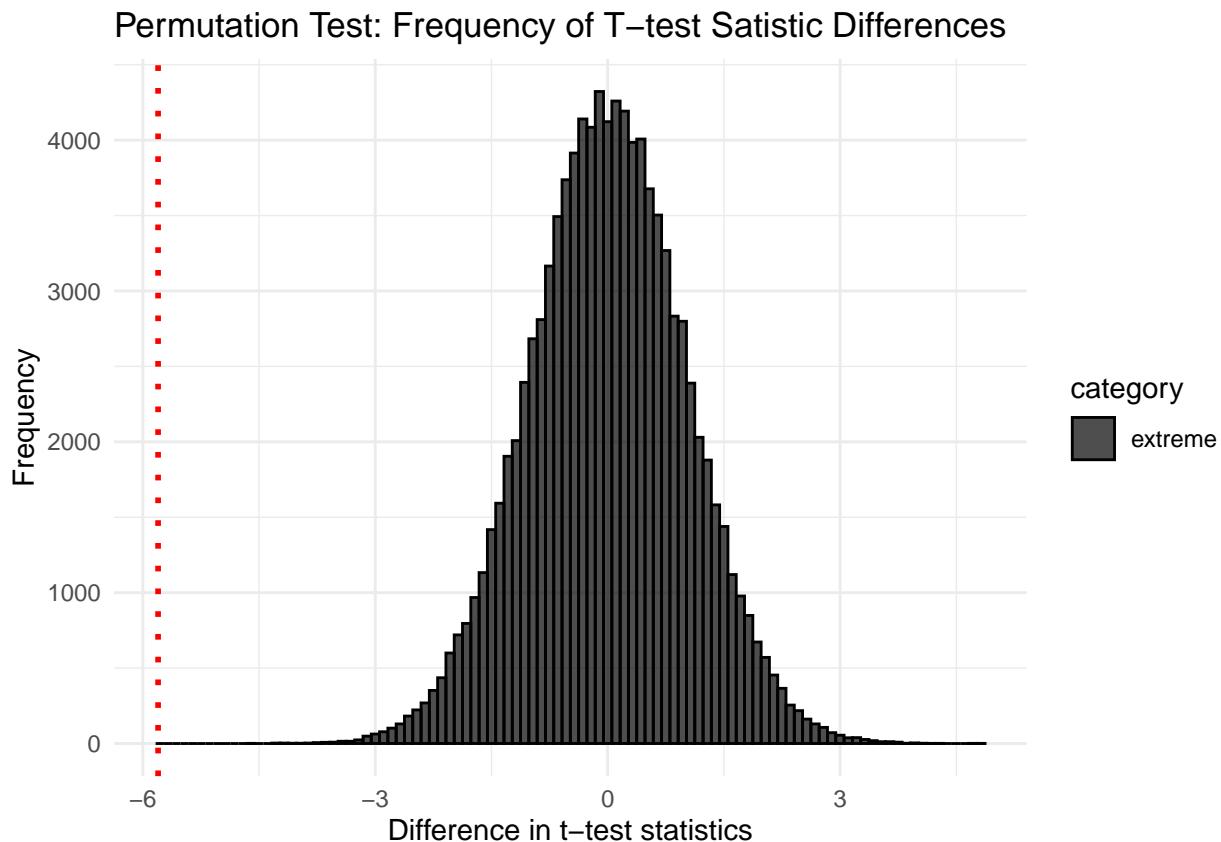
```
validation_accuracy_summary(filter(task_data, split == '95%'))
```

```
## # A tibble: 2 x 8
##   selection  count  na_cnt  min  median  mean   max    IQR
##   <fct>      <int>   <int>  <dbl>   <dbl>  <dbl> <dbl>   <dbl>
## 1 tournament    40      0  0.842  0.882  0.880  0.911  0.0281
## 2 lexicase      40      0  0.864  0.906  0.904  0.942  0.0298
```

The permutation test revealed that the results are:

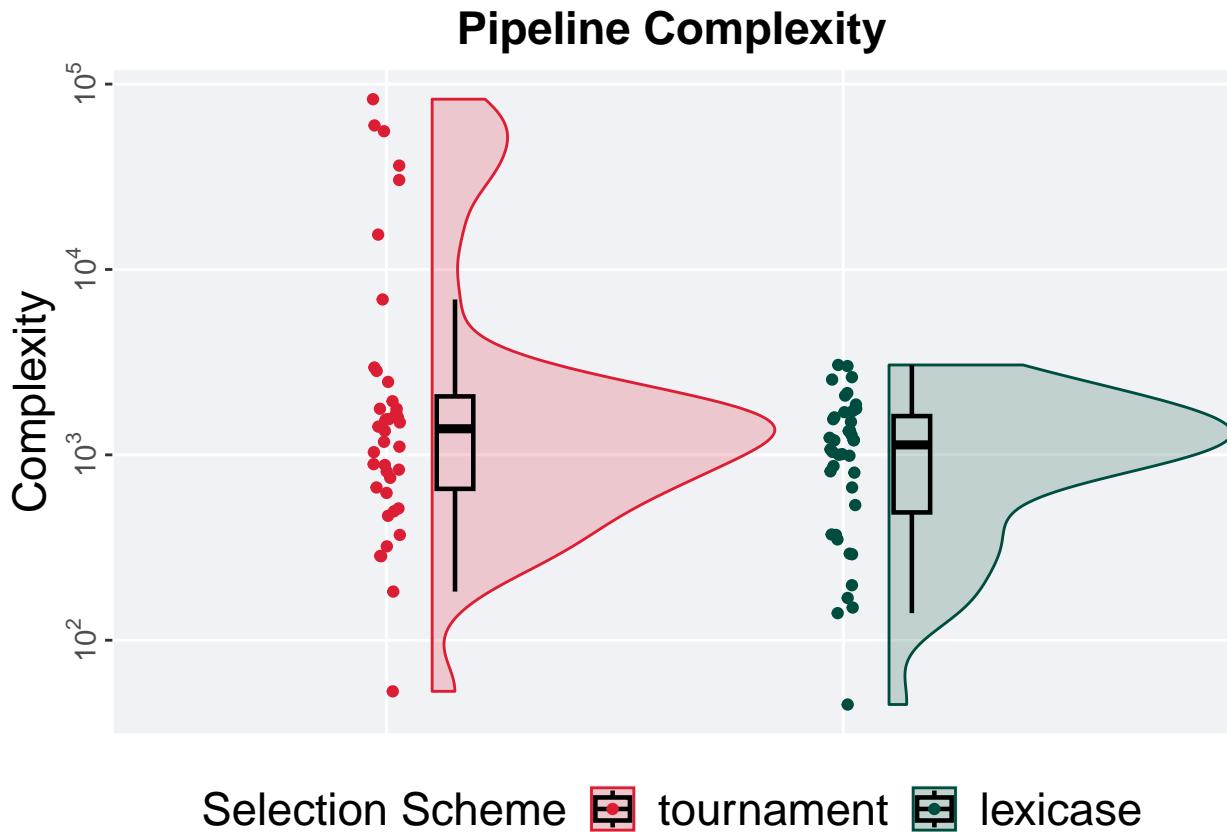
```
tournament_results <- filter(task_data, split == '95%' & selection == 'tournament')
lexicase_results <- filter(task_data, split == '95%' & selection == 'lexicase')
permutation_test(tournament_results$training_performance,
                 lexicase_results$training_performance,
                 seed = 110,
                 alternative = "1")
```

```
## [1] "observed_diff: -5.80303164089155"
## [1] "permutation_diffs[0.05 * n_permutations]: -1.66315447368821"
## [1] "reject null hypothesis"
## [1] "p-value: 1e-05"
```



12.5.3 Pipeline complexity

```
complexity_plot(filter(task_data, split == '95%'))
```



Summary statistics for the pipeline complexity of the selection schemes at the 5% selection set split:

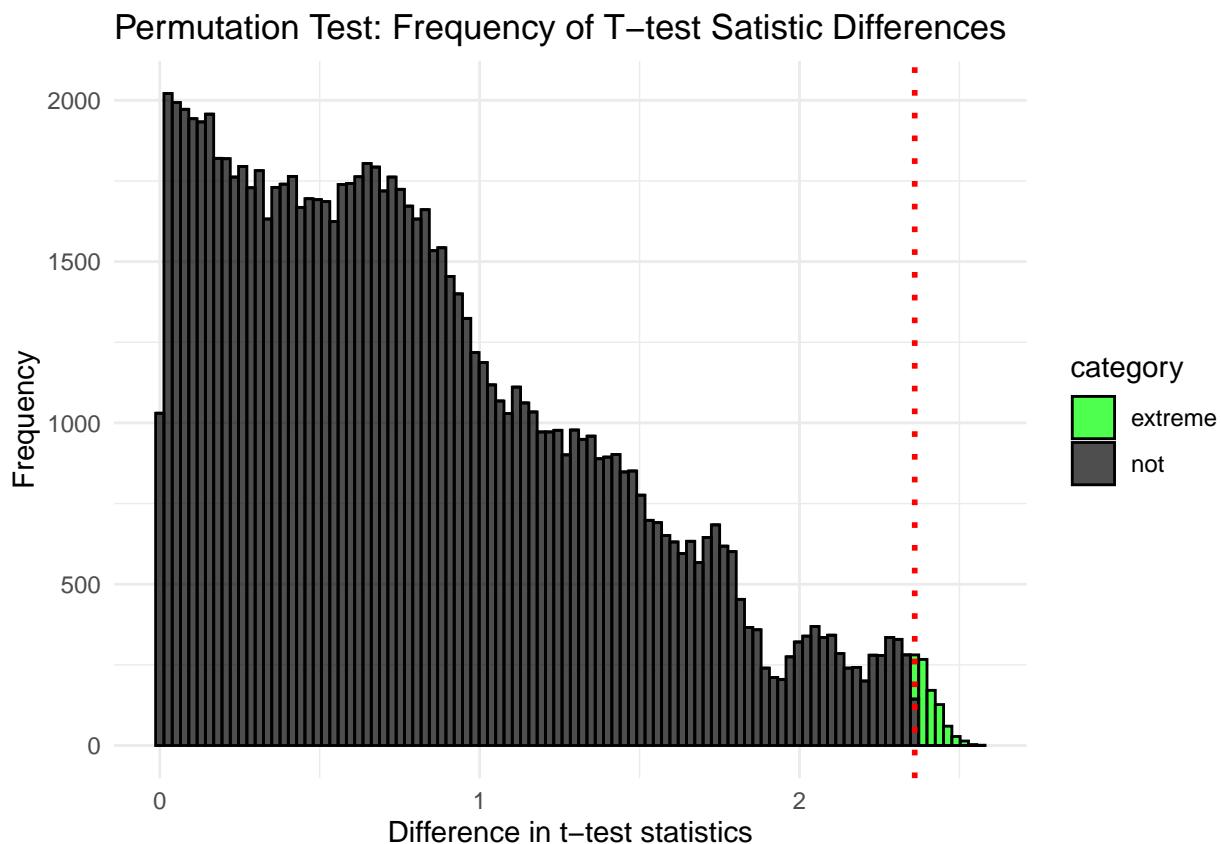
```
complexity_summary(filter(task_data, split == '95%'))
```

```
## # A tibble: 2 x 8
##   selection count na_cnt  min median  mean   max   IQR
##   <fct>     <int> <int> <dbl> <dbl> <dbl> <dbl>
## 1 tournament     40     0    53  1384.  8132. 82926 1423
## 2 lexicase       40     0    45  1134   1180.  3055 1124.
```

The permutation test revealed that the results are:

```
tournament_results <- filter(task_data, split == '95%' & selection == 'tournament')
lexicase_results <- filter(task_data, split == '95%' & selection == 'lexicase')
permutation_test(tournament_results$testing_complexity,
                 lexicase_results$testing_complexity,
                 seed = 242,
                 alternative = "t")
```

```
## [1] "observed_diff: 2.36021849572633"
## [1] "lower: -1.99771775061279"
## [1] "upper: 1.99052606641906"
## [1] "reject null hypothesis"
## [1] "p-value: 0.00808"
```



Chapter 13

Task 168784

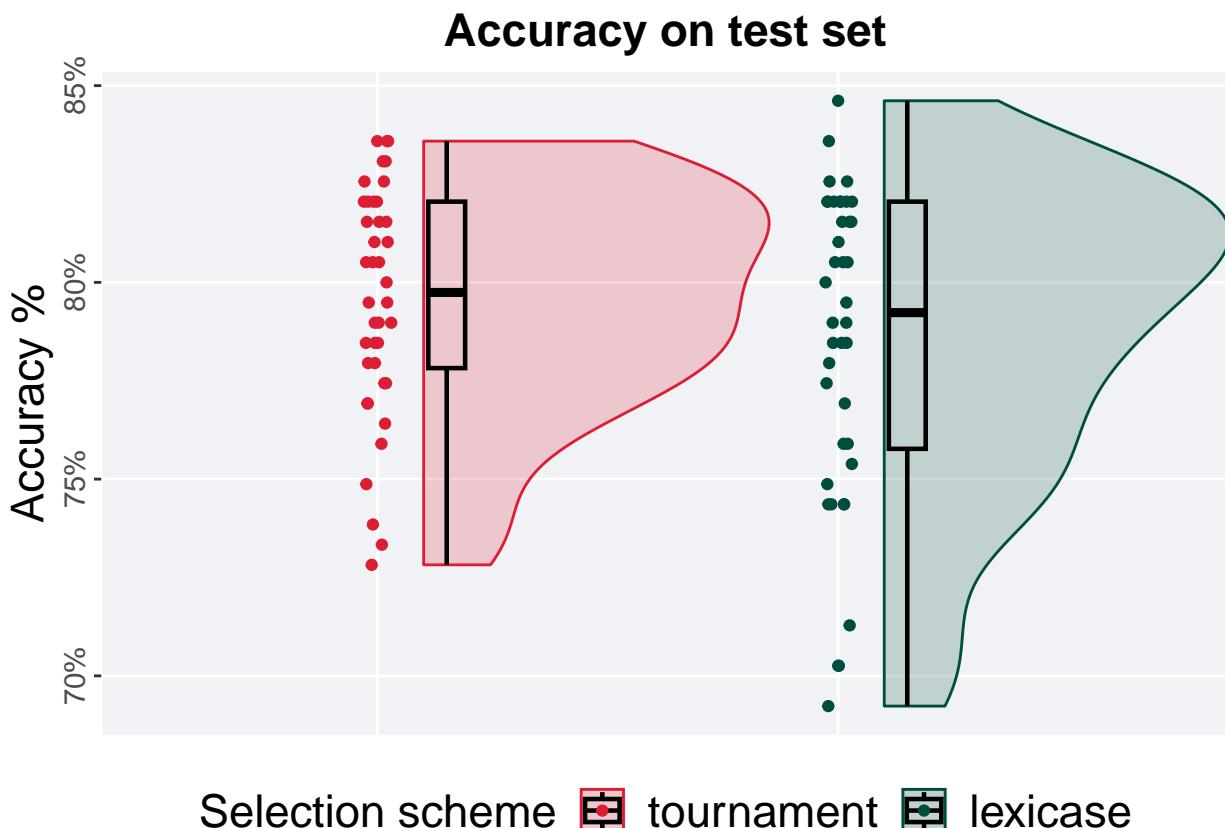
We present the results of our analysis of task 168784 with the different selection set splits used in our study.

```
task_data <- filter(results, task_id == 168784)
```

13.1 5%

13.1.1 Test accuracy

```
test_plot(filter(task_data, split == '5%'))
```



Summary statistics for the testing performance of the selection schemes at the 5% selection set split:

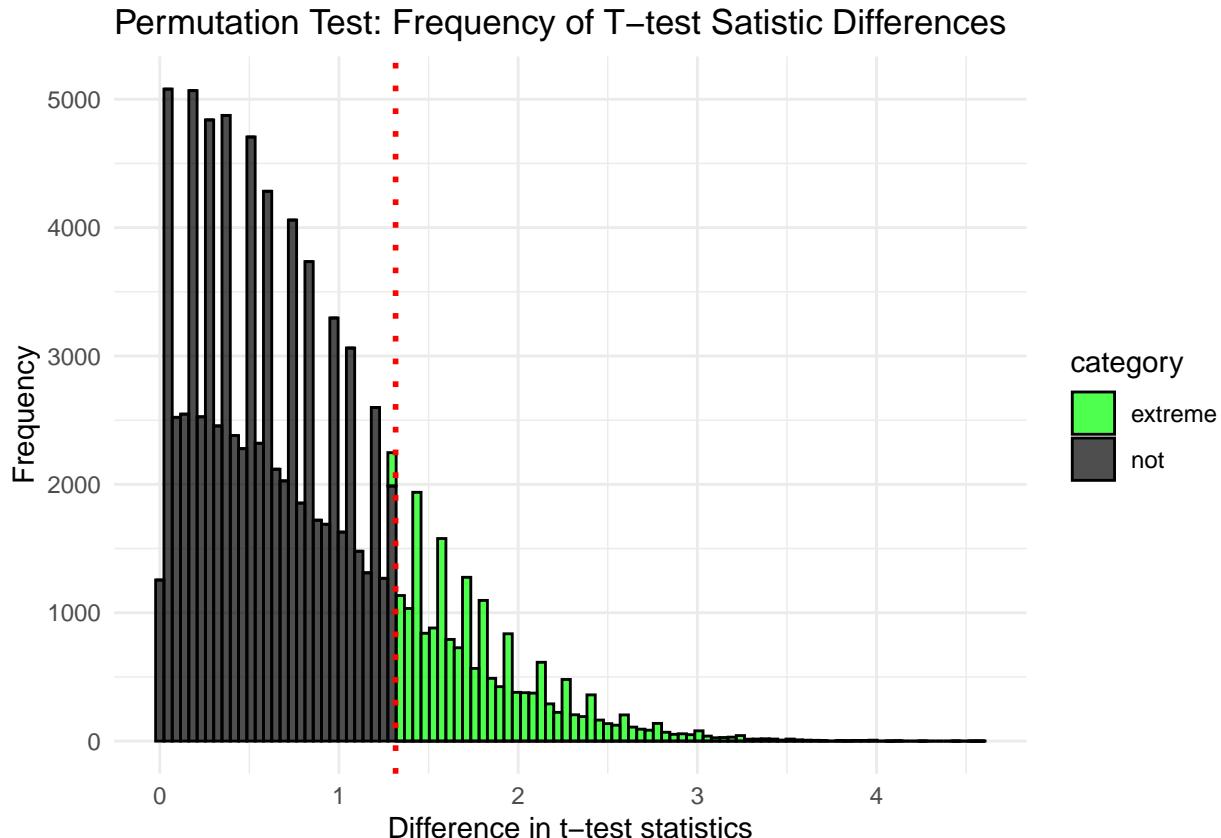
```
test_results_summary(filter(task_data, split == '5%'))
```

```
## # A tibble: 2 x 8
##   selection count na_cnt  min median  mean   max    IQR
##   <fct>     <int> <int> <dbl> <dbl> <dbl> <dbl> <dbl>
## 1 tournament     40     0 0.728 0.797 0.795 0.836 0.0423
## 2 lexicase       40     0 0.692 0.792 0.785 0.846 0.0628
```

The permutation test revealed that the results are:

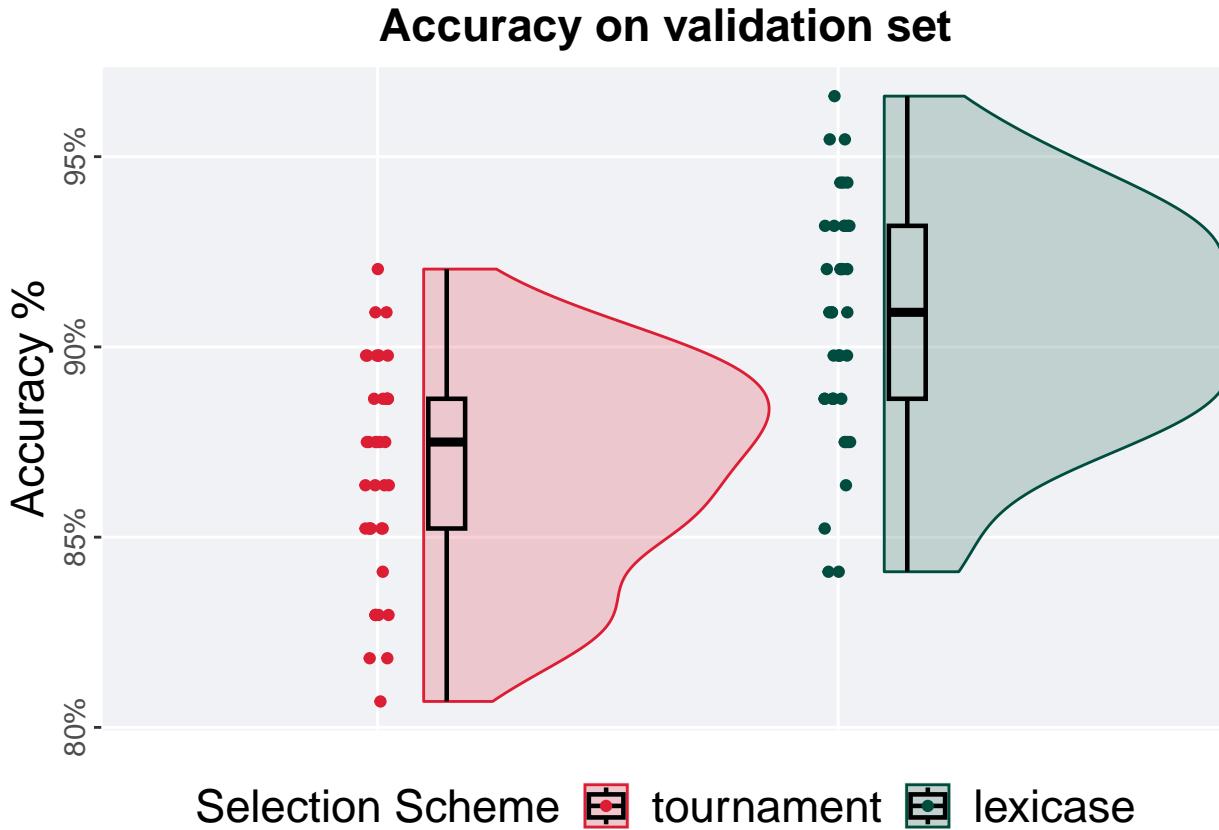
```
tournament_results <- filter(task_data, split == '5%' & selection == 'tournament')
lexicase_results <- filter(task_data, split == '5%' & selection == 'lexicase')
permutation_test(tournament_results$testing_performance,
                 lexicase_results$testing_performance,
                 seed = 111,
                 alternative = "t")
```

```
## [1] "observed_diff: 1.31597665591326"
## [1] "lower: -2.00194003515967"
## [1] "upper: 2.00194017197189"
## [1] "fail to reject null hypothesis"
## [1] "p-value: 0.19036"
```



13.1.2 Validation accuracy

```
validation_plot(filter(task_data, split == '5%'))
```



Summary statistics for the testing performance of the selection schemes at the 5% selection set split:

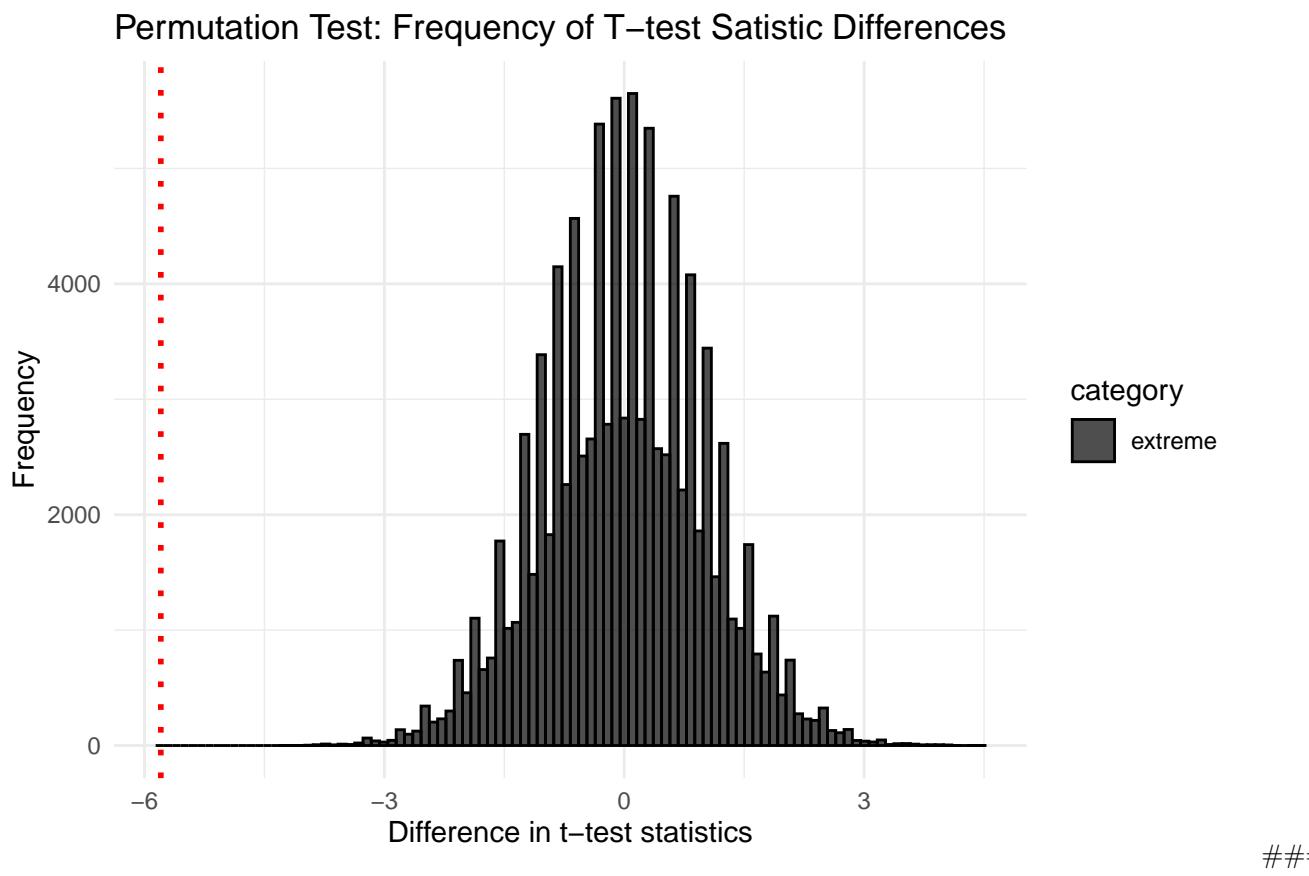
```
validation_accuracy_summary(filter(task_data, split == '5%'))
```

```
## # A tibble: 2 x 8
##   selection count na_cnt  min median  mean   max     IQR
##   <fct>     <int> <int> <dbl> <dbl> <dbl> <dbl> <dbl>
## 1 tournament     40     0 0.807 0.875 0.867 0.920 0.0341
## 2 lexicase       40     0 0.841 0.909 0.906 0.966 0.0455
```

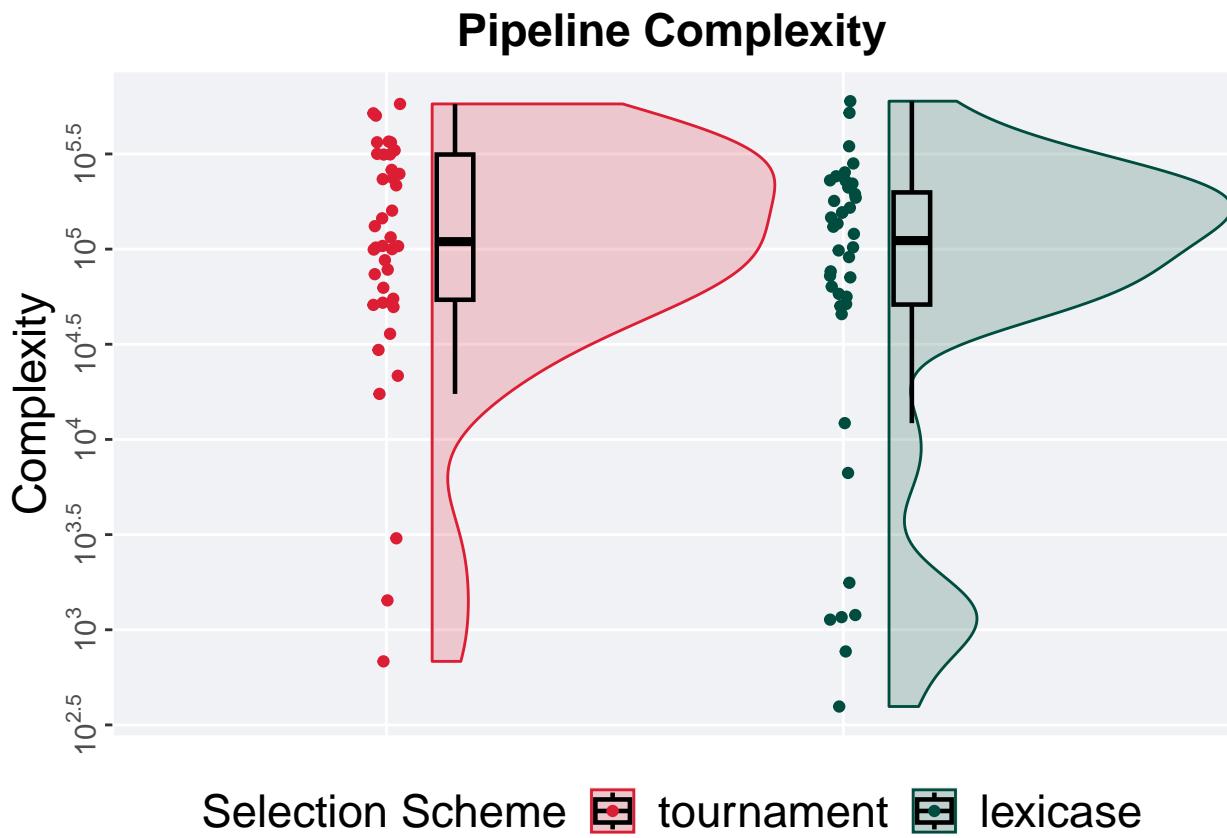
The permutation test revealed that the results are:

```
tournament_results <- filter(task_data, split == '5%' & selection == 'tournament')
lexicase_results <- filter(task_data, split == '5%' & selection == 'lexicase')
permutation_test(tournament_results$training_performance,
                 lexicase_results$training_performance,
                 seed = 112,
                 alternative = "1")
```

```
## [1] "observed_diff: -5.79635757388812"
## [1] "permutation_diffs[0.05 * n_permutations]: -1.66803276720504"
## [1] "reject null hypothesis"
## [1] "p-value: 1e-05"
```



```
complexity_plot(filter(task_data, split == '5%'))
```



Summary statistics for the pipeline complexity of the selection schemes at the 5% selection set split:

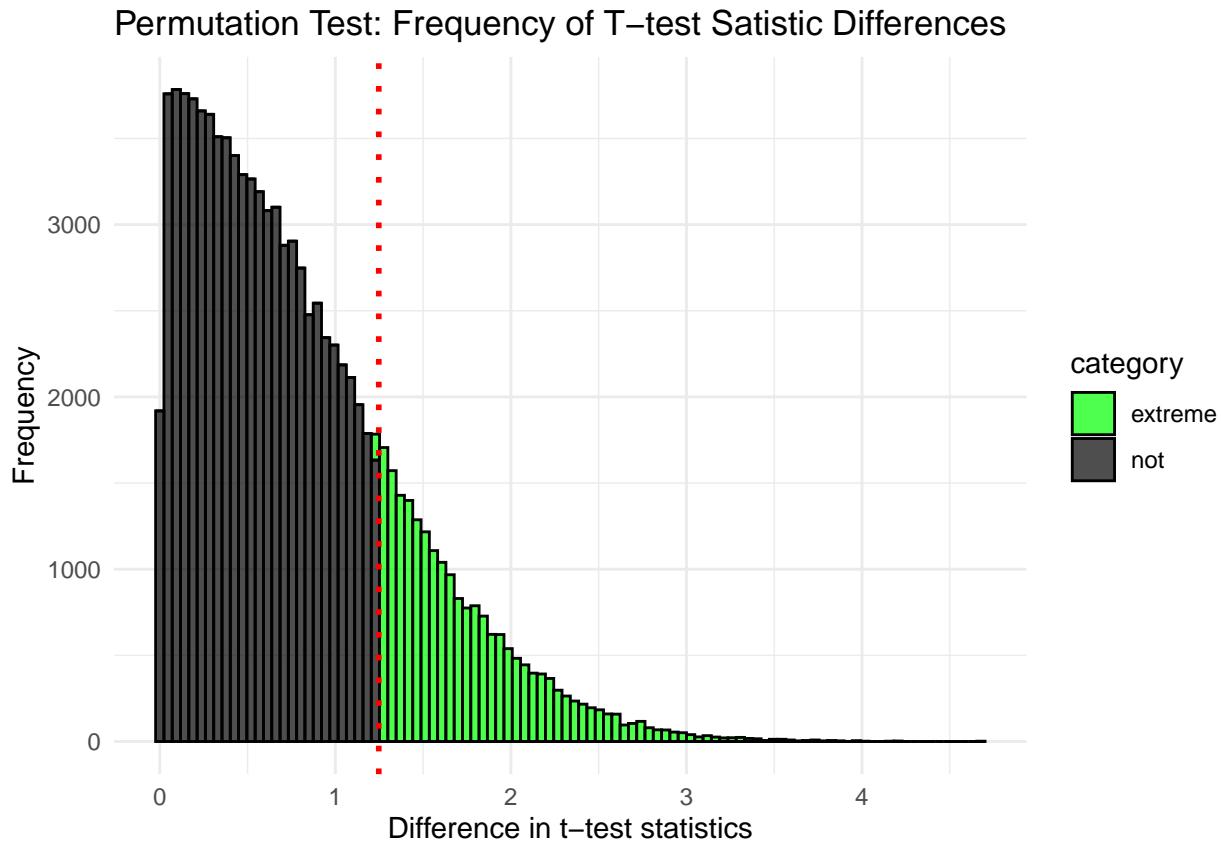
```
complexity_summary(filter(task_data, split == '5%'))
```

```
## # A tibble: 2 x 8
##   selection count na_cnt   min   median     mean     max     IQR
##   <fct>     <int> <int> <dbl>   <dbl>   <dbl>   <dbl>
## 1 tournament     40     0  682 109626. 179412. 579281 259926.
## 2 lexicase       40     0  395 111254. 139120. 598851 147555
```

The permutation test revealed that the results are:

```
tournament_results <- filter(task_data, split == '5%' & selection == 'tournament')
lexicase_results <- filter(task_data, split == '5%' & selection == 'lexicase')
permutation_test(tournament_results$testing_complexity,
                 lexicase_results$testing_complexity,
                 seed = 243,
                 alternative = "t")
```

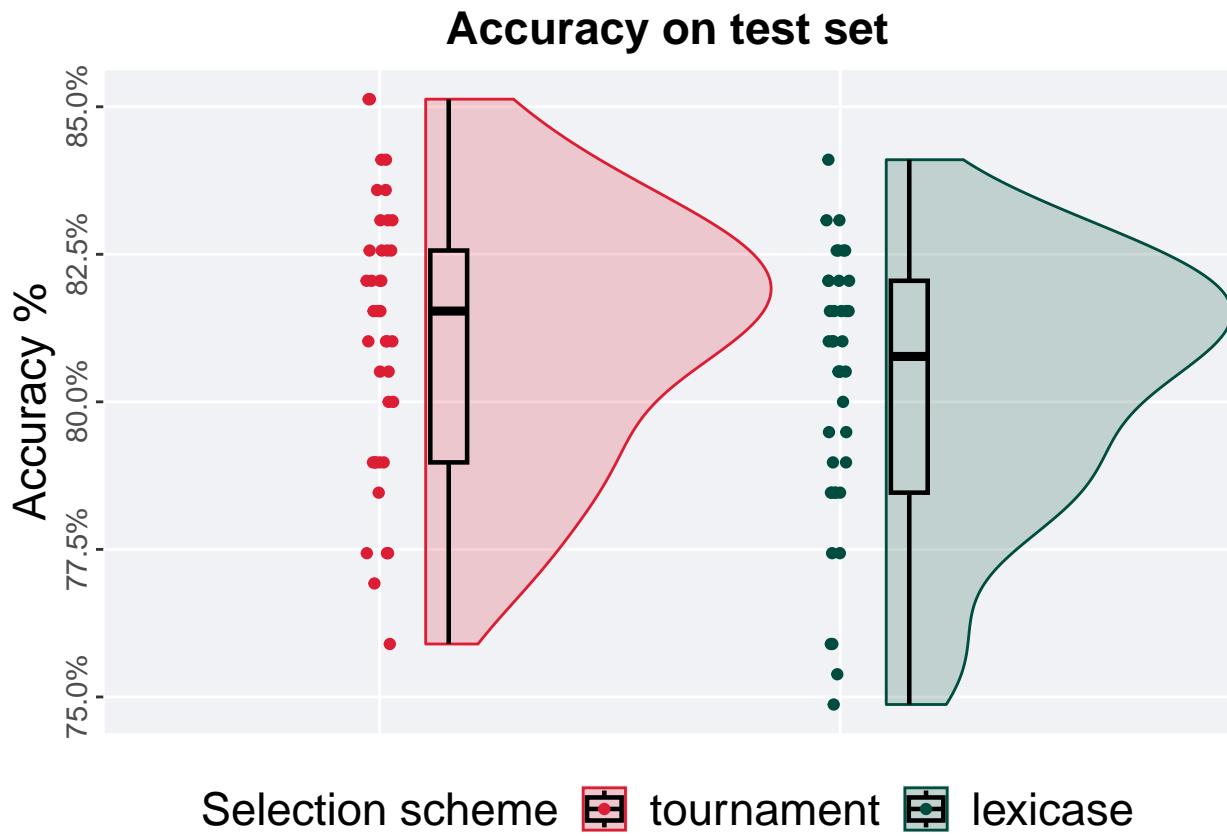
```
## [1] "observed_diff: 1.24740815644827"
## [1] "lower: -1.99242273351428"
## [1] "upper: 1.97905452947959"
## [1] "fail to reject null hypothesis"
## [1] "p-value: 0.21532"
```



13.2 10%

13.2.1 Test accuracy

```
test_plot(filter(task_data, split == '10%'))
```



Summary statistics for the testing performance of the selection schemes at the 5% selection set split:

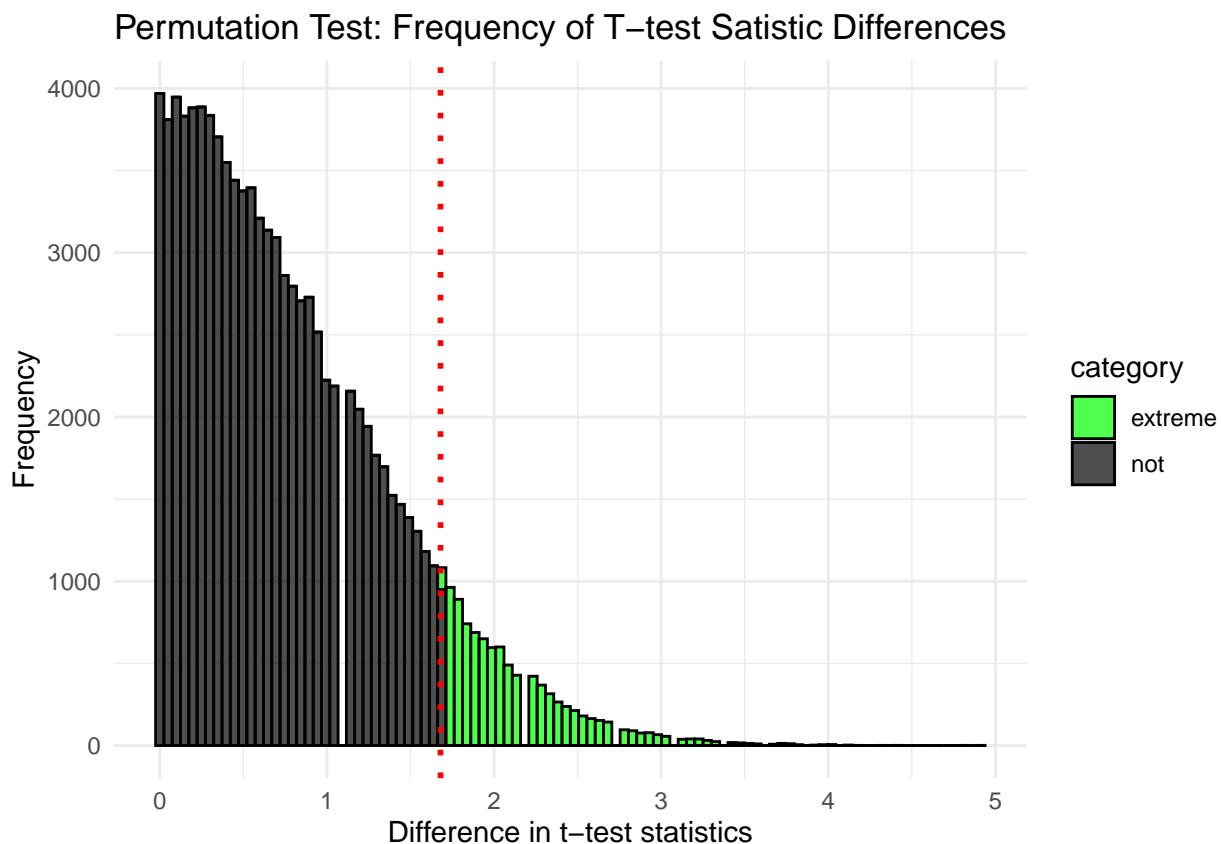
```
test_results_summary(filter(task_data, split == '10%'))
```

```
## # A tibble: 2 x 8
##   selection count na_cnt  min median  mean   max     IQR
##   <fct>     <int> <int> <dbl> <dbl> <dbl> <dbl> <dbl>
## 1 tournament     40     0 0.759 0.815 0.811 0.851 0.0359
## 2 lexicase       40     0 0.749 0.808 0.802 0.841 0.0359
```

The permutation test revealed that the results are:

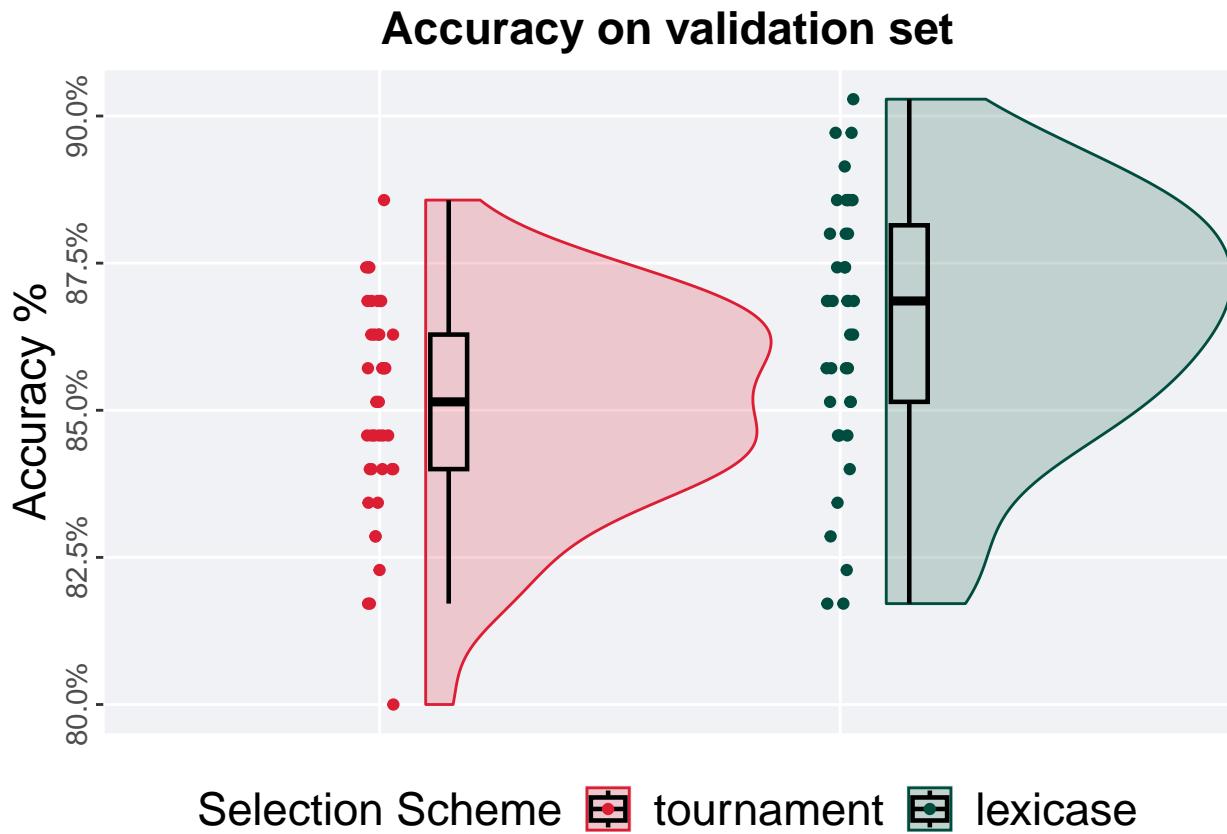
```
tournament_results <- filter(task_data, split == '10%' & selection == 'tournament')
lexicase_results <- filter(task_data, split == '10%' & selection == 'lexicase')
permutation_test(tournament_results$testing_performance,
                 lexicase_results$testing_performance,
                 seed = 113,
                 alternative = "t")
```

```
## [1] "observed_diff: 1.6796503157697"
## [1] "lower: -1.99461117576958"
## [1] "upper: 1.99461096878831"
## [1] "fail to reject null hypothesis"
## [1] "p-value: 0.09392"
```



13.2.2 Validation accuracy

```
validation_plot(filter(task_data, split == '10%'))
```



Summary statistics for the testing performance of the selection schemes at the 5% selection set split:

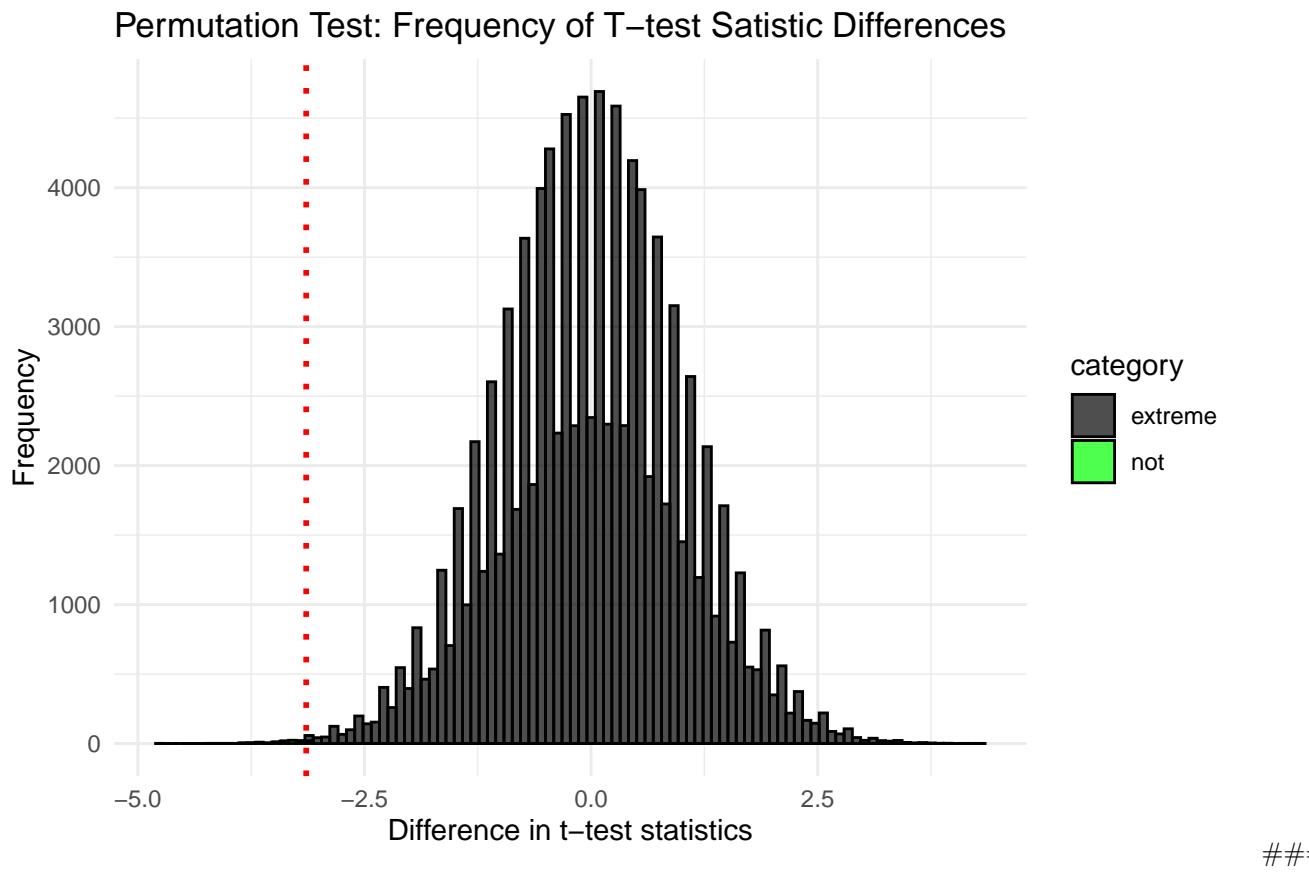
```
validation_accuracy_summary(filter(task_data, split == '10%'))
```

```
## # A tibble: 2 x 8
##   selection count na_cnt  min median  mean   max   IQR
##   <fct>     <int>  <int> <dbl>  <dbl> <dbl> <dbl> <dbl>
## 1 tournament     40      0  0.8   0.851  0.850  0.886  0.0229
## 2 lexicase       40      0  0.817  0.869  0.865  0.903  0.0300
```

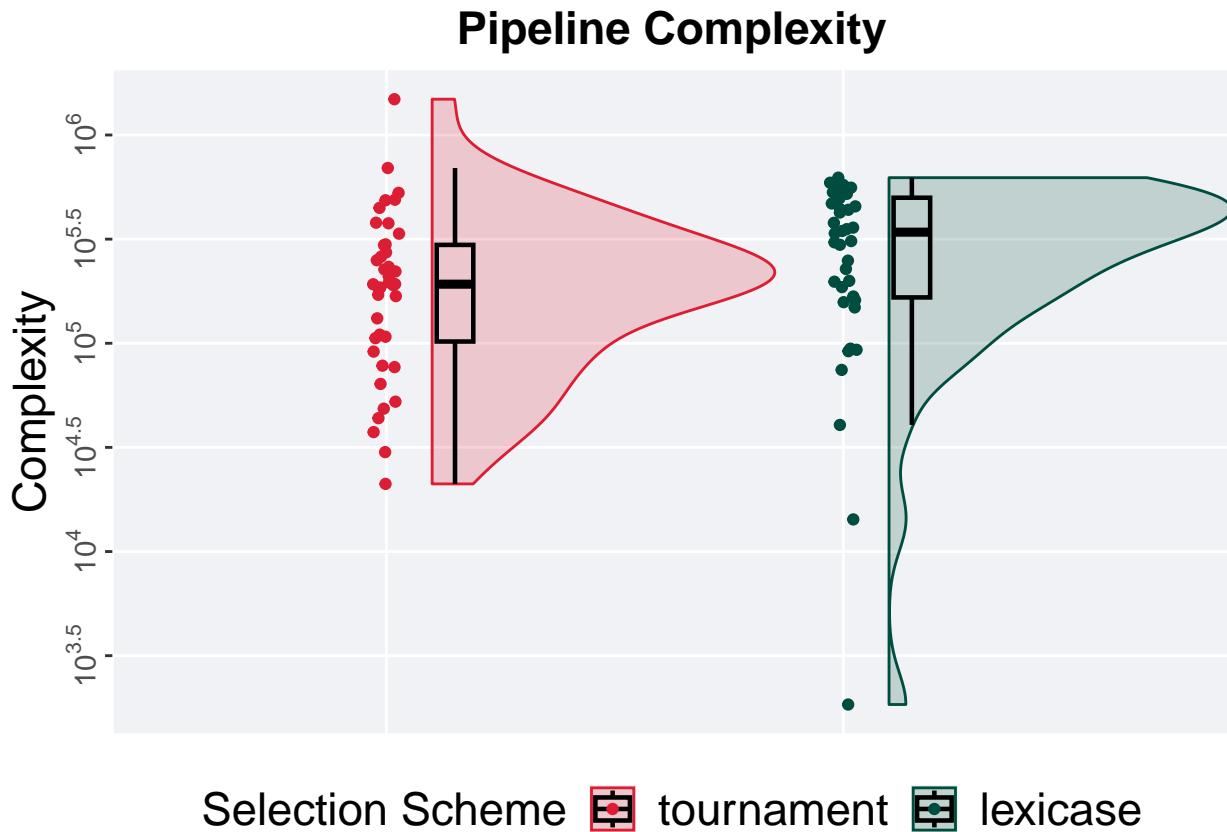
The permutation test revealed that the results are:

```
tournament_results <- filter(task_data, split == '10%' & selection == 'tournament')
lexicase_results <- filter(task_data, split == '10%' & selection == 'lexicase')
permutation_test(tournament_results$training_performance,
                 lexicase_results$training_performance,
                 seed = 114,
                 alternative = "1")
```

```
## [1] "observed_diff: -3.1433131617963"
## [1] "permutation_diffs[0.05 * n_permutations]: -1.68838346854448"
## [1] "reject null hypothesis"
## [1] "p-value: 0.00117"
```



```
complexity_plot(filter(task_data, split == '10%'))
```



Summary statistics for the pipeline complexity of the selection schemes at the 5% selection set split:

```
complexity_summary(filter(task_data, split == '10%'))
```

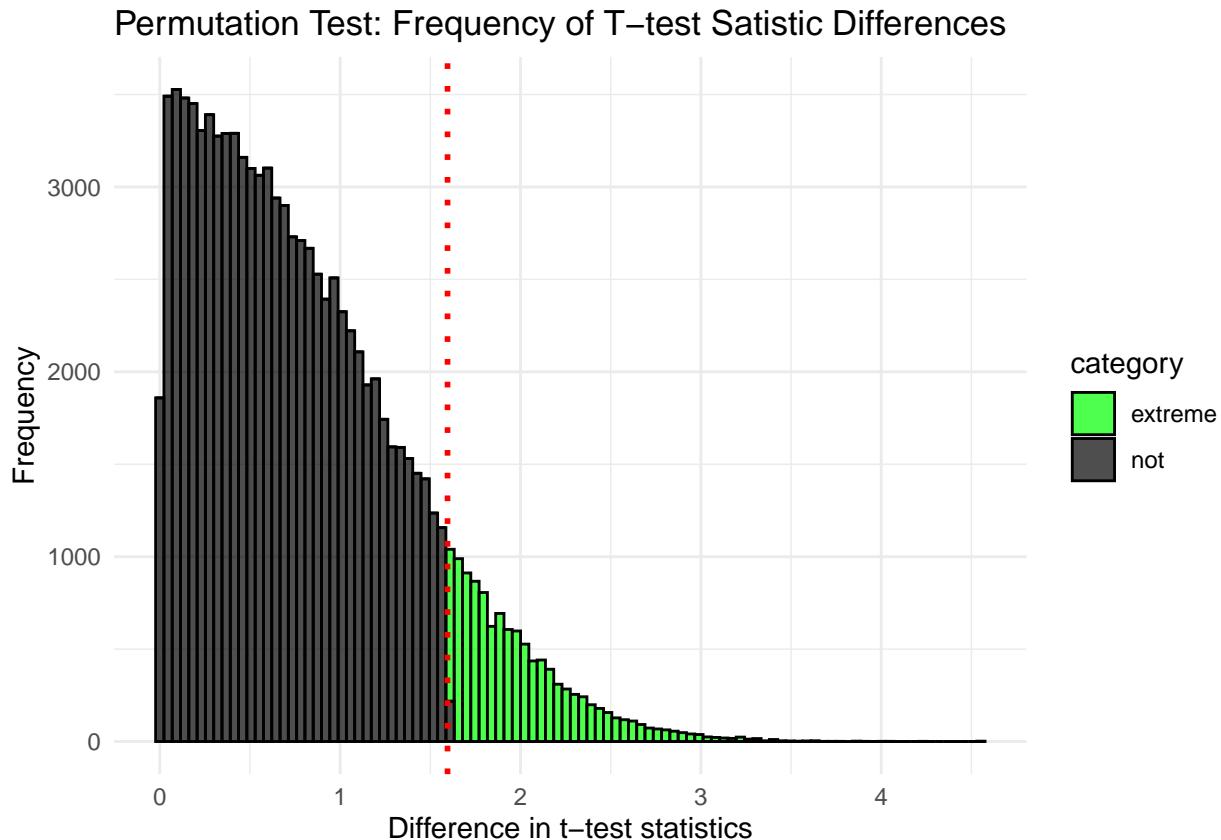
```
## # A tibble: 2 x 8
##   selection count na_cnt  min median   mean     max     IQR
##   <fct>     <int> <int> <dbl> <dbl>   <dbl> <int>   <dbl>
## 1 tournament     40     0 21125 192180 249169. 1485571 194540.
## 2 lexicase       40     0 1843 341611 328614.  623961 333838.
```

The permutation test revealed that the results are:

```
tournament_results <- filter(task_data, split == '10%' & selection == 'tournament')
lexicase_results <- filter(task_data, split == '10%' & selection == 'lexicase')
```

```
permutation_test(tournament_results$testing_complexity,
                  lexicase_results$testing_complexity,
                  seed = 244,
                  alternative = "t")
```

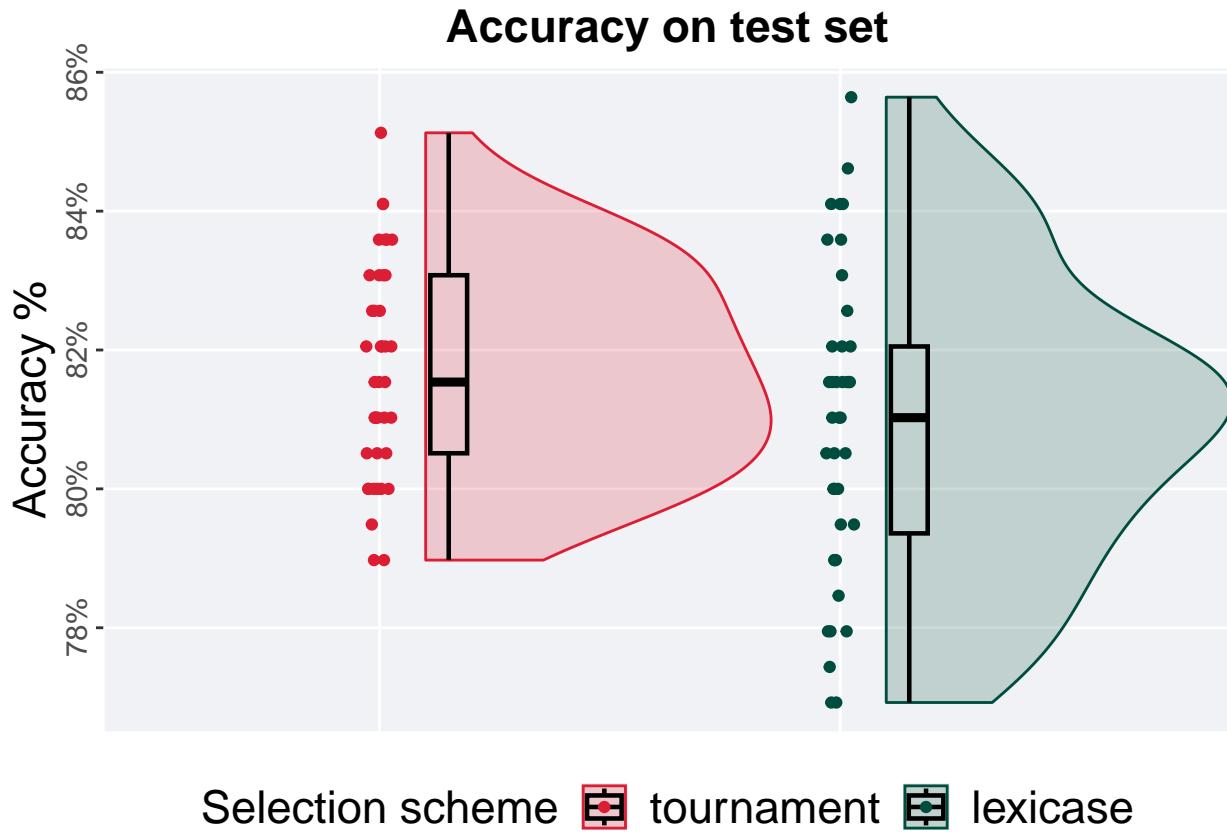
```
## [1] "observed_diff: -1.59530594860251"
## [1] "lower: -1.96562926914469"
## [1] "upper: 1.9472021242306"
## [1] "fail to reject null hypothesis"
## [1] "p-value: 0.11351"
```



13.3 50%

13.3.1 Test accuracy

```
test_plot(filter(task_data, split == '50%'))
```



Summary statistics for the testing performance of the selection schemes at the 5% selection set split:

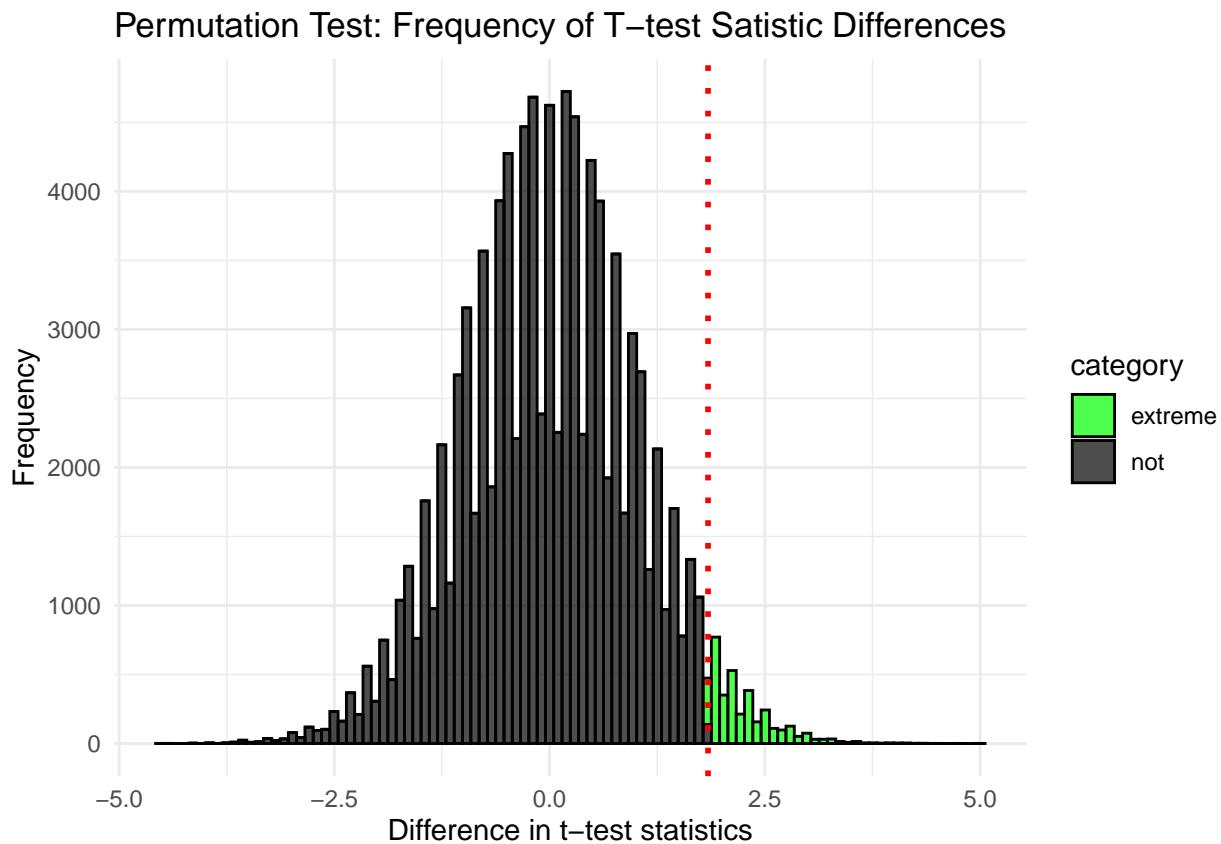
```
test_results_summary(filter(task_data, split == '50%'))
```

```
## # A tibble: 2 x 8
##   selection  count na_cnt   min median   mean   max   IQR
##   <fct>      <int> <int> <dbl> <dbl> <dbl> <dbl> <dbl>
## 1 tournament    40     0 0.790  0.815  0.817  0.851  0.0256
## 2 lexicase      40     0 0.769  0.810  0.809  0.856  0.0269
```

The permutation test revealed that the results are:

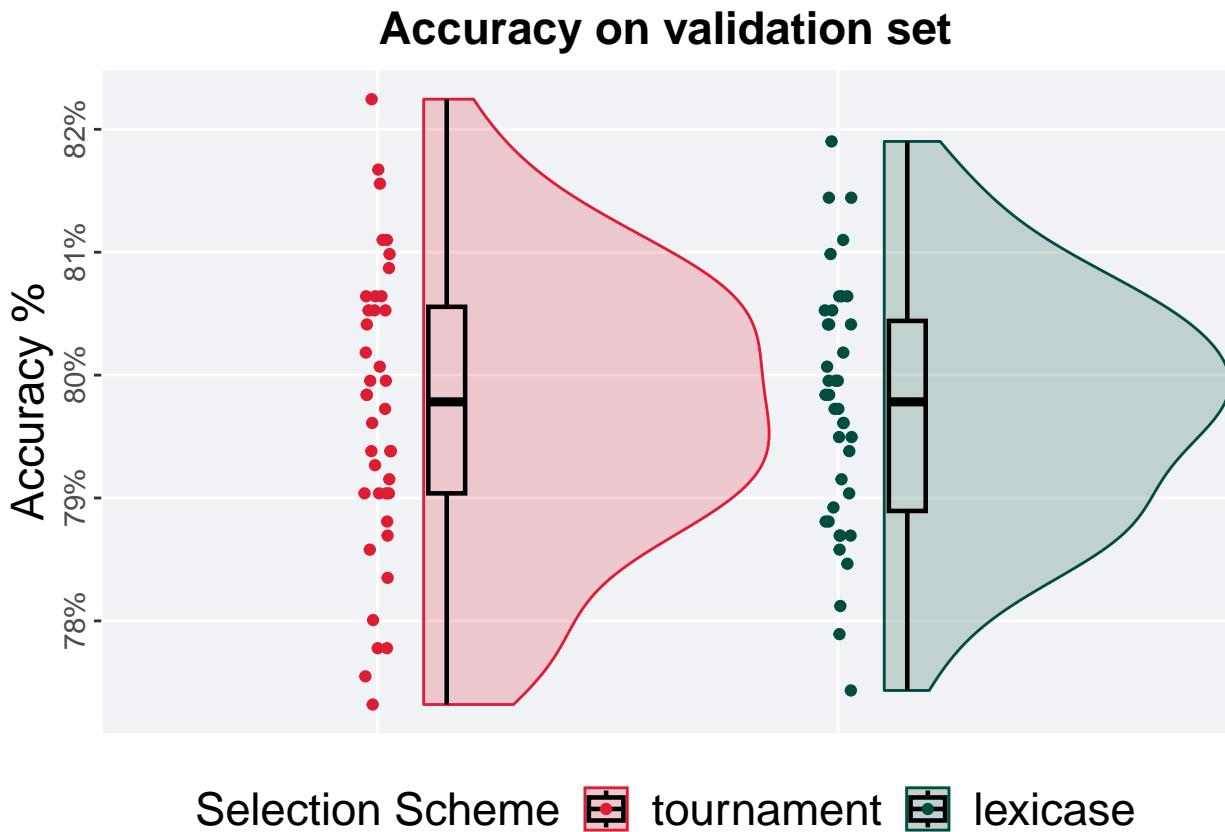
```
tournament_results <- filter(task_data, split == '50%' & selection == 'tournament')
lexicase_results <- filter(task_data, split == '50%' & selection == 'lexicase')
permutation_test(tournament_results$testing_performance,
                 lexicase_results$testing_performance,
                 seed = 115,
                 alternative = "g")
```

```
## [1] "observed_diff: 1.84018428427782"
## [1] "permutation_diffs[0.95 * n_permutations]: 1.65248732220136"
## [1] "reject null hypothesis"
## [1] "p-value: 0.03586"
```



13.3.2 Validation accuracy

```
validation_plot(filter(task_data, split == '50%'))
```



Summary statistics for the testing performance of the selection schemes at the 5% selection set split:

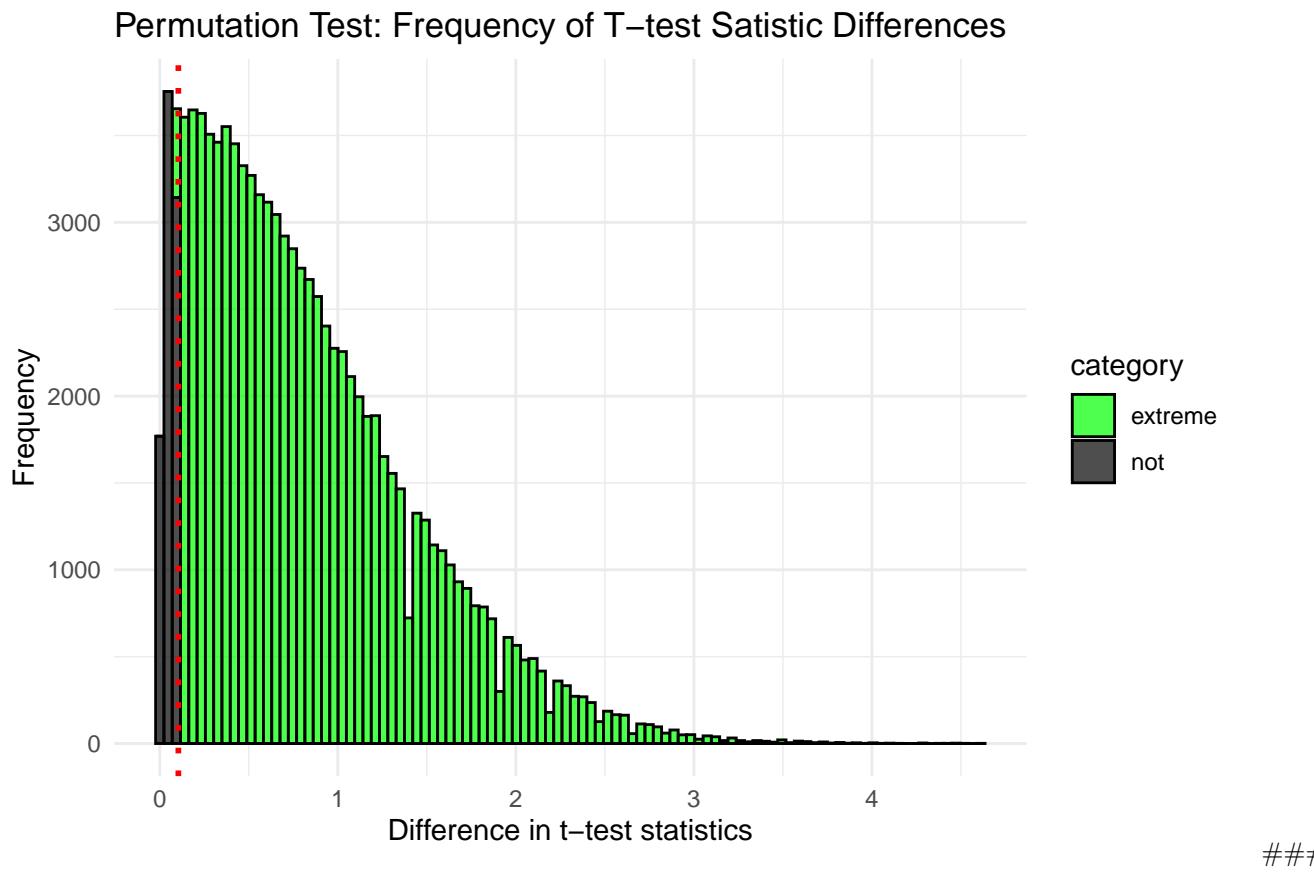
```
validation_accuracy_summary(filter(task_data, split == '50%'))
```

```
## # A tibble: 2 x 8
##   selection count na_cnt  min median  mean  max    IQR
##   <fct>     <int> <int> <dbl> <dbl> <dbl> <dbl> <dbl>
## 1 tournament     40     0 0.773 0.798 0.797 0.822 0.0152
## 2 lexicase       40     0 0.774 0.798 0.797 0.819 0.0155
```

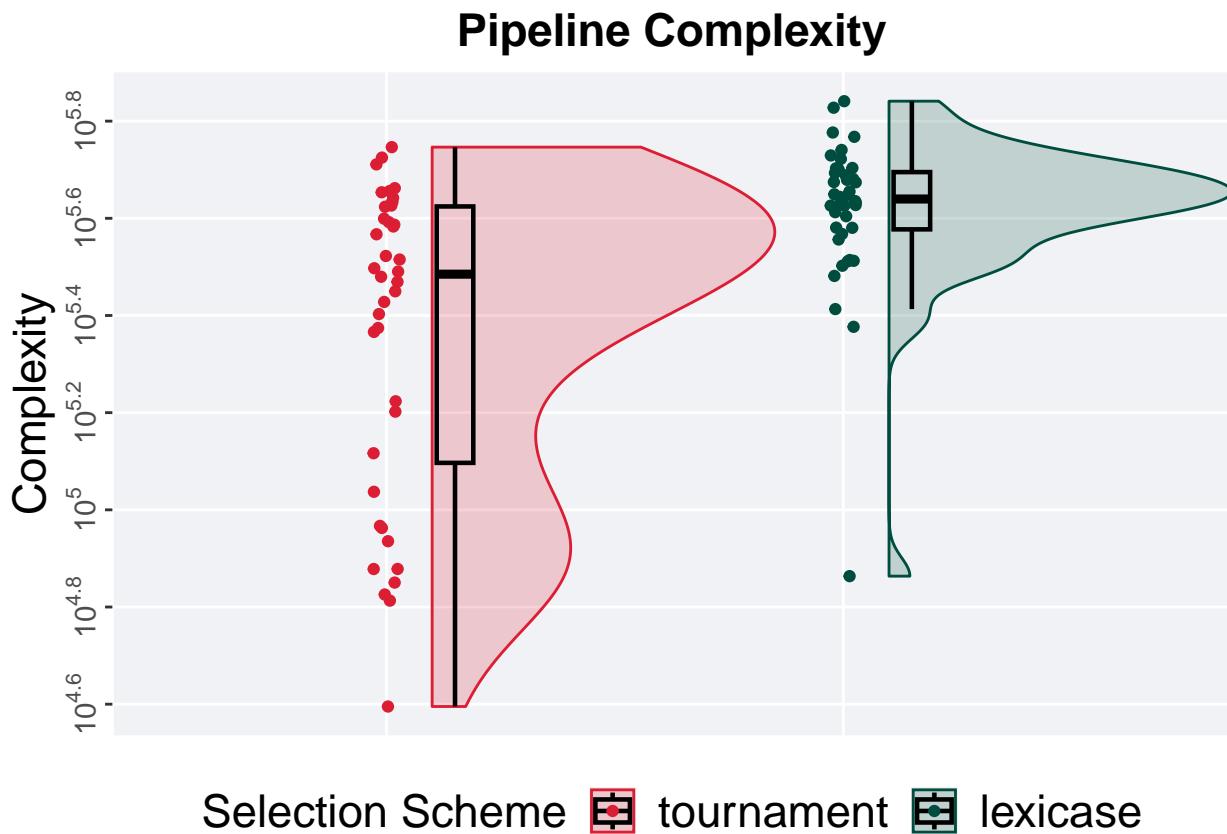
The permutation test revealed that the results are:

```
tournament_results <- filter(task_data, split == '50%' & selection == 'tournament')
lexicase_results <- filter(task_data, split == '50%' & selection == 'lexicase')
permutation_test(tournament_results$training_performance,
                 lexicase_results$training_performance,
                 seed = 116,
                 alternative = "t")
```

```
## [1] "observed_diff: -0.104445353599085"
## [1] "lower: -1.98629617822013"
## [1] "upper: 1.98629465100794"
## [1] "fail to reject null hypothesis"
## [1] "p-value: 0.91335"
```



```
complexity_plot(filter(task_data, split == '50%'))
```



Summary statistics for the pipeline complexity of the selection schemes at the 5% selection set split:

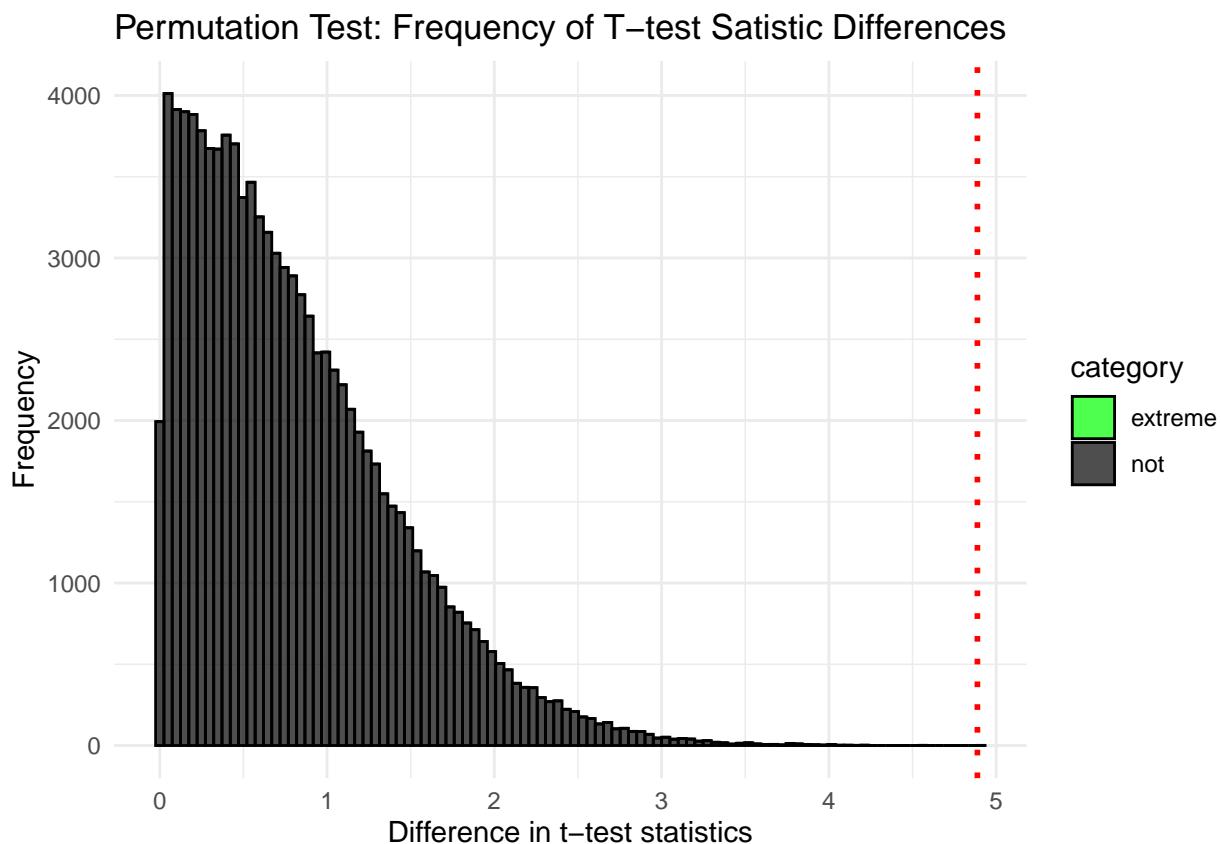
```
complexity_summary(filter(task_data, split == '50%'))
```

```
## # A tibble: 2 x 8
##   selection count na_cnt   min   median     mean     max     IQR
##   <fct>     <int> <int> <dbl>   <dbl>   <dbl>   <dbl>
## 1 tournament     40     0 39371 305526. 285357. 557811 295822.
## 2 lexicase       40     0 73031 436271 434134. 693521 117745
```

The permutation test revealed that the results are:

```
tournament_results <- filter(task_data, split == '50%' & selection == 'tournament')
lexicase_results <- filter(task_data, split == '50%' & selection == 'lexicase')
permutation_test(tournament_results$testing_complexity,
                 lexicase_results$testing_complexity,
                 seed = 245,
                 alternative = "t")
```

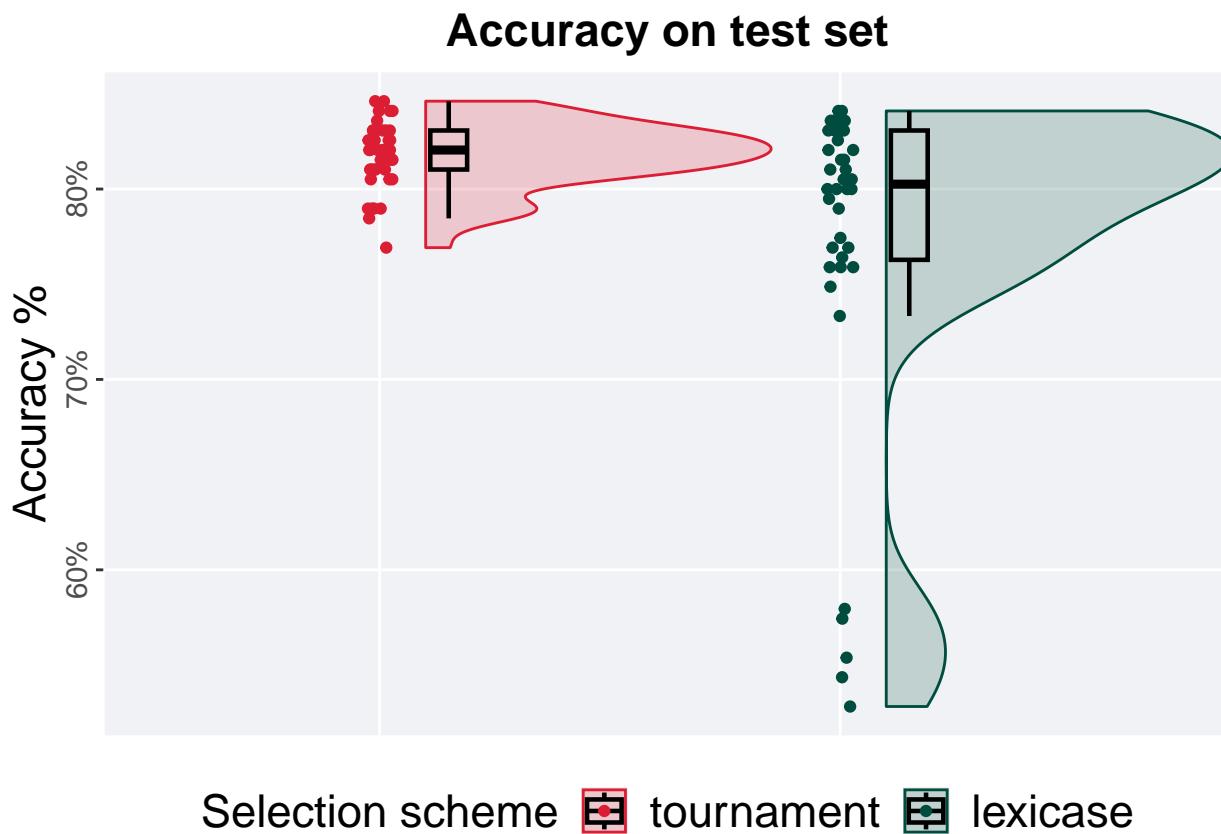
```
## [1] "observed_diff: -4.88784263883817"
## [1] "lower: -1.97975389901524"
## [1] "upper: 2.00588458675597"
## [1] "reject null hypothesis"
## [1] "p-value: 1e-05"
```



13.4 90%

13.4.1 Test accuracy

```
test_plot(filter(task_data, split == '90%'))
```



Summary statistics for the testing performance of the selection schemes at the 5% selection set split:

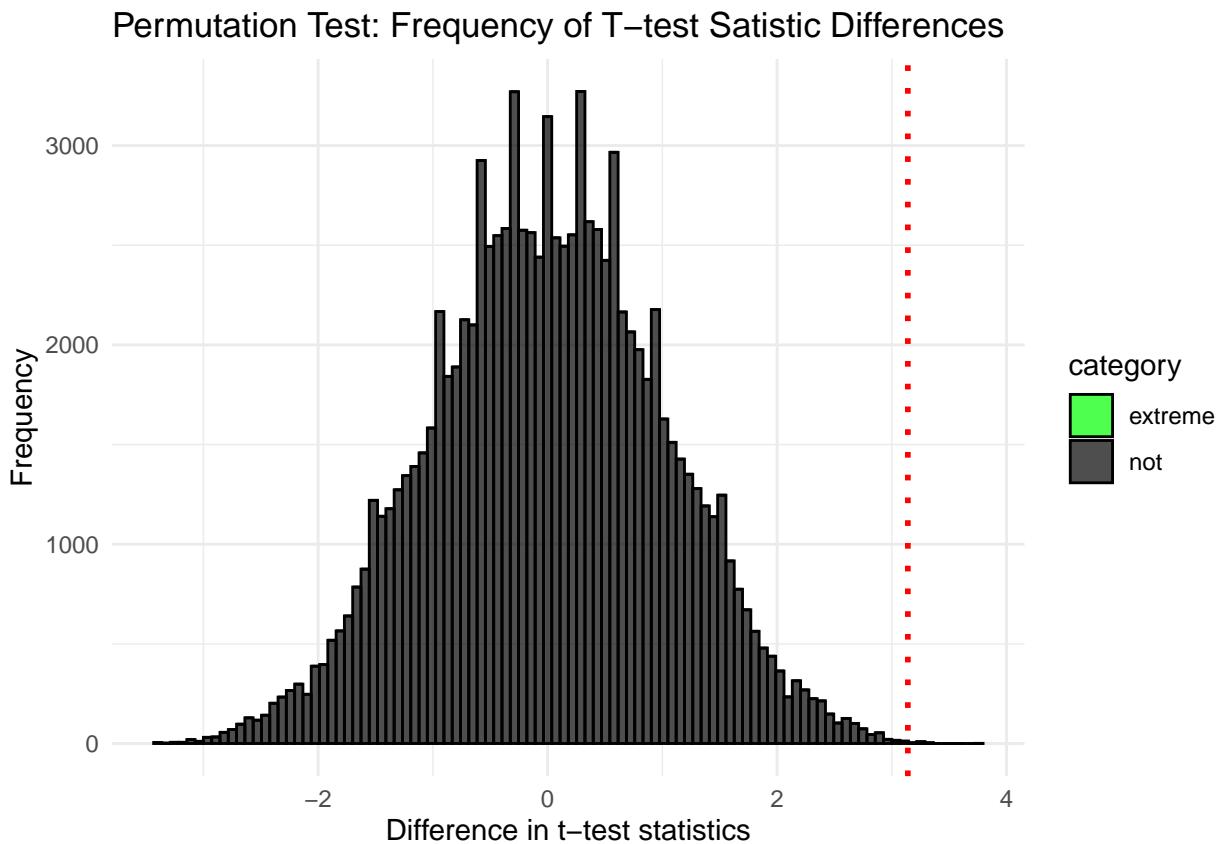
```
test_results_summary(filter(task_data, split == '90%'))
```

```
## # A tibble: 2 x 8
##   selection  count  na_cnt  min  median  mean   max   IQR
##   <fct>      <int>   <int> <dbl>  <dbl> <dbl> <dbl> <dbl>
## 1 tournament     40      0 0.769  0.821 0.818 0.846 0.0205
## 2 lexicase       40      0 0.528  0.803 0.773 0.841 0.0679
```

The permutation test revealed that the results are:

```
tournament_results <- filter(task_data, split == '90%' & selection == 'tournament')
lexicase_results <- filter(task_data, split == '90%' & selection == 'lexicase')
permutation_test(tournament_results$testing_performance,
                 lexicase_results$testing_performance,
                 seed = 117,
                 alternative = "g")
```

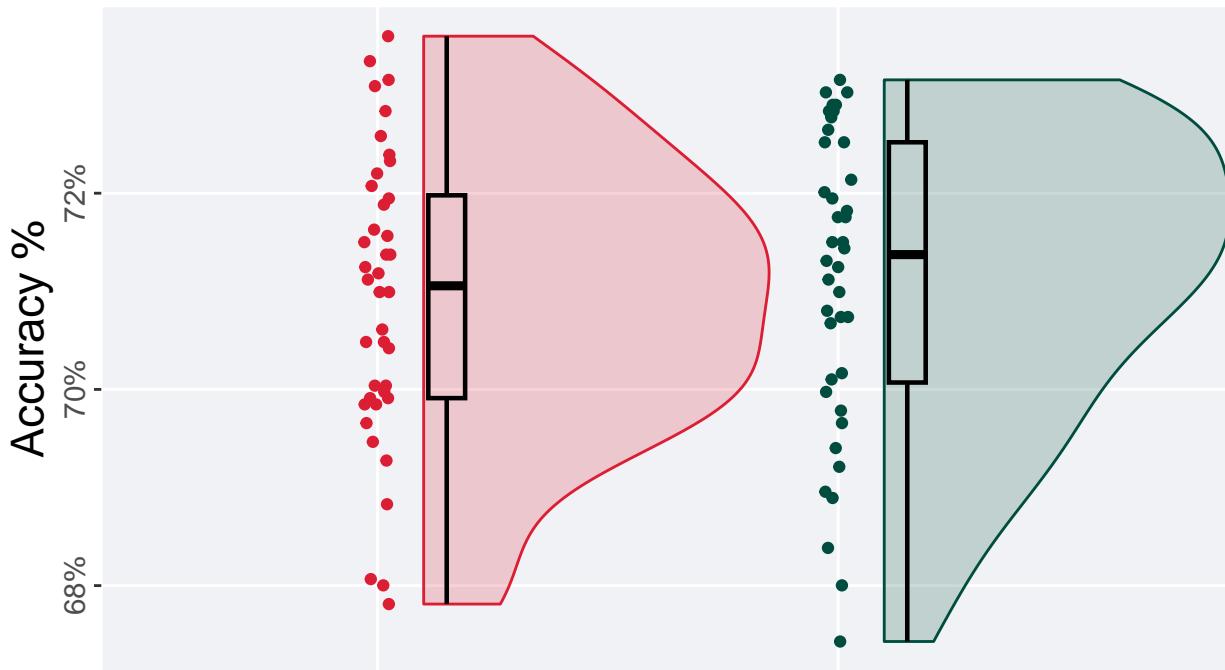
```
## [1] "observed_diff: 3.13974144978373"
## [1] "permutation_diffs[0.95 * n_permutations]: 1.66080565220089"
## [1] "reject null hypothesis"
## [1] "p-value: 0.00024"
```



13.4.2 Validation accuracy

```
validation_plot(filter(task_data, split == '90%'))
```

Accuracy on validation set



Selection Scheme tournament lexicase

Summary statistics for the testing performance of the selection schemes at the 5% selection set split:

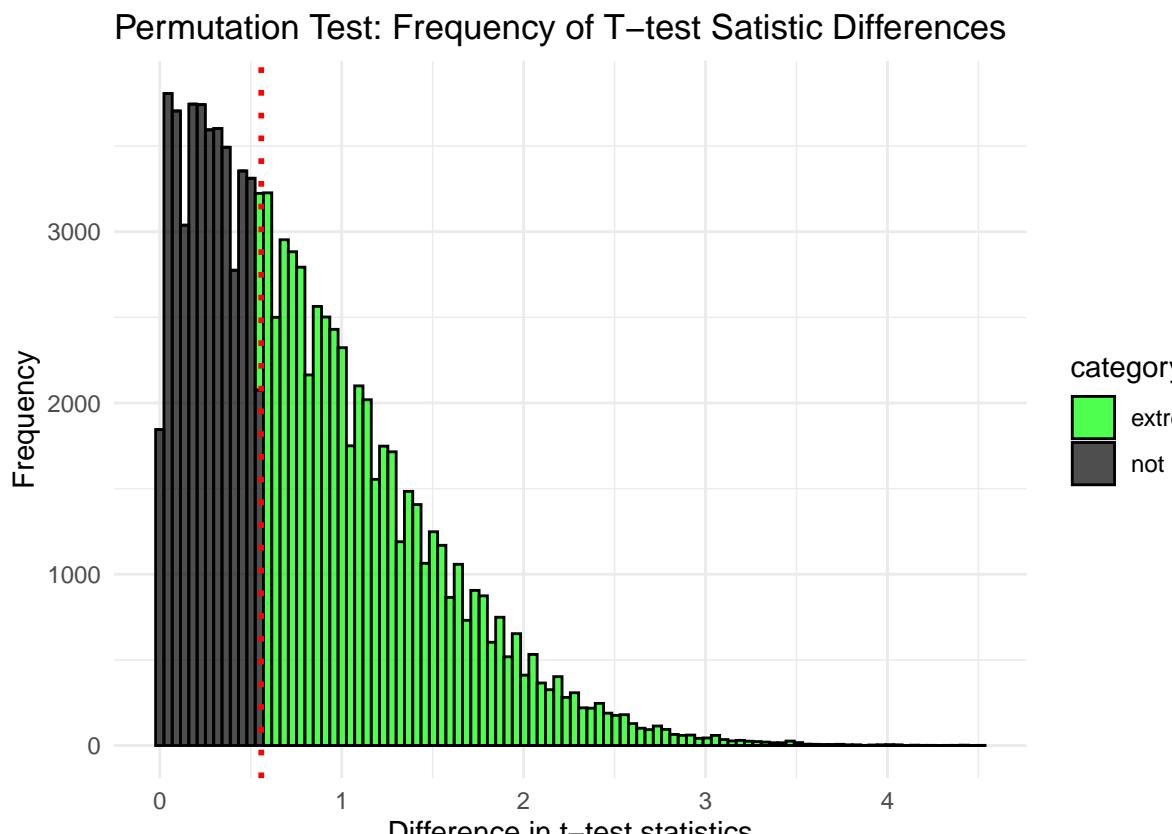
```
validation_accuracy_summary(filter(task_data, split == '90%'))
```

```
## # A tibble: 2 x 8
##   selection count na_cnt  min median  mean   max     IQR
##   <fct>     <int> <int> <dbl> <dbl> <dbl> <dbl> <dbl>
## 1 tournament     40     0 0.678 0.711 0.709 0.736 0.0207
## 2 lexicase       40     0 0.674 0.714 0.711 0.732 0.0245
```

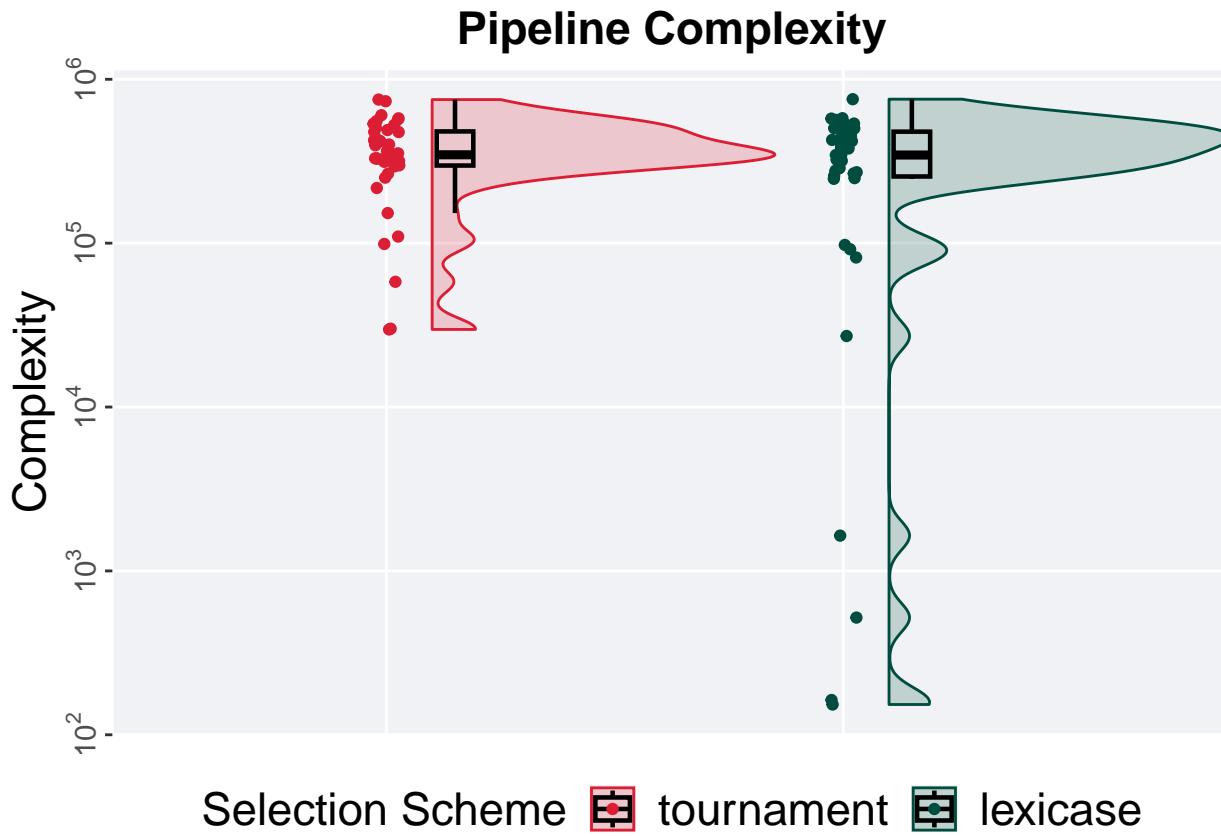
The permutation test revealed that the results are:

```
tournament_results <- filter(task_data, split == '90%' & selection == 'tournament')
lexicase_results <- filter(task_data, split == '90%' & selection == 'lexicase')
permutation_test(tournament_results$training_performance,
                 lexicase_results$training_performance,
                 seed = 118,
                 alternative = "t")
```

```
## [1] "observed_diff: -0.557779127920981"
## [1] "lower: -1.99247802744732"
## [1] "upper: 1.99247866659252"
## [1] "fail to reject null hypothesis"
## [1] "p-value: 0.5791"
```



```
complexity_plot(filter(task_data, split == '90%'))
```



Summary statistics for the pipeline complexity of the selection schemes at the 5% selection set split:

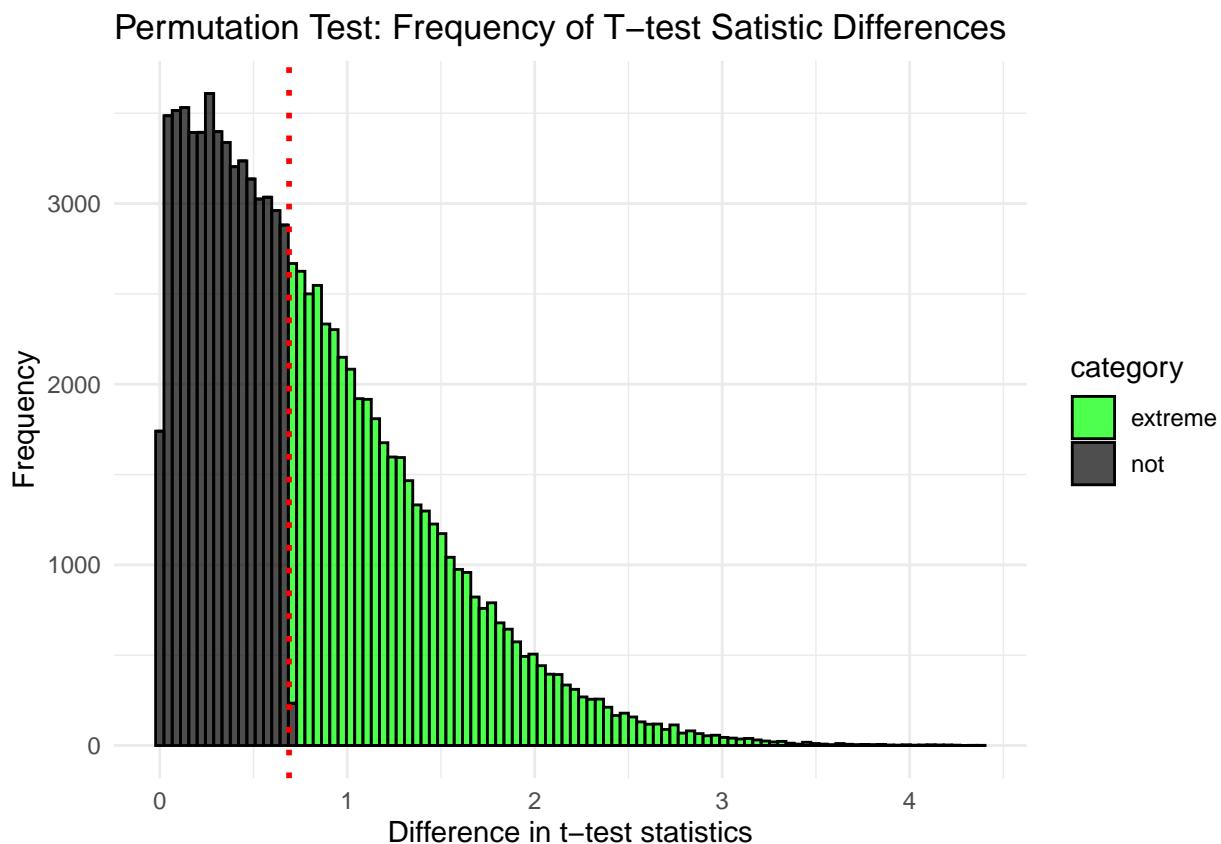
```
complexity_summary(filter(task_data, split == '90%'))
```

```
## # A tibble: 2 x 8
##   selection count na_cnt   min   median     mean     max     IQR
##   <fct>     <int> <int> <dbl>   <dbl>   <dbl>   <dbl>
## 1 tournament     40     0 29751 346266. 368281. 751991 182978.
## 2 lexicase       40     0    153 345032. 340339. 756061 223842.
```

The permutation test revealed that the results are:

```
tournament_results <- filter(task_data, split == '90%' & selection == 'tournament')
lexicase_results <- filter(task_data, split == '90%' & selection == 'lexicase')
permutation_test(tournament_results$testing_complexity,
                  lexicase_results$testing_complexity,
                  seed = 246,
                  alternative = "t")
```

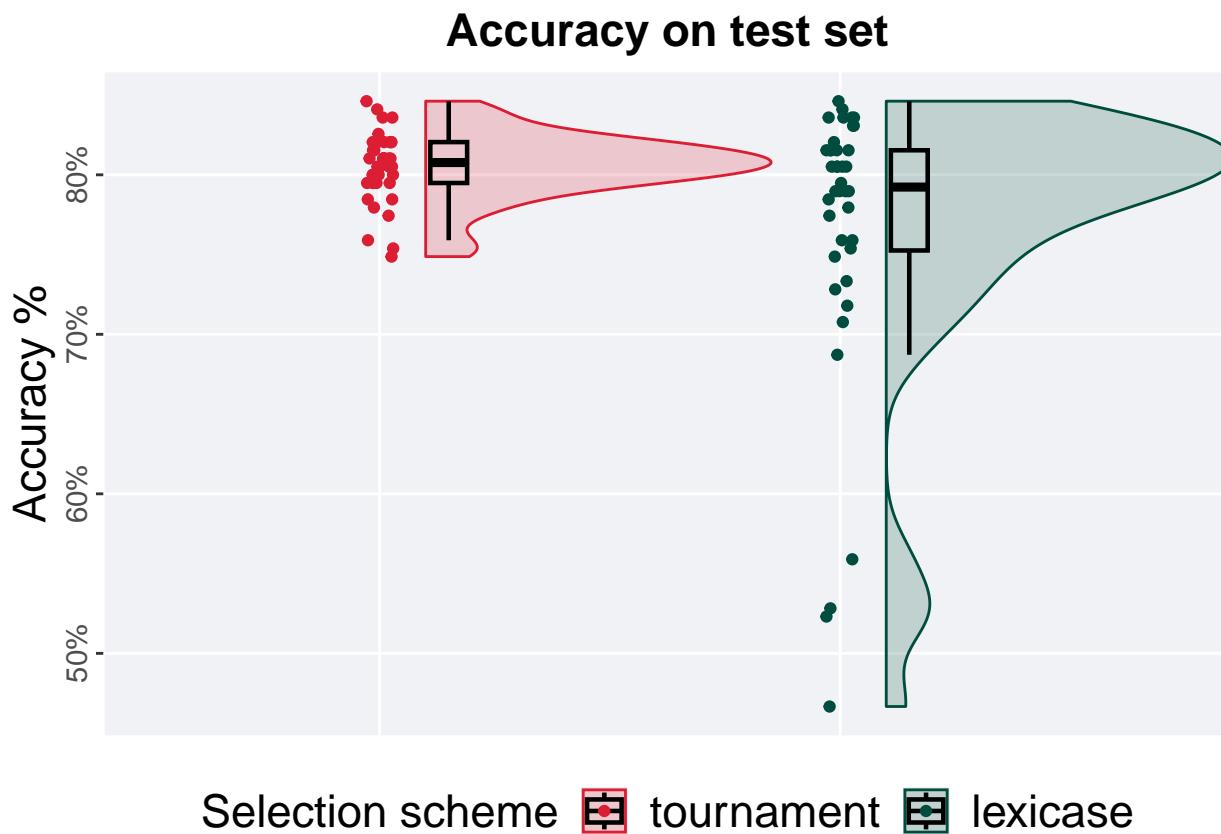
```
## [1] "observed_diff: 0.689619350378453"
## [1] "lower: -1.97421372944794"
## [1] "upper: 1.99319528999097"
## [1] "fail to reject null hypothesis"
## [1] "p-value: 0.48884"
```



13.5 95%

13.5.1 Test accuracy

```
test_plot(filter(task_data, split == '95%'))
```



Summary statistics for the testing performance of the selection schemes at the 5% selection set split:

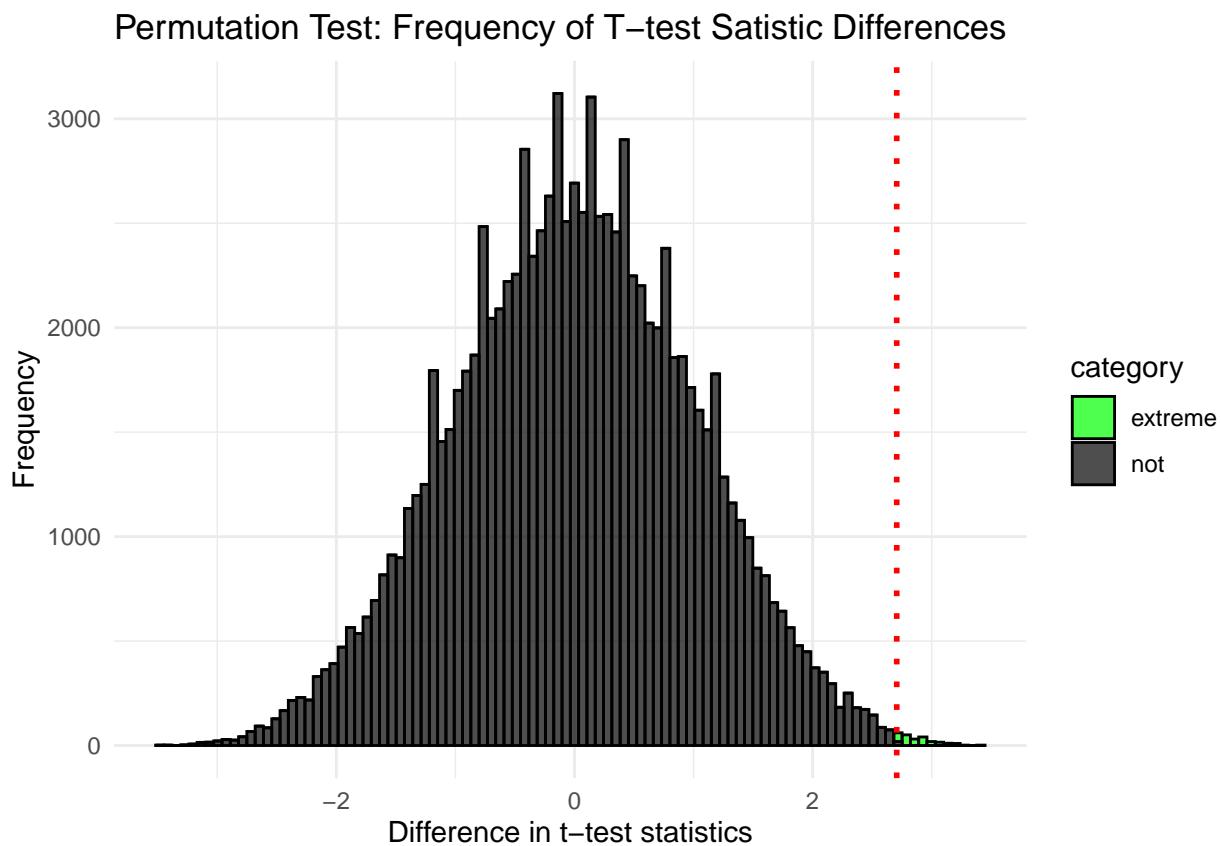
```
test_results_summary(filter(task_data, split == '95%'))
```

```
## # A tibble: 2 x 8
##   selection  count  na_cnt  min  median  mean  max  IQR
##   <fct>      <int>   <int> <dbl>  <dbl> <dbl> <dbl> <dbl>
## 1 tournament     40      0 0.749  0.808 0.805 0.846 0.0256
## 2 lexicase       40      0 0.467  0.792 0.764 0.846 0.0628
```

The permutation test revealed that the results are:

```
tournament_results <- filter(task_data, split == '95%' & selection == 'tournament')
lexicase_results <- filter(task_data, split == '95%' & selection == 'lexicase')
permutation_test(tournament_results$testing_performance,
                 lexicase_results$testing_performance,
                 seed = 119,
                 alternative = "g")
```

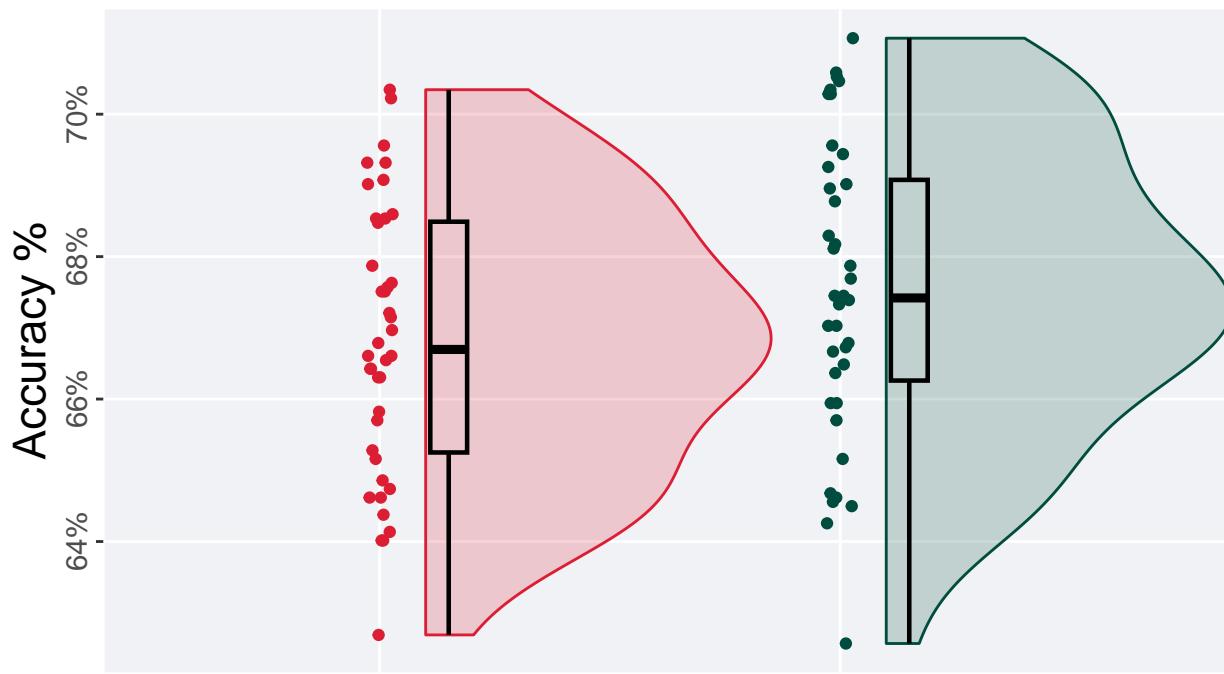
```
## [1] "observed_diff: 2.70614045169234"
## [1] "permutation_diffs[0.95 * n_permutations]: 1.66331762825335"
## [1] "reject null hypothesis"
## [1] "p-value: 0.00225"
```



13.5.2 Validation accuracy

```
validation_plot(filter(task_data, split == '95%'))
```

Accuracy on validation set



Selection Scheme  tournament  lexicase

Summary statistics for the testing performance of the selection schemes at the 5% selection set split:

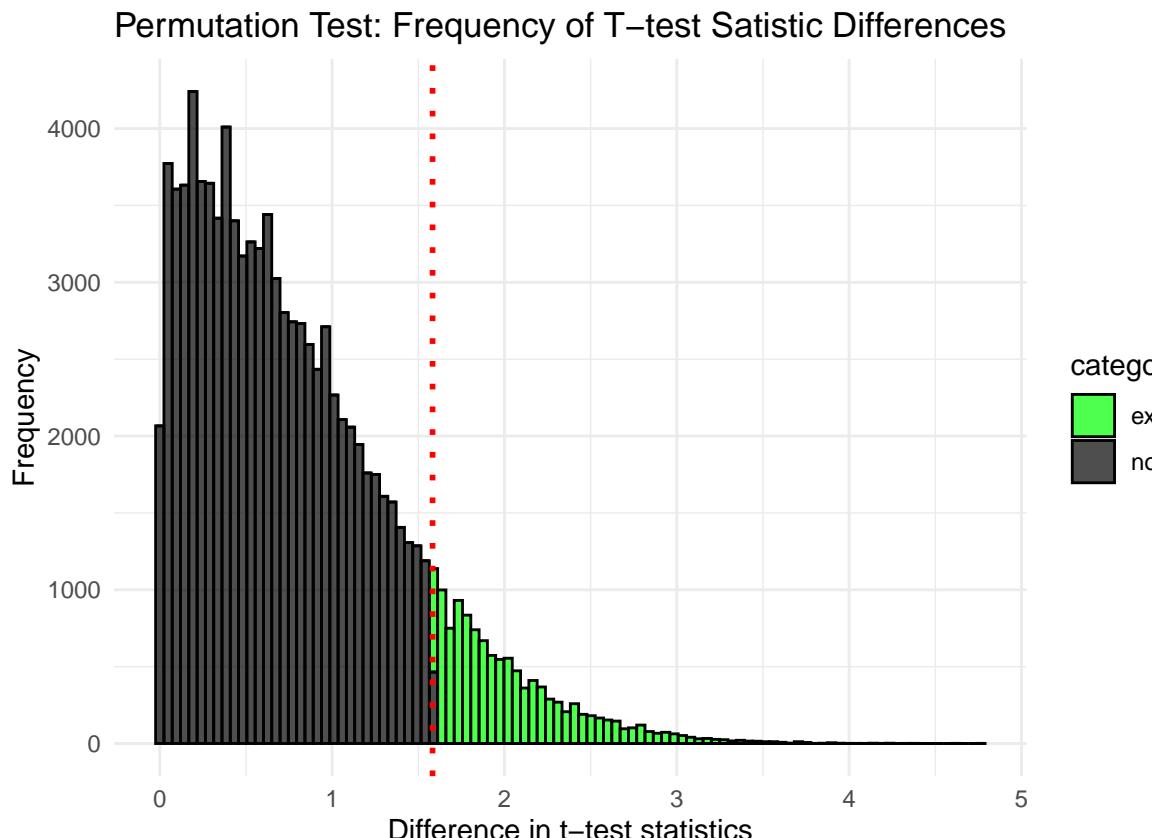
```
validation_accuracy_summary(filter(task_data, split == '95%'))
```

```
## # A tibble: 2 x 8
##   selection count na_cnt  min median  mean  max    IQR
##   <fct>     <int> <int> <dbl> <dbl> <dbl> <dbl> <dbl>
## 1 tournament     40     0 0.627 0.667 0.668 0.703 0.0324
## 2 lexicase       40     0 0.626 0.674 0.675 0.711 0.0282
```

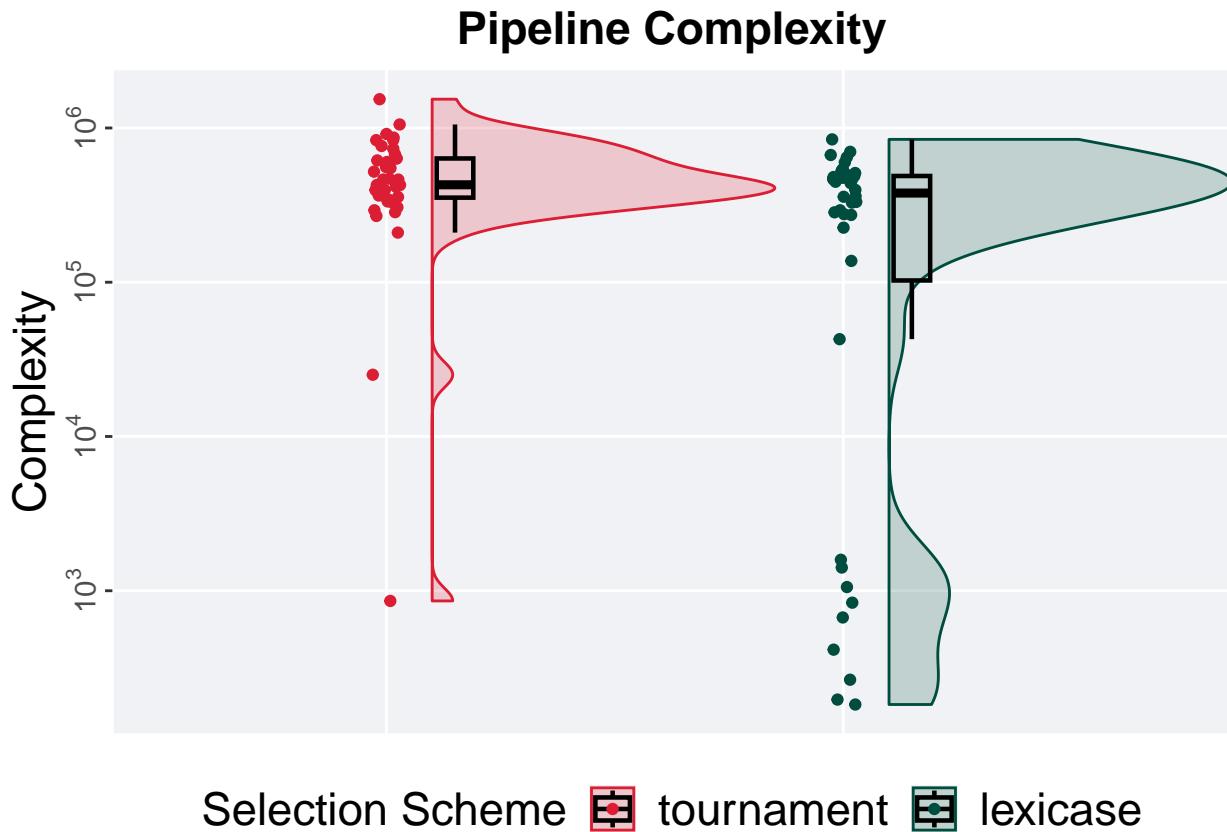
The permutation test revealed that the results are:

```
tournament_results <- filter(task_data, split == '95%' & selection == 'tournament')
lexicase_results <- filter(task_data, split == '95%' & selection == 'lexicase')
permutation_test(tournament_results$training_performance,
                  lexicase_results$training_performance,
                  seed = 120,
                  alternative = "t")
```

```
## [1] "observed_diff: -1.5828085773587"
## [1] "lower: -1.99931288855232"
## [1] "upper: 1.99215361272119"
## [1] "fail to reject null hypothesis"
## [1] "p-value: 0.11697"
```



```
complexity_plot(filter(task_data, split == '95%'))
```



Summary statistics for the pipeline complexity of the selection schemes at the 5% selection set split:

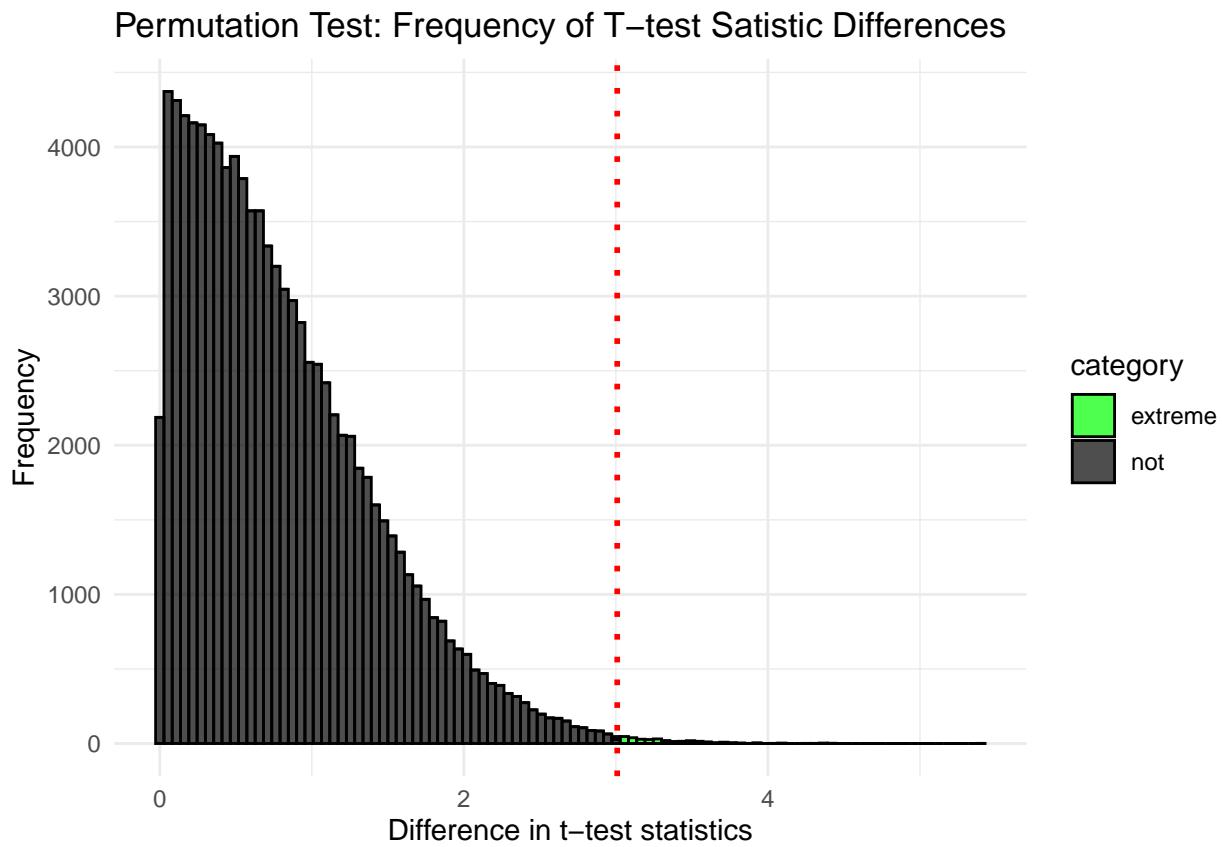
```
complexity_summary(filter(task_data, split == '95%'))
```

```
## # A tibble: 2 x 8
##   selection count na_cnt   min   median     mean     max     IQR
##   <fct>     <int> <int> <dbl> <dbl>     <dbl> <dbl> <dbl>
## 1 tournament     40     0  859 428256. 513593. 1536741 280965
## 2 lexicase       40     0  183 378956 339531.  843511 374528
```

The permutation test revealed that the results are:

```
tournament_results <- filter(task_data, split == '95%' & selection == 'tournament')
lexicase_results <- filter(task_data, split == '95%' & selection == 'lexicase')
permutation_test(tournament_results$testing_complexity,
                 lexicase_results$testing_complexity,
                 seed = 247,
                 alternative = "t")
```

```
## [1] "observed_diff: 3.00876915113078"
## [1] "lower: -1.99486071474436"
## [1] "upper: 1.98667921694081"
## [1] "reject null hypothesis"
## [1] "p-value: 0.0033"
```



Chapter 14

Task 359962

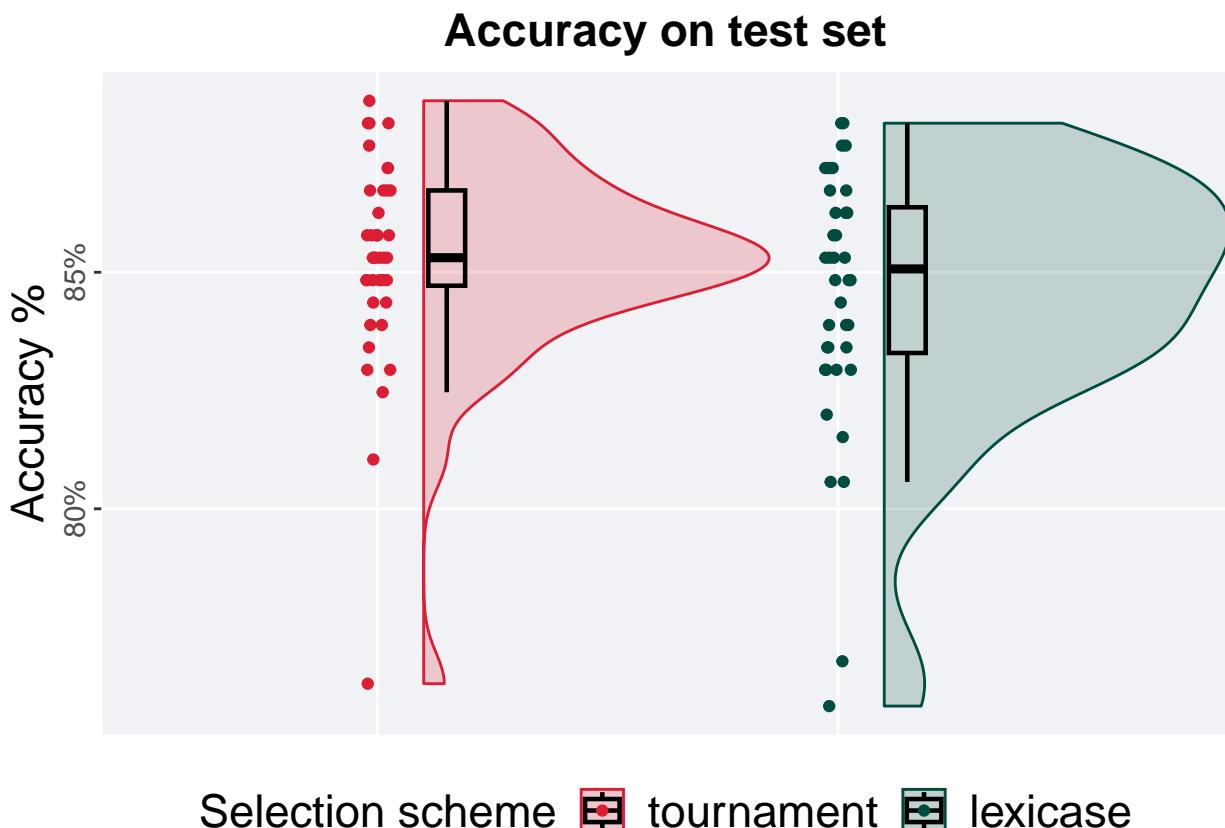
We present the results of our analysis of task 359962 with the different selection set splits used in our study.

```
task_data <- filter(results, task_id == 359962)
```

14.1 5%

14.1.1 Test accuracy

```
test_plot(filter(task_data, split == '5%'))
```



Summary statistics for the testing performance of the selection schemes at the 5% selection set split:

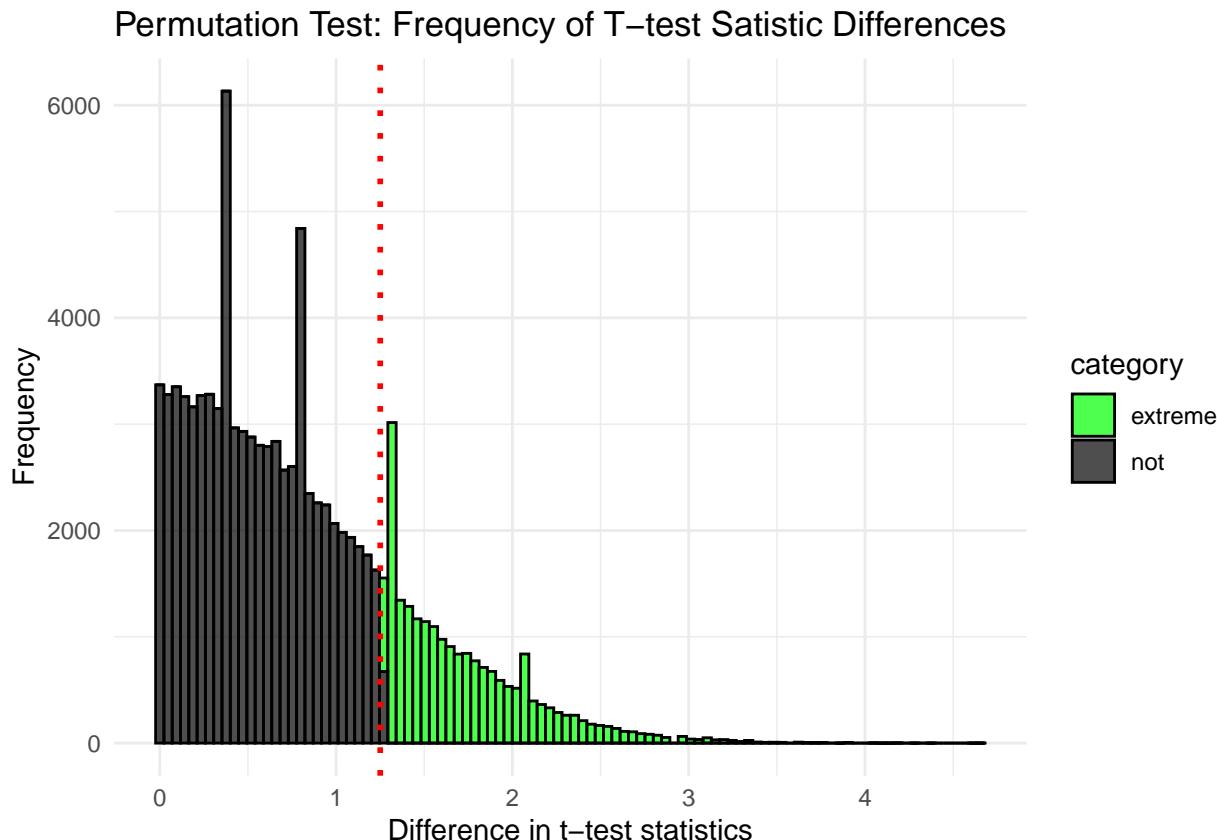
```
test_results_summary(filter(task_data, split == '5%'))
```

```
## # A tibble: 2 x 8
##   selection count na_cnt   min median   mean   max   IQR
##   <fct>     <int> <int> <dbl> <dbl> <dbl> <dbl> <dbl>
## 1 tournament     40     0 0.763 0.853 0.852 0.886 0.0201
## 2 lexicase       40     0 0.758 0.851 0.845 0.882 0.0308
```

The permutation test revealed that the results are:

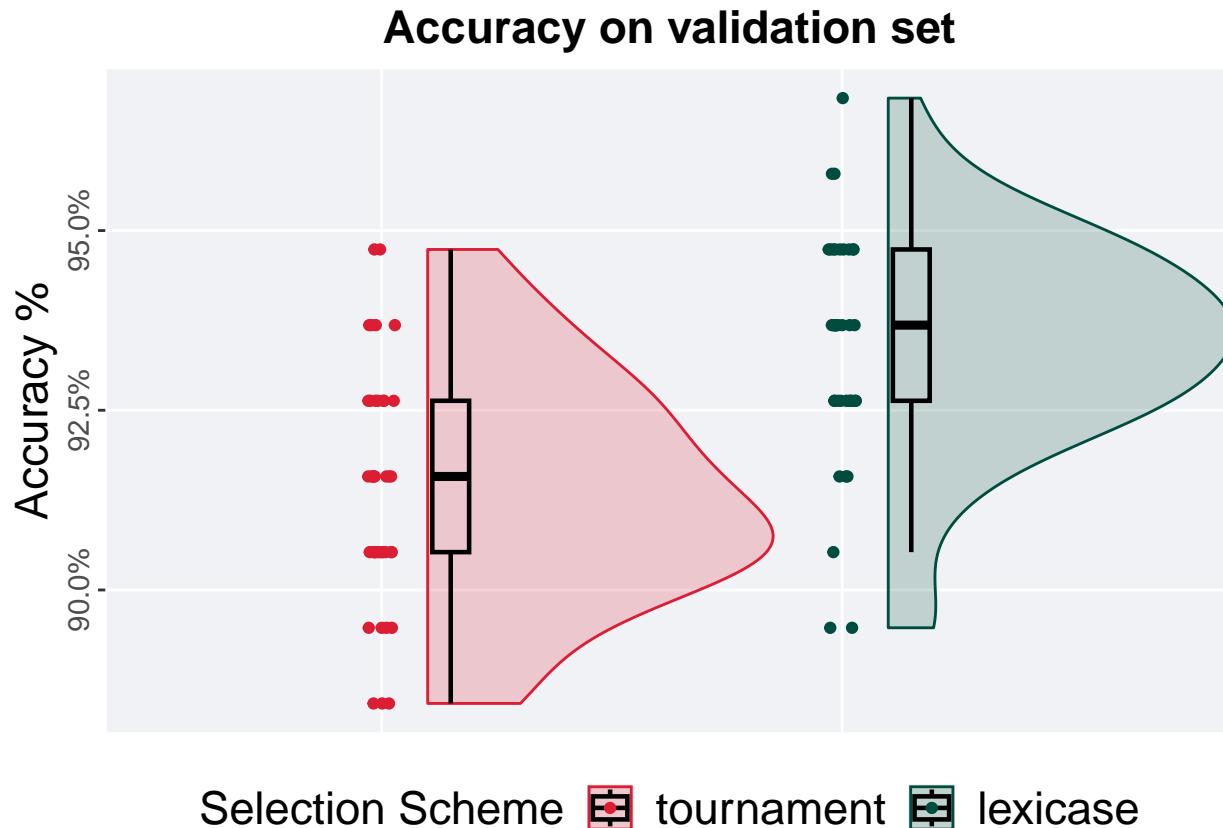
```
tournament_results <- filter(task_data, split == '5%' & selection == 'tournament')
lexicase_results <- filter(task_data, split == '5%' & selection == 'lexicase')
permutation_test(tournament_results$testing_performance,
                 lexicase_results$testing_performance,
                 seed = 121,
                 alternative = "t")
```

```
## [1] "observed_diff: 1.25048032725024"
## [1] "lower: -2.00109902223407"
## [1] "upper: 1.95593273813578"
## [1] "fail to reject null hypothesis"
## [1] "p-value: 0.21784"
```



14.1.2 Validation accuracy

```
validation_plot(filter(task_data, split == '5%'))
```



Selection Scheme tournament lexicase

Summary statistics for the testing performance of the selection schemes at the 5% selection set split:

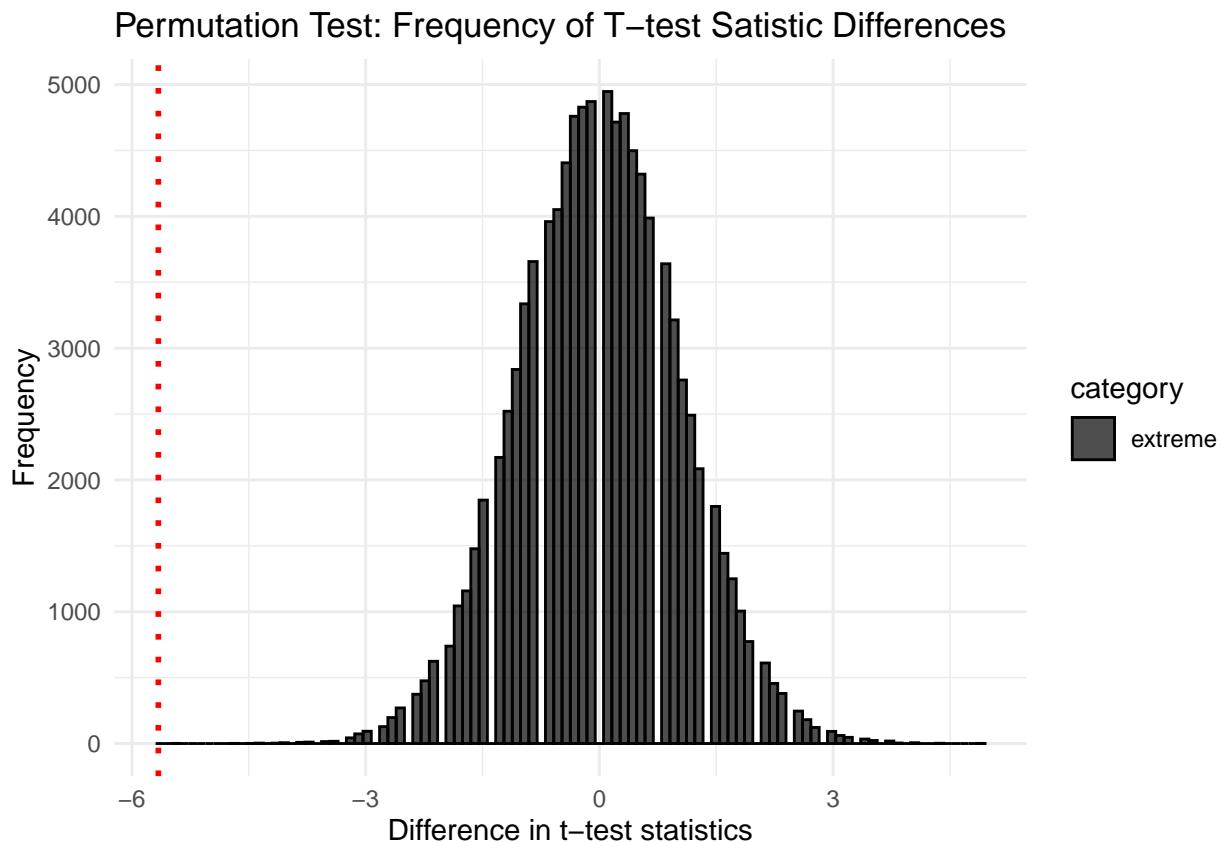
```
validation_accuracy_summary(filter(task_data, split == '5%'))
```

```
## # A tibble: 2 x 8
##   selection count na_cnt  min median  mean   max     IQR
##   <fct>     <int> <int> <dbl> <dbl> <dbl> <dbl> <dbl>
## 1 tournament     40     0 0.884 0.916 0.914 0.947 0.0211
## 2 lexicase       40     0 0.895 0.937 0.934 0.968 0.0211
```

The permutation test revealed that the results are:

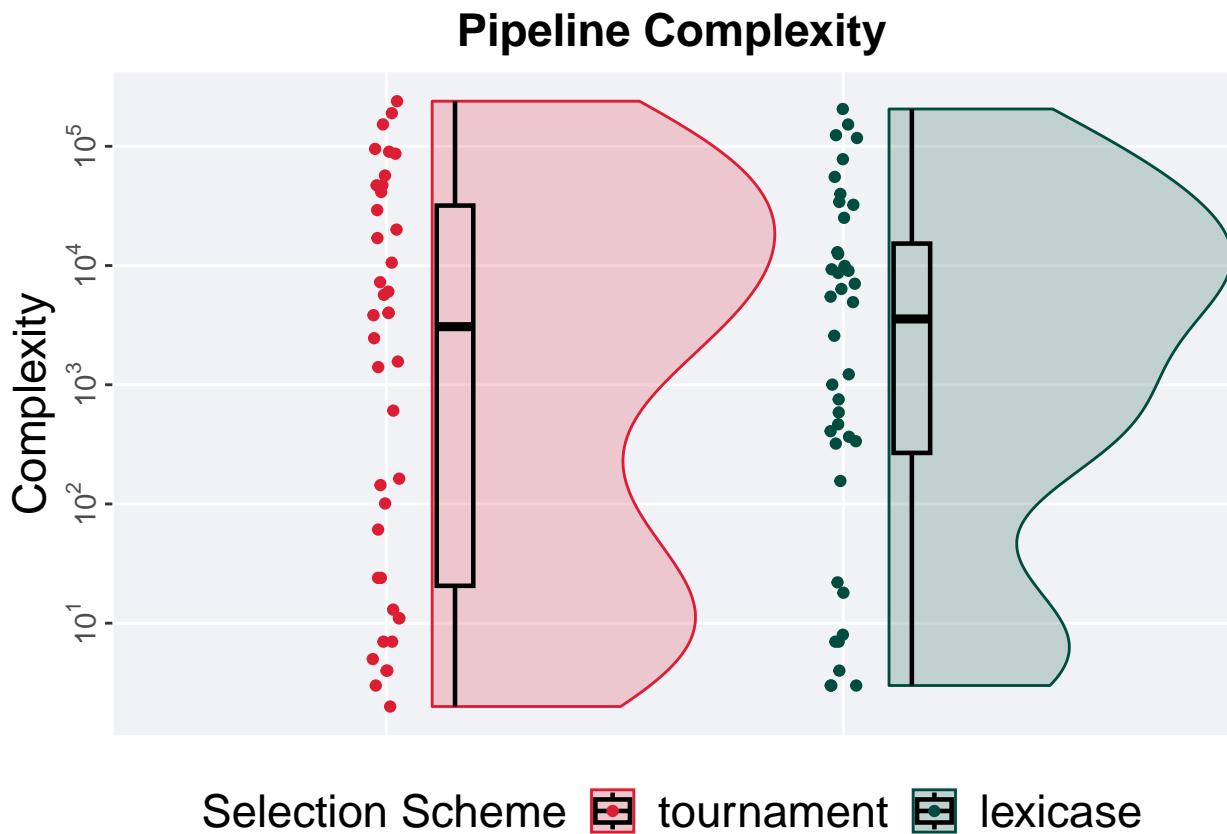
```
tournament_results <- filter(task_data, split == '5%' & selection == 'tournament')
lexicase_results <- filter(task_data, split == '5%' & selection == 'lexicase')
permutation_test(tournament_results$training_performance,
                 lexicase_results$training_performance,
                 seed = 122,
                 alternative = "1")
```

```
## [1] "observed_diff: -5.66114893846717"
## [1] "permutation_diffs[0.05 * n_permutations]: -1.70196523619733"
## [1] "reject null hypothesis"
## [1] "p-value: 1e-05"
```



14.1.3 Pipeline complexity

```
complexity_plot(filter(task_data, split == '5%'))
```



Summary statistics for the pipeline complexity of the selection schemes at the 5% selection set split:

```
complexity_summary(filter(task_data, split == '5%'))
```

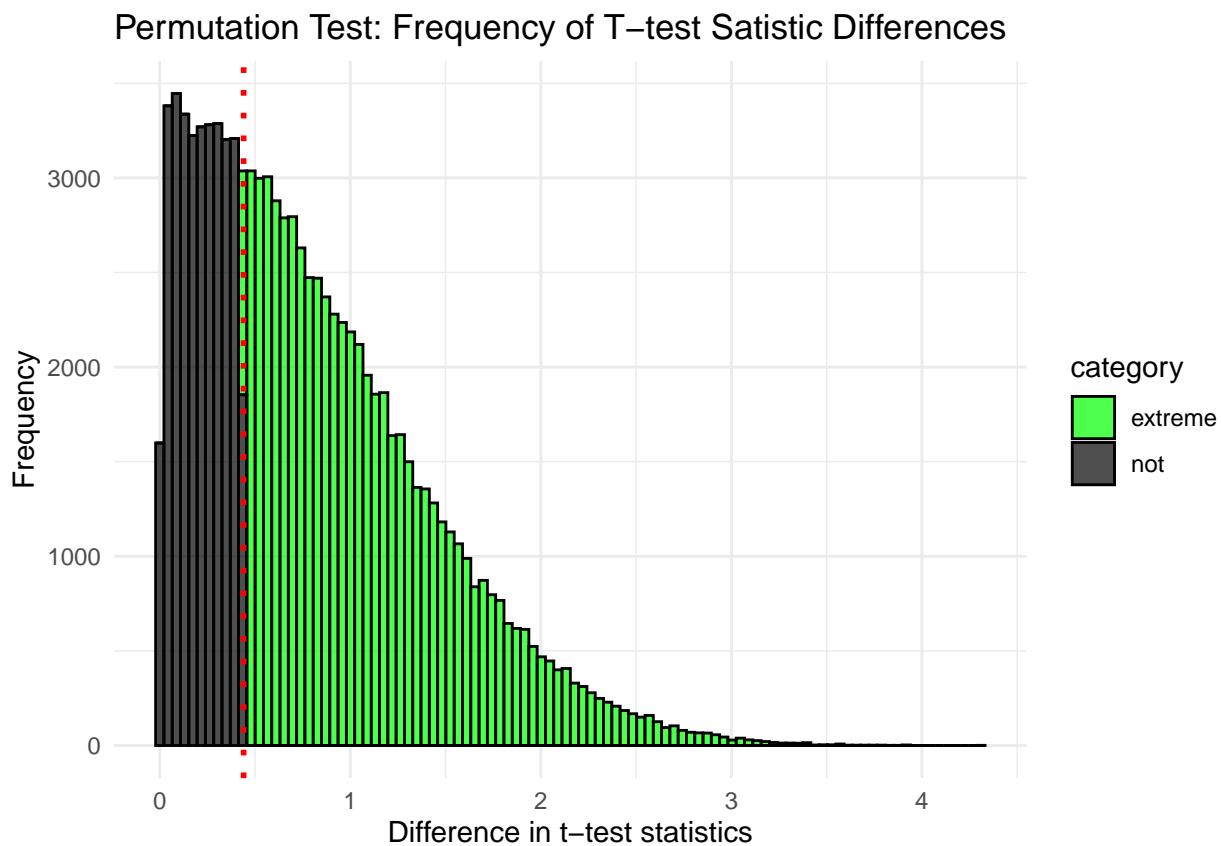
```
## # A tibble: 2 x 8
##   selection count na_cnt  min median   mean   max   IQR
##   <fct>     <int> <int> <dbl> <dbl> <dbl> <int> <dbl>
## 1 tournament     40     0     2  3146  29003. 239041 32212.
## 2 lexicase       40     0     3 3752.  23963. 205611 15711.
```

The permutation test revealed that the results are:

```
tournament_results <- filter(task_data, split == '5%' & selection == 'tournament')
lexicase_results <- filter(task_data, split == '5%' & selection == 'lexicase')
```

```
permutation_test(tournament_results$testing_complexity,
                  lexicase_results$testing_complexity,
                  seed = 248,
                  alternative = "t")
```

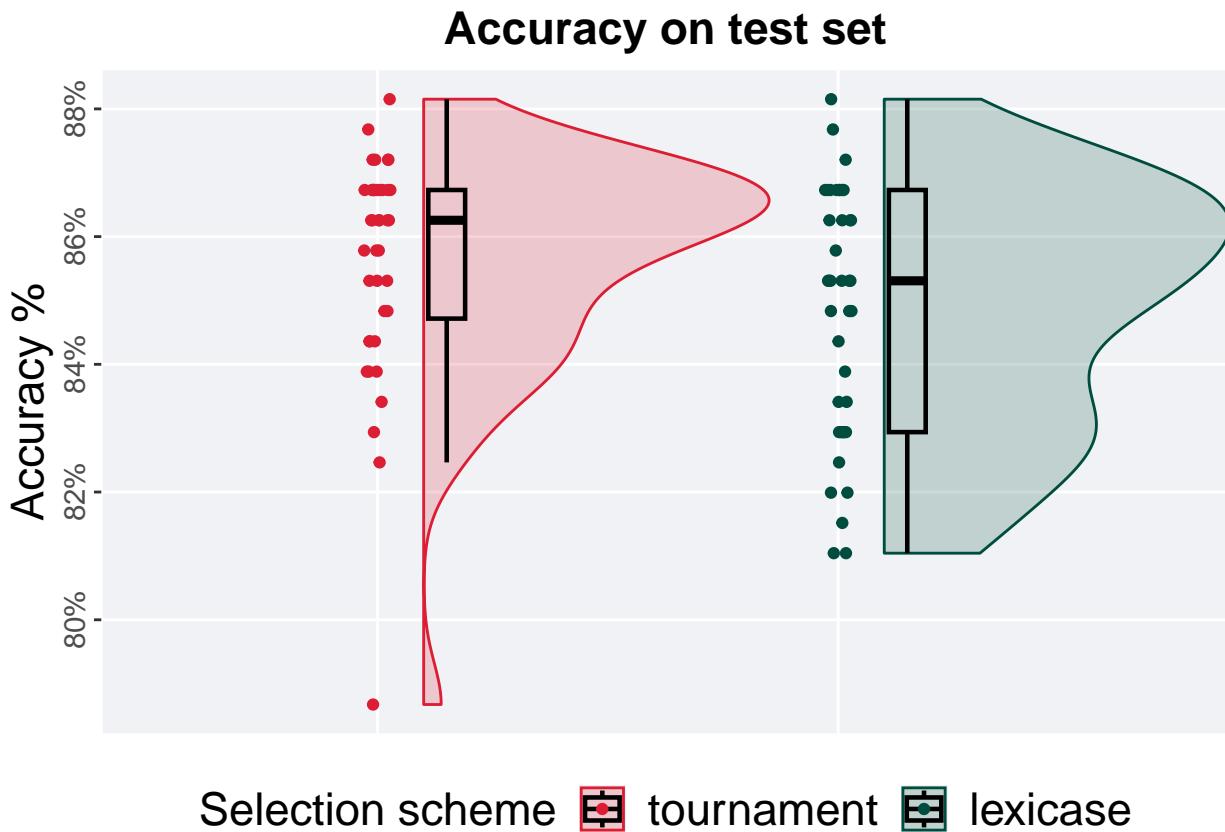
```
## [1] "observed_diff: 0.440007126157174"
## [1] "lower: -1.97871300636883"
## [1] "upper: 1.97101626762399"
## [1] "fail to reject null hypothesis"
## [1] "p-value: 0.6691"
```



14.2 10%

14.2.1 Test accuracy

```
test_plot(filter(task_data, split == '10%'))
```



Summary statistics for the testing performance of the selection schemes at the 5% selection set split:

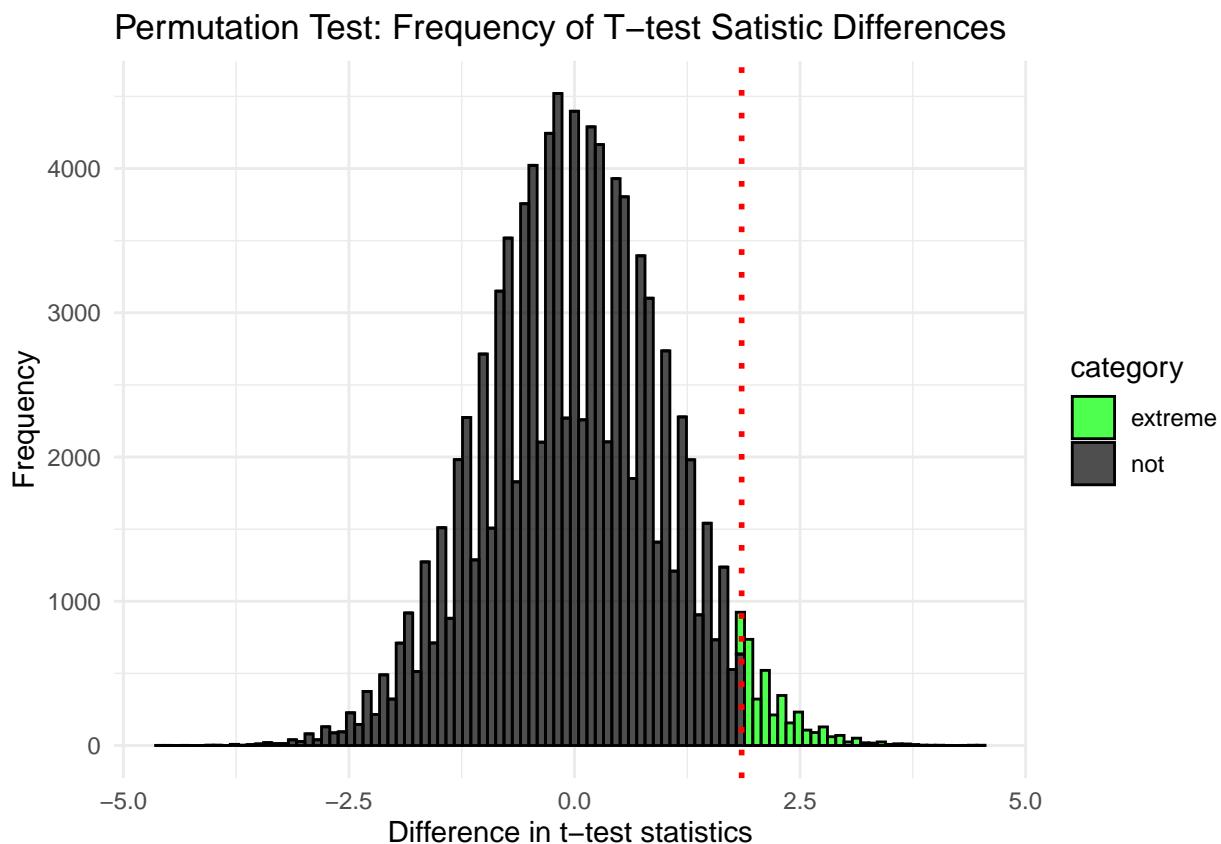
```
test_results_summary(filter(task_data, split == '10%'))
```

```
## # A tibble: 2 x 8
##   selection  count na_cnt   min median   mean   max   IQR
##   <fct>      <int> <int> <dbl> <dbl> <dbl> <dbl> <dbl>
## 1 tournament    40     0 0.787  0.863  0.856  0.882  0.0201
## 2 lexicase      40     0 0.810  0.853  0.848  0.882  0.0379
```

The permutation test revealed that the results are:

```
tournament_results <- filter(task_data, split == '10%' & selection == 'tournament')
lexicase_results <- filter(task_data, split == '10%' & selection == 'lexicase')
permutation_test(tournament_results$testing_performance,
                 lexicase_results$testing_performance,
                 seed = 123,
                 alternative = "g")
```

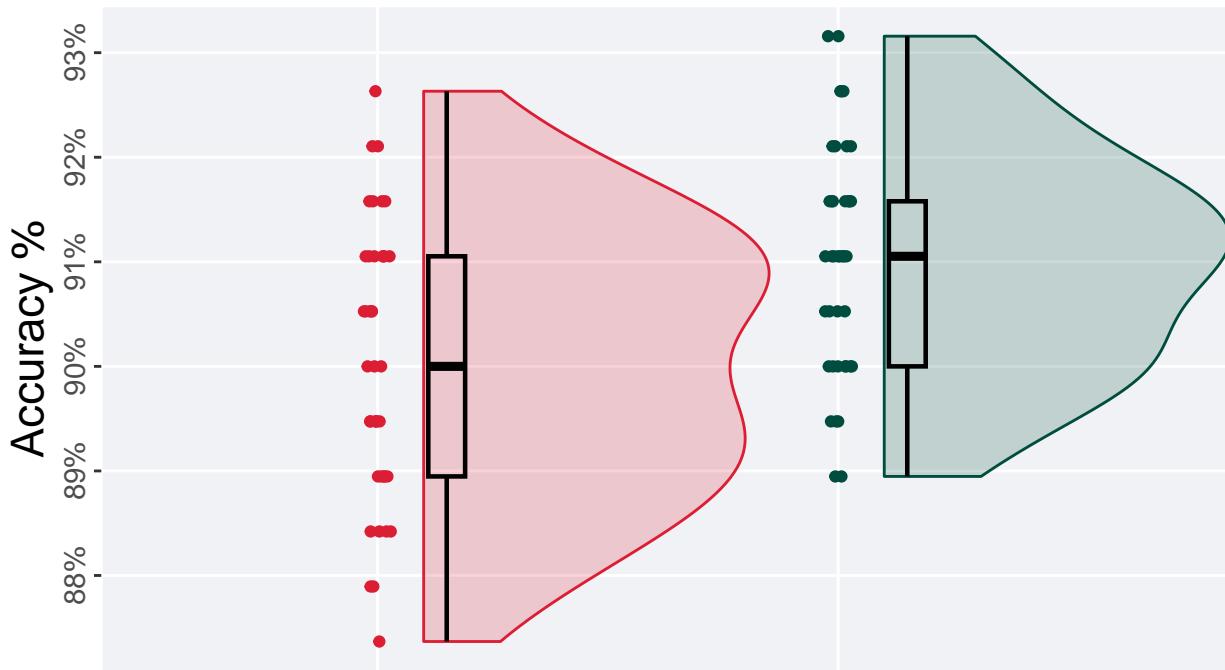
```
## [1] "observed_diff: 1.85272669685502"
## [1] "permutation_diffs[0.95 * n_permutations]: 1.67522717697368"
## [1] "reject null hypothesis"
## [1] "p-value: 0.0345"
```



14.2.2 Validation accuracy

```
validation_plot(filter(task_data, split == '10%'))
```

Accuracy on validation set



Selection Scheme  tournament  lexicase

Summary statistics for the testing performance of the selection schemes at the 5% selection set split:

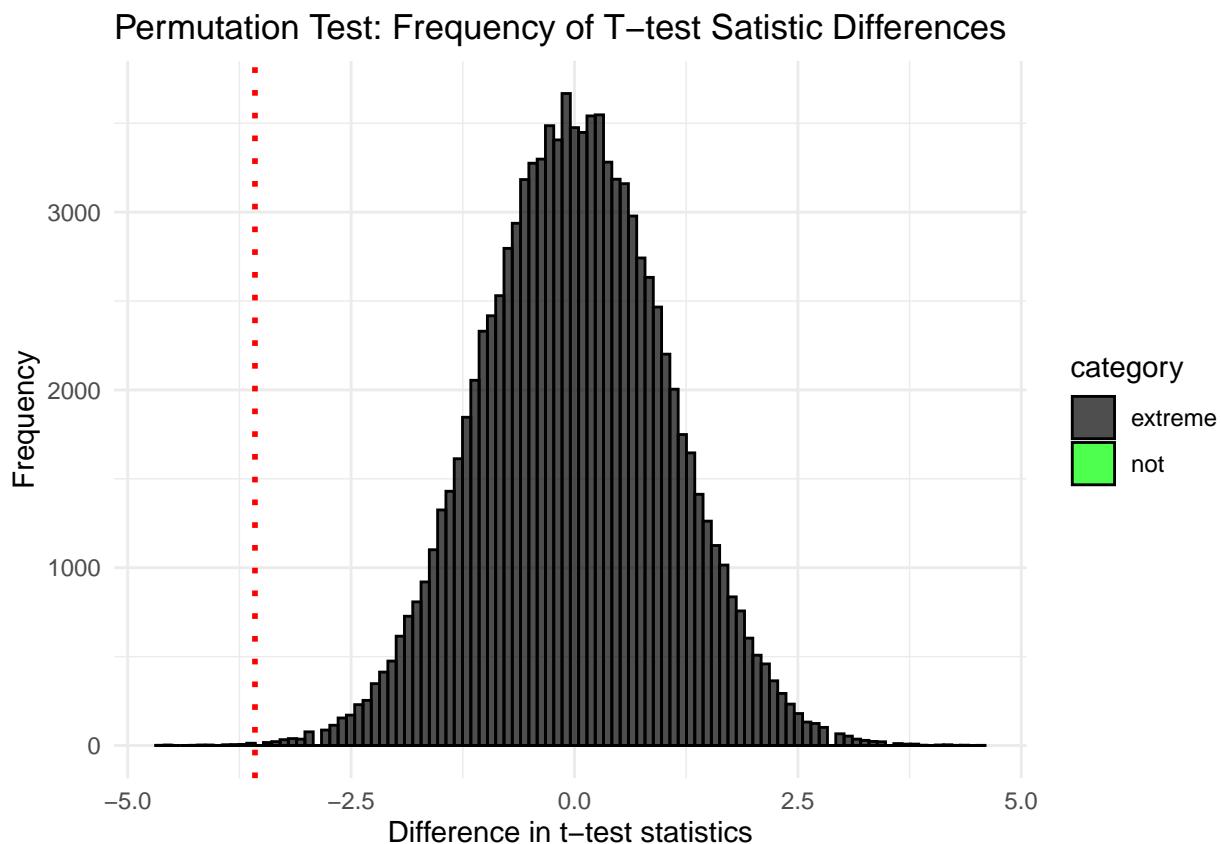
```
validation_accuracy_summary(filter(task_data, split == '10%'))
```

```
## # A tibble: 2 x 8
##   selection  count  na_cnt  min  median  mean   max    IQR
##   <fct>      <int>  <int>  <dbl>  <dbl>  <dbl>  <dbl>  <dbl>
## 1 tournament    40      0  0.874  0.900  0.926  0.926  0.0211
## 2 lexicase      40      0  0.889  0.911  0.910  0.932  0.0158
```

The permutation test revealed that the results are:

```
tournament_results <- filter(task_data, split == '10%' & selection == 'tournament')
lexicase_results <- filter(task_data, split == '10%' & selection == 'lexicase')
permutation_test(tournament_results$training_performance,
                 lexicase_results$training_performance,
                 seed = 124,
                 alternative = "1")
```

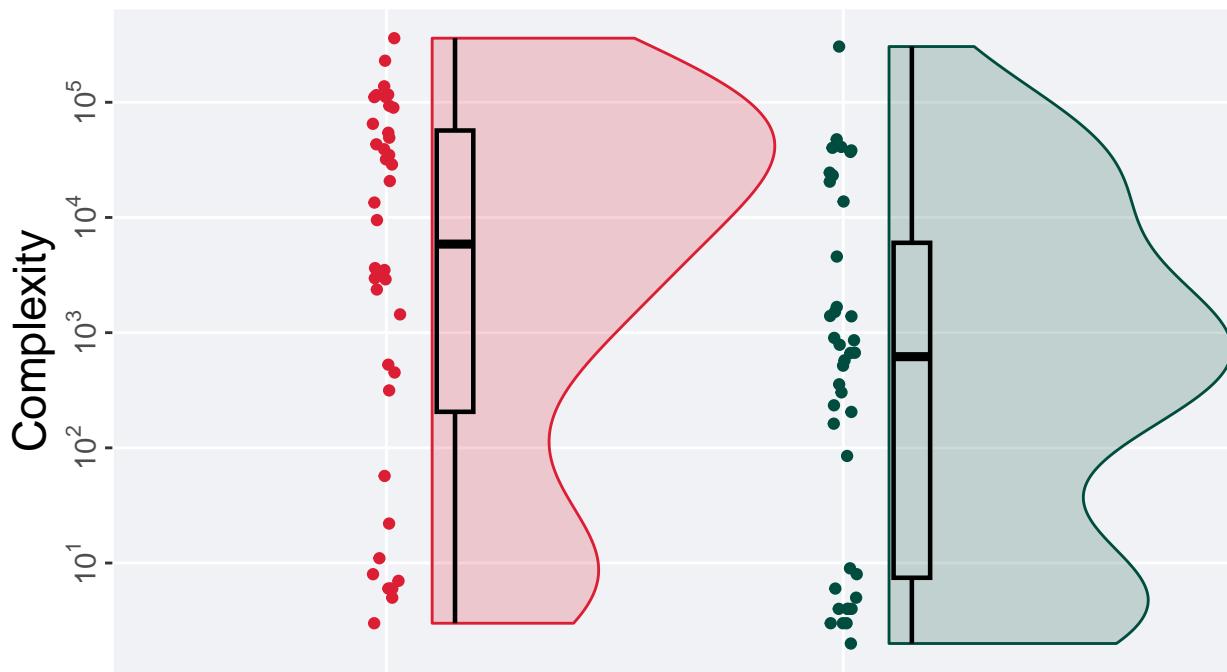
```
## [1] "observed_diff: -3.57585347444171"
## [1] "permutation_diffs[0.05 * n_permutations]: -1.64001739781141"
## [1] "reject null hypothesis"
## [1] "p-value: 0.00031"
```



14.2.3 Pipeline complexity

```
complexity_plot(filter(task_data, split == '10%'))
```

Pipeline Complexity



Selection Scheme tournament lexicase

Summary statistics for the pipeline complexity of the selection schemes at the 5% selection set split:

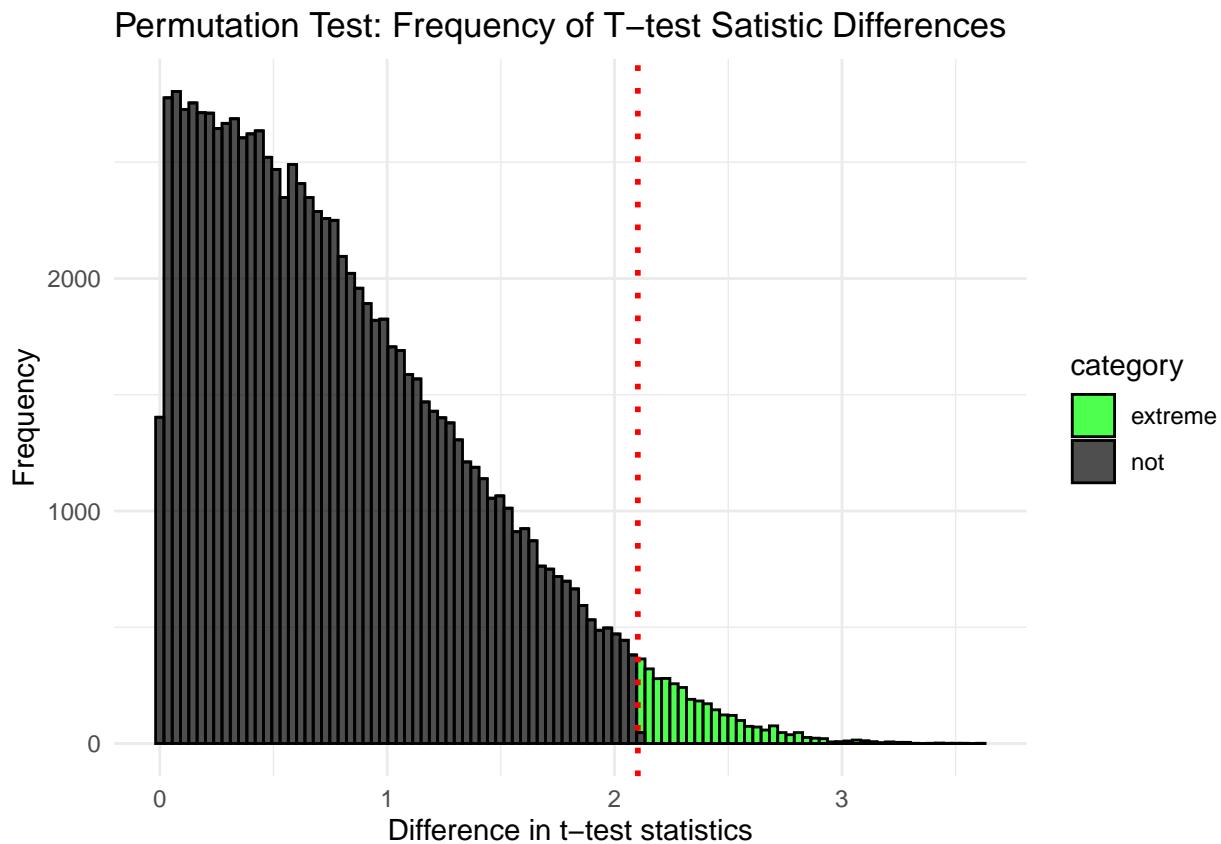
```
complexity_summary(filter(task_data, split == '10%'))
```

```
## # A tibble: 2 x 8
##   selection  count  na_cnt  min  median  mean  max    IQR
##   <fct>     <int>   <int> <dbl>  <dbl>  <dbl> <int>  <dbl>
## 1 tournament     40      0      3  6560. 44422. 360121 56807.
## 2 lexicase       40      0      2   620. 15194. 304721  6865.
```

The permutation test revealed that the results are:

```
tournament_results <- filter(task_data, split == '10%' & selection == 'tournament')
lexicase_results <- filter(task_data, split == '10%' & selection == 'lexicase')
permutation_test(tournament_results$testing_complexity,
                 lexicase_results$testing_complexity,
                 seed = 249,
                 alternative = "t")
```

```
## [1] "observed_diff: 2.10243937846225"
## [1] "lower: -1.96069369649229"
## [1] "upper: 1.96005815554816"
## [1] "reject null hypothesis"
## [1] "p-value: 0.03298"
```

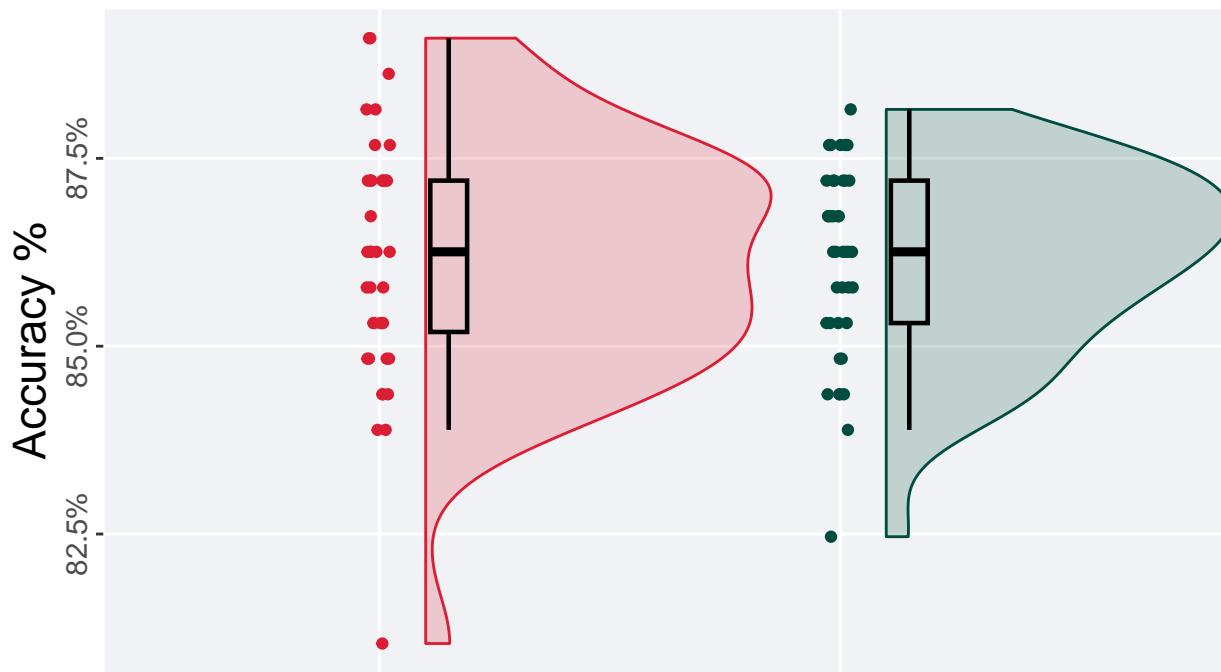


14.3 50%

14.3.1 Test accuracy

```
test_plot(filter(task_data, split == '50%'))
```

Accuracy on test set



Selection scheme tournament lexicase

Summary statistics for the testing performance of the selection schemes at the 5% selection set split:

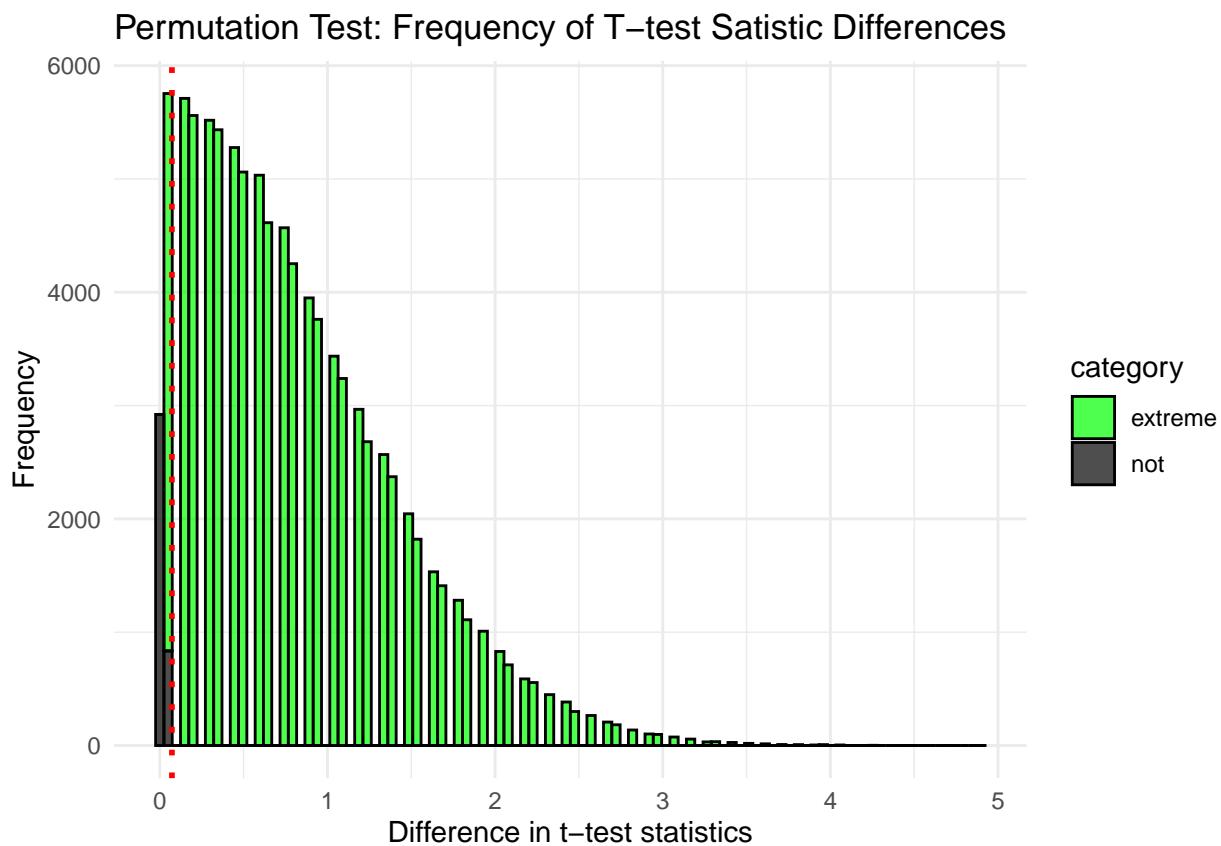
```
test_results_summary(filter(task_data, split == '50%'))
```

```
## # A tibble: 2 x 8
##   selection count na_cnt  min median  mean   max     IQR
##   <fct>     <int> <int> <dbl> <dbl> <dbl> <dbl> <dbl>
## 1 tournament     40     0 0.810 0.863 0.862 0.891 0.0201
## 2 lexicase       40     0 0.825 0.863 0.861 0.882 0.0190
```

The permutation test revealed that the results are:

```
tournament_results <- filter(task_data, split == '50%' & selection == 'tournament')
lexicase_results <- filter(task_data, split == '50%' & selection == 'lexicase')
permutation_test(tournament_results$testing_performance,
                 lexicase_results$testing_performance,
                 seed = 125,
                 alternative = "t")
```

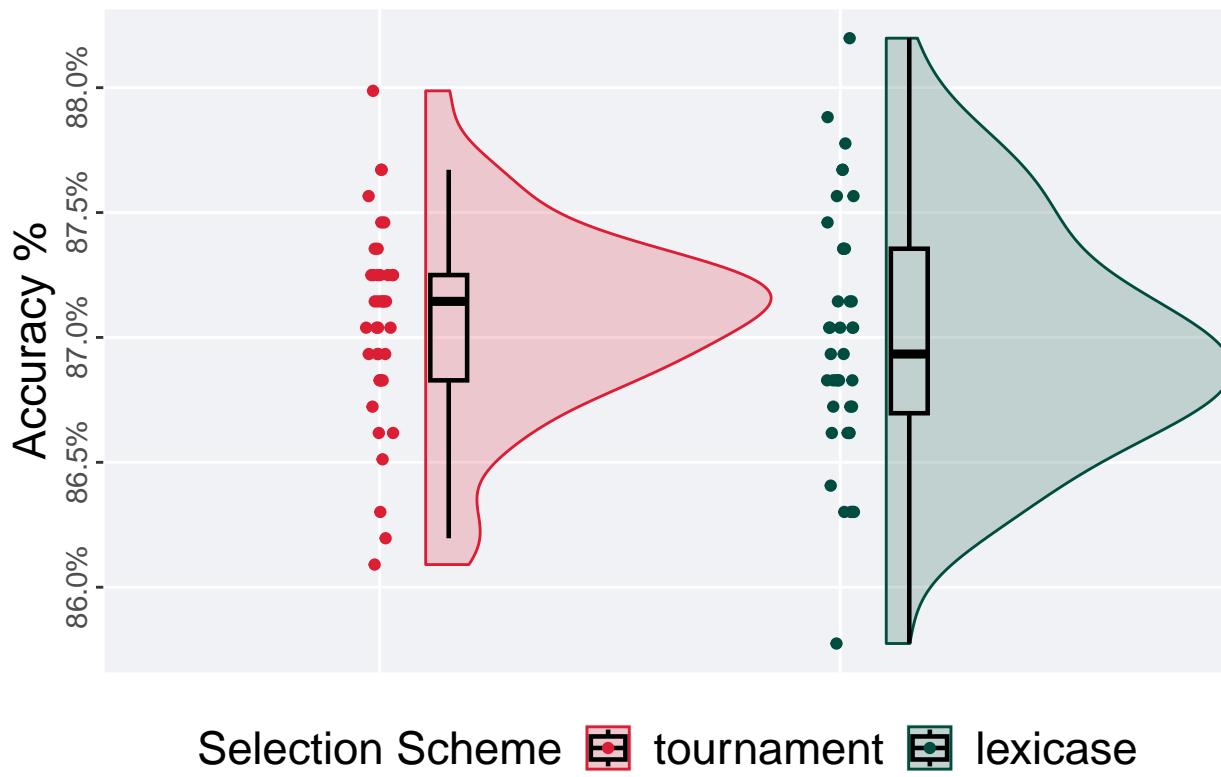
```
## [1] "observed_diff: 0.0725999889735864"
## [1] "lower: -2.01031173065856"
## [1] "upper: 2.010313713565"
## [1] "fail to reject null hypothesis"
## [1] "p-value: 0.96243"
```



14.3.2 Validation accuracy

```
validation_plot(filter(task_data, split == '50%'))
```

Accuracy on validation set



Summary statistics for the testing performance of the selection schemes at the 5% selection set split:

```
validation_accuracy_summary(filter(task_data, split == '50%'))
```

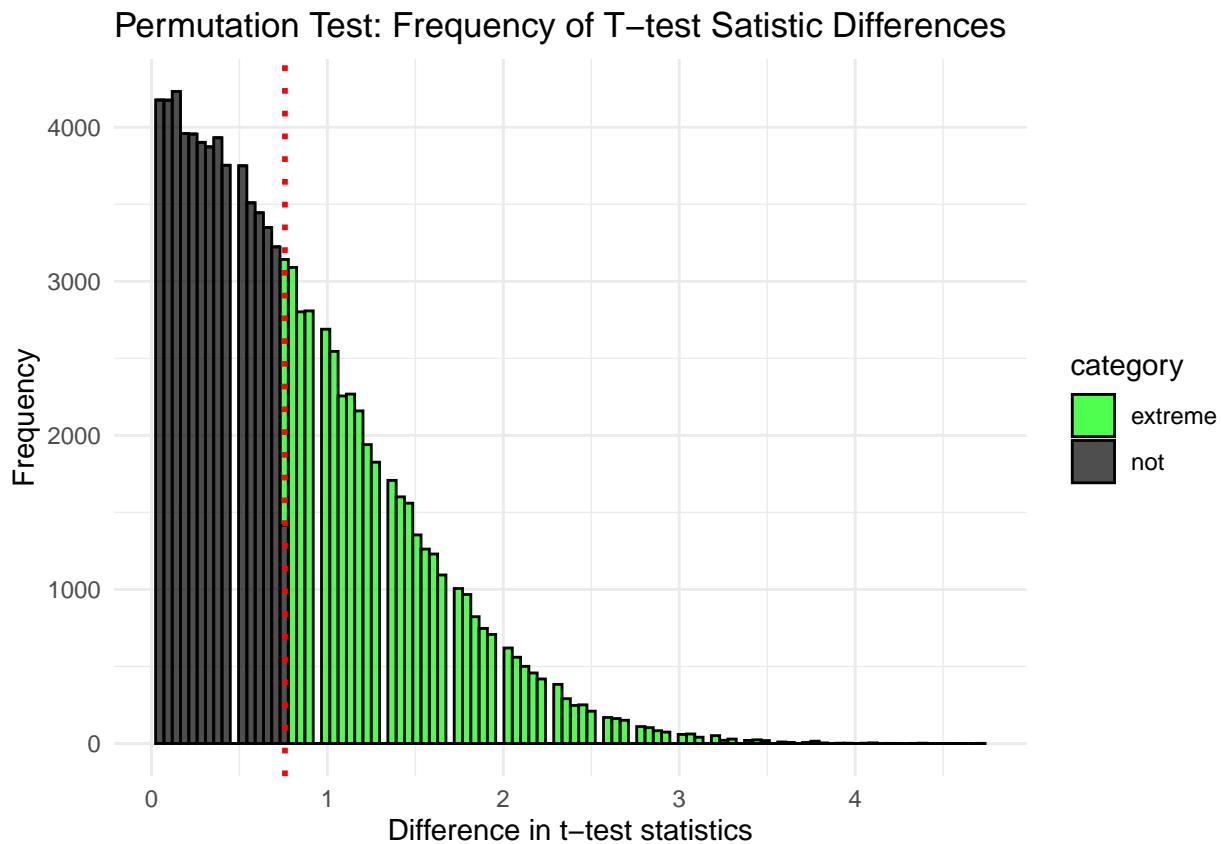
```
## # A tibble: 2 x 8
##   selection count na_cnt  min median  mean   max      IQR
##   <fct>     <int> <int> <dbl> <dbl> <dbl> <dbl> <dbl>
## 1 tournament     40     0 0.861 0.871 0.871 0.880 0.00421
## 2 lexicase       40     0 0.858 0.869 0.870 0.882 0.00659
```

The permutation test revealed that the results are:

```
tournament_results <- filter(task_data, split == '50%' & selection == 'tournament')
lexicase_results <- filter(task_data, split == '50%' & selection == 'lexicase')
```

```
permutation_test(tournament_results$training_performance,
                  lexicase_results$training_performance,
                  seed = 126,
                  alternative = "t")
```

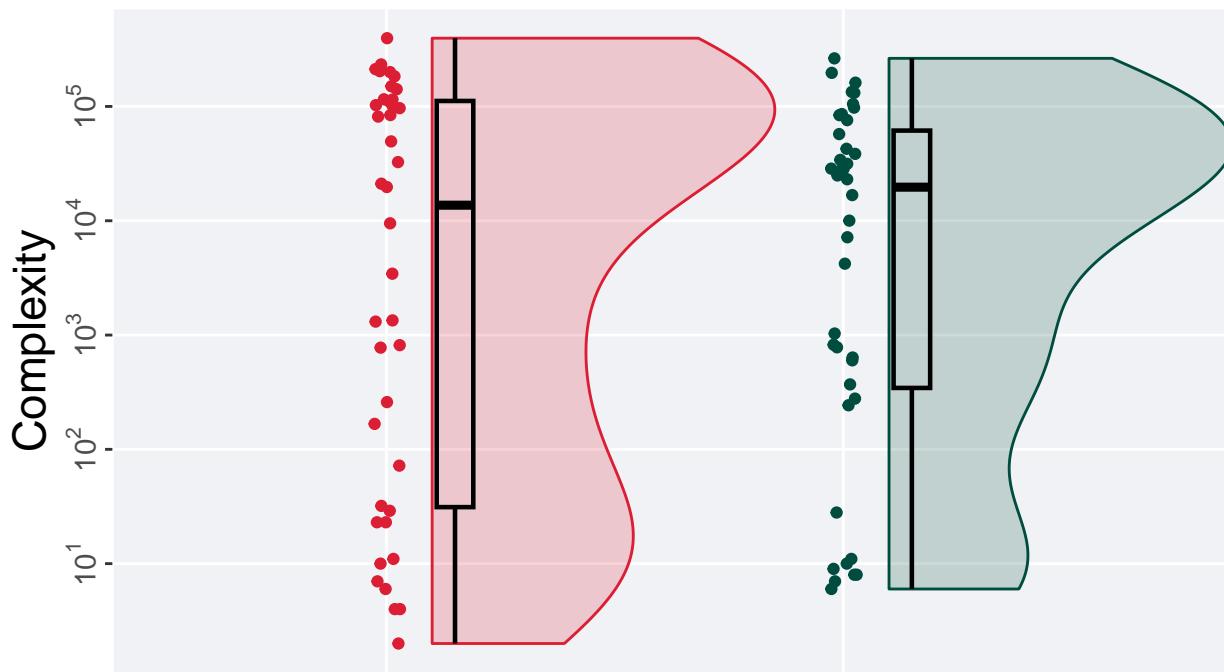
```
## [1] "observed_diff: 0.758730494171471"
## [1] "lower: -2.00476759324351"
## [1] "upper: 2.00476866020914"
## [1] "fail to reject null hypothesis"
## [1] "p-value: 0.45346"
```



14.3.3 Pipeline complexity

```
complexity_plot(filter(task_data, split == '50%'))
```

Pipeline Complexity



Selection Scheme  tournament  lexicase

Summary statistics for the pipeline complexity of the selection schemes at the 5% selection set split:

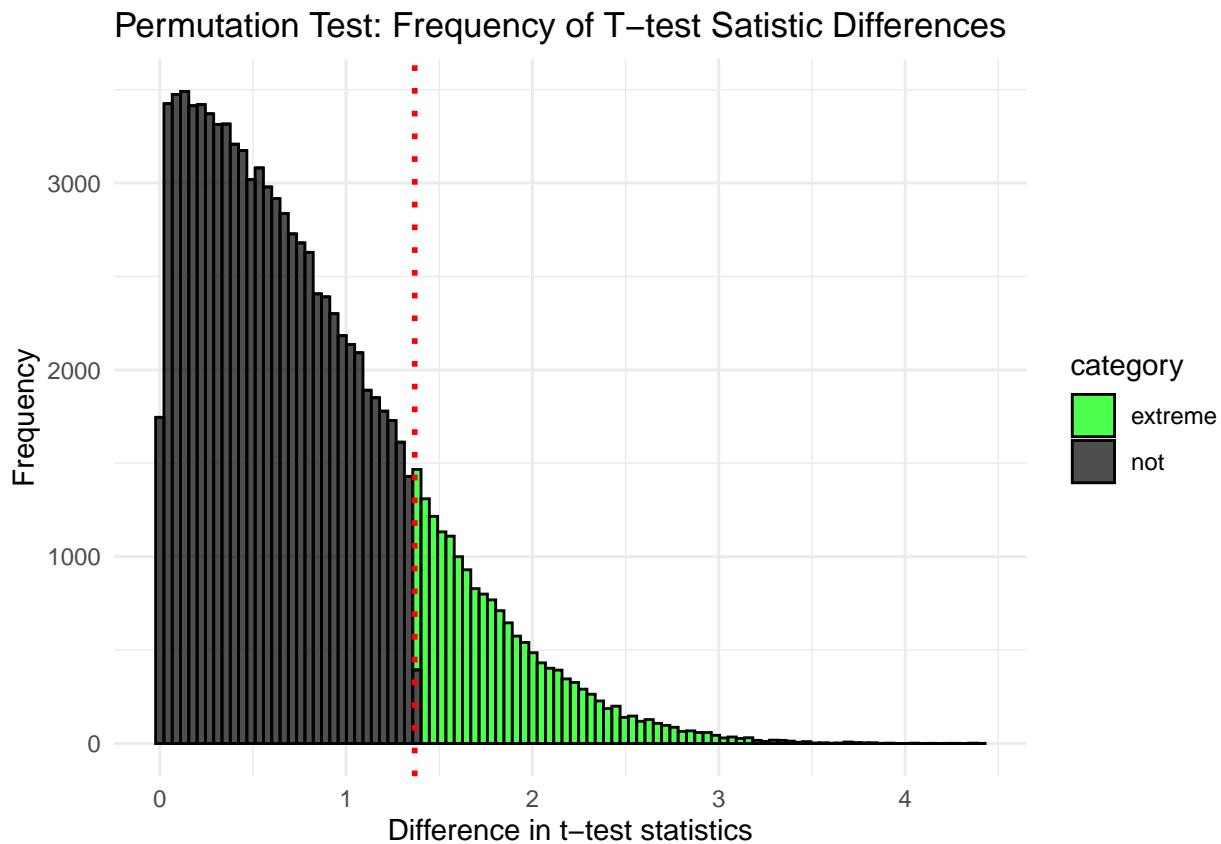
```
complexity_summary(filter(task_data, split == '50%'))
```

```
## # A tibble: 2 x 8
##   selection  count  na_cnt  min median  mean    max      IQR
##   <fct>     <int>   <int> <dbl> <dbl> <dbl> <int>    <dbl>
## 1 tournament     40      0     2 14611  66710. 395651 111684.
## 2 lexicase       40      0     6 19920. 42898. 263621  61630.
```

The permutation test revealed that the results are:

```
tournament_results <- filter(task_data, split == '50%' & selection == 'tournament')
lexicase_results <- filter(task_data, split == '50%' & selection == 'lexicase')
permutation_test(tournament_results$testing_complexity,
                 lexicase_results$testing_complexity,
                 seed = 250,
                 alternative = "t")
```

```
## [1] "observed_diff: 1.36830843807089"
## [1] "lower: -1.98108732120777"
## [1] "upper: 1.96791502944089"
## [1] "fail to reject null hypothesis"
## [1] "p-value: 0.17581"
```

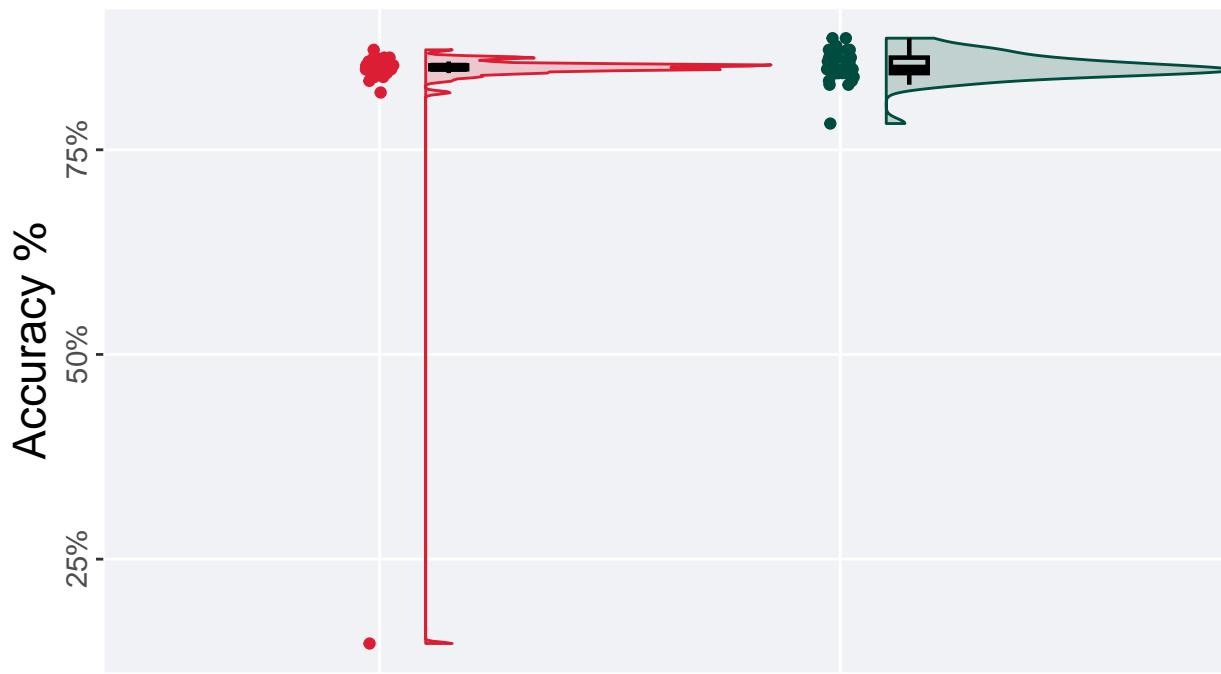


14.4 90%

14.4.1 Test accuracy

```
test_plot(filter(task_data, split == '90%'))
```

Accuracy on test set



Selection scheme tournament lexicase

Summary statistics for the testing performance of the selection schemes at the 5% selection set split:

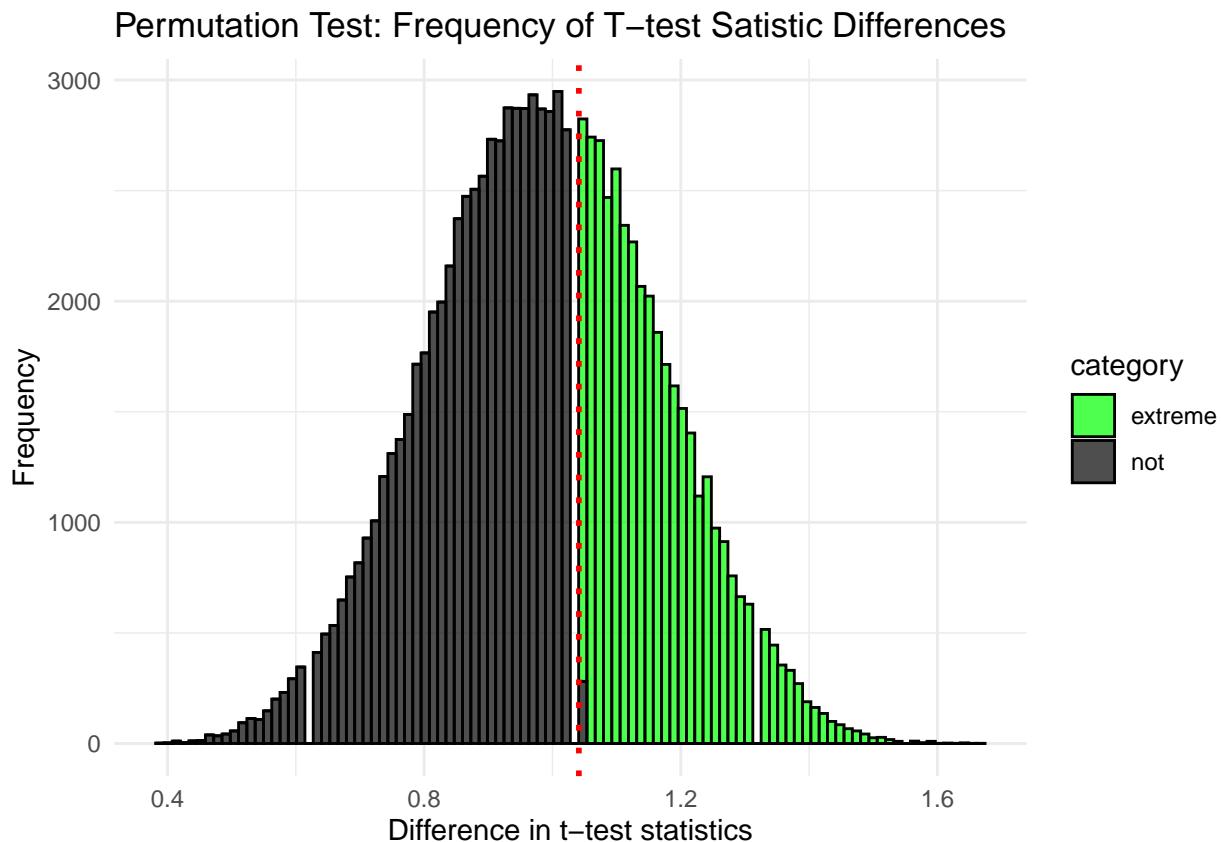
```
test_results_summary(filter(task_data, split == '90%'))
```

```
## # A tibble: 2 x 8
##   selection count na_cnt  min median  mean   max      IQR
##   <fct>     <int>  <int> <dbl>  <dbl> <dbl> <dbl>    <dbl>
## 1 tournament     40      0 0.147  0.851  0.833  0.872  0.00474
## 2 lexicase       40      0 0.782  0.848  0.851  0.886  0.0190
```

The permutation test revealed that the results are:

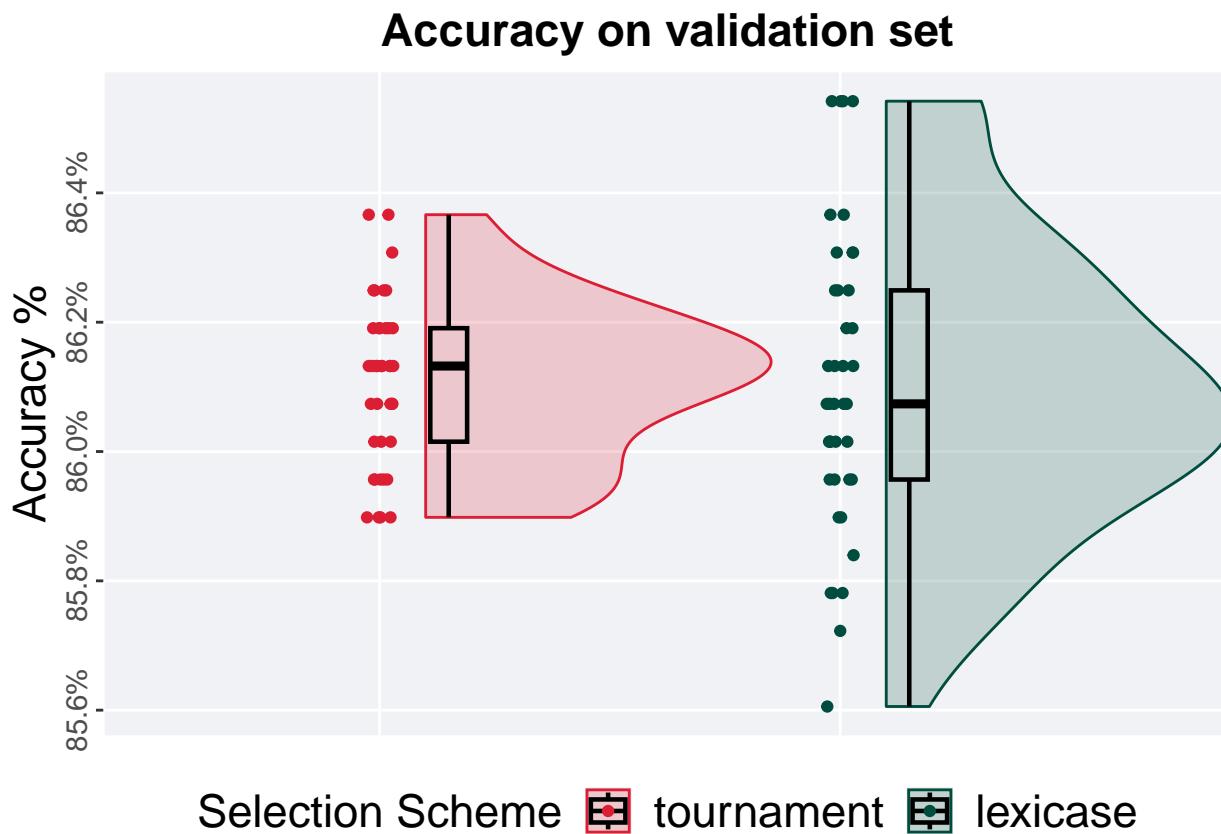
```
tournament_results <- filter(task_data, split == '90%' & selection == 'tournament')
lexicase_results <- filter(task_data, split == '90%' & selection == 'lexicase')
permutation_test(tournament_results$testing_performance,
                 lexicase_results$testing_performance,
                 seed = 127,
                 alternative = "t")
```

```
## [1] "observed_diff: -1.04113830032943"
## [1] "lower: -1.27085455535858"
## [1] "upper: 1.27085472731831"
## [1] "fail to reject null hypothesis"
## [1] "p-value: 0.39026"
```



14.4.2 Validation accuracy

```
validation_plot(filter(task_data, split == '90%'))
```



Summary statistics for the testing performance of the selection schemes at the 5% selection set split:

```
validation_accuracy_summary(filter(task_data, split == '90%'))
```

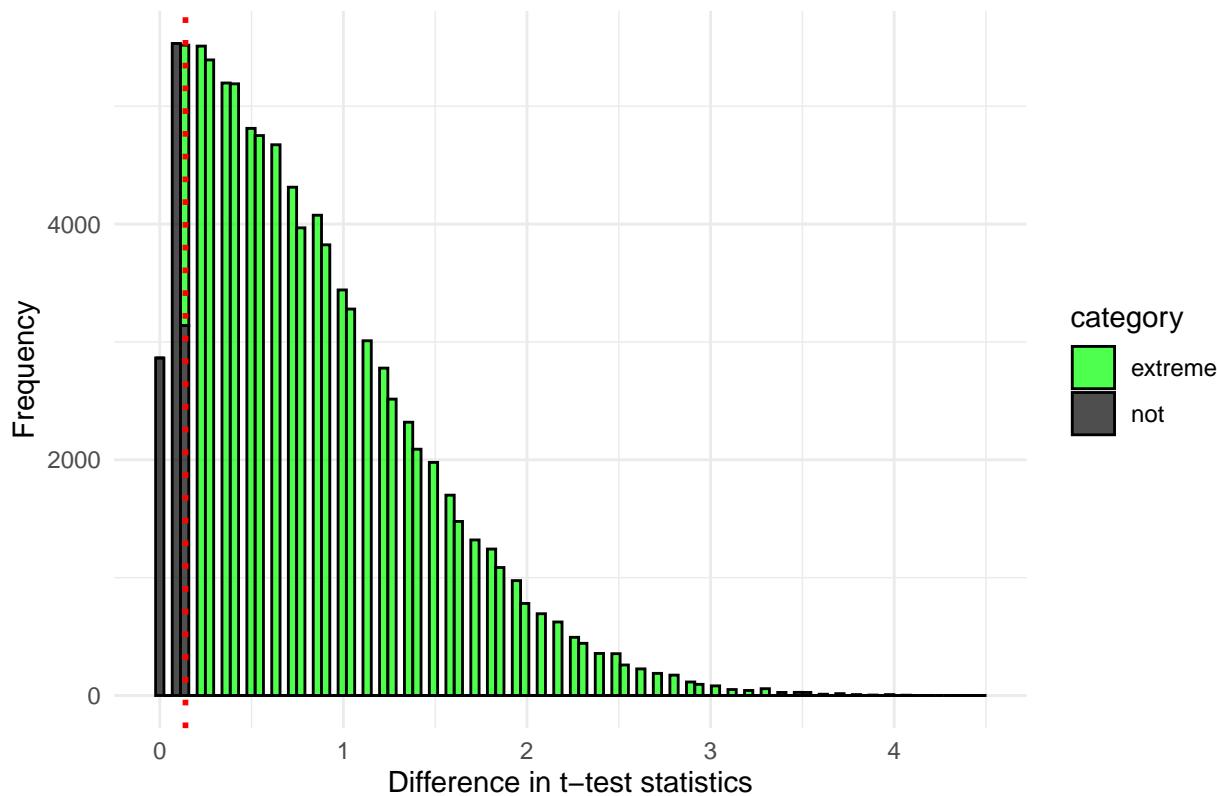
```
## # A tibble: 2 x 8
##   selection count na_cnt  min median  mean   max      IQR
##   <fct>     <int> <int> <dbl> <dbl> <dbl> <dbl>    <dbl>
## 1 tournament     40     0 0.859 0.861 0.861 0.864 0.00176
## 2 lexicase       40     0 0.856 0.861 0.861 0.865 0.00293
```

The permutation test revealed that the results are:

```
tournament_results <- filter(task_data, split == '90%' & selection == 'tournament')
lexicase_results <- filter(task_data, split == '90%' & selection == 'lexicase')
permutation_test(tournament_results$training_performance,
                 lexicase_results$training_performance,
                 seed = 128,
                 alternative = "t")
```

```
## [1] "observed_diff: 0.140015429975094"
## [1] "lower: -2.01007682745605"
## [1] "upper: 2.01007940828714"
## [1] "fail to reject null hypothesis"
## [1] "p-value: 0.88465"
```

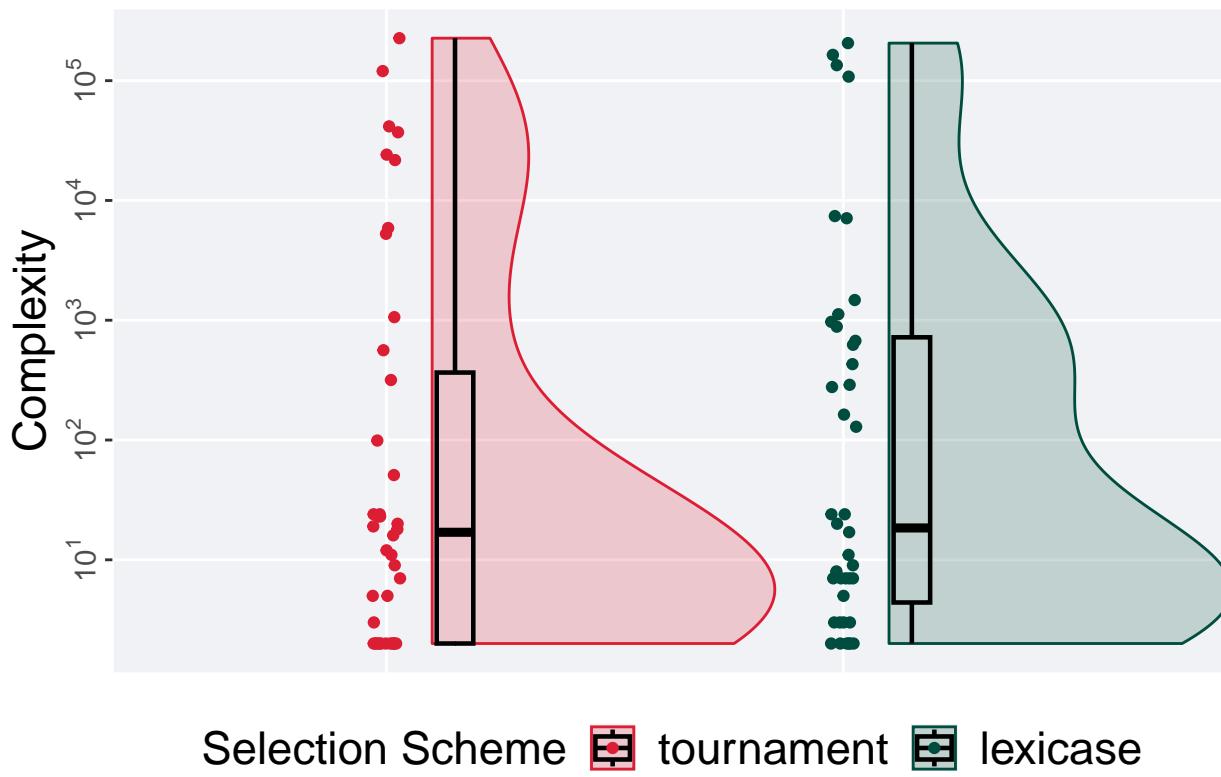
Permutation Test: Frequency of T-test Statistic Differences



14.4.3 Pipeline complexity

```
complexity_plot(filter(task_data, split == '90%'))
```

Pipeline Complexity



Summary statistics for the pipeline complexity of the selection schemes at the 5% selection set split:

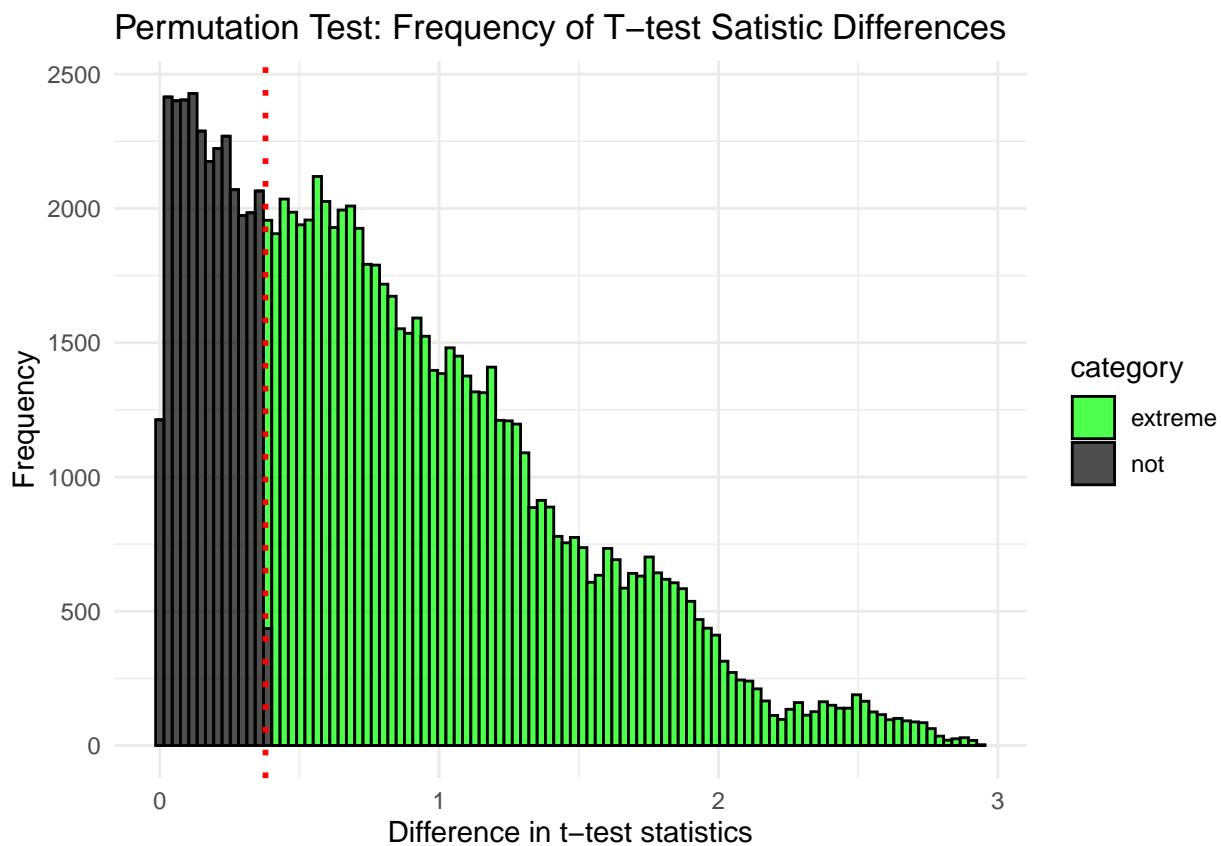
```
complexity_summary(filter(task_data, split == '90%'))
```

```
## # A tibble: 2 x 8
##   selection count na_cnt  min median   mean   max   IQR
##   <fct>     <int> <int> <dbl> <dbl> <dbl> <int> <dbl>
## 1 tournament     40     0     2    17  12114. 226291   376.
## 2 lexicase       40     0     2   18.5 15866. 205871   721
```

The permutation test revealed that the results are:

```
tournament_results <- filter(task_data, split == '90%' & selection == 'tournament')
lexicase_results <- filter(task_data, split == '90%' & selection == 'lexicase')
permutation_test(tournament_results$testing_complexity,
                 lexicase_results$testing_complexity,
                 seed = 251,
                 alternative = "t")
```

```
## [1] "observed_diff: -0.378507777635726"
## [1] "lower: -1.94090914169516"
## [1] "upper: 1.934819501353"
## [1] "fail to reject null hypothesis"
## [1] "p-value: 0.71655"
```

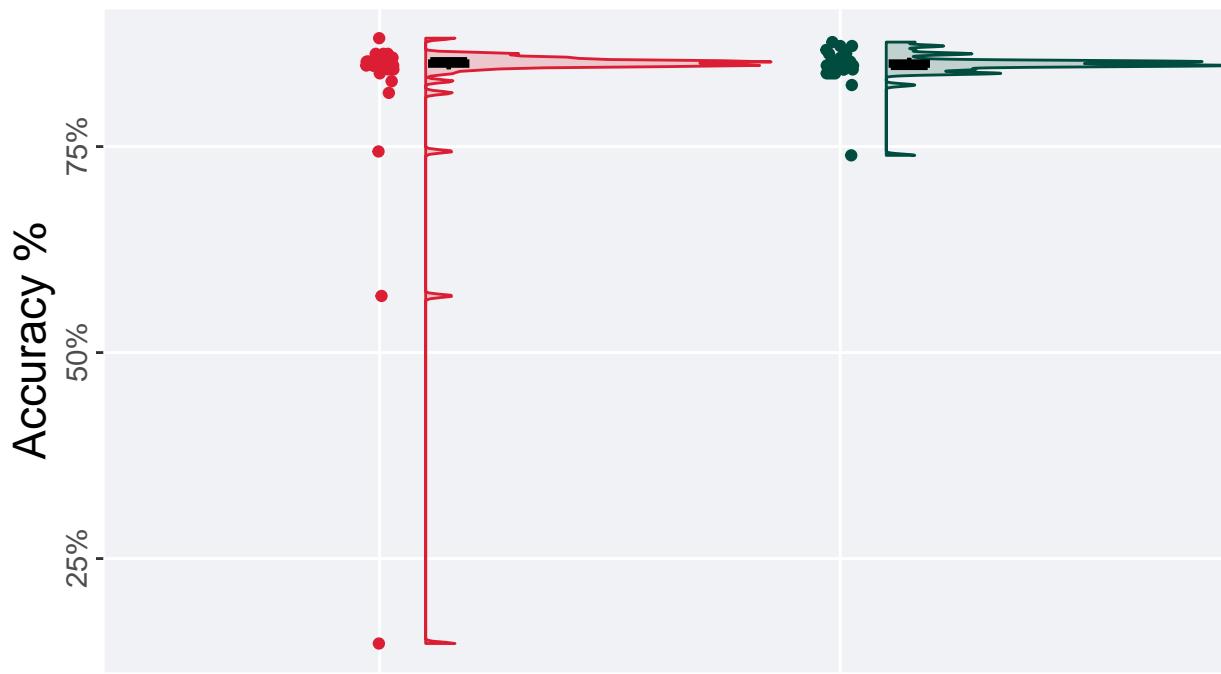


14.5 95%

14.5.1 Test accuracy

```
test_plot(filter(task_data, split == '95%'))
```

Accuracy on test set



Selection scheme tournament lexicase

Summary statistics for the testing performance of the selection schemes at the 5% selection set split:

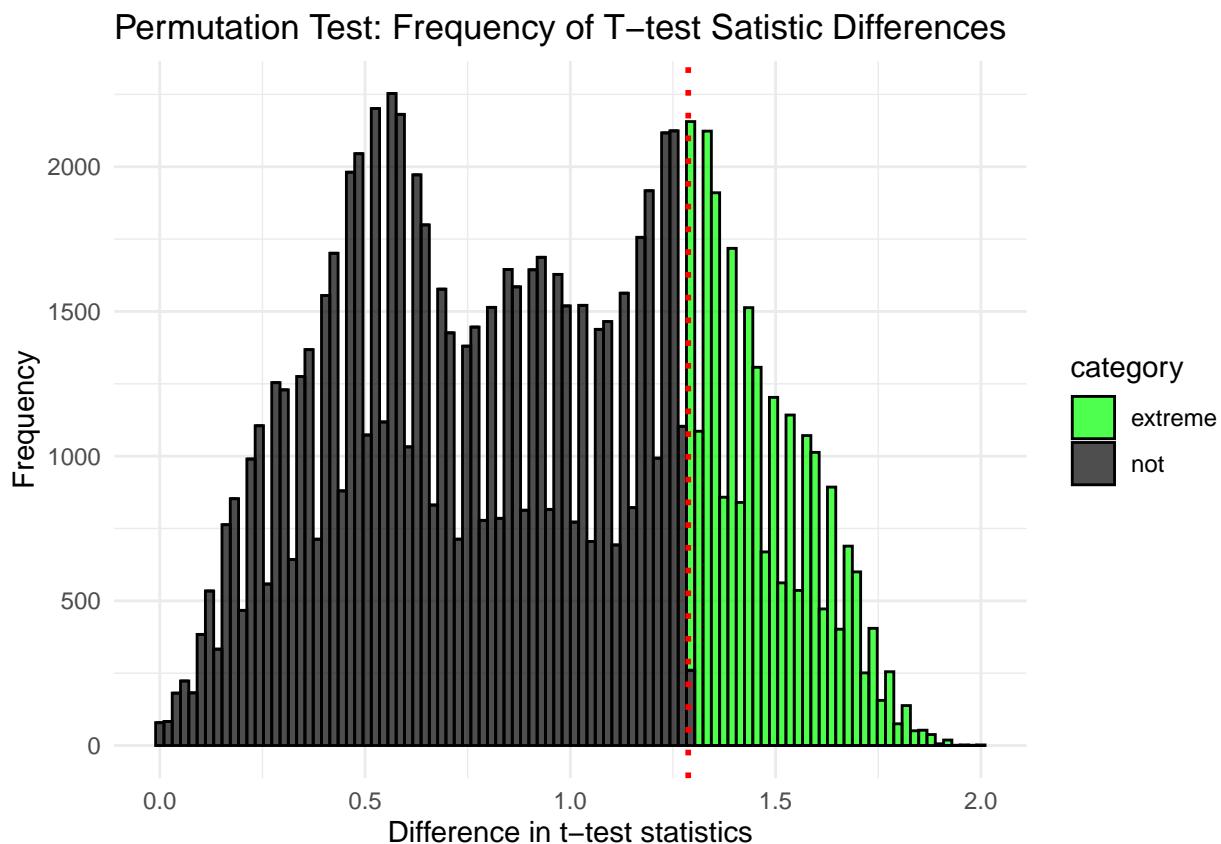
```
test_results_summary(filter(task_data, split == '95%'))
```

```
## # A tibble: 2 x 8
##   selection count na_cnt  min median  mean   max      IQR
##   <fct>     <int> <int> <dbl> <dbl> <dbl> <dbl>    <dbl>
## 1 tournament     40     0 0.147  0.853  0.824  0.882  0.00474
## 2 lexicase       40     0 0.739  0.848  0.849  0.877  0.00474
```

The permutation test revealed that the results are:

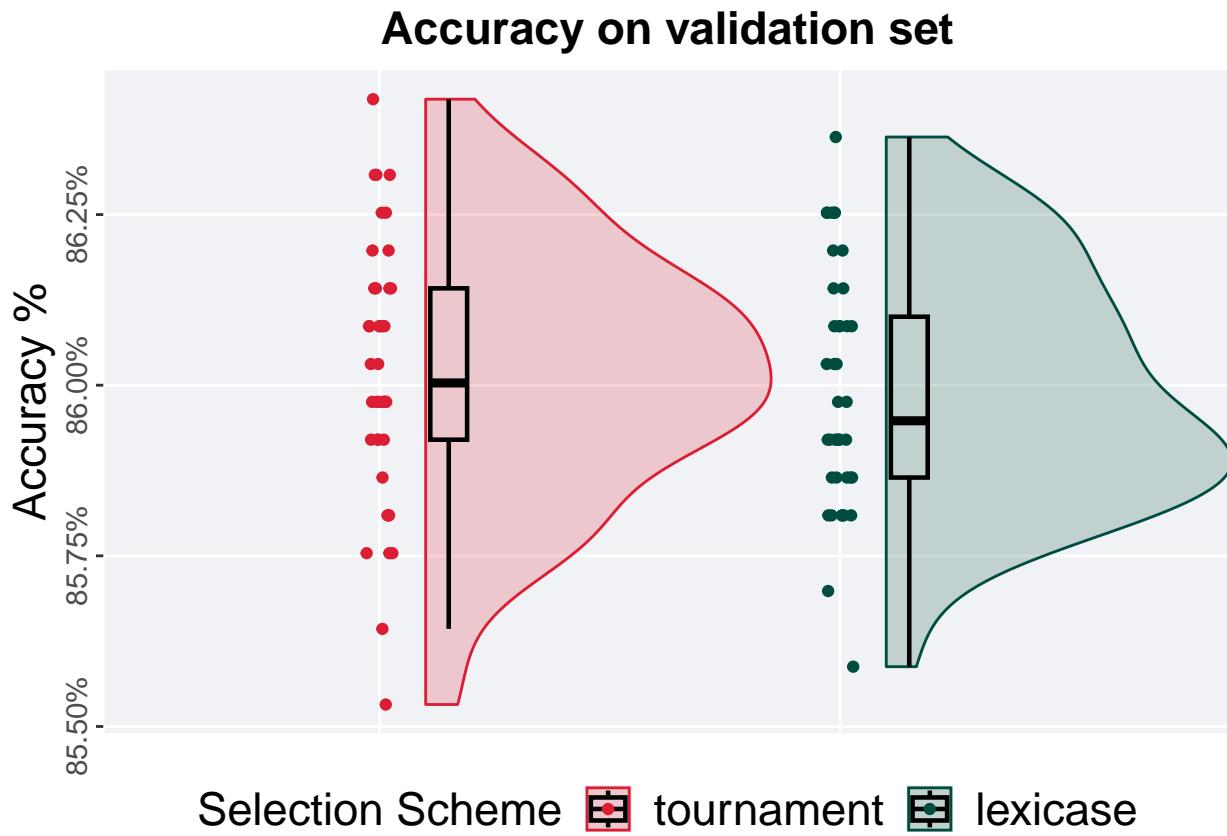
```
tournament_results <- filter(task_data, split == '95%' & selection == 'tournament')
lexicase_results <- filter(task_data, split == '95%' & selection == 'lexicase')
permutation_test(tournament_results$testing_performance,
                 lexicase_results$testing_performance,
                 seed = 129,
                 alternative = "t")
```

```
## [1] "observed_diff: -1.28720945209971"
## [1] "lower: -1.60430871420884"
## [1] "upper: 1.59151668419804"
## [1] "fail to reject null hypothesis"
## [1] "p-value: 0.23956"
```



14.5.2 Validation accuracy

```
validation_plot(filter(task_data, split == '95%'))
```



Summary statistics for the testing performance of the selection schemes at the 5% selection set split:

```
validation_accuracy_summary(filter(task_data, split == '95%'))
```

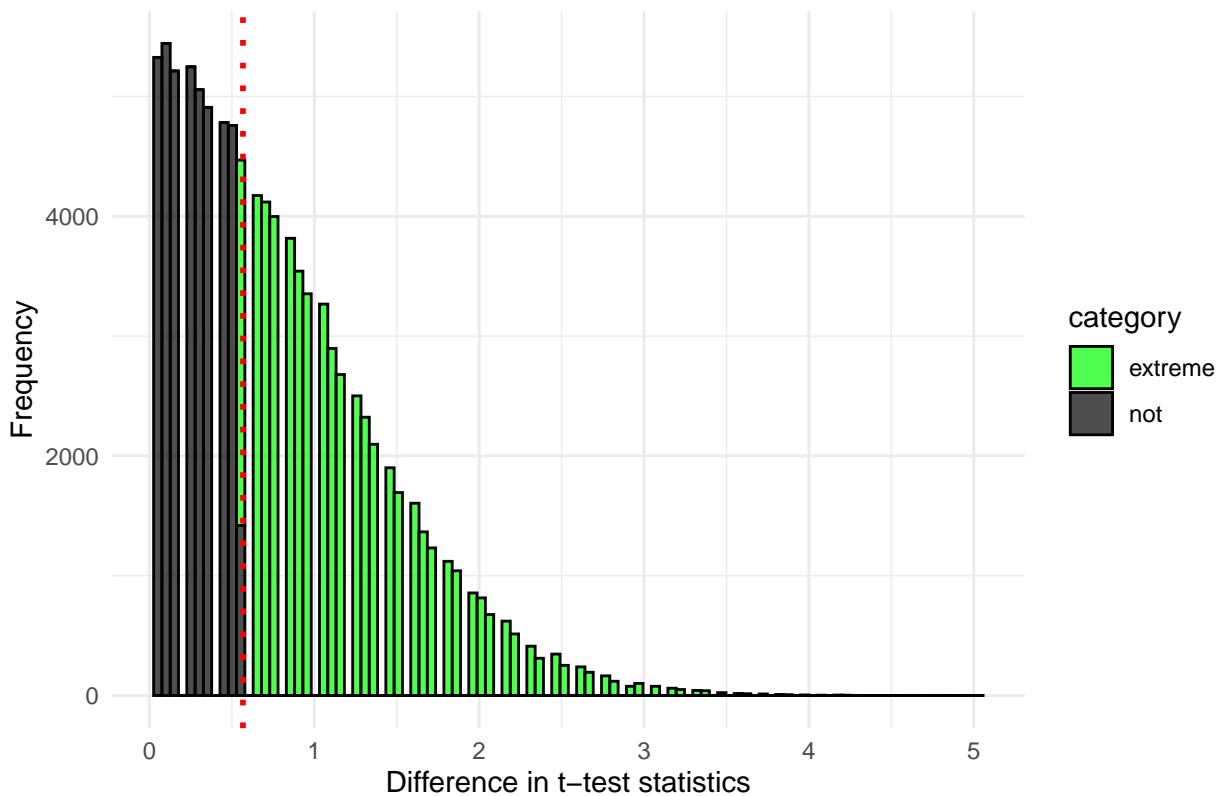
```
## # A tibble: 2 x 8
##   selection count na_cnt  min median  mean   max      IQR
##   <fct>     <int> <int> <dbl> <dbl> <dbl> <dbl>    <dbl>
## 1 tournament     40     0 0.855 0.860 0.860 0.864 0.00222
## 2 lexicase       40     0 0.856 0.859 0.860 0.864 0.00236
```

The permutation test revealed that the results are:

```
tournament_results <- filter(task_data, split == '95%' & selection == 'tournament')
lexicase_results <- filter(task_data, split == '95%' & selection == 'lexicase')
permutation_test(tournament_results$training_performance,
                 lexicase_results$training_performance,
                 seed = 130,
                 alternative = "t")
```

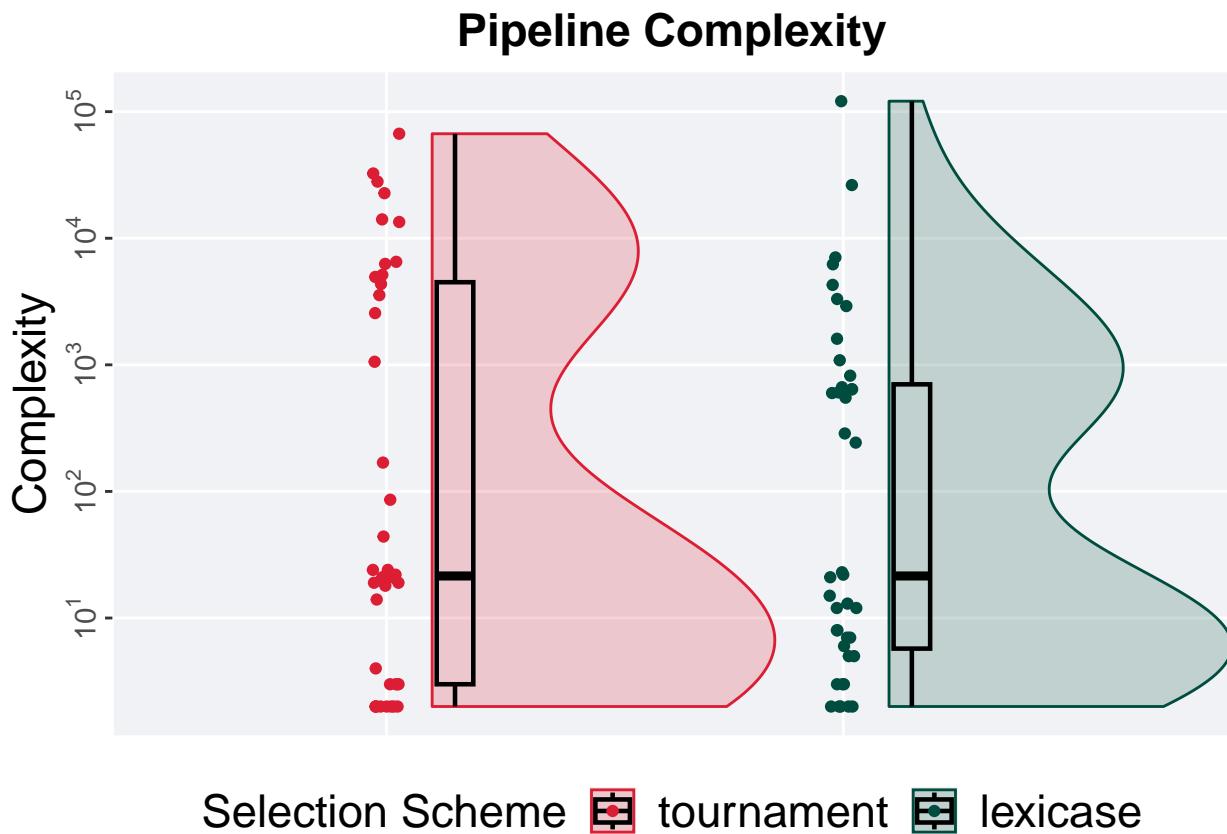
```
## [1] "observed_diff: 0.567026129247492"
## [1] "lower: -2.01431137808478"
## [1] "upper: 2.01431267396974"
## [1] "fail to reject null hypothesis"
## [1] "p-value: 0.57846"
```

Permutation Test: Frequency of T-test Statistic Differences



14.5.3 Pipeline complexity

```
complexity_plot(filter(task_data, split == '95%'))
```



Summary statistics for the pipeline complexity of the selection schemes at the 5% selection set split:

```
complexity_summary(filter(task_data, split == '95%'))
```

```
## # A tibble: 2 x 8
##   selection count na_cnt  min median  mean     max   IQR
##   <fct>     <int> <int> <dbl> <dbl> <dbl> <int> <dbl>
## 1 tournament     40     0     2  21.5  5315.  66917  4498.
## 2 lexicase       40     0     2  21.5  4474. 120951    699
```

The permutation test revealed that the results are:

```
tournament_results <- filter(task_data, split == '95%' & selection == 'tournament')
lexicase_results <- filter(task_data, split == '95%' & selection == 'lexicase')
permutation_test(tournament_results$testing_complexity,
                 lexicase_results$testing_complexity,
                 seed = 252,
                 alternative = "t")

## [1] "observed_diff: 0.229521488372537"
## [1] "lower: -1.7911392796415"
## [1] "upper: 1.78081606429978"
## [1] "fail to reject null hypothesis"
## [1] "p-value: 0.84587"
```

