

# Supplemental Material for Phylogeny-informed fitness estimation

Alexander Lalejini, Matthew Andres Moreno, Jose Guadalupe Hernandez, and Emily Dolson

2023-05-15



# Contents

<b>1</b>	<b>Introduction</b>	<b>5</b>
1.1	About our supplemental material . . . . .	5
1.2	Contributing authors . . . . .	5
<b>2</b>	<b>Data Availability</b>	<b>7</b>
2.1	Source code . . . . .	7
2.2	Training and testing sets . . . . .	7
2.3	Experimental results . . . . .	7
<b>3</b>	<b>SignalGP instruction set</b>	<b>9</b>
3.1	Default Instructions . . . . .	10
3.2	Problem-specific instructions . . . . .	11
<b>4</b>	<b>Selection scheme diagnostic experiments</b>	<b>13</b>
4.1	Dependencies . . . . .	13
4.2	Setup . . . . .	14
4.3	Contradictory objectives diagnostic . . . . .	20
4.4	Multi-path exploration diagnostic . . . . .	32
4.5	Manuscript figures . . . . .	43
<b>5</b>	<b>Program synthesis experiments</b>	<b>51</b>
5.1	Dependencies . . . . .	51
5.2	Setup . . . . .	52
5.3	Problem-solving success . . . . .	58



# Chapter 1

## Introduction

This is the supplemental material for our submission to Genetic Programming Theory and Practice XX. 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.

The source code/configuration files for this supplemental material can be found in this GitHub repository.

Our supplemental material includes the following:

- Data availability (Section 2)
- GP instruction set (Section 3)
- Analysis notebooks for each experiment (including source code)
  - Selection scheme diagnostics (Section 4)
  - Program synthesis problems (Section TODO)

### 1.2 Contributing authors

- Alexander Lalejini
- Matthew Andres Moreno
- Jose Guadalupe Hernandez

- Emily Dolson

## Chapter 2

# Data Availability

### 2.1 Source code

The source code for this work is publicly accessible on GitHub: <https://github.com/amlalejini/phylogeny-informed-evaluation>.

#### 2.1.1 Experiment software dependencies

- SignalGP: <https://github.com/amlalejini/SignalGP>
  - commit hash: 8cc56a79aa6b2655bacefcc7b63ee6a859c730f3
- Empirical: <https://github.com/devosoft/Empirical>
  - commit hash: 5955a1cae2a5de36aa3a65df060a56b38f575bd0

### 2.2 Training and testing sets

The training and testing sets used for program synthesis problems can be found on GitHub: <https://github.com/amlalejini/phylogeny-informed-evaluation/tree/main/experiments/2023-05-08-psynth/hpc/config>.

### 2.3 Experimental results

All of our experimental data is available online from our OSF repository: <https://osf.io/wxckn/>





## Chapter 3

# SignalGP instruction set

Below, we document the instruction set used in our GP system for our 2023 GPTP experiments.

Abbreviations:

- EOP: End of program
- Reg: local register
  - Reg[0] indicates the value at the register specified by an instruction's first *argument*, Reg[1] indicates the value at the register specified by an instruction's second argument, and Reg[2] indicates the value at the register specified by the instruction's third argument.
  - Reg[0], Reg[1], *etc.*: Register 0, Register 1, *etc.*
- Input: input buffer
  - Follows same scheme as Reg
- Output: output buffer
  - Follows same scheme as Reg
- Global: global memory buffer
  - Follows same scheme as Reg
- Arg: Instruction argument
  - Arg[i] indicates the i'th instruction argument (an integer encoded in the genome)
  - E.g., Arg[0] is an instruction's first argument

Instructions that would produce undefined behavior (e.g., division by zero) are treated as no operations.

### 3.1 Default Instructions

I.e., instructions used across all diagnostic tasks.

Instruction	Arguments Used	Description
Nop	0	No operation
Not	1	$\text{Reg}[0] = \neg \text{Reg}[0]$
Inc	1	$\text{Reg}[0] = \text{Reg}[0] + 1$
Dec	1	$\text{Reg}[0] = \text{Reg}[0] - 1$
Add	3	$\text{Reg}[2] = \text{Reg}[0] +$ $\text{Reg}[1]$
Sub	3	$\text{Reg}[2] = \text{Reg}[0] -$ $\text{Reg}[1]$
Mult	3	$\text{Reg}[2] = \text{Reg}[0] *$ $\text{Reg}[1]$
Div	3	$\text{Reg}[2] = \text{Reg}[0] /$ $\text{Reg}[1]$
Mod	3	$\text{Reg}[2] = \text{Reg}[0] \%$ $\text{Reg}[1]$
Nand	2	$\text{Reg}[2] = \neg(\text{Reg}[0] \&$ $\text{Reg}[1])$
TestEqu	3	$\text{Reg}[2] = \text{Reg}[0] ==$ $\text{Reg}[1]$
TestNEqu	3	$\text{Reg}[2] = \text{Reg}[0] !=$ $\text{Reg}[1]$
TestLess	3	$\text{Reg}[2] = \text{Reg}[0] <$ $\text{Reg}[1]$
TestLessEqu	3	$\text{Reg}[2] = \text{Reg}[0] <=$ $\text{Reg}[1]$
TestGreater	3	$\text{Reg}[2] = \text{Reg}[0] >$ $\text{Reg}[1]$
TestGreaterEqu	3	$\text{Reg}[2] = \text{Reg}[0] >=$ $\text{Reg}[1]$
SetMem	2	$\text{Reg}[0] = \text{Arg}[1]$
Terminal	1	$\text{Reg}[0] = \text{double value}$ encoded by instruction tag
CopyMem	2	$\text{Reg}[0] = \text{Reg}[1]$
SwapMem	2	$\text{Swap}(\text{Reg}[0], \text{Reg}[1])$
InputToWorking	2	$\text{Reg}[1] = \text{Input}[0]$
WorkingToOutput	2	$\text{Output}[1] = \text{Reg}[0]$
If	1	If $\text{Reg}[0] != 0$ , proceed. Otherwise skip to the next <code>Close</code> or <code>EOP</code> .

Instruction	Arguments Used	Description
<b>While</b>	1	While <code>Reg[0] != 0</code> , loop. Otherwise skip to next <b>Close</b> or EOP.
<b>Close</b>	0	Indicate the end of a control block of code (e.g., loop, if).
<b>Break</b>	0	Break out of current control flow (e.g., loop).
<b>Call</b>	0	Call a function, using this instruction's tag to determine which function is called.
<b>Routine</b>	0	Same as call, but local memory is shared. Sort of like a jump that will jump back when the routine ends.
<b>Return</b>	0	Return from the current function call.
<b>WorkingToGlobal</b>	2	<code>Global[1] = Reg[0]</code>
<b>GlobalToWorking</b>	2	<code>Reg[1] = Global[0]</code>
<b>FullGlobalToWorking</b>	0	Copy entire global memory buffer into working memory buffer
<b>FullWorkingToGlobal</b>	0	Copy entire working memory buffer into global memory buffer

Note that **Nand** performs a bitwise operation.

## 3.2 Problem-specific instructions

Each problem has problem-specific instructions for producing output.

### 3.2.1 Fizz Buzz

- `SubmitFizz`
- `SubmitBuzz`

- SubmitFizzBuzz
- SubmitEcho

### **3.2.2 Median**

- SubmitOutput

### **3.2.3 Grade**

- SubmitA
- SubmitB
- SubmitC
- SubmitD
- SubmitF

### **3.2.4 Small or large**

- SubmitSmall
- SubmitLarge
- SubmitNeither

## Chapter 4

# Selection scheme diagnostic experiments

```
experiment_slug <- "2023-05-10-diagnostics"

working_directory <- paste0(
  "experiments/",
  experiment_slug,
  "/analysis/"
)

if (exists("bookdown_wd_prefix")) {
  working_directory <- paste0(
    bookdown_wd_prefix,
    working_directory
  )
}
```

### 4.1 Dependencies

```
library(tidyverse)
```

```
## -- Attaching packages -----
## v ggplot2 3.3.6      v purrr   1.0.1
## v tibble  3.1.8      v dplyr  1.1.0
## v tidyr   1.3.0      v stringr 1.5.0
```

```
## v readr 2.1.3 v forcats 0.5.2
## -- Conflicts -----
## x dplyr::filter() masks stats::filter()
## x dplyr::lag() masks stats::lag()

library(ggplot2)
library(cowplot)
library(RColorBrewer)
library(khroma)
library(rstatix)

##
## Attaching package: 'rstatix'
##
## The following object is masked from 'package:stats':
##
## filter

library(knitr)
source("https://gist.githubusercontent.com/benmarwick/2a1bb0133ff568cbe28d/raw/fb53bd9")

print(version)

##
## platform aarch64-apple-darwin20
## arch aarch64
## os darwin20
## system aarch64, darwin20
## status
## major 4
## minor 2.1
## year 2022
## month 06
## day 23
## svn rev 82513
## language R
## version.string R version 4.2.1 (2022-06-23)
## nickname Funny-Looking Kid
```

## 4.2 Setup

```
# Configure our default graphing theme
theme_set(theme_cowplot())
# Create a directory to store plots
plot_directory <- paste0(working_directory, "plots/")
dir.create(plot_directory, showWarnings=FALSE)
```

### 4.2.1 Load experiment summary data

```
summary_data_loc <- paste0(working_directory, "data/aggregate.csv")
summary_data <- read_csv(summary_data_loc)
```

```
## Rows: 520 Columns: 49
## -- Column specification -----
## Delimiter: ","
## chr (5): DIAGNOSTIC, EVAL_FIT_EST_MODE, EVAL_MODE, SELECTION, STOP_MODE
## dbl (44): ACCURACY, CREDIT, DIAGNOSTIC_DIMENSIONALITY, GENE_LOWER_BND, GENE...
##
## i Use 'spec()' to retrieve the full column specification for this data.
## i Specify the column types or set 'show_col_types = FALSE' to quiet this message.
```

```
summary_data <- summary_data %>%
  mutate(
    eval_mode_row = case_when(
      EVAL_MODE == "full" & TEST_DOWNSAMPLE_RATE == "1" ~ "down-sample",
      EVAL_MODE == "full" & NUM_COHORTS == "1" ~ "cohort",
      .default = EVAL_MODE
    ),
    evals_per_gen = case_when(
      EVAL_MODE == "cohort-full-compete" ~ 1.0 / NUM_COHORTS,
      EVAL_MODE == "cohort" ~ 1.0 / NUM_COHORTS,
      EVAL_MODE == "down-sample" ~ TEST_DOWNSAMPLE_RATE,
      EVAL_MODE == "full" ~ 1.0
    ),
    EVAL_FIT_EST_MODE = case_when(
      EVAL_FIT_EST_MODE == "ancestor-opt" ~ "ancestor",
      EVAL_FIT_EST_MODE == "relative-opt" ~ "relative",
      .default = EVAL_FIT_EST_MODE
    ),
    .keep = "all"
  ) %>%
  mutate(
    evals_per_gen = as.factor(evals_per_gen),
```

```

eval_mode_row = as.factor(eval_mode_row),
DIAGNOSTIC = as.factor(DIAGNOSTIC),
SELECTION = as.factor(SELECTION),
EVAL_MODE = as.factor(EVAL_MODE),
NUM_COHORTS = as.factor(NUM_COHORTS),
TEST_DOWNSAMPLE_RATE = as.factor(TEST_DOWNSAMPLE_RATE),
EVAL_FIT_EST_MODE = factor(
  EVAL_FIT_EST_MODE,
  levels = c(
    "none",
    "ancestor",
    "relative"
  ),
  labels = c(
    "None",
    "Ancestor",
    "Relative"
  )
)
)
)

# Split summary data on diagnostic
con_obj_summary_data <- filter(
  summary_data,
  DIAGNOSTIC == "contradictory-objectives"
)

explore_summary_data <- filter(
  summary_data,
  DIAGNOSTIC == "multipath-exploration"
)

```

## 4.2.2 Load experiment time series data

```

ts_data_loc <- paste0(working_directory, "data/time_series.csv")
ts_data <- read_csv(ts_data_loc)

```

```

## Rows: 104000 Columns: 19
## -- Column specification -----
## Delimiter: ","
## chr  (4): DIAGNOSTIC, EVAL_FIT_EST_MODE, EVAL_MODE, SELECTION
## dbl  (15): NUM_COHORTS, SEED, TEST_DOWNSAMPLE_RATE, entropy_selected_ids, eva...
##

```



```
## i Use 'spec()' to retrieve the full column specification for this data.
## i Specify the column types or set 'show_col_types = FALSE' to quiet this message.
```

```
ts_data <- ts_data %>%
  mutate(
    eval_mode_row = case_when(
      EVAL_MODE == "full" & TEST_DOWNSAMPLE_RATE == "1" ~ "down-sample",
      EVAL_MODE == "full" & NUM_COHORTS == "1" ~ "cohort",
      .default = EVAL_MODE
    ),
    evals_per_gen = case_when(
      EVAL_MODE == "cohort-full-compete" ~ 1.0 / NUM_COHORTS,
      EVAL_MODE == "cohort" ~ 1.0 / NUM_COHORTS,
      EVAL_MODE == "down-sample" ~ TEST_DOWNSAMPLE_RATE,
      EVAL_MODE == "full" ~ 1.0
    ),
    EVAL_FIT_EST_MODE = case_when(
      EVAL_FIT_EST_MODE == "ancestor-opt" ~ "ancestor",
      EVAL_FIT_EST_MODE == "relative-opt" ~ "relative",
      .default = EVAL_FIT_EST_MODE
    ),
    .keep = "all"
  ) %>%
  mutate(
    evals_per_gen = as.factor(evals_per_gen),
    DIAGNOSTIC = as.factor(DIAGNOSTIC),
    SELECTION = as.factor(SELECTION),
    EVAL_MODE = as.factor(EVAL_MODE),
    NUM_COHORTS = as.factor(NUM_COHORTS),
    TEST_DOWNSAMPLE_RATE = as.factor(TEST_DOWNSAMPLE_RATE),
    EVAL_FIT_EST_MODE = factor(
      EVAL_FIT_EST_MODE,
      levels = c(
        "none",
        "ancestor",
        "relative"
      ),
      labels = c(
        "None",
        "Ancestor",
        "Relative"
      )
    )
  )
```

```

con_obj_ts_data <- ts_data %>%
  filter(DIAGNOSTIC == "contradictory-objectives")

explore_ts_data <- ts_data %>%
  filter(DIAGNOSTIC == "multipath-exploration")

```

### 4.2.3 Load estimate source distributions

```

est_source_data <- read_csv(
  paste0(working_directory, "data/phylo-est-info.csv")
)

```

```

## Rows: 520 Columns: 38
## -- Column specification -----
## Delimiter: ","
## chr (6): OUTPUT_DIR, DIAGNOSTIC, STOP_MODE, SELECTION, EVAL_MODE, EVAL_FIT_...
## dbl (32): SNAPSHOT_INTERVAL, OUTPUT_SUMMARY_DATA_INTERVAL, MUTATE_STD, TARGE...
##
## i Use 'spec()' to retrieve the full column specification for this data.
## i Specify the column types or set 'show_col_types = FALSE' to quiet this message.

```

```

est_source_data <- est_source_data %>%
  mutate(
    eval_mode_row = case_when(
      EVAL_MODE == "full" & TEST_DOWNSAMPLE_RATE == "1" ~ "down-sample",
      EVAL_MODE == "full" & NUM_COHORTS == "1" ~ "cohort",
      .default = EVAL_MODE
    ),
    evals_per_gen = case_when(
      EVAL_MODE == "cohort-full-compete" ~ 1.0 / NUM_COHORTS,
      EVAL_MODE == "cohort" ~ 1.0 / NUM_COHORTS,
      EVAL_MODE == "down-sample" ~ TEST_DOWNSAMPLE_RATE,
      EVAL_MODE == "full" ~ 1.0
    ),
    EVAL_FIT_EST_MODE = case_when(
      EVAL_FIT_EST_MODE == "ancestor-opt" ~ "ancestor",
      EVAL_FIT_EST_MODE == "relative-opt" ~ "relative",
      .default = EVAL_FIT_EST_MODE
    ),
    .keep = "all"
  ) %>%
  mutate(

```

```

evals_per_gen = as.factor(evals_per_gen),
eval_mode_row = as.factor(eval_mode_row),
DIAGNOSTIC = as.factor(DIAGNOSTIC),
SELECTION = as.factor(SELECTION),
EVAL_MODE = as.factor(EVAL_MODE),
NUM_COHORTS = as.factor(NUM_COHORTS),
TEST_DOWNSAMPLE_RATE = as.factor(TEST_DOWNSAMPLE_RATE),
EVAL_FIT_EST_MODE = factor(
  EVAL_FIT_EST_MODE,
  levels = c(
    "none",
    "ancestor",
    "relative"
  ),
  labels = c(
    "None",
    "Ancestor",
    "Relative"
  )
)
) %>%
mutate(
  prop_self_lookups = case_when(
    (EVAL_MODE != "full" & EVAL_FIT_EST_MODE != "None") ~
      self_count / (other_count + ancestor_count + descendant_count + self_count + outside_count)
    .default = 0
  ),
  prop_other_lookups = case_when(
    (EVAL_MODE != "full" & EVAL_FIT_EST_MODE != "None") ~
      other_count / (other_count + ancestor_count + descendant_count + self_count + outside_count)
    .default = 0
  ),
  prop_ancestor_lookups = case_when(
    (EVAL_MODE != "full" & EVAL_FIT_EST_MODE != "None") ~
      ancestor_count / (other_count + ancestor_count + descendant_count + self_count + outside_count)
    .default = 0
  ),
  prop_descendant_lookups = case_when(
    (EVAL_MODE != "full" & EVAL_FIT_EST_MODE != "None") ~
      descendant_count / (other_count + ancestor_count + descendant_count + self_count + outside_count)
    .default = 0
  ),
  prop_outside_lookups = case_when(
    (EVAL_MODE != "full" & EVAL_FIT_EST_MODE != "None") ~
      outside_count / (other_count + ancestor_count + descendant_count + self_count + outside_count)

```

```

      .default = 0
    )
  )

```

## 4.3 Contradictory objectives diagnostic

### 4.3.1 Population-wide satisfactory trait coverage (final)

Satisfactory trait coverage after 50,000 generations:

```

contradictory_obj_final_plt <- ggplot(
  con_obj_summary_data,
  aes(
    x = EVAL_FIT_EST_MODE,
    y = pop_optimal_trait_coverage,
    fill = EVAL_FIT_EST_MODE
  )
) +
geom_flat_violin(
  position = position_nudge(x = .2, y = 0),
  alpha = .8,
  adjust=1.5
) +
geom_point(
  mapping=aes(color=EVAL_FIT_EST_MODE),
  position = position_jitter(width = .15),
  size = .5,
  alpha = 0.8
) +
geom_boxplot(
  width = .1,
  outlier.shape = NA,
  alpha = 0.5
) +
scale_y_continuous(
  # limits = c(-0.5, 100)
) +
scale_fill_bright() +
scale_color_bright() +
facet_grid(
  eval_mode_row~evals_per_gen,
  # nrow=2,
  labeller=label_both
)

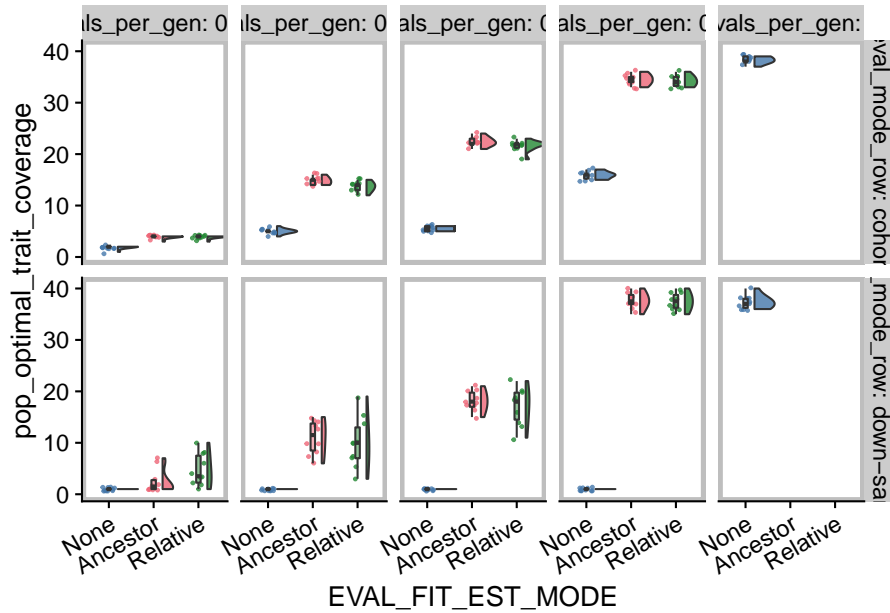
```

```

) +
theme(
  legend.position = "none",
  axis.text.x = element_text(
    angle = 30,
    hjust = 1
  ),
  panel.border = element_rect(color="gray", size=2)
)
ggsave(
  filename = paste0(plot_directory, "contra-obj-final.pdf"),
  plot = contradictory_obj_final_plt + labs(title="Contradictory objectives"),
  width = 15,
  height = 10
)

```

```
contradictory_obj_final_plt
```



#### 4.3.1.1 Statistical analysis

First, we create a table of summary values for satisfactory trait coverage in the final generation.

```

con_obj_summary_data %>%
  filter(EVAL_MODE != "full") %>%
  group_by(EVAL_MODE, evals_per_gen, EVAL_FIT_EST_MODE) %>%
  summarize(
    cov_median = median(pop_optimal_trait_coverage),
    cov_mean = mean(pop_optimal_trait_coverage),
    n = n()
  ) %>%
  kable()

```

## 'summarise()' has grouped output by 'EVAL\_MODE', 'evals\_per\_gen'. You can  
## override using the '.groups' argument.

EVAL_MODE	evals_per_gen	EVAL_FIT_EST_MODE	cov_median	cov_mean	n
cohort	0.01	None	2.0	1.9	10
cohort	0.01	Ancestor	4.0	3.9	10
cohort	0.01	Relative	4.0	3.9	10
cohort	0.05	None	5.0	5.0	10
cohort	0.05	Ancestor	15.0	14.8	10
cohort	0.05	Relative	14.0	13.7	10
cohort	0.1	None	5.5	5.5	10
cohort	0.1	Ancestor	22.0	22.4	10
cohort	0.1	Relative	22.0	21.6	10
cohort	0.5	None	16.0	15.9	10
cohort	0.5	Ancestor	34.5	34.5	10
cohort	0.5	Relative	34.0	34.2	10
down-sample	0.01	None	1.0	1.0	10
down-sample	0.01	Ancestor	1.5	2.5	10
down-sample	0.01	Relative	3.5	4.7	10
down-sample	0.05	None	1.0	1.0	10
down-sample	0.05	Ancestor	11.5	11.0	10
down-sample	0.05	Relative	10.0	10.0	10
down-sample	0.1	None	1.0	1.0	10
down-sample	0.1	Ancestor	18.0	18.1	10
down-sample	0.1	Relative	18.0	17.1	10
down-sample	0.5	None	1.0	1.0	10
down-sample	0.5	Ancestor	37.5	37.6	10
down-sample	0.5	Relative	37.5	37.5	10

Next, we perform a Kruskal-Wallis test to determine which comparisons contain statistically significant differences among treatments.

```

con_obj_kw_test <- con_obj_summary_data %>%
  filter(EVAL_MODE != "full") %>%

```

```

group_by(EVAL_MODE, evals_per_gen) %>%
kruskal_test(pop_optimal_trait_coverage ~ EVAL_FIT_EST_MODE) %>%
unite(
  "comparison_group",
  EVAL_MODE,
  evals_per_gen,
  sep = "_",
  remove = FALSE
)

kable(con_obj_kw_test)

```

comparison_group	EVAL_MODE	evals_per_gen	.y.	n	statistic	df	
cohort_0.01	cohort	0.01	pop_optimal_trait_coverage	30	25.55066	2	2.80
cohort_0.05	cohort	0.05	pop_optimal_trait_coverage	30	22.72918	2	1.16
cohort_0.1	cohort	0.1	pop_optimal_trait_coverage	30	21.76615	2	1.88
cohort_0.5	cohort	0.5	pop_optimal_trait_coverage	30	20.05082	2	4.43
down-sample_0.01	down-sample	0.01	pop_optimal_trait_coverage	30	15.17863	2	5.06
down-sample_0.05	down-sample	0.05	pop_optimal_trait_coverage	30	20.38430	2	3.75
down-sample_0.1	down-sample	0.1	pop_optimal_trait_coverage	30	20.29663	2	3.91
down-sample_0.5	down-sample	0.5	pop_optimal_trait_coverage	30	20.31895	2	3.87

Finally, we perform a pairwise Wilcoxon rank-sum test (using a Holm-Bonferroni correction for multiple comparisons). Note that only results from signific

```

sig_kw_groups <- filter(con_obj_kw_test, p < 0.05)$comparison_group

con_obj_stats <- con_obj_summary_data %>%
  unite(
    "comparison_group",
    EVAL_MODE,
    evals_per_gen,
    sep = "_",
    remove = FALSE
  ) %>%
  filter(EVAL_MODE != "full" & comparison_group %in% sig_kw_groups) %>%
  group_by(EVAL_MODE, evals_per_gen) %>%
  pairwise_wilcox_test(pop_optimal_trait_coverage ~ EVAL_FIT_EST_MODE) %>%
  adjust_pvalue(method = "holm") %>%
  add_significance("p.adj")

kable(con_obj_stats)

```

EVAL_MODE	evals_per_gen	.y.	group1	group2	n1	n2	s
cohort	0.01	pop_optimal_trait_coverage	None	Ancestor	10	10	
cohort	0.01	pop_optimal_trait_coverage	None	Relative	10	10	
cohort	0.01	pop_optimal_trait_coverage	Ancestor	Relative	10	10	
cohort	0.05	pop_optimal_trait_coverage	None	Ancestor	10	10	
cohort	0.05	pop_optimal_trait_coverage	None	Relative	10	10	
cohort	0.05	pop_optimal_trait_coverage	Ancestor	Relative	10	10	
cohort	0.1	pop_optimal_trait_coverage	None	Ancestor	10	10	
cohort	0.1	pop_optimal_trait_coverage	None	Relative	10	10	
cohort	0.1	pop_optimal_trait_coverage	Ancestor	Relative	10	10	
cohort	0.5	pop_optimal_trait_coverage	None	Ancestor	10	10	
cohort	0.5	pop_optimal_trait_coverage	None	Relative	10	10	
cohort	0.5	pop_optimal_trait_coverage	Ancestor	Relative	10	10	
down-sample	0.01	pop_optimal_trait_coverage	None	Ancestor	10	10	
down-sample	0.01	pop_optimal_trait_coverage	None	Relative	10	10	
down-sample	0.01	pop_optimal_trait_coverage	Ancestor	Relative	10	10	
down-sample	0.05	pop_optimal_trait_coverage	None	Ancestor	10	10	
down-sample	0.05	pop_optimal_trait_coverage	None	Relative	10	10	
down-sample	0.05	pop_optimal_trait_coverage	Ancestor	Relative	10	10	
down-sample	0.1	pop_optimal_trait_coverage	None	Ancestor	10	10	
down-sample	0.1	pop_optimal_trait_coverage	None	Relative	10	10	
down-sample	0.1	pop_optimal_trait_coverage	Ancestor	Relative	10	10	
down-sample	0.5	pop_optimal_trait_coverage	None	Ancestor	10	10	
down-sample	0.5	pop_optimal_trait_coverage	None	Relative	10	10	
down-sample	0.5	pop_optimal_trait_coverage	Ancestor	Relative	10	10	

```
# con_obj_stats %>%
#   filter(p.adj <= 0.05) %>%
#   arrange(
#     desc(p.adj)
#   ) %>%
#   kable()
```

### 4.3.2 Population-wide satisfactory trait coverage (over time)

```
contradictory_obj_pop_cov_ts <- ggplot(
  con_obj_ts_data,
  aes(
    x = ts_step,
    y = pop_optimal_trait_coverage,
    fill = EVAL_FIT_EST_MODE,
    color = EVAL_FIT_EST_MODE
  )
)
```



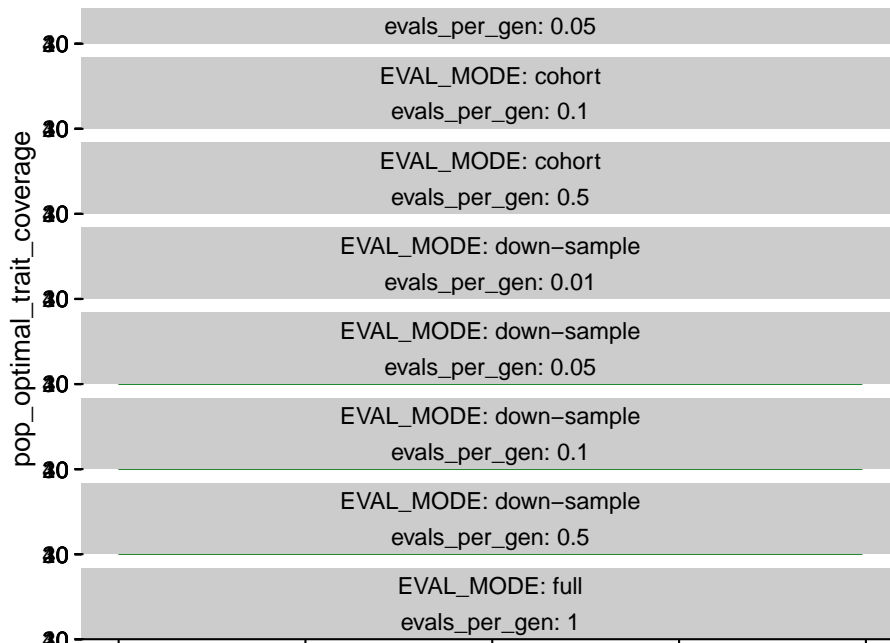
```

    )
  ) +
  stat_summary(
    geom = "line",
    fun = mean
  ) +
  stat_summary(
    geom = "ribbon",
    fun.data = "mean_cl_boot",
    fun.args = list(conf.int = 0.95),
    alpha = 0.2,
    linetype = 0
  ) +
  scale_fill_bright() +
  scale_color_bright() +
  facet_wrap(
    EVAL_MODE ~ evals_per_gen,
    ncol = 1,
    labeller = label_both
  ) +
  theme(
    legend.position = "bottom"
  )
)

ggsave(
  filename = paste0(plot_directory, "contra-obj-ts.pdf"),
  plot = contradictory_obj_pop_cov_ts + labs(title="Contradictory objectives"),
  width = 10,
  height = 15
)

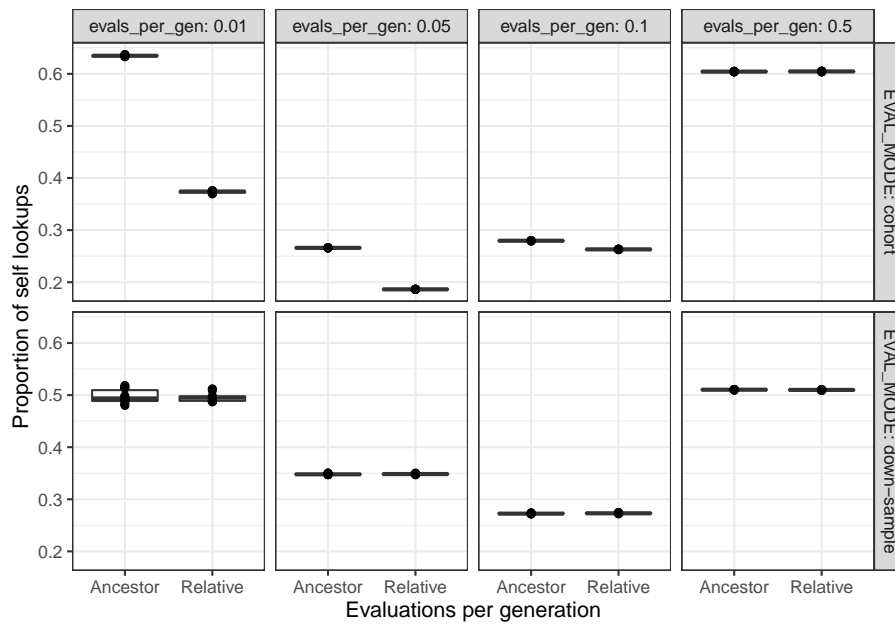
```

```
contradictory_obj_pop_cov_ts
```



### 4.3.3 Phylogeny estimate source distributions

```
est_source_data %>%
  filter(DIAGNOSTIC == "contradictory-objectives") %>%
  filter(EVAL_MODE != "full" & EVAL_FIT_EST_MODE != "None") %>%
  ggplot(
    aes(
      x = EVAL_FIT_EST_MODE,
      y = prop_self_lookups
    )
  ) +
  geom_boxplot() +
  geom_point() +
  facet_grid(
    cols = vars(eval_per_gen),
    rows = vars(EVAL_MODE),
    labeller = label_both
  ) +
  scale_y_continuous("Proportion of self lookups") +
  scale_x_discrete("Evaluations per generation") +
  theme_bw() +
  theme(legend.position = "none")
```

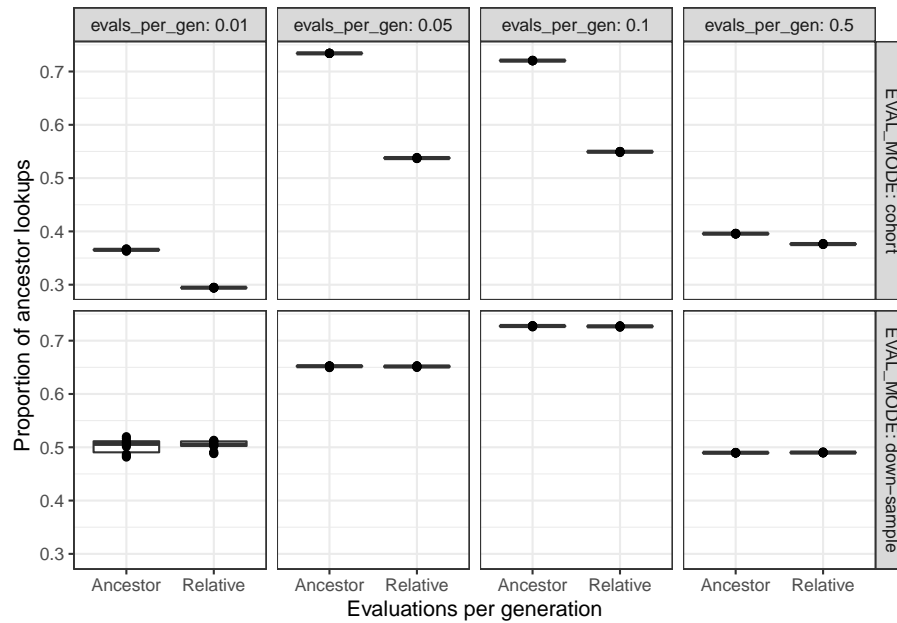


```
ggsave(
  filename=paste0(plot_directory, "contra-obj-self-lookups.pdf")
)
```

```
## Saving 6.5 x 4.5 in image
```

```
est_source_data %>%
  filter(DIAGNOSTIC == "contradictory-objectives") %>%
  filter(EVAL_MODE != "full" & EVAL_FIT_EST_MODE != "None") %>%
  ggplot(
    aes(
      x = EVAL_FIT_EST_MODE,
      y = prop_ancestor_lookups
    )
  ) +
  geom_boxplot() +
  geom_point() +
  facet_grid(
    cols = vars(eval_per_gen),
    rows = vars(EVAL_MODE),
    labeller = label_both
  ) +
  scale_y_continuous("Proportion of ancestor lookups") +
  scale_x_discrete("Evaluations per generation") +
```

```
theme_bw() +
theme(legend.position = "none")
```



```
ggsave(
  filename=paste0(plot_directory, "contra-obj-ancestor-lookups.pdf")
)
```

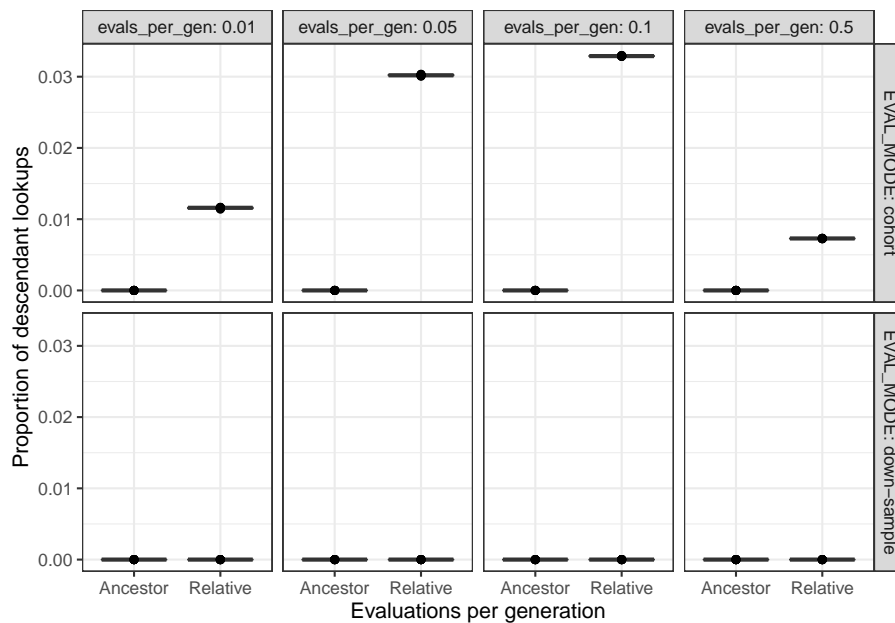
```
## Saving 6.5 x 4.5 in image
```

```
est_source_data %>%
  filter(DIAGNOSTIC == "contradictory-objectives") %>%
  filter(EVAL_MODE != "full" & EVAL_FIT_EST_MODE != "None") %>%
  ggplot(
    aes(
      x = EVAL_FIT_EST_MODE,
      y = prop_descendant_lookups
    )
  ) +
  geom_boxplot() +
  geom_point() +
  facet_grid(
    cols = vars(evals_per_gen),
    rows = vars(EVAL_MODE),
```

```

    labeller = label_both
  ) +
  scale_y_continuous("Proportion of descendant lookups") +
  scale_x_discrete("Evaluations per generation") +
  theme_bw() +
  theme(legend.position = "none")

```



```

ggsave(
  filename=paste0(plot_directory, "contra-obj-descendant-lookups.pdf")
)

```

## Saving 6.5 x 4.5 in image

```

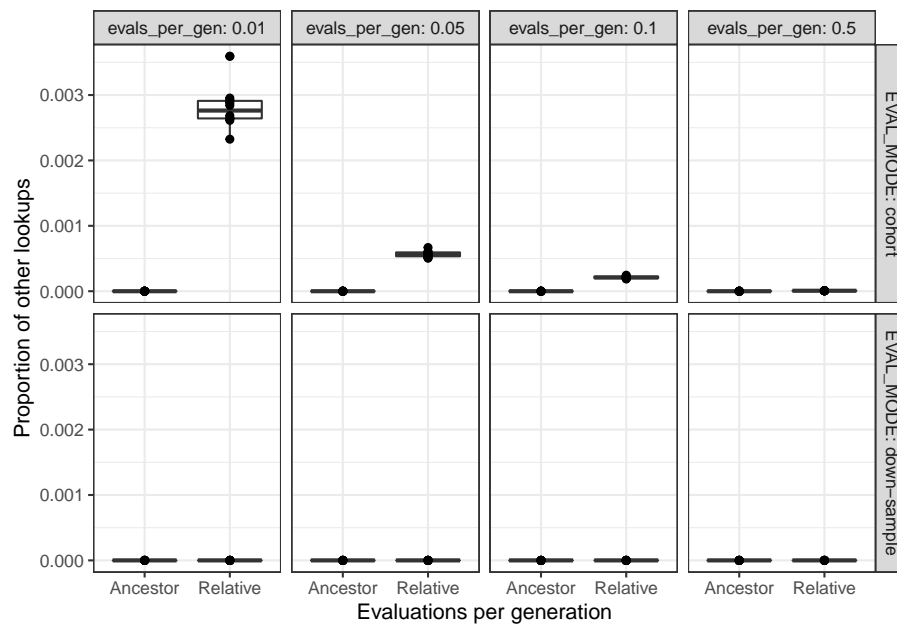
est_source_data %>%
  filter(DIAGNOSTIC == "contradictory-objectives") %>%
  filter(EVAL_MODE != "full" & EVAL_FIT_EST_MODE != "None") %>%
  ggplot(
    aes(
      x = EVAL_FIT_EST_MODE,
      y = prop_other_lookups
    )
  ) +
  geom_boxplot() +

```

```

geom_point() +
facet_grid(
  cols = vars(evals_per_gen),
  rows = vars(EVAL_MODE),
  labeller = label_both
) +
scale_y_continuous("Proportion of other lookups") +
scale_x_discrete("Evaluations per generation") +
theme_bw() +
theme(legend.position = "none")

```



```

ggsave(
  filename=paste0(plot_directory, "contra-obj-other-lookups.pdf")
)

```

## Saving 6.5 x 4.5 in image

```

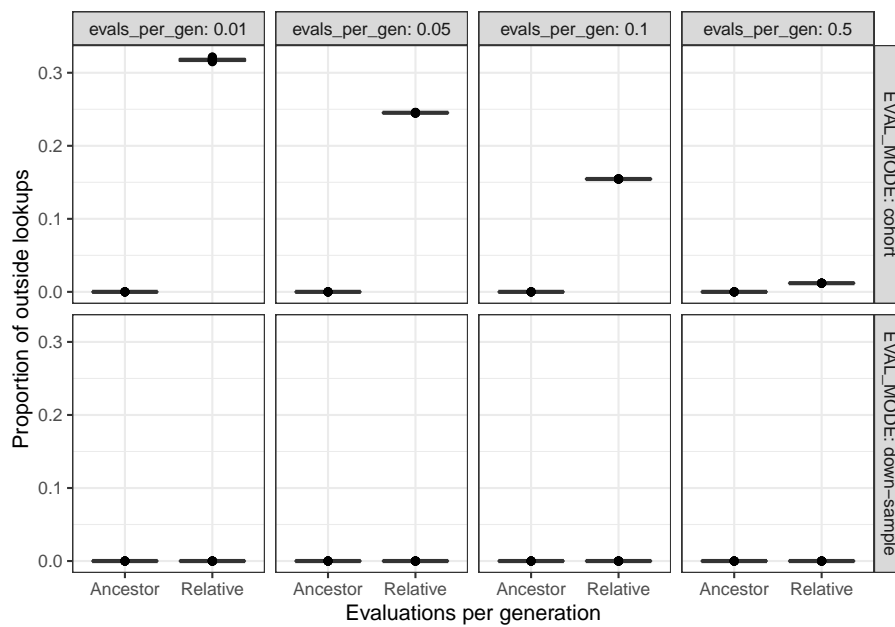
est_source_data %>%
  filter(DIAGNOSTIC == "contradictory-objectives") %>%
  filter(EVAL_MODE != "full" & EVAL_FIT_EST_MODE != "None") %>%
  ggplot(
    aes(
      x = EVAL_FIT_EST_MODE,

```

```

    y = prop_outside_lookups
  )
) +
geom_boxplot() +
geom_point() +
facet_grid(
  cols = vars(evals_per_gen),
  rows = vars(EVAL_MODE),
  labeller = label_both
) +
scale_y_continuous("Proportion of outside lookups") +
scale_x_discrete("Evaluations per generation") +
theme_bw() +
theme(legend.position = "none")

```



```

ggsave(
  filename=paste0(plot_directory, "contra-obj-outside-lookups.pdf")
)

```

```
## Saving 6.5 x 4.5 in image
```

## 4.4 Multi-path exploration diagnostic

### 4.4.1 Maximum aggregate score (final)

```

explore_final_score_plt <- ggplot(
  explore_summary_data,
  aes(
    x = EVAL_FIT_EST_MODE,
    y = max_agg_score,
    fill = EVAL_FIT_EST_MODE
  )
) +
geom_flat_violin(
  position = position_nudge(x = .2, y = 0),
  alpha = .8,
  adjust=1.5
) +
geom_point(
  mapping=aes(color=EVAL_FIT_EST_MODE),
  position = position_jitter(width = .15),
  size = .5,
  alpha = 0.8
) +
geom_boxplot(
  width = .1,
  outlier.shape = NA,
  alpha = 0.5
) +
scale_y_continuous(
  # limits = c(-0.5, 100)
) +
scale_fill_bright() +
scale_color_bright() +
facet_grid(
  eval_mode_row~evals_per_gen,
  # nrow=2,
  labeller=label_both
) +
theme(
  legend.position = "none",
  axis.text.x = element_text(
    angle = 30,
    hjust = 1
  ),

```



```

    panel.border = element_rect(color="gray", size=2)
  )
  ggsave(
    filename = paste0(plot_directory, "explore-final.pdf"),
    plot = explore_final_score_plt + labs(title="Multi-path exploration"),
    width = 15,
    height = 10
  )

```

#### 4.4.1.1 Statistical analysis

```

explore_summary_data %>%
  filter(EVAL_MODE != "full") %>%
  group_by(EVAL_MODE, evals_per_gen, EVAL_FIT_EST_MODE) %>%
  summarize(
    score_median = median(max_agg_score),
    score_mean = mean(max_agg_score),
    n = n()
  ) %>%
  kable()

```

## 'summarise()' has grouped output by 'EVAL\_MODE', 'evals\_per\_gen'. You can  
## override using the '.groups' argument.

EVAL_MODE	evals_per_gen	EVAL_FIT_EST_MODE	score_median	score_mean	n
cohort	0.01	None	1971.7450	1900.3130	10
cohort	0.01	Ancestor	2316.5800	1971.5663	10
cohort	0.01	Relative	2182.5700	2006.6843	10
cohort	0.05	None	2401.0150	2373.2040	10
cohort	0.05	Ancestor	2858.6950	2747.0960	10
cohort	0.05	Relative	3471.8500	3389.0910	10
cohort	0.1	None	3075.5150	3076.6120	10
cohort	0.1	Ancestor	4508.1150	4383.9440	10
cohort	0.1	Relative	5144.5350	5163.0130	10
cohort	0.5	None	8187.5000	8198.1150	10
cohort	0.5	Ancestor	8591.7150	8708.0110	10
cohort	0.5	Relative	8684.2050	8652.1500	10
down-sample	0.01	None	580.4215	532.1152	10
down-sample	0.01	Ancestor	434.8545	430.0114	10
down-sample	0.01	Relative	449.3640	465.0957	10
down-sample	0.05	None	396.0890	445.1163	10
down-sample	0.05	Ancestor	2007.3700	1982.4690	10
down-sample	0.05	Relative	1777.9000	1762.3250	10
down-sample	0.1	None	692.7270	690.7322	10
down-sample	0.1	Ancestor	2423.2200	2451.7950	10
down-sample	0.1	Relative	2529.6100	2542.1340	10
down-sample	0.5	None	1499.9800	1658.0837	10
down-sample	0.5	Ancestor	6976.5950	6972.2630	10
down-sample	0.5	Relative	7309.9450	7120.4160	10

```

explore_kw_test <- explore_summary_data %>%
  filter(EVAL_MODE != "full") %>%
  group_by(EVAL_MODE, evals_per_gen) %>%
  kruskal_test(max_agg_score ~ EVAL_FIT_EST_MODE) %>%
  mutate(
    sig = (p <= 0.05)
  ) %>%
  unite(
    "comparison_group",
    EVAL_MODE,
    evals_per_gen,
    sep = "_",
    remove = FALSE
  )

kable(explore_kw_test)

```

comparison_group	EVAL_MODE	evals_per_gen	.y.	n	statistic	df	p	met
cohort_0.01	cohort	0.01	max_agg_score	30	0.7045161	2	7.03e-01	Kru
cohort_0.05	cohort	0.05	max_agg_score	30	15.5380645	2	4.23e-04	Kru
cohort_0.1	cohort	0.1	max_agg_score	30	25.5509677	2	2.80e-06	Kru
cohort_0.5	cohort	0.5	max_agg_score	30	5.0348387	2	8.07e-02	Kru
down-sample_0.01	down-sample	0.01	max_agg_score	30	2.6090323	2	2.71e-01	Kru
down-sample_0.05	down-sample	0.05	max_agg_score	30	22.3380645	2	1.41e-05	Kru
down-sample_0.1	down-sample	0.1	max_agg_score	30	19.3780645	2	6.20e-05	Kru
down-sample_0.5	down-sample	0.5	max_agg_score	30	19.4047630	2	6.11e-05	Kru

```
expl_sig_kw_groups <- filter(explore_kw_test, p < 0.05)$comparison_group

explore_stats <- explore_summary_data %>%
  unite(
    "comparison_group",
    EVAL_MODE,
    evals_per_gen,
    sep = "_",
    remove = FALSE
  ) %>%
  filter(EVAL_MODE != "full" & comparison_group %in% expl_sig_kw_groups) %>%
  group_by(EVAL_MODE, evals_per_gen) %>%
  pairwise_wilcox_test(max_agg_score ~ EVAL_FIT_EST_MODE) %>%
  adjust_pvalue(method = "holm") %>%
  add_significance("p.adj")

kable(explore_stats)
```

EVAL_MODE	evals_per_gen	.y.	group1	group2	n1	n2	statistic	p	
cohort	0.05	max_agg_score	None	Ancestor	10	10	27	8.90e-02	0.267
cohort	0.05	max_agg_score	None	Relative	10	10	4	1.30e-04	0.003
cohort	0.05	max_agg_score	Ancestor	Relative	10	10	12	3.00e-03	0.015
cohort	0.1	max_agg_score	None	Ancestor	10	10	0	1.08e-05	0.000
cohort	0.1	max_agg_score	None	Relative	10	10	0	1.08e-05	0.000
cohort	0.1	max_agg_score	Ancestor	Relative	10	10	1	2.17e-05	0.000
down-sample	0.05	max_agg_score	None	Ancestor	10	10	0	1.08e-05	0.000
down-sample	0.05	max_agg_score	None	Relative	10	10	0	1.08e-05	0.000
down-sample	0.05	max_agg_score	Ancestor	Relative	10	10	84	9.00e-03	0.030
down-sample	0.1	max_agg_score	None	Ancestor	10	10	0	1.08e-05	0.000
down-sample	0.1	max_agg_score	None	Relative	10	10	0	1.08e-05	0.000
down-sample	0.1	max_agg_score	Ancestor	Relative	10	10	47	8.53e-01	1.000
down-sample	0.5	max_agg_score	None	Ancestor	10	10	0	1.81e-04	0.003
down-sample	0.5	max_agg_score	None	Relative	10	10	0	1.81e-04	0.003
down-sample	0.5	max_agg_score	Ancestor	Relative	10	10	46	7.96e-01	1.000

```
# explore_stats %>%
#   filter(p.adj <= 0.05) %>%
#   arrange(
#     desc(p.adj)
#   ) %>%
#   kable()
```

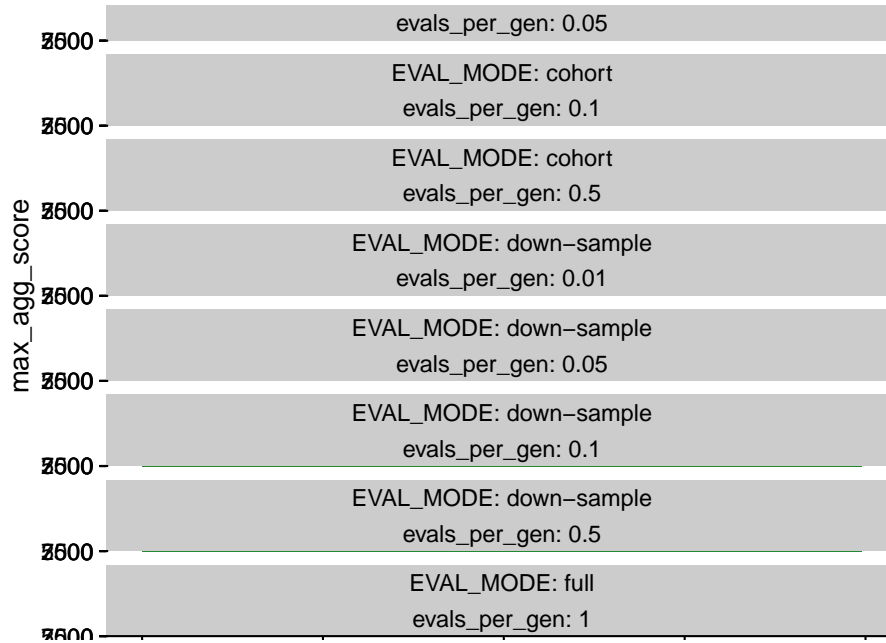
#### 4.4.2 Maximum aggregate score (over time)

```
explore_score_ts <- ggplot(
  explore_ts_data,
  aes(
    x = ts_step,
    y = max_agg_score,
    fill = EVAL_FIT_EST_MODE,
    color = EVAL_FIT_EST_MODE
  )
) +
stat_summary(
  geom = "line",
  fun = mean
) +
stat_summary(
  geom = "ribbon",
  fun.data = "mean_cl_boot",
  fun.args = list(conf.int = 0.95),
  alpha = 0.2,
  linetype = 0
) +
scale_fill_bright() +
scale_color_bright() +
facet_wrap(
  EVAL_MODE ~ evals_per_gen,
  ncol = 1,
  labeller = label_both
) +
theme(
  legend.position = "bottom"
)

ggsave(
  filename = paste0(plot_directory, "explore-ts.pdf"),
  plot = explore_score_ts + labs(title="Multi-path exploration"),
```

```
width = 10,
height = 15
)
```

```
explore_score_ts
```



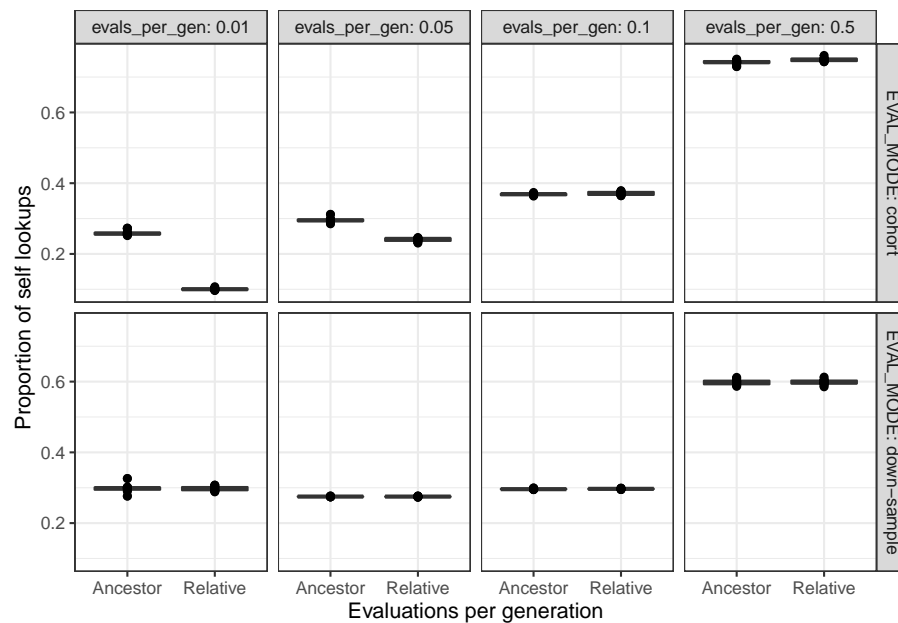
#### 4.4.3 Phylogeny estimate source distributions

```
est_source_data %>%
  filter(DIAGNOSTIC == "multipath-exploration") %>%
  filter(EVAL_MODE != "full" & EVAL_FIT_EST_MODE != "None") %>%
  ggplot(
    aes(
      x = EVAL_FIT_EST_MODE,
      y = prop_self_lookups
    )
  ) +
  geom_boxplot() +
  geom_point() +
  facet_grid(
    cols = vars(eval_per_gen),
    rows = vars(EVAL_MODE),
```

```

labeller = label_both
) +
scale_y_continuous("Proportion of self lookups") +
scale_x_discrete("Evaluations per generation") +
theme_bw() +
theme(legend.position = "none")

```



```

ggsave(
  filename=paste0(plot_directory, "explore-self-lookups.pdf")
)

```

## Saving 6.5 x 4.5 in image

```

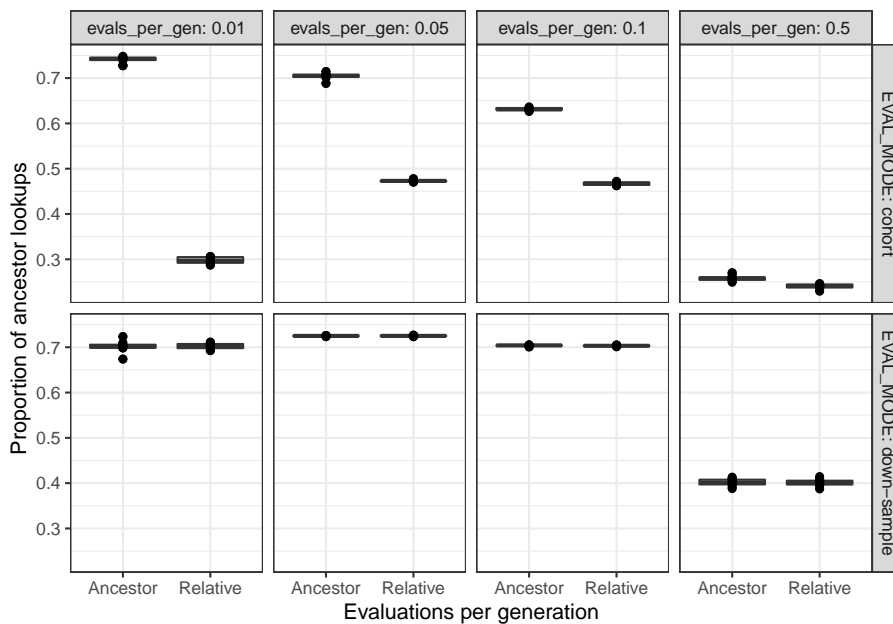
est_source_data %>%
  filter(DIAGNOSTIC == "multipath-exploration") %>%
  filter(EVAL_MODE != "full" & EVAL_FIT_EST_MODE != "None") %>%
  ggplot(
    aes(
      x = EVAL_FIT_EST_MODE,
      y = prop_ancestor_lookups
    )
  ) +
  geom_boxplot() +

```

```

geom_point() +
facet_grid(
  cols = vars(evals_per_gen),
  rows = vars(EVAL_MODE),
  labeller = label_both
) +
scale_y_continuous("Proportion of ancestor lookups") +
scale_x_discrete("Evaluations per generation") +
theme_bw() +
theme(legend.position = "none")

```



```

ggsave(
  filename=paste0(plot_directory, "explore-ancestor-lookups.pdf")
)

```

## Saving 6.5 x 4.5 in image

```

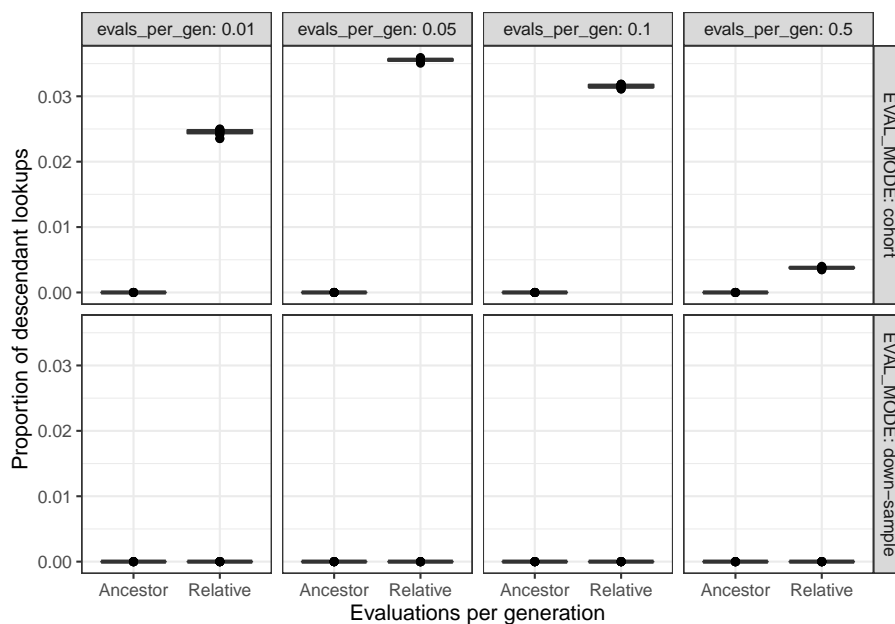
est_source_data %>%
  filter(DIAGNOSTIC == "multipath-exploration") %>%
  filter(EVAL_MODE != "full" & EVAL_FIT_EST_MODE != "None") %>%
  ggplot(
    aes(
      x = EVAL_FIT_EST_MODE,

```

```

    y = prop_descendant_lookups
  )
) +
geom_boxplot() +
geom_point() +
facet_grid(
  cols = vars(evals_per_gen),
  rows = vars(EVAL_MODE),
  labeller = label_both
) +
scale_y_continuous("Proportion of descendant lookups") +
scale_x_discrete("Evaluations per generation") +
theme_bw() +
theme(legend.position = "none")

```



```

ggsave(
  filename=paste0(plot_directory, "explore-descendant-lookups.pdf")
)

```

```
## Saving 6.5 x 4.5 in image
```

```

est_source_data %>%
  filter(DIAGNOSTIC == "multipath-exploration") %>%

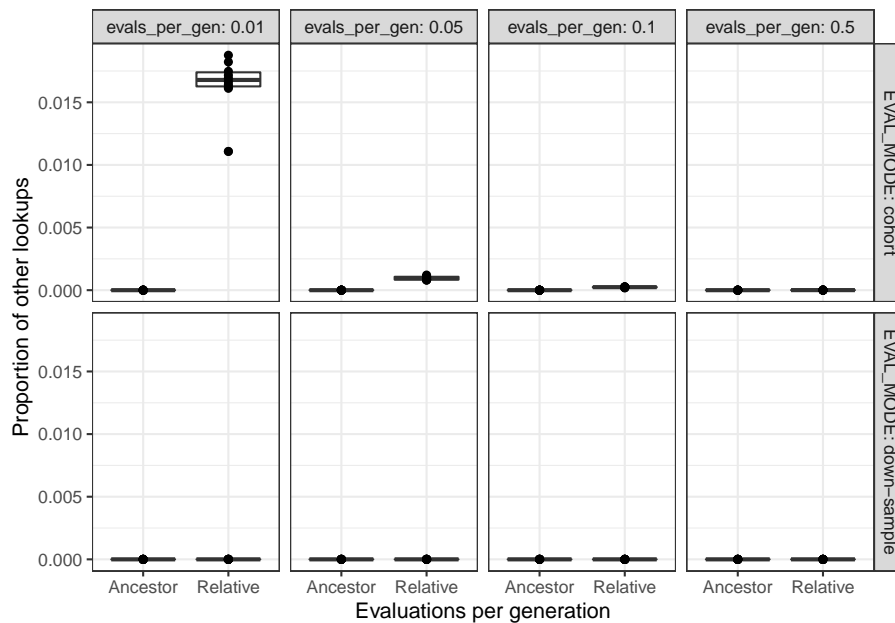
```



```

filter(EVAL_MODE != "full" & EVAL_FIT_EST_MODE != "None") %>%
ggplot(
  aes(
    x = EVAL_FIT_EST_MODE,
    y = prop_other_lookups
  )
) +
geom_boxplot() +
geom_point() +
facet_grid(
  cols = vars(evals_per_gen),
  rows = vars(EVAL_MODE),
  labeller = label_both
) +
scale_y_continuous("Proportion of other lookups") +
scale_x_discrete("Evaluations per generation") +
theme_bw() +
theme(legend.position = "none")

```



```

ggsave(
  filename=paste0(plot_directory, "explore-other-lookups.pdf")
)

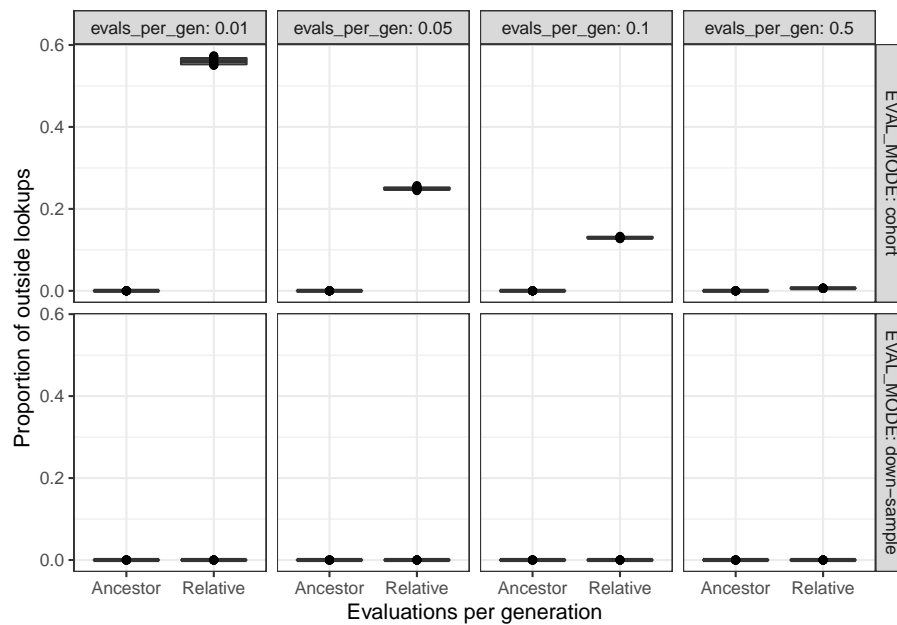
```

```
## Saving 6.5 x 4.5 in image
```

```

est_source_data %>%
  filter(DIAGNOSTIC == "multipath-exploration") %>%
  filter(EVAL_MODE != "full" & EVAL_FIT_EST_MODE != "None") %>%
  ggplot(
    aes(
      x = EVAL_FIT_EST_MODE,
      y = prop_outside_lookups
    )
  ) +
  geom_boxplot() +
  geom_point() +
  facet_grid(
    cols = vars(evals_per_gen),
    rows = vars(EVAL_MODE),
    labeller = label_both
  ) +
  scale_y_continuous("Proportion of outside lookups") +
  scale_x_discrete("Evaluations per generation") +
  theme_bw() +
  theme(legend.position = "none")

```



```

ggsave(
  filename=paste0(plot_directory, "explore-outside-lookups.pdf")
)

```

```
## Saving 6.5 x 4.5 in image
```

## 4.5 Manuscript figures

```
full_median_size = 1.5

subsample_labeller <- function(subsample_level) {
  return(paste("Subsample level:", subsample_level))
}
```

### 4.5.1 Contradictory objectives

Build plot panels (1 cohort, 1 down-sample)

```
build_con_obj_plot <- function(eval_mode) {

  full_median <- median(
    filter(
      con_obj_summary_data,
      eval_mode_row == eval_mode & EVAL_MODE == "full"
    )$pop_optimal_trait_coverage
  )

  p <- con_obj_summary_data %>%
    filter(eval_mode_row == eval_mode & EVAL_MODE != "full") %>%
    ggplot(
      aes(
        x = EVAL_FIT_EST_MODE,
        y = pop_optimal_trait_coverage,
        fill = EVAL_FIT_EST_MODE
      )
    ) +
    geom_hline(
      yintercept = full_median,
      size = full_median_size,
      alpha = 0.7,
      color = "black"
    ) +
    geom_flat_violin(
      position = position_nudge(x = .2, y = 0),
      alpha = .8,
      adjust=1.5
    )
}
```

```

) +
geom_point(
  mapping=aes(color=EVAL_FIT_EST_MODE),
  position = position_jitter(width = .15),
  size = .5,
  alpha = 0.8
) +
geom_boxplot(
  width = .1,
  outlier.shape = NA,
  alpha = 0.5
) +
scale_y_continuous(
  limits = c(-0.5, 50)
) +
scale_fill_bright() +
scale_color_bright() +
facet_wrap(
  ~ evals_per_gen,
  nrow = 1,
  labeller = as_labeller(
    subsample_labeller
  )
) +
labs(
  x = "Estimation mode",
  y = "Satisfactory trait coverage"
) +
theme(
  legend.position = "none",
  axis.text.x = element_text(
    angle = 30,
    hjust = 1
  ),
  panel.border = element_rect(color="gray", size=2)
)

return(p)
}

con_obj_ds_plot <- build_con_obj_plot("down-sample")
con_obj_cohort_plot <- build_con_obj_plot("cohort")

```

Combine panels into single plot.

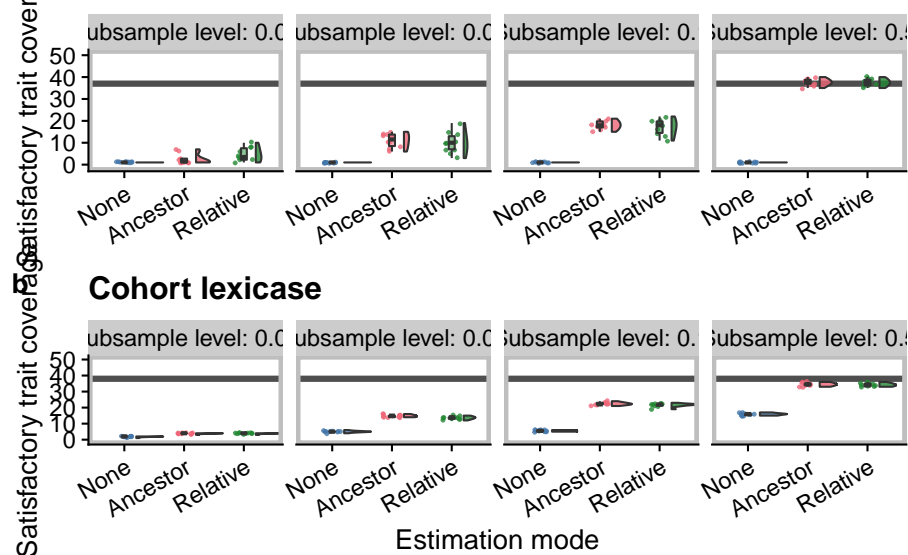
```

# Joint title: https://wilkelab.org/cowplot/articles/plot\_grid.html
con_obj_title <- ggdraw() +
  draw_label(
    "Contradictory objectives diagnostic",
    fontface = 'bold',
    x = 0,
    hjust = 0
  ) +
  theme(
    # add margin on the left of the drawing canvas,
    # so title is aligned with left edge of first plot
    plot.margin = margin(0, 0, 0, 7)
  )

con_obj_grid <- plot_grid(
  con_obj_title,
  con_obj_ds_plot +
    labs(
      title = "Down-sampled lexicase"
    ) +
    theme(axis.title.x = element_blank()),
  con_obj_cohort_plot +
    labs(
      title = "Cohort lexicase"
    ),
  nrow = 3,
  ncol = 1,
  # align = "h",
  labels = c("", "a", "b"),
  rel_heights = c(0.075, 1, 1)
)
con_obj_grid

```

### Contradictory objectives diagnostic Down-sampled lexicase



```
save_plot(
  filename = paste0(plot_directory, "2023-05-10-diagnostics-con-obj-final-fig.pdf"),
  plot = con_obj_grid,
  base_width = 10,
  base_height = 8,
  dpi = 600
)
```

### 4.5.2 Multi-path exploration

```
build_explore_plot <- function(eval_mode) {

  full_median <- median(
    filter(
      explore_summary_data,
      eval_mode_row == eval_mode & EVAL_MODE == "full"
    )$max_agg_score
  )

  p <- explore_summary_data %>%
    filter(eval_mode_row == eval_mode & EVAL_MODE != "full") %>%
    ggplot(
```

```

    aes(
      x = EVAL_FIT_EST_MODE,
      y = max_agg_score,
      fill = EVAL_FIT_EST_MODE
    )
  ) +
  geom_hline(
    yintercept = full_median,
    size = full_median_size,
    alpha = 0.7,
    color = "black"
  ) +
  geom_flat_violin(
    position = position_nudge(x = .2, y = 0),
    alpha = .8,
    adjust=1.5
  ) +
  geom_point(
    mapping=aes(color=EVAL_FIT_EST_MODE),
    position = position_jitter(width = .15),
    size = .5,
    alpha = 0.8
  ) +
  geom_boxplot(
    width = .1,
    outlier.shape = NA,
    alpha = 0.5
  ) +
  scale_y_continuous(
    limits = c(-0.5, 10005)
  ) +
  scale_fill_bright() +
  scale_color_bright() +
  facet_wrap(
    ~ evals_per_gen,
    nrow = 1,
    labeller = as_labeller(
      subsample_labeller
    )
  ) +
  labs(
    x = "Estimation mode",
    y = "Max aggregate score"
  ) +
  theme(

```

```

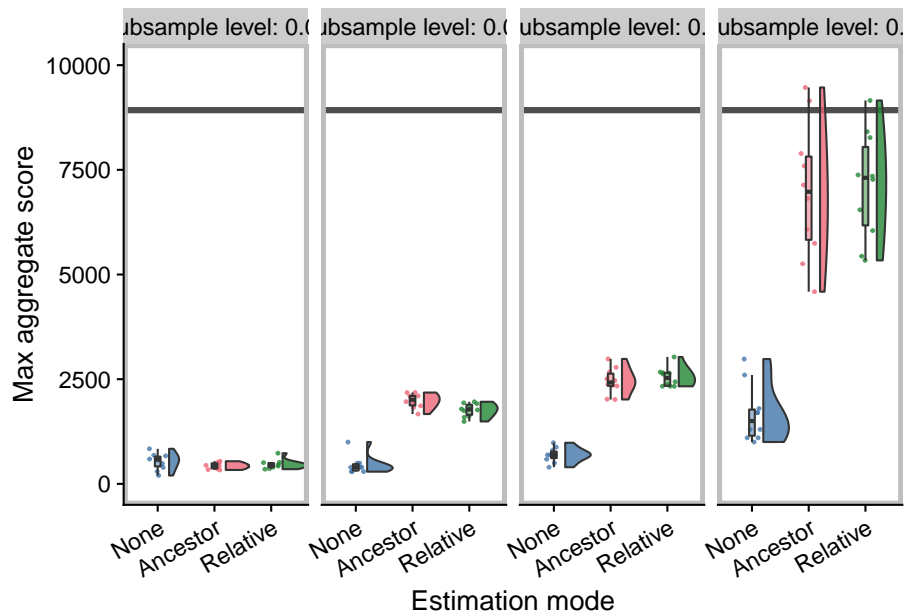
    legend.position = "none",
    axis.text.x = element_text(
      angle = 30,
      hjust = 1
    ),
    panel.border = element_rect(color="gray", size=2)
  )

  return(p)
}

explore_ds_plot <- build_explore_plot("down-sample")
explore_cohort_plot <- build_explore_plot("cohort")

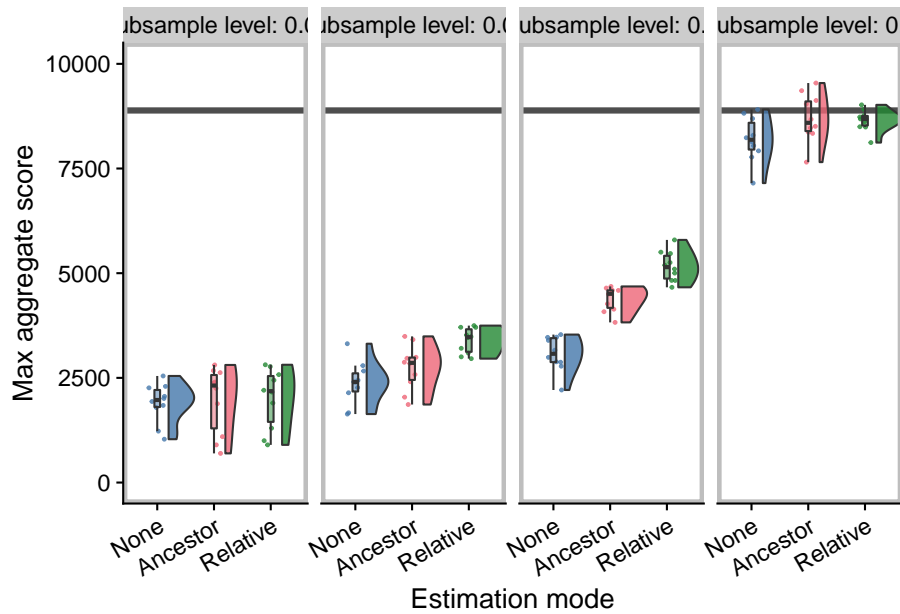
explore_ds_plot

```



```
explore_cohort_plot
```





Combine panels into single plot.

```
# Joint title: https://wilkelab.org/cowplot/articles/plot\_grid.html
explore_title <- ggdraw() +
  draw_label(
    "Multi-path exploration diagnostic",
    fontface = 'bold',
    x = 0,
    hjust = 0
  ) +
  theme(
    # add margin on the left of the drawing canvas,
    # so title is aligned with left edge of first plot
    plot.margin = margin(0, 0, 0, 7)
  )

explore_grid <- plot_grid(
  explore_title,
  explore_ds_plot +
    labs(
      title = "Down-sampled lexicase"
    ) +
    theme(axis.title.x = element_blank()),
  explore_cohort_plot +
    labs(
```

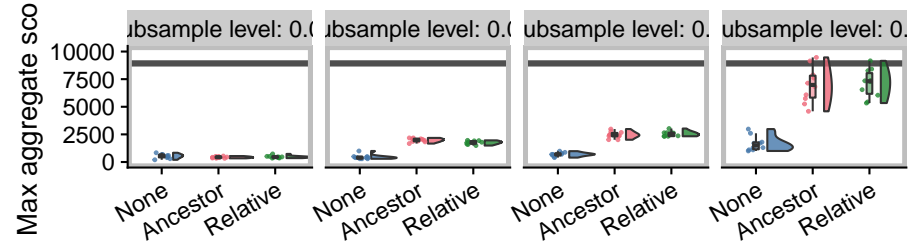
```

    title = "Cohort lexibase"
  ),
  nrow = 3,
  ncol = 1,
  # align = "h",
  labels = c("", "a", "b"),
  rel_heights = c(0.075, 1, 1)
)
explore_grid

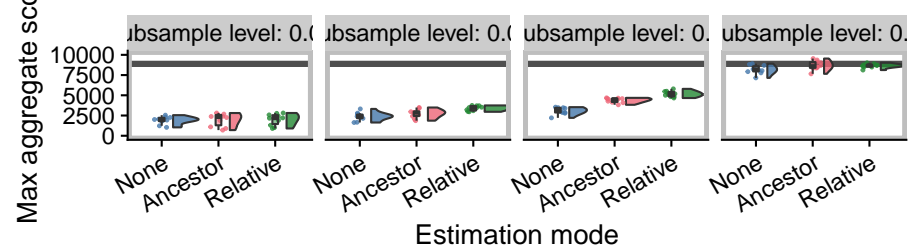
```

### Multi-path exploration diagnostic

#### a Down-sampled lexibase



#### b Cohort lexibase



```

save_plot(
  filename = paste0(plot_directory, "2023-05-10-diagnostics-explore-final-fig.pdf"),
  plot = explore_grid,
  base_width = 10,
  base_height = 8,
  dpi = 600
)

```

## Chapter 5

# Program synthesis experiments

```
experiment_slug <- "2023-05-08-psynth"

working_directory <- paste0(
  "experiments/",
  experiment_slug,
  "/analysis/"
)

if (exists("bookdown_wd_prefix")) {
  working_directory <- paste0(
    bookdown_wd_prefix,
    working_directory
  )
}
```

### 5.1 Dependencies

```
library(tidyverse)
library(ggplot2)
library(cowplot)
library(RColorBrewer)
library(khroma)
library(rstatix)
```

```

library(knitr)
source("https://gist.githubusercontent.com/benmarwick/2a1bb0133ff568cbe28d/raw/fb53bd9

print(version)

##
## platform      -
## arch          aarch64-apple-darwin20
## arch          aarch64
## os            darwin20
## system        aarch64, darwin20
## status
## major         4
## minor         2.1
## year          2022
## month         06
## day           23
## svn rev       82513
## language      R
## version.string R version 4.2.1 (2022-06-23)
## nickname      Funny-Looking Kid

```

## 5.2 Setup

```

# Configure our default graphing theme
theme_set(theme_cowplot())
# Create a directory to store plots
plot_directory <- paste0(working_directory, "plots/")
dir.create(plot_directory, showWarnings=FALSE)

```

### 5.2.1 Load summary data

```

summary_data_loc <- paste0(working_directory, "data/aggregate.csv")
summary_data <- read_csv(summary_data_loc)

## Rows: 3120 Columns: 73
## -- Column specification -----
## Delimiter: ","
## chr (11): ANCESTOR_FILE_PATH, EVAL_FIT_EST_MODE, EVAL_MODE, POP_INIT_MODE, P...
## dbl (62): EVAL_CPU_CYCLES_PER_TEST, EVAL_MAX_PHYLO_SEARCH_DEPTH, MAX_ACTIVE_...

```

```
##
## i Use 'spec()' to retrieve the full column specification for this data.
## i Specify the column types or set 'show_col_types = FALSE' to quiet this message.
```

```
summary_data <- summary_data %>%
  mutate(
    eval_mode_row = case_when(
      EVAL_MODE == "full" & TEST_DOWNSAMPLE_RATE == "1" ~ "down-sample",
      EVAL_MODE == "full" & NUM_COHORTS == "1" ~ "cohort",
      .default = EVAL_MODE
    ),
    evals_per_gen = case_when(
      EVAL_MODE == "cohort" ~ 1.0 / NUM_COHORTS,
      EVAL_MODE == "down-sample" ~ TEST_DOWNSAMPLE_RATE,
      EVAL_MODE == "full" ~ 1.0
    ),
    EVAL_FIT_EST_MODE = case_when(
      EVAL_FIT_EST_MODE == "ancestor-opt" ~ "ancestor",
      EVAL_FIT_EST_MODE == "relative-opt" ~ "relative",
      .default = EVAL_FIT_EST_MODE
    ),
    .keep = "all"
  ) %>%
  mutate(
    evals_per_gen = as.factor(evals_per_gen),
    PROBLEM = as.factor(PROBLEM),
    SELECTION = as.factor(SELECTION),
    EVAL_MODE = as.factor(EVAL_MODE),
    NUM_COHORTS = as.factor(NUM_COHORTS),
    TEST_DOWNSAMPLE_RATE = as.factor(TEST_DOWNSAMPLE_RATE),
    EVAL_FIT_EST_MODE = factor(
      EVAL_FIT_EST_MODE,
      levels = c(
        "none",
        "ancestor",
        "relative"
      ),
      labels = c(
        "None",
        "Ancestor",
        "Relative"
      )
    ),
    .keep = "all"
  )
```

```

solution_counts <- summary_data %>%
  group_by(
    PROBLEM,
    evals_per_gen,
    eval_mode_row,
    EVAL_FIT_EST_MODE,
    EVAL_MODE
  ) %>%
  summarize(
    solution_count = sum(found_solution == "1"),
    replicates = n(),
    no_solution_count = n() - sum(found_solution == "1")
  )

```

## 'summarise()' has grouped output by 'PROBLEM', 'evals\_per\_gen', 'eval\_mode\_row',  
## 'EVAL\_FIT\_EST\_MODE'. You can override using the '.groups' argument.

```
print(solution_counts, n=140)
```

```

## # A tibble: 104 x 8
## # Groups:   PROBLEM, evals_per_gen, eval_mode_row, EVAL_FIT_EST_MODE [104]
##   PROBLEM      evals_per_gen eval_~1 EVAL_~2 EVAL_~3 solut~4 repli~5 no_so~6
##   <fct>         <fct>         <chr>  <fct>  <fct>      <int>  <int>  <int>
## 1 fizz-buzz      0.01      cohort None   cohort      0      30     30
## 2 fizz-buzz      0.01      cohort Ancest~ cohort      2      30     28
## 3 fizz-buzz      0.01      cohort Relati~ cohort      3      30     27
## 4 fizz-buzz      0.01    down-s~ None   down-s~      0      30     30
## 5 fizz-buzz      0.01    down-s~ Ancest~ down-s~      0      30     30
## 6 fizz-buzz      0.01    down-s~ Relati~ down-s~      0      30     30
## 7 fizz-buzz      0.05      cohort None   cohort      5      30     25
## 8 fizz-buzz      0.05      cohort Ancest~ cohort      3      30     27
## 9 fizz-buzz      0.05      cohort Relati~ cohort      7      30     23
## 10 fizz-buzz     0.05    down-s~ None   down-s~     20      30     10
## 11 fizz-buzz     0.05    down-s~ Ancest~ down-s~      2      30     28
## 12 fizz-buzz     0.05    down-s~ Relati~ down-s~      2      30     28
## 13 fizz-buzz      0.1      cohort None   cohort      1      30     29
## 14 fizz-buzz      0.1      cohort Ancest~ cohort      3      30     27
## 15 fizz-buzz      0.1      cohort Relati~ cohort      9      30     21
## 16 fizz-buzz      0.1    down-s~ None   down-s~      8      30     22
## 17 fizz-buzz      0.1    down-s~ Ancest~ down-s~      8      30     22
## 18 fizz-buzz      0.1    down-s~ Relati~ down-s~      7      30     23
## 19 fizz-buzz      0.5      cohort None   cohort      0      30     30
## 20 fizz-buzz      0.5      cohort Ancest~ cohort      9      30     21
## 21 fizz-buzz      0.5      cohort Relati~ cohort      6      30     24

```

##	22	fizz-buzz	0.5	down-s~	None	down-s~	0	30	30
##	23	fizz-buzz	0.5	down-s~	Ancest~	down-s~	7	30	23
##	24	fizz-buzz	0.5	down-s~	Relati~	down-s~	7	30	23
##	25	fizz-buzz	1	cohort	None	full	0	30	30
##	26	fizz-buzz	1	down-s~	None	full	0	30	30
##	27	grade	0.01	cohort	None	cohort	20	30	10
##	28	grade	0.01	cohort	Ancest~	cohort	18	30	12
##	29	grade	0.01	cohort	Relati~	cohort	23	30	7
##	30	grade	0.01	down-s~	None	down-s~	1	30	29
##	31	grade	0.01	down-s~	Ancest~	down-s~	10	30	20
##	32	grade	0.01	down-s~	Relati~	down-s~	11	30	19
##	33	grade	0.05	cohort	None	cohort	18	30	12
##	34	grade	0.05	cohort	Ancest~	cohort	13	30	17
##	35	grade	0.05	cohort	Relati~	cohort	19	30	11
##	36	grade	0.05	down-s~	None	down-s~	22	30	8
##	37	grade	0.05	down-s~	Ancest~	down-s~	12	30	18
##	38	grade	0.05	down-s~	Relati~	down-s~	11	30	19
##	39	grade	0.1	cohort	None	cohort	12	30	18
##	40	grade	0.1	cohort	Ancest~	cohort	20	30	10
##	41	grade	0.1	cohort	Relati~	cohort	15	30	15
##	42	grade	0.1	down-s~	None	down-s~	22	30	8
##	43	grade	0.1	down-s~	Ancest~	down-s~	13	30	17
##	44	grade	0.1	down-s~	Relati~	down-s~	11	30	19
##	45	grade	0.5	cohort	None	cohort	4	30	26
##	46	grade	0.5	cohort	Ancest~	cohort	2	30	28
##	47	grade	0.5	cohort	Relati~	cohort	4	30	26
##	48	grade	0.5	down-s~	None	down-s~	5	30	25
##	49	grade	0.5	down-s~	Ancest~	down-s~	9	30	21
##	50	grade	0.5	down-s~	Relati~	down-s~	4	30	26
##	51	grade	1	cohort	None	full	1	30	29
##	52	grade	1	down-s~	None	full	1	30	29
##	53	median	0.01	cohort	None	cohort	0	30	30
##	54	median	0.01	cohort	Ancest~	cohort	22	30	8
##	55	median	0.01	cohort	Relati~	cohort	27	30	3
##	56	median	0.01	down-s~	None	down-s~	8	30	22
##	57	median	0.01	down-s~	Ancest~	down-s~	13	30	17
##	58	median	0.01	down-s~	Relati~	down-s~	14	30	16
##	59	median	0.05	cohort	None	cohort	12	30	18
##	60	median	0.05	cohort	Ancest~	cohort	23	30	7
##	61	median	0.05	cohort	Relati~	cohort	25	30	5
##	62	median	0.05	down-s~	None	down-s~	4	30	26
##	63	median	0.05	down-s~	Ancest~	down-s~	19	30	11
##	64	median	0.05	down-s~	Relati~	down-s~	23	30	7
##	65	median	0.1	cohort	None	cohort	15	30	15
##	66	median	0.1	cohort	Ancest~	cohort	26	30	4
##	67	median	0.1	cohort	Relati~	cohort	24	30	6

##	68	median	0.1	down-s~	None	down-s~	16	30	14
##	69	median	0.1	down-s~	Ancest~	down-s~	21	30	9
##	70	median	0.1	down-s~	Relati~	down-s~	22	30	8
##	71	median	0.5	cohort	None	cohort	5	30	25
##	72	median	0.5	cohort	Ancest~	cohort	12	30	18
##	73	median	0.5	cohort	Relati~	cohort	13	30	17
##	74	median	0.5	down-s~	None	down-s~	2	30	28
##	75	median	0.5	down-s~	Ancest~	down-s~	15	30	15
##	76	median	0.5	down-s~	Relati~	down-s~	13	30	17
##	77	median	1	cohort	None	full	1	30	29
##	78	median	1	down-s~	None	full	1	30	29
##	79	small-or-large	0.01	cohort	None	cohort	0	30	30
##	80	small-or-large	0.01	cohort	Ancest~	cohort	1	30	29
##	81	small-or-large	0.01	cohort	Relati~	cohort	0	30	30
##	82	small-or-large	0.01	down-s~	None	down-s~	0	30	30
##	83	small-or-large	0.01	down-s~	Ancest~	down-s~	0	30	30
##	84	small-or-large	0.01	down-s~	Relati~	down-s~	0	30	30
##	85	small-or-large	0.05	cohort	None	cohort	0	30	30
##	86	small-or-large	0.05	cohort	Ancest~	cohort	0	30	30
##	87	small-or-large	0.05	cohort	Relati~	cohort	1	30	29
##	88	small-or-large	0.05	down-s~	None	down-s~	0	30	30
##	89	small-or-large	0.05	down-s~	Ancest~	down-s~	0	30	30
##	90	small-or-large	0.05	down-s~	Relati~	down-s~	0	30	30
##	91	small-or-large	0.1	cohort	None	cohort	0	30	30
##	92	small-or-large	0.1	cohort	Ancest~	cohort	0	30	30
##	93	small-or-large	0.1	cohort	Relati~	cohort	0	30	30
##	94	small-or-large	0.1	down-s~	None	down-s~	0	30	30
##	95	small-or-large	0.1	down-s~	Ancest~	down-s~	0	30	30
##	96	small-or-large	0.1	down-s~	Relati~	down-s~	0	30	30
##	97	small-or-large	0.5	cohort	None	cohort	0	30	30
##	98	small-or-large	0.5	cohort	Ancest~	cohort	0	30	30
##	99	small-or-large	0.5	cohort	Relati~	cohort	0	30	30
##	100	small-or-large	0.5	down-s~	None	down-s~	0	30	30
##	101	small-or-large	0.5	down-s~	Ancest~	down-s~	0	30	30
##	102	small-or-large	0.5	down-s~	Relati~	down-s~	0	30	30
##	103	small-or-large	1	cohort	None	full	0	30	30
##	104	small-or-large	1	down-s~	None	full	0	30	30
##	... with abbreviated variable names 1: eval_mode_row, 2: EVAL_FIT_EST_MODE,								
##	3: EVAL_MODE, 4: solution_count, 5: replicates, 6: no_solution_count								

### 5.2.2 Load time series data

Because runs stop when a solution is found, the over time data aren't completely fair to compare across replicates.



```

ts_data_loc <- paste0(working_directory, "data/time_series.csv")
ts_data <- read_csv(ts_data_loc)

## Rows: 262694 Columns: 24
## -- Column specification -----
## Delimiter: ","
## chr (6): EVAL_FIT_EST_MODE, EVAL_MODE, PROBLEM, SELECTION, TESTING_SET_PATH...
## dbl (18): EVAL_MAX_PHYLO_SEARCH_DEPTH, NUM_COHORTS, SEED, TEST_DOWNSAMPLE_RA...
##
## i Use 'spec()' to retrieve the full column specification for this data.
## i Specify the column types or set 'show_col_types = FALSE' to quiet this message.

ts_data <- ts_data %>%
  mutate(
    eval_mode_row = case_when(
      EVAL_MODE == "full" & TEST_DOWNSAMPLE_RATE == "1" ~ "down-sample",
      EVAL_MODE == "full" & NUM_COHORTS == "1" ~ "cohort",
      .default = EVAL_MODE
    ),
    evals_per_gen = case_when(
      EVAL_MODE == "cohort" ~ 1.0 / NUM_COHORTS,
      EVAL_MODE == "down-sample" ~ TEST_DOWNSAMPLE_RATE,
      EVAL_MODE == "full" ~ 1.0
    ),
    EVAL_FIT_EST_MODE = case_when(
      EVAL_FIT_EST_MODE == "ancestor-opt" ~ "ancestor",
      EVAL_FIT_EST_MODE == "relative-opt" ~ "relative",
      .default = EVAL_FIT_EST_MODE
    ),
    .keep = "all"
  ) %>%
  mutate(
    evals_per_gen = as.factor(evals_per_gen),
    PROBLEM = as.factor(PROBLEM),
    SELECTION = as.factor(SELECTION),
    EVAL_MODE = as.factor(EVAL_MODE),
    NUM_COHORTS = as.factor(NUM_COHORTS),
    TEST_DOWNSAMPLE_RATE = as.factor(TEST_DOWNSAMPLE_RATE),
    EVAL_FIT_EST_MODE = factor(
      EVAL_FIT_EST_MODE,
      levels = c(
        "none",
        "ancestor",
        "relative"
      )
    ),

```

```

        labels = c(
            "None",
            "Ancestor",
            "Relative"
        )
    ),
    .keep = "all"
)

avg_across_time_data <- ts_data %>%
  group_by(
    PROBLEM,
    evals_per_gen,
    eval_mode_row,
    EVAL_FIT_EST_MODE,
    EVAL_MODE,
    SEED
  ) %>%
  summarize(
    avg_entropy_selected_ids = mean(entropy_selected_ids),
    avg_genotype_pairwise_distance = mean(mean_genotype_pairwise_distance),
    avg_num_unique_selected = mean(num_unique_selected),
    max_genotype_pairwise_distance = max(mean_genotype_pairwise_distance)
  )

```

## 'summarise()' has grouped output by 'PROBLEM', 'evals\_per\_gen', 'eval\_mode\_row',  
 ## 'EVAL\_FIT\_EST\_MODE', 'EVAL\_MODE'. You can override using the '.groups' argument.

### 5.3 Problem-solving success

```

plt_solutions <- function(data, problem) {
  data %>%
    filter(PROBLEM == problem) %>%
    ggplot(
      aes(
        x = EVAL_FIT_EST_MODE,
        y = solution_count,
        fill = EVAL_FIT_EST_MODE
      )
    ) +
    geom_col() +
    scale_y_continuous(

```

```

    limits = c(0, 30),
    breaks = seq(0, 30),
    labels = seq(0, 30)
  ) +
  scale_fill_bright() +
  scale_color_bright() +
  facet_grid(
    eval_mode_row ~ evals_per_gen
  ) +
  labs(title = problem) +
  theme(
    legend.position = "none",
    axis.text.x = element_text(angle = 45, hjust = 1)
  )
ggsave(
  filename = paste0(plot_directory, problem, "-final.pdf"),
  height = 15,
  width = 7
)
}
# plt_solutions(solution_counts, "small-or-large")
plt_solutions(solution_counts, "median")
plt_solutions(solution_counts, "grade")
plt_solutions(solution_counts, "fizz-buzz")

sol_stats_data <- summary_data %>%
  filter(EVAL_MODE != "full" & EVAL_MODE != "cohort-full-competee") %>%
  # filter(PROBLEM != "small-or-large") %>%
  group_by(
    PROBLEM,
    evals_per_gen,
    EVAL_FIT_EST_MODE,
    EVAL_MODE
  ) %>%
  summarize(
    solution_count = sum(found_solution == "1"),
    replicates = n(),
    no_solution_count = n() - sum(found_solution == "1")
  ) %>%
  unite(
    "grouping",
    PROBLEM,
    EVAL_MODE,
    evals_per_gen,
    sep = "__"
  )

```

```

) %>%
select(
  !replicates,
) %>%
mutate(
  grouping = as.factor(grouping)
)

```

## 'summarise()' has grouped output by 'PROBLEM', 'evals\_per\_gen',  
## 'EVAL\_FIT\_EST\_MODE'. You can override using the '.groups' argument.

```

fisher_results <- data.frame(
  comparison = character(),
  group1 = character(),
  group2 = character(),
  n = integer(),
  p = double(),
  p.adj = double(),
  p.adj.signif = character()
)
groupings <- levels(sol_stats_data$grouping)
for (g in groupings) {

  ft_results <- sol_stats_data %>%
    filter(grouping == g) %>%
    select(!grouping) %>%
    column_to_rownames(var = "EVAL_FIT_EST_MODE") %>%
    pairwise_fisher_test(
      p.adjust.method = "holm"
    ) %>%
    add_significance("p.adj")

  ft_results <- ft_results %>%
    mutate(
      comparison = rep(g, nrow(ft_results)),
      .keep = "all"
    ) %>%
    relocate(comparison)

  fisher_results <- rbind(
    fisher_results,
    ft_results
  )
}
kable(fisher_results)

```

comparison	group1	group2	n	p	p.adj	p.adj.signif
fizz-buzz__cohort__0.01	None	Ancestor	60	4.92e-01	9.84e-01	ns
fizz-buzz__cohort__0.01	None	Relative	60	2.37e-01	7.11e-01	ns
fizz-buzz__cohort__0.01	Ancestor	Relative	60	1.00e+00	1.00e+00	ns
fizz-buzz__cohort__0.05	None	Ancestor	60	7.06e-01	1.00e+00	ns
fizz-buzz__cohort__0.05	None	Relative	60	7.48e-01	1.00e+00	ns
fizz-buzz__cohort__0.05	Ancestor	Relative	60	2.99e-01	8.97e-01	ns
fizz-buzz__cohort__0.1	None	Ancestor	60	6.12e-01	6.12e-01	ns
fizz-buzz__cohort__0.1	None	Relative	60	1.22e-02	3.66e-02	*
fizz-buzz__cohort__0.1	Ancestor	Relative	60	1.04e-01	2.08e-01	ns
fizz-buzz__cohort__0.5	None	Ancestor	60	1.94e-03	5.82e-03	**
fizz-buzz__cohort__0.5	None	Relative	60	2.37e-02	4.74e-02	*
fizz-buzz__cohort__0.5	Ancestor	Relative	60	5.52e-01	5.52e-01	ns
fizz-buzz__down-sample__0.01	None	Ancestor	60	1.00e+00	1.00e+00	ns
fizz-buzz__down-sample__0.01	None	Relative	60	1.00e+00	1.00e+00	ns
fizz-buzz__down-sample__0.01	Ancestor	Relative	60	1.00e+00	1.00e+00	ns
fizz-buzz__down-sample__0.05	None	Ancestor	60	1.90e-06	5.70e-06	****
fizz-buzz__down-sample__0.05	None	Relative	60	1.90e-06	5.70e-06	****
fizz-buzz__down-sample__0.05	Ancestor	Relative	60	1.00e+00	1.00e+00	ns
fizz-buzz__down-sample__0.1	None	Ancestor	60	1.00e+00	1.00e+00	ns
fizz-buzz__down-sample__0.1	None	Relative	60	1.00e+00	1.00e+00	ns
fizz-buzz__down-sample__0.1	Ancestor	Relative	60	1.00e+00	1.00e+00	ns
fizz-buzz__down-sample__0.5	None	Ancestor	60	1.05e-02	3.15e-02	*
fizz-buzz__down-sample__0.5	None	Relative	60	1.05e-02	3.15e-02	*
fizz-buzz__down-sample__0.5	Ancestor	Relative	60	1.00e+00	1.00e+00	ns
grade__cohort__0.01	None	Ancestor	60	7.89e-01	1.00e+00	ns
grade__cohort__0.01	None	Relative	60	5.67e-01	1.00e+00	ns
grade__cohort__0.01	Ancestor	Relative	60	2.67e-01	8.01e-01	ns
grade__cohort__0.05	None	Ancestor	60	3.01e-01	6.02e-01	ns
grade__cohort__0.05	None	Relative	60	1.00e+00	1.00e+00	ns
grade__cohort__0.05	Ancestor	Relative	60	1.95e-01	5.85e-01	ns
grade__cohort__0.1	None	Ancestor	60	6.92e-02	2.08e-01	ns
grade__cohort__0.1	None	Relative	60	6.04e-01	6.04e-01	ns
grade__cohort__0.1	Ancestor	Relative	60	2.95e-01	5.90e-01	ns
grade__cohort__0.5	None	Ancestor	60	6.71e-01	1.00e+00	ns
grade__cohort__0.5	None	Relative	60	1.00e+00	1.00e+00	ns
grade__cohort__0.5	Ancestor	Relative	60	6.71e-01	1.00e+00	ns
grade__down-sample__0.01	None	Ancestor	60	5.58e-03	1.12e-02	*
grade__down-sample__0.01	None	Relative	60	2.47e-03	7.41e-03	**
grade__down-sample__0.01	Ancestor	Relative	60	1.00e+00	1.00e+00	ns
grade__down-sample__0.05	None	Ancestor	60	1.82e-02	3.64e-02	*
grade__down-sample__0.05	None	Relative	60	8.87e-03	2.66e-02	*
grade__down-sample__0.05	Ancestor	Relative	60	1.00e+00	1.00e+00	ns
grade__down-sample__0.1	None	Ancestor	60	3.52e-02	7.04e-02	ns
grade__down-sample__0.1	None	Relative	60	8.87e-03	2.66e-02	*
grade__down-sample__0.1	Ancestor	Relative	60	7.92e-01	7.92e-01	ns
grade__down-sample__0.5	None	Ancestor	60	3.60e-01	7.20e-01	ns
grade__down-sample__0.5	None	Relative	60	1.00e+00	1.00e+00	ns
grade__down-sample__0.5	Ancestor	Relative	60	2.09e-01	6.27e-01	ns
median__cohort__0.01	None	Ancestor	60	0.00e+00	0.00e+00	****
median__cohort__0.01	None	Relative	60	0.00e+00	0.00e+00	****
median__cohort__0.01	Ancestor	Relative	60	1.81e-01	1.81e-01	ns
median__cohort__0.05	None	Ancestor	60	8.21e-03	1.64e-02	*
median__cohort__0.05	None	Relative	60	1.19e-03	3.57e-03	**

```
# temp <- sol_stats_data %>%
#   filter(grouping == "fizz-buzz__cohort__0.5") %>%
#   select(!grouping) %>%
#   column_to_rownames(var = "EVAL_FIT_EST_MODE")

# pairwise_fisher_test(temp, p.adjust.method = "holm")
# print(
#   solution_counts %>% filter(eval_mode_row == "cohort" & PROBLEM == "fizz-buzz"),
#   n = 150
# )

# kable(
#   fisher_results %>%
#   filter(grepl("cohort", comparison) & grepl("fizz-buzz", comparison))
# )
```