Perturb-seq inference: `sceptre`

Gene Katsevich

March 5, 2024

1 Overview

In the `sceptreIGVF` package, I have implemented a function called `inference_sceptre()`, which takes as input a `MuData` object and optional additional analysis parameters and outputs another `MuData` object with the inference results (p-value and log fold change) added.

Below are demonstrations of using `inference_sceptre()` with all default arguments and with some additional arguments.

```
library(sceptreIGVF)
library(MultiAssayExperiment)
library(ggplot2)
data("mudata_inference")        # this is the Gasperini subset
```

2 Default arguments

```
mudata_out <- inference_sceptre(mudata_inference)
```

```
## Welcome to sceptre.
## Submit issues on the sceptre website: github.com/Katsevich-Lab/sceptre
## Read the sceptre manual: timothy-barry.github.io/sceptre-book/
```

```
metadata(mudata_out)$test_results |>
    as.data.frame() |>
    head()
```

```
##           gene_id intended_target_name        pair_type      p_value    log2_fc
## 1 ENSG00000187109      ENSG00000187109 positive_control 4.941655e-83 -0.7542542
## 2 ENSG00000114850      ENSG00000114850 positive_control 5.065197e-82 -1.8660050
## 3 ENSG00000134851      ENSG00000134851 positive_control 8.874414e-53 -0.9552938
## 4 ENSG00000163866      ENSG00000163866 positive_control 7.684199e-51 -1.2229365
## 5 ENSG00000181610      ENSG00000181610 positive_control 2.736911e-47 -1.3298467
## 6 ENSG00000113552      ENSG00000113552 positive_control 1.222514e-36 -1.7534278
```

```
metadata(mudata_out)$test_results |>
    as.data.frame() |>
    na.omit() |>
    ggplot(aes(x = pair_type, color = pair_type, y = p_value)) +
    geom_jitter() +
    scale_y_log10() +
    theme_bw() +
    theme(legend.position = "none")
```
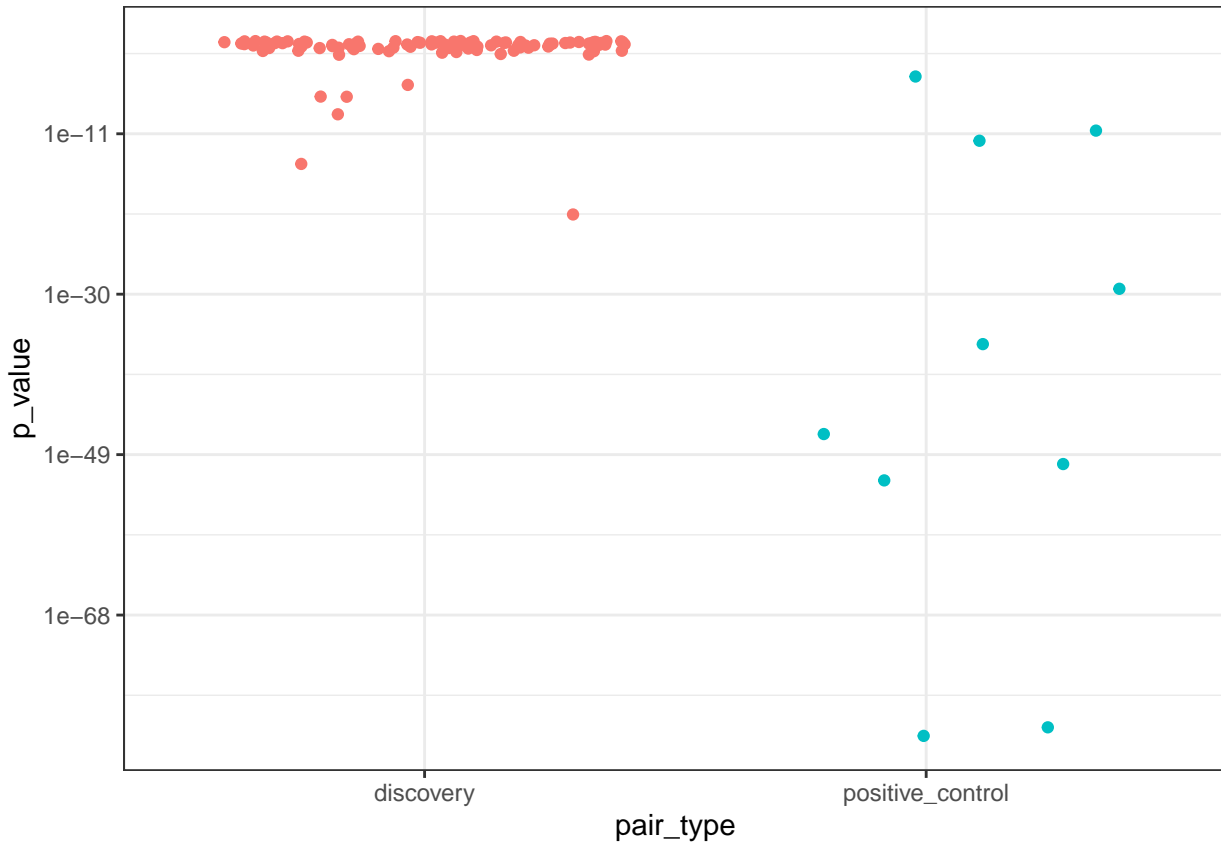
3 Optional arguments

`sceptre` has several optional arguments that can be used to customize the analysis, including the following:

- `side`: The sidedness of the test
- `grna_integration_strategy`: The strategy for integrating information across gRNAs with the same target
- `formula_object`: The formula object to use for the negative binomial model
- `fit_parametric_curve`: Whether to fit a parametric curve to the data for computational speed
- `control_group`: Which set of cells to use to contrast with cells having a perturbation of interest
- `resampling_mechanism`: Whether to use conditional randomization or permutations

These arguments are documented in detail in the sceptre manual. While all of these arguments have sensible defaults, users might want to customize the analysis by changing one or more of them. Any of the above arguments can be specified to `inference_sceptre()` as named arguments. Below is an example:

```
# let's run a left-sided test, with a different formula:
mudata_out <- inference_sceptre(
  mudata_inference,
  side = "left",
  formula_object = formula(~ prep_batch + log(response_n_nonzero) + log(response_n_umis))
)
```

```
metadata(mudata_out)$test_results |>
    as.data.frame() |>
    head()
```

```
##            gene_id intended_target_name        pair_type      p_value     log2_fc
```

```
## 1 ENSG00000187109    ENSG00000187109 positive_control 5.775438e-77 -0.7544941
## 2 ENSG00000114850    ENSG00000114850 positive_control 7.718576e-77 -1.8672587
## 3 ENSG00000134851    ENSG00000134851 positive_control 9.716034e-47 -0.9553815
## 4 ENSG00000163866    ENSG00000163866 positive_control 5.692775e-46 -1.2229287
## 5 ENSG00000181610    ENSG00000181610 positive_control 1.050974e-45 -1.3302300
## 6 ENSG00000113552    ENSG00000113552 positive_control 5.280541e-40 -1.7519115
```

```r
metadata(mudata_out)$test_results |>
    as.data.frame() |>
    na.omit() |>
    ggplot(aes(x = pair_type, color = pair_type, y = p_value)) +
    geom_jitter() +
    scale_y_log10() +
    theme_bw() +
    theme(legend.position = "none")
```