sSNAPPY: an R/Bioconductor package for single-sample directional pathway perturbation analysis

Wenjun Liu*[1], Ville-Petteri Mäkinen[2,3], Wayne D. Tilley[1], and Stephen M. Pederson[1,4,5]

[1]**Dame Roma Mitchell Cancer Research Laboratories, Adelaide Medical School, Faculty of Health and Medical Sciences, University of Adelaide, Adelaide, Australia**
[2]**Computational Medicine, Faculty of Medicine, University of Oulu, Oulu, Finland**
[3]**Center for Life Course Health Research, Faculty of Medicine, University of Oulu, Oulu, Finland**
[4]**Black Ochre Data Laboratories, Telethon Kids Institute, Adelaide, Australia**
[5]**John Curtin School of Medical Research, Australian National University, Canberra, Australia**

Abstract A common outcome of analysing RNA-Seq data is the detection of biological pathways with significantly altered activity between the conditions under investigation. Whilst many strategies test for over-representation within pre-defined gene-sets for genes showing changed expression, these analyses typically do not account for gene-gene interactions encoded by pathway topologies, and are not able to directly predict the directional change of pathway activity. To address these issues, we have developed a single-sample pathway perturbation analysis method *sSNAPPY*, now available as an R/Bioconductor package, which leverages pathway topology information to compute pathway perturbation scores, and predicts the direction of change across a set of pathways. Here, we demonstrate the use of *sSNAPPY* by applying the method to public scRNA-seq data, derived from ovarian cancer patient tissues collected before and after chemotherapy. Not only were we able to predict the directions of significant perturbations of pathways discussed in the original study, but *sSNAPPY* was also able to detect significant changes of other biological processes, yielding far greater insight into the response to treatment. *sSNAPPY* represents a novel pathway analysis strategy that takes into consideration of pathway topology to predict impacted biology pathways, both within related samples and across treatment groups. In addition to not relying on the detection of differentially expressed genes, the method and associated R package offer important flexibility and provide powerful visualisation tools.

Keywords

RNA-seq, pathway enrichment, R package, topology, KEGG, Reactome, scRNA-seq

*Corresponding Author wenjun.liu@adelaide.edu.au

R version: R version 4.3.0 (2023-04-21)

Bioconductor version: 3.17

Package: 1.4.4

Introduction

Using pathway enrichment analysis to gain biological insights from gene expression data is a pivotal step in the analysis and interpretation of RNA-seq data, for which numerous methods have been developed (reviewed in [1, 2]). Many existing methods tend to view pathways simply as a collection of gene names, as seen in those relying on the detection of differentially expressed genes and applying over-representation analysis (ORA) strategies, and those scoring all genes using functional class scoring (FCS), such as in Gene Set Enrichment Analysis (GSEA)[3], arguably the most widely-used approach. However, databases such as the Kyoto Encyclopaedia of Genes and Genomes (KEGG)[4] and WikiPathways[5] capture not only which genes are implicated in a certain biological process but also their interactions, activating or inhibitory roles, and their relative importance within the pathway, all of which are overlooked in ORA- and FCS-based approaches.

To fully utilise that additional information, the latest generation of pathway analysis approaches include many which are topology-based such as SPIA[6], DEGraph[7], NetGSA[8] and PRS[9], as well as others which explicitly model inter-gene correlations[10]. Despite differences in the null hypotheses tested across these approaches, overall, they have demonstrated enhanced sensitivity and specificity due to their abilities to take gene-gene interconnections into account[11, 12]. Nevertheless, most topology-based methods focus only on comparing activities of pathways between two treatment groups and cannot be used to score individual samples (Figure 1). However, in heterogenous data where more than one factor may be influencing observations[13], incorporating scoring within paired samples may be desirable and may be able to reveal more nuanced insights. To address this gap, we present a Single-Sample directioNAl Pathway Perturbation analYsis methodology called *sSNAPPY*, available as an R/Bioconductor package. This article defines how *sSNAPPY* computes changes in gene expression within paired samples, and propagates this through gene-set topologies to predict the perturbation in pathway activities within paired samples, before providing summarised results across an entire dataset (Figure 1). The practical usage of the *sSNAPPY* R/Bioconductor package is illustrated through the analysis of a public scRNA-seq dataset using the pseudo-bulk strategy.

[Figure 1 about here.]

Methods

Implementation

sSNAPPY is an R package that has been reviewed and published on the open-source bioinformatics software platform Bioconductor with all source code available via GitHub. The methodology itself is topology-based, designed to compute directional, single-sample, pathway perturbation scores for gene expression datasets with a matched-pair, or nested design (eg. samples collected before and after treatment). This allows for the detection of pathway perturbations within all samples from a treatment group, but also within individual samples. The only data required to run *sSNAPPY*, is a log-transformed expression matrix (e.g. logCPM) with matching sample metadata describing treatment groups and the nested structure. It is assumed that all preprocessing has been performed beforehand, such as the exclusion of low-signal genes or normalisation to minimise technical artefacts like GC-bias.

The first step performed by *sSNAPPY*, is to estimate sample-specific log fold-change ($\delta_{ghi} = \mu_{ghi} - \mu_{g0i}$) across all genes g for each treatment h within each set of nested replicates i, by subtracting expression estimates for the baseline samples μ_{g0i} from those in the treatment group h. Each set of nested replicates may be drawn from treated or control samples within cell-line passages, or from treatments applied to the same donor tissue. It should also be noted that *sSNAPPY* is applicable to any number of treatment/condition levels and sample numbers within each treatment group are not required to be balanced.

It is well known that in RNA-seq data, genes with lower expression tend to have greater variability in signal and more broadly spread estimates of change[14]. As such, we utilise a gene-level weighting strategy to down-weight fold-change estimates for low-abundance genes prior to passing these estimates to sSNAPPY. Gene-level weights w_g are obtained in a treatment-agnostic manner by fitting a loess curve through the relationship between observed gene-level variance (σ^2_g) and average signal ($\bar{\mu}_g$)(Figure 2), and taking the inverse of the loess-predicted variance as the weight $w_g = a/f(\bar{\mu}_g)$, where $f(\bar{\mu}_g)$ is the predicted value from the loess curve and the constant a ensures $\sum w_g = 1$. We then use these weighted estimates of log fold-change ($\delta^*_{ghi} = w_g \delta_{ghi}$) in the calculation of all subsequent pathway perturbation scores.

[Figure 2 about here.]

sSNAPPY extends the topology-based scoring algorithm initially proposed in SPIA[6] which propagates fold-change estimates from genes considered as differentially expressed through pathway topologies, to compute a perturbation score for each pathway. In contrast to SPIA, *sSNAPPY* uses fold-change estimates from all detected genes. By modifying the algorithm to incorporate single-sample, weighted estimates of fold-change, we are able to numerically represent changes in a pathway within a given sample, and subsequently model these across all samples within a treatment group. Thus, we define the single-sample perturbation score (S_{hip}) for a given pathway p and treatment h for a set of nested samples i:

$$S_{hip} = \sum_{g \in G_p} [S_{ghip} - \delta^*_{ghi}], \text{ where}$$

$$S_{ghip} = \delta^*_{ghi} + \sum_{g' \in U_{gp}} \beta_{gg'p} \frac{S_{g'hip}}{N_{g'p}}$$

where:

- G_p represents the set of genes in pathway p, such that $g \in G_p$
- S_{ghip} is the gene-, treatment- and sample-specific perturbation score for pathway p
- $\delta^*_{ghi} = w_g \delta_{ghi}$ is the weighted log fold-change of gene g as described above
- U_{gp} is the subset of G_p containing only the genes directly upstream of gene g
- $\beta_{gg'p}$ is the pair-wise gene-gene interactions[6] encoded by the topology matrix for genes g and g'
- N_{gp} is the number of downstream genes from any gene g
- S_{hip} is the accumulated pathway perturbation score for pathway p in treatment h within sample i

To scale single-sample pathway perturbation scores (S_{hip}) so they are comparable across pathways, and to test for significance of individual scores, null distributions of perturbation scores for each pathway are generated through a sample permutation strategy, which retains any existing correlation structures between genes within a pathway. During permutation, all sample labels are randomly shuffled and permuted pseudo-pairs formed from the re-shuffled labels. Single-sample fold-changes are then calculated for each pseudo-pair of permuted samples while the rest of the scoring algorithm remains unchanged. The median and median absolute deviation (MAD) are calculated from the set of permuted perturbation scores within each pathway, and used to normalise the raw perturbation scores to robust Z-scores. All possible permuted pseudo-pairs are sampled unless otherwise specified, such that in an experiment with I total samples, the maximum number of unique permuted pairs is ${}^{I}P_2 = \frac{I!}{(I-2)!} = I \times (I-1)$. Permutation p-values for individual scores, indicating the approximate significance of pathway perturbation at the single-sample level, are also derived by assessing the proportion of permuted scores with absolute values as extreme, or more extreme, than the absolute value of test perturbation within each pathway[15]. Since the smallest achievable permutation p-value is $1/NP$, where NP is the number of permuted pairs, accurate estimation of small p-value requires a large number of permutation that is only feasible in data with large sample size. As a guideline, GSEA recommends a minimum of 7 samples in each treatment group for utilizing their phenotype permutation approach[16].

Apart from assessing whether a pathway's activity changed significantly within an individual sample, users may also be interested in detecting changes across all samples within a treatment, which can be performed by modelling scores using regression models, and incorporating Smyth's moderated t-statistic[17] as implemented in *limma*[18]. The single-sample nature of *sSNAPPY*'s pathway perturbation scores is particularly helpful for datasets with complex experimental designs or known confounding factors as these can also be incorporated into the final regression models.

The Bioconductor package graphite[19] provides functions that can be used to retrieve pathway topologies from a database and convert topology information to adjacency matrices. In order to streamline this process we have implemented a convenience function, where users only need to provide the name of the desired database and species to retrieve all topology information in the format required by the scoring algorithm with the correct type of gene identifiers (ie. Entrez ID).

Operation

The package has been tested on all operating systems, requiring R > 4.3.0, and can be installed using BiocManager as follows.

```
if (!requireNamespace("BiocManager", quietly = TRUE))
  install.packages("BiocManager")
BiocManager::install("sSNAPPY")
```

Use Cases

Data

To demonstrate the application of *sSNAPPY*, we used pre-processed counts from a publicly available scRNA-seq dataset, retrieved from Gene Expression Omnibus (GEO) with accession code GSE165897. This dataset consists of 11 high-grade serous ovarian cancer (HGSOC) patients samples taken before and after chemotherapy[20]. *sSNAPPY* was used to re-analyse data from the epithelial cells as they were the primary focus of the original study. Since *sSNAPPY* was designed primarily for bulk RNA-seq data, and as such, counts from epithelial cells within the same samples were first summed into pseudo-bulk profiles, giving rise to a total of 22 samples. We considered a gene detectable if we observed >1.5 counts per million in >11 samples out of 22, ideally representing all samples from a complete treatment group. 11,101 (33.8%) of the 32,847 annotated genes passed this selection criteria and were included for downstream analyses. Conditional quantile normalisation[21] was then applied to mitigate potential biases introduced by gene length and GC content. The normalised logCPM matrix of the processed dataset and sample metadata can be downloaded from here.

The following packages are required for this workflow

```
library(sSNAPPY)
```

```
## Warning: package 'sSNAPPY' was built under R version 4.3.1
```

```
library(tidyverse)
library(magrittr)
library(ggplot2)
library(patchwork)
library(kableExtra)
library(AnnotationHub)
library(edgeR)
library(patchwork)
library(colorspace)
```

To begin running the *sSNAPPY* workflow, we first load our expression matrix and define our sample-level metadata. Importantly, the row names of the expression matrix must be specified as EntrezGene IDs, for compatibility with pathway databases. Genes without EntrezGene IDs were excluded during pre-processing, leaving 10,098 genes in the example expression matrix. The treatment column within our metadata is expected to be a factor, with the reference level interpreted as the control treatment.

```
logCPM <- read_tsv(here::here("data/logCPM.tsv")) %>%
    column_to_rownames("entrezid")
sample_meta <- read_tsv(here::here("data/sample_meta.tsv"), col_types = "cfccncnc")
head(sample_meta)
```

```
## # A tibble: 6 x 8
##   sample        treatment patient_id anatomical_location   Age Stage   PFI CRS
##   <chr>         <fct>     <chr>      <chr>                <dbl> <chr> <dbl> <chr>
## 1 EOC372_treat~ treatmen~ EOC372     Peritoneum              68 IIIC    460 1
## 2 EOC372_post-~ post-NACT EOC372     Peritoneum              68 IIIC    460 1
## 3 EOC443_post-~ post-NACT EOC443     Omentum                 54 IVA     177 3
## 4 EOC443_treat~ treatmen~ EOC443     Omentum                 54 IVA     177 3
## 5 EOC540_treat~ treatmen~ EOC540     Omentum                 62 IIIC    126 2
## 6 EOC540_post-~ post-NACT EOC540     Omentum                 62 IIIC    126 2
```

Retrieval of Pathway Topology

Next, pathway topology information needs to be retrieved from a chosen database, and this is the only step requiring internet access. Using the Reactome database[22] as an example, the retrieved topology information will be stored as a list where each element corresponds to a pathway and the numbers in the matrices encode gene-gene interactions.

```
gsTopology <- retrieve_topology(database = "reactome", species = "hsapiens")
```

In addition to downloading topology matrices for all pathways, it is also possible to provide a restricted set of keywords for a targeted analysis. For example, passing the argument `keyword = c("metabolism", "estrogen")` would only return the subset of pathways which match either of these keywords. Multiple databases are also able to be searched by passing a vector of database names to the `database` argument.

Score Single-Sample Pathway Perturbation

To compute the single-sample fold-changes (i.e. logFC) required for the set of perturbation scores, samples must be 'matched pairs' or nested, as would be found when analysing biopsies pre- vs post-treatment, or untreated vs treated cell lines nested by passage. The factor defining the nested structure is passed to the `weight_ss_fc()` function through the `groupBy` parameter. In our example dataset, pre- and post-treatment samples are matched by the "patient_id" column.

```
weightedFC <- weight_ss_fc(
  as.matrix(logCPM), metadata = sample_meta,
  sampleColumn = "sample", groupBy = "patient_id", treatColumn = "treatment"
)
glimpse(weightedFC)
```

The output of `weight_ss_fc()` is a `list` where one element is a matrix of weighted single-sample fold-changes (δ^*_{ghi}), with rows corresponding to genes and columns to samples, and the other element is the vector of gene-wise weights (w_g) used to calculate the weighted log fold-change (δ^*_{ghi}), as described above. By default, the string `ENTREZID:` is added to all row names of the δ^*_{ghi} matrix to be compatible with the format Reactome pathway topologies are retrieved in.

The matrix of δ^*_{ghi} values are then passed to pathway topologies to compute gene-wise perturbation scores for all genes within a pathway, before being summed into a single score for each pathway. `raw_gene_pert()` returns a list, with each element containing the gene-level perturbation scores for a given pathway, with each matrix able to be used during downstream analysis to identify which genes play the most significant roles in each pathway, as demonstrated in later sections. Pathway-level perturbation scores (S_{hip}) are then returned as a data.frame containing sample and gene-set names after calling `pathway_pert()`. Pathways with zero perturbation scores across all genes and samples are dropped at this step.

```
genePertScore <- raw_gene_pert(weightedFC$weighted_logFC, gsTopology)
ssPertScore <- pathway_pert(genePertScore, weightedFC$weighted_logFC)
head(ssPertScore)
```

```
##             sample         score                          gs_name
## 1 EOC372_post-NACT -2.292688e-04 reactome.Interleukin-6 signaling
## 2 EOC443_post-NACT -2.447003e-04 reactome.Interleukin-6 signaling
## 3 EOC540_post-NACT -1.848758e-04 reactome.Interleukin-6 signaling
## 4   EOC3_post-NACT -1.229489e-04 reactome.Interleukin-6 signaling
## 5  EOC87_post-NACT  3.427132e-05 reactome.Interleukin-6 signaling
## 6 EOC136_post-NACT  2.822155e-04 reactome.Interleukin-6 signaling
```

Sample Permutation for Normalisation and Significance Testing

The range of values obtained from each pathway will vary greatly due to the variability in topology structures. To determine the significance of individual scores and transform scores to ensure they are comparable across pathways, sSNAPPY utilises a sample-permutation strategy to estimate the null distributions of perturbation scores. Since sample labels will be permuted randomly to put samples into pseudo-pairs, sample metadata is not required by the `generate_permuted _scores()` function. All possible random pairs between samples will be sampled by default, unless otherwise specified. In this example dataset with a total of 22 samples, the full set of 462 (i.e. 22×21) permuted scores will be computed for each pathway.

```
permutedScore <- generate_permuted_scores(
    as.matrix(logCPM), gsTopology = gsTopology, weight = weightedFC$weight
)
```

Apart from pathways whose permuted perturbation scores are consistently zero, the empirical distributions of remaining pathways are expected to be approximately normally distributed with $\mu = 0$, but with the scale of distributions heavily impacted by both the number of genes within each pathway and the overall topology. To demonstrate this, we randomly selected 6 pathways to demonstrate their quantile-quantile (q-q) plot and visualised the distributions of their permuted perturbation scores as boxplots (Figure 3).

```
set.seed(123)
random_pathways <- permutedScore %>%
    keep(~all(.!=0)) %>%
    .[sample(seq_along(.), 6)] %>%
```

```
    as.data.frame() %>%
    pivot_longer(
      cols = everything(), names_to = "gs_name", values_to = "score"
    ) %>%
    mutate(
        gs_name = str_replace_all(gs_name, "\\.", " "),
        gs_name = str_remove_all(gs_name, "reactome ")
    )
p1 <- random_pathways %>%
  ggplot(aes(sample = score, colour = gs_name)) +
  stat_qq() +
  stat_qq_line(colour = "black") +
  facet_wrap(~str_wrap(gs_name, width = 25), scales= "free") +
  labs(y = "Permuted Perturbation Score", x = "Theoretical Quantiles") +
  theme_bw() +
  theme(
      legend.position = "none",
      text = element_text(size = 14),
      strip.text = element_text(size = 16))
p2 <- random_pathways %>%
    ggplot(aes(gs_name, score, fill = gs_name)) +
    geom_boxplot() +
    scale_x_discrete(labels = function(x) str_wrap(x, width = 10)) +
    scale_fill_discrete(name = "Gene-set Name") +
    labs(x = "Pathway", y = "Permuted Perturbation Score") +
    theme_bw() +
    theme(
      legend.position = "none",
      axis.title = element_text(size = 16),
      axis.text = element_text(size = 14)
    )
(p1 / p2) +
    plot_annotation(tag_levels = "A") +
    plot_layout(heights = c(0.6, 0.4))
```

[Figure 3 about here.]

The distributions obtained from label permutations are then used to convert each pathway-level score into a robust Z-score using the function `normalise_by_permu()`. Two-sided p-values for individual scores are computed based on how extreme test scores are in comparison to permuted scores for each pathway, and corrected for multiple testing using any of the available methods, returning the FDR-adjusted values by default. In our example data, no pathways would be considered as significantly perturbed at the single-sample level using an FDR adjustment with $\alpha = 0.05$.

```
normalisedScores <- normalise_by_permu(permutedScore, ssPertScore,
                                       sortBy = "pvalue")
head(normalisedScores)
```

```
##               MAD MEDIAN
## 2306 0.0006067519      0
## 2525 0.0002909911      0
## 5869 0.0001652241      0
## 5871 0.0001652241      0
## 5872 0.0001652241      0
## 7721 0.0275863198      0
##                                                          gs_name
## 2306 reactome.Golgi Cisternae Pericentriolar Stack Reorganization
## 2525       reactome.DNA Damage/Telomere Stress Induced Senescence
## 5869                          reactome.Defective CHST6 causes MCDC1
## 5871             reactome.Defective ST3GAL3 causes MCT12 and EIEE15
## 5872       reactome.Defective B4GALT1 causes B4GALT1-CDG (CDG-2d)
## 7721                          reactome.Mitochondrial protein import
##                 sample         score   robustZ      pvalue adjPvalue
## 2306 EOC153_post-NACT -0.0013632057 -2.246727 0.004329004 1.0000000
```

```
## 2525 EOC153_post-NACT  0.0006149637  2.113342 0.004329004 1.0000000
## 5869 EOC349_post-NACT -0.0003304944 -2.000279 0.004329004 0.9993248
## 5871 EOC349_post-NACT -0.0003304944 -2.000279 0.004329004 0.9993248
## 5872 EOC349_post-NACT -0.0003304944 -2.000279 0.004329004 0.9993248
## 7721 EOC443_post-NACT -0.0598366717 -2.169070 0.004329004 0.9471861
```

A key question of interest in our example dataset is to identify which biological processes were impacted by chemotherapy across the entire group of patients. Using the sample-level output obtained above, we can explore this by applying t-tests or regression models across all samples. In order to minimise spurious results, Smyth's moderated t-statistics[17] are able to be applied across the complete dataset, with a constant variance assumed across all pathways, given that we are using Z-scores. To perform this analysis, robust Z-scores were converted to a matrix and standard *limma* methodologies were used. For our use case here, where only one treatment group is present, no design matrix is required and a simple t-test is appropriate.

```
z_matrix <- normalisedScores %>%
    dplyr::select(robustZ, gs_name, sample) %>%
    pivot_wider(names_from = "sample", values_from = "robustZ") %>%
    column_to_rownames("gs_name") %>%
    as.matrix()
z_fits <- lmFit(z_matrix, design = rep(1, ncol(z_matrix))
                ) %>%
  eBayes()
top_table <- topTable(z_fits, number = Inf) %>%
  as_tibble(rownames = "gs_name")
sigPathway <- top_table %>%
    dplyr::filter(adj.P.Val < 0.05)
```

121 out of the 1094 tested Reactome pathways have an FDR < 0.05 in the moderated t-test, hence were considered to be significantly perturbed at the group level. The Table 1 presents the top 10 significantly inhibited and activated pathways, along with their predicted direction of change.

```
table1 <- sigPathway %>%
    mutate(
        Direction = ifelse(logFC < 0, "Inhibited", "Activated"),
        gs_name = str_remove_all(gs_name, "reactome.")
    ) %>%
    split(f = .$Direction) %>%
    lapply(function(x)x[1:10,]) %>%
    bind_rows() %>%
    dplyr::select(
    Pathway = gs_name, Change = logFC,  P.Value, FDR = adj.P.Val, Direction
)
```

[Table 1 about here.]

For enrichment analysis in the original study[20], unsupervised clustering was performed on all cells labelled as cancer cells. Clusters were then annotated manually by performing pathway enrichment testing on cluster marker genes. Two clusters, associated with proliferative DNA repair signatures and stress-related markers, contained significantly higher numbers of post-chemotherapy cells than pre-treatment ones[20]. The representative pathways enriched in the stress-associated cluster were *IL6-mediated signaling events*, *TNF signaling pathway*, and *cellular responses to stress*, and the other post-chemotherapy cell dominated cluster in the original study was enriched for pathways associated with cell proliferation and DNA repair, such as the Cell cycle, DNA repair, Homology directed repair (HDR) through homologous recombination, and the Fanconi anaemia pathway. *sSNAPPY* not only detected many significant perturbed pathways that are highly concordant with the pathways reported to be enriched in the original study but also predicted their directions of changes. For example, the DNT repair pathway *SUMOylation of DNA damage response and repair proteins* pathway was predicted to be significantly inhibited by the chemotherapy. In comparison, we also performed pathway analysis on this example dataset using two non-topological-based approaches: 1) *GSEA* basing on the ranking statistics derived from the differentially expression (DE) analysis and 2) the rotation gene set testing for linear models (*roast*), which does not rely on DE analysis results. Although the *SUMOylation of DNA damage response and repair proteins* pathway was also defined as significantly impacted by the chemotherapy using the two non-topological-based methods(see extended data), the directionality of pathway perturbation can only be predicted by *sSNAPPY*. The existing topology-based method *SPIA*, on the other hand, only considers pathways

containing differentially expressed genes, hence has a lower sensitivity in detecting pathway perturbation and was only able to capture changes in immune-related processes(see extended data).

The single-sample nature of the *sSNAPPY* output also provides great flexibility: apart from considering all treated samples as biological replicates, users may elect to perform an analysis incorporating other phenotypic traits which may impact a patients responses to chemotherapy, such as disease stages or tumour grades. To perform this step using the moderated t-statistic strategy and extend the above analysis, an appropriate design matrix is the only additional requirement for model-fitting, or alternatively, samples may be subset as may be appropriate.

Visualising Perturbed Pathways as Networks

A valuable feature of sSNAPPY is the provision of several visualisation functions to assist in the interpretation of results. Biological pathways are not independent of each other with many genes playing a role across multiple pathways, and as such, viewing pathway analysis results as a network can be a powerful way to intuitively summarise the results and facilitate interpretation of the underlying biology. The `plot_gs_network()` function allows users to easily convert a list of relevant biological pathways to a network where edges between pathway nodes represent overlapping genes. Defined by the colorBy parameter, pathway nodes can be coloured by either the predicted direction of change or by significance levels (Figure 4). The returned plot is a ggplot2 [23] object, meaning that components of the plotting theme and other parameters can be customized as for any other ggplot2 objects.

In the following example, we'll inspect the 10 most significantly inhibited and 10 most significantly activated pathways, which involved four steps to prepare the data: 1) rename the logFC column to reflect the true meaning of the value and, 2) create a categorical variable with the pathway status, 3) transform p-values for simpler visualisation and 4) obtain a subset of pathways to visualise.

```
sigPathway <- sigPathway %>%
    dplyr::rename(Z = logFC) %>%
    mutate(
        status = ifelse(
            Z > 0, "Activated", "Inhibited"),
        status = ifelse(
            adj.P.Val < 0.05, status, "Unchanged"
        ),
        status = as.factor(status),
        `-log10(p)` = -log10(P.Value)
    )  %>%
    split(f = .$status) %>%
    lapply(function(x)x[1:10,]) %>%
    bind_rows()
```

```
set.seed(123)
# Plot the network structure
p1 <- plot_gs_network(
  normalisedScores = sigPathway, gsTopology = gsTopology, colorBy = "status",
  gsNameSize = 3
) +
  scale_colour_manual(values = c("red", "blue", "grey30")) +
  theme_void() +
  theme(legend.text = element_text(size = 10))
set.seed(123)
p2 <- plot_gs_network(
  normalisedScores = sigPathway,
  gsTopology = gsTopology,
  colorBy  = "-log10(p)",
  gsNameSize = 3,
  gsLegTitle = expression(paste(-log[10], "p"))
) +
  scale_colour_viridis_c() +
  theme_void() +
  theme(
    legend.text = element_text(size = 8),
    legend.title = element_text(size = 10)
  )
p1 / p2 + plot_annotation(tag_levels = "A")
```

[Figure 4 about here.]

Any advantage of visualising pathway analysis results using network structures is that it allows the identification of highly connected pathways (Figure 4). To summarise related pathways and further enable interpretation, we can apply community detection[24] to group related pathways into 'communities'. sSNAPPY's `plot_community()` function is a "one-stop shop" for applying a community detection algorithm of the user's choice to the network structure and annotating identified communities by the most common pathway category, denoting the main biological processes perturbed in that community. The most recent categories for both KEGG and Reactome databases were curated from their respective website (KEGG website & Reactome website) and included as parts of sSNAPPY. Analyses involving other pathway databases may require user-provided pathway categories. When the information about pathway categorisations is available, annotation of pathway communities is automatically completed. In the current dataset, the Louvain method was applied to the network of biological pathways and revealed five primary communities: 1) Adaptive Immune System; 2) Cell Cycle, Mitotic; 3) Chromatin modifying enzymes & Epigenetic regulation of gene expression; 4) Post-translational protein modification and 5) The citric acid (TCA) cycle and respiratory electron transport (Figure 5). The largest community formed was the Adaptive Immune System pathway, indicating a clear immune-signalling aspect to these results.

```
#load(system.file("extdata", "gsAnnotation_df_wiki.rda", package = "sSNAPPY"))
set.seed(456)
plot_community(
    normalisedScores = sigPathway,
    gsTopology = gsTopology,
    gsAnnotation = gsAnnotation_df_wiki,
    colorBy = "status",
    lb_size = 3
) +
    scale_colour_manual(values = c("red", "blue")) +
    scale_fill_viridis_d() +
    scale_x_continuous(expand = expansion(0.25)) +
    scale_y_continuous(expand = expansion(0.25)) +
    guides(fill = FALSE) +
    theme_void() +
    theme(
        legend.text = element_text(size = 8),
        legend.title = element_text(size = 10)
    )
```

```
## Warning: The `<scale>` argument of `guides()` cannot be `FALSE`. Use "none" instead as
## of ggplot2 3.3.4.
## This warning is displayed once every 8 hours.
## Call `lifecycle::last_lifecycle_warnings()` to see where this warning was
## generated.
```

[Figure 5 about here.]

A key advantage of sSNAPPY is that it does not require the prior identification of differentially expressed genes, as this is a common challenge within clinical datasets. However, knowing which genes are implicated in the perturbation of pathways, particularly those which influence multiple pathways, can provide valuable insights for hypothesis generation and the underlying biological mechanisms. Therefore, sSNAPPY provides another visualisation feature called `plot_gs2gene`, which enables the inclusion of select genes from each pathway using network structures. Users can provide a vector of fold-change estimates to visualise genes within pathways, showing their estimated change in expression. As pathways often include hundreds of genes, it is recommended to filter for genes most likely to be playing a significant role. In this example dataset, only genes within the top 500 when ranking by the magnitude of the mean log fold-change were included (Figure 6). An alternative strategy will be to select genes based on test-statistics, however, this decision is up to the individual researcher.

```
meanFC <- rowMeans(weightedFC$weighted_logFC) / weightedFC$weight
top500 <- rank(1/abs(meanFC)) <= 500
dirFC <- ifelse(meanFC > 0, "Up-Regulated", "Down-Regulated")
```

Since Reactome pathway topologies were retrieved using Entrez IDs, users can provide a data.frame mapping Entrez IDs to their chosen identifiers, such as gene names, through the mapEntrezID parameter, in order to

make the visualisations more informative. A data.frame converting Entrez IDs to Ensembl gene names was derived from the Ensembl Release 101[25] and has been made available as part of the package and serves as a helpful template for future mapping operations by users.

```
load(system.file("extdata", "entrez2name.rda", package = "sSNAPPY"))
head(entrez2name)
```

```
## # A tibble: 6 x 2
##   entrezid           mapTo
##   <chr>              <chr>
## 1 ENTREZID:84771     DDX11L1
## 2 ENTREZID:727856    DDX11L1/DDX11L9/DDX11L10
## 3 ENTREZID:100287102 DDX11L1
## 4 ENTREZID:100287596 DDX11L1/DDX11L9
## 5 ENTREZID:102725121 DDX11L1
## 6 ENTREZID:653635    WASH7P
```

```
set.seed(123)
plot_gs2gene(
    normalisedScores = sigPathway,
    gsTopology = gsTopology,
    colorGsBy = "status",
    mapEntrezID = entrez2name,
    geneFC = meanFC[top500],
    layout = "kk",
    edgeAlpha = 1,
    gsNameSize = 4,
    gsNodeSize = 4,
    geneNameSize = 4
) +
    scale_colour_gradient2(name = "logFC") +
    scale_fill_manual(values = c("red", "blue", "grey50")) +
    theme_void() +
    theme(
        legend.text = element_text(size = 12),
        legend.title = element_text(size = 12)
    )
```

[Figure 6 about here.]

Identifying Key Gene Contributions

To further investigate a specific pathway and elucidate which are the key genes contributing to the final perturbation score, we can generate a heatmap via `plot_gene_contribution()` which shows the gene-level perturbation scores for the top-ranked members of a given pathway. This function takes advantage of the plotting capabilities of the `pheatmap` package[26], and as such, other annotations are also able to be easily included, such as patient response, or which general ranges the pathway-level normalised Z-Scores are in. Inclusion of the Z-Scores enabled the assessment of the level of perturbation predicted in each sample and key genes involved (Figure 7).

```
annotation_df <- normalisedScores %>%
    dplyr::filter(gs_name == "reactome.SUMOylation of DNA replication proteins") %>%
    left_join(dplyr::select(sample_meta, sample,  CRS), by = "sample") %>%
    mutate(
      `Z Range` = cut(
        robustZ, breaks = seq(-2, 2, length.out = 6), include.lowest = TRUE
      ),
      sample = str_remove_all(sample, "_post-NACT")
    ) %>%
    dplyr::select(sample, `Z Range`, CRS)
z_levels <- levels(annotation_df$`Z Range`)
annotation_col <- list(
  CRS = c("3" = "#4B0055", "2" = "#009B95", "1" = "#FDE333"),
```

```
  `Z Range` = setNames(
    colorRampPalette(c("navyblue", "white", "darkred"))(length(z_levels)),
    z_levels
  )
)
plot_gene_contribution(
    genePertMatr = genePertScore$`reactome.SUMOylation of DNA replication proteins` %>%
        set_colnames(str_remove_all(colnames(.), "_post-NACT")) %>%
        .[rownames(.) %in% rownames(weightedFC$weighted_logFC),],
    color = rev(colorspace::divergex_hcl(100, palette = "RdBu")),
    breaks = seq(-0.002, 0.002, length.out = 100),
    annotation_df =  annotation_df,
    topGene = 15, filterBy = "mean",
    mapEntrezID = entrez2name,
    annotation_colors = annotation_col,
    cutree_rows = 2,
    cutree_cols = 2,
    main = "SUMOylation of DNA replication proteins [REACTOME]"
)
```

[Figure 7 about here.]

From this heatmap we can identify candidate genes which are likely to be making the biggest contribution to the inhibition of the SUMOylation of DNA replication proteins pathway upon chemotherapy, such as *CDCA8*, *TOP2A*, *UBE2I*, *BIRC5* (Figure 7). The four genes are all associated with tumour progression and invasiveness and have been studied in the context of ovarian cancer. Both ubiquitin conjugating enzyme E2I (*UBE2I*) and cell division cycle associated 8 (*CDCA8*) genes have been identified as oncogenes in numerous cancer types, including ovarian cancer[27, 28]. Notably, in ovarian cancer, elevated *UBE2I* expression has been associated with poorer clinical outcomes[29]. Similarly, expression of *BIRC5* that encodes human survivin protein is also a predictor of inferior ovarian cancer patient outcome[30]. Lastly, Topoisomerase IIα (*TOP2A*), which encodes DNA topoisomerase, has been identified as a gene that promotes the tumorigenesis of HGSOC tumours[31]. Aligning with the report by Chekerov et al.[32] that expression of *TOP2A* in ovarian tumour cells decreases as a response to chemotherapy[32], the median single-sample logFC of *TOP2A* was negative among the HGSOC post-chemotherapy samples included in this study (Figure 8). The other three selected potential driver genes (*CDCA8*, *UBE2I*, and *BIRC5*) also had negative median single-sample logFC in post-chemotherapy samples (Figure 8). Considering the implication of these four genes in ovarian cancer, decreases in their expression after chemotherapy treatment potentially indicate a favorable response to therapy. By annotating the heatmap of gene-wise perturbation scores with patient chemotherapy response score (CRS), we noticed that the strongest inhibition of the SUMOylation of DNA replication proteins pathway was in the patient with the highest CRS score of 3 (i.e sample EOC443). CRS is an indicator of the relative length of progression-free survival after chemotherapy, where a score of 3 represents the longest survival. Hence inhibition of the SUMOylation of DNA replication proteins pathway might mediate favorable response to chemotherapy in ovarian cancer patients. We acknowledge that our analysis was limited to a small number of patients, which restricts the generalizability of the results. However, despite this limitation, these findings underscore the strength of sSNAPPY as a valuable tool for hypothesis generation. Not only can sSNAPPY predict directional pathway perturbations, but it also enables the identification of key driver genes underlying these perturbations.

```
gene2plot <- entrez2name %>%
    dplyr::filter(
        mapTo %in% c("CDCA8", "TOP2A", "UBE2I", "BIRC5")
    )
(weightedFC$weighted_logFC / weightedFC$weight) %>%
    as.data.frame() %>%
    rownames_to_column("entrezid") %>%
    pivot_longer(
        cols = -"entrezid",
        names_to = "sample",
        values_to = "ssFC"
    ) %>%
    left_join(entrez2name) %>%
    dplyr::filter(mapTo %in% c("CDCA8", "TOP2A", "UBE2I", "BIRC5")) %>%
    ggplot(
        aes(mapTo, ssFC, fill = mapTo)
    ) +
```

```
    geom_boxplot() +
    labs(
        x = "",
        fill = "Gene"
    ) +
    geom_hline(yintercept = 0, color = "red", linetype = "dashed") +
    theme_bw() +
    theme(
        text = element_text(size = 10)
    )
```

[Figure 8 about here.]

Discussion

In conclusion, we have presented and provided a demonstration for the R/Bioconductor package *sSNAPPY* which offers a novel single-sample pathway perturbation testing approach, tailored for heterogeneous tissue samples in matched-pair design. In contrast to many common enrichment methods, sSNAPPY uses pathway topology information to compute perturbation scores which indicate the likely impact on the activity of a pathway, by predicting direction of change and enabling deeper characterisation of biological responses. By applying sSNAPPY to a public scRNA-seq data collected before and after HGSOC patients were subjected to chemotherapy, we demonstrated its ability to detect significant pathway perturbations of various interesting biological processes consistent with, and far beyond what was shown in the original study. Whilst initially conceived for bulk-RNA studies, this demonstration has also provided clear applicability to scRNA datasets. *sSNAPPY* addresses the limitations of alternative strategies which fail to account for gene-gene interactions encoded by pathway topologies and are unable to predict the directionality of pathway activities. In addition, the single-sample nature of the method can be utilised to address the increasing demand for personalised medicine. Through identifying shared and divergent responses between individuals, *sSNAPPY* can provide valuable insights into the heterogeneous responses across clinical samples. Overall, we believe sSNAPPY represents a valuable addition to the existing body of pathway analysis methods.

Data availability

The dataset analysed in this manuscript are stored in the data directory of this GitHub repository.

Software availability

- Software available from: `https://bioconductor.org/packages/release/bioc/html/sSNAPPY.html`
- Source code available from: `https://github.com/Wenjun-Liu/sSNAPPY`
- Archived source code at time of publication: `https://doi.org/10.5281/zenodo.8185451`
- License: GNU General Public License v3.0 (GPL-3

Author Contributions

WL's contributions include Conceptualization, Data Curation, Formal Analysis, Investigation, Methodology, Project Administration, Software, Validation, Visualisation, Writing - Original Draft Preparation, and Writing - Review & Editing. VM was involved with Conceptualization, Methodology and Writing - Review & Editing. WDT contributed to Writing - Review & Editing. SMP's contributions include Conceptualization, Methodology, Project Administration, Software, Supervision, Writing - Original Draft Preparation, and Writing - Review & Editing.

Competing interests

No competing interests were disclosed

Grant information

W.D. Tilley's research is supported by the National Health and Medical Research Council of Australia (ID 1186647) and the National Breast Cancer Foundation Australia (ID IIRS-23-069)

References

[1] Farhad Maleki, Katie Ovens, Daniel J. Hogan, and Anthony J. Kusalik. Gene set analysis: Challenges, opportunities, and future research. *Frontiers in Genetics*, 11, June 2020. doi: 10.3389/fgene.2020.00654.

[2] Sarah Mubeen, Alpha Tom Kodamullil, Martin Hofmann-Apitius, and Daniel Domingo-Fernández. On the influence of several factors on pathway enrichment analysis. *Briefings in Bioinformatics*, 23(3):bbac143, April 2022. doi: 10.1093/bib/bbac143.

[3] Aravind Subramanian, Pablo Tamayo, Vamsi K Mootha, Sayan Mukherjee, Benjamin L Ebert, Michael A Gillette, Amanda Paulovich, Scott L Pomeroy, Todd R Golub, Eric S Lander, and Jill P Mesirov. Gene set enrichment analysis: a knowledge-based approach for interpreting genome-wide expression profiles. *Proc. Natl. Acad. Sci. U. S. A.*, 102 (43):15545–15550, October 2005. doi: 10.1073/pnas.0506580102.

[4] Hiroyuki Ogata, Susumu Goto, Kazushige Sato, Wataru Fujibuchi, Hidemasa Bono, and Minoru Kanehisa. KEGG: Kyoto Encyclopedia of Genes and Genomes. *Nucleic Acids Research*, 27(1):29–34, January 1999. doi: 10.1093/nar/27.1.29.

[5] Marvin Martens, Ammar Ammar, Anders Riutta, Andra Waagmeester, Denise N Slenter, Kristina Hanspers, Ryan A. Miller, Daniela Digles, Elisson N Lopes, Friederike Ehrhart, Lauren J Dupuis, Laurent A Winckers, Susan L Coort, Egon L Willighagen, Chris T Evelo, Alexander R Pico, and Martina Kutmon. WikiPathways: connecting communities. *Nucleic Acids Research*, 49(D1):D613–D621, January 2021. doi: 10.1093/nar/gkaa1024.

[6] Adi Laurentiu Tarca, Sorin Draghici, Purvesh Khatri, Sonia S. Hassan, Pooja Mittal, Jung-sun Kim, Chong Jai Kim, Juan Pedro Kusanovic, and Roberto Romero. A novel signaling pathway impact analysis. *Bioinformatics*, 25(1):75–82, January 2009. doi: 10.1093/bioinformatics/btn577.

[7] Laurent Jacob, Pierre Neuvial, and Sandrine Dudoit. More power via graph-structured tests for differential expression of gene networks. *The Annals of Applied Statistics*, 6(2):561 – 600, June 2012. doi: 10.1214/11-AOAS528.

[8] Jing Ma, Ali Shojaie, and George Michailidis. Network-based pathway enrichment analysis with incomplete network information. *Bioinformatics*, 32(20):3165–3174, June 2016. doi: 10.1093/bioinformatics/btw410.

[9] Maysson Al-Haj Ibrahim, Sabah Jassim, Michael Anthony Cawthorne, and Kenneth Langlands. A topology-based score for pathway enrichment. *Journal of Computational Biology*, 19(5):563–573, May 2012. doi: 10.1089/cmb.2011.0182.

[10] Di Wu and Gordon K. Smyth. Camera: a competitive gene set test accounting for inter-gene correlation. *Nucleic Acids Research*, 40(17):e133–e133, May 2012. doi: 10.1093/nar/gks461.

[11] Tuan-Minh Nguyen, Adib Shafi, Tin Nguyen, and Sorin Draghici. Identifying significantly impacted pathways: a comprehensive review and assessment. *Genome Biol.*, 20(1):203, October 2019. doi: 10.1186/s13059-019-1790-4.

[12] Jing Ma, Ali Shojaie, and George Michailidis. A comparative study of topology-based pathway enrichment analysis methods. *BMC Bioinformatics*, 20(1):546, December 2019. doi: 10.1186/s12859-019-3146-1.

[13] Sonja Hänzelmann, Robert Castelo, and Justin Guinney. GSVA: gene set variation analysis for microarray and RNA-Seq data. *BMC Bioinformatics*, 14(1):7, December 2013. doi: 10.1186/1471-2105-14-7.

[14] Charity W Law, Yunshun Chen, Wei Shi, and Gordon K Smyth. voom: precision weights unlock linear model analysis tools for RNA-seq read counts. *Genome Biol*, 15(2):R29, 2014. doi: 10.1186/gb-2014-15-2-r29.

[15] Theo A. Knijnenburg, Lodewyk F. A. Wessels, Marcel J. T. Reinders, and Ilya Shmulevich. Fewer permutations, more accurate P-values. *Bioinformatics*, 25(12):i161–i168, May 2009. doi: 10.1093/bioinformatics/btp211.

[16] Gene set enrichment analysis (gsea) user guide. `https://www.gsea-msigdb.org/gsea/doc/GSEAUserGuideFrame.html?Run_GSEA_Page`. Accessed: [029/09/2023].

[17] Gordon K Smyth. Linear models and empirical bayes methods for assessing differential expression in microarray experiments. *Statistical Applications in Genetics and Molecular Biology*, 3(1), February 2004. doi: 10.2202/1544-6115.1027.

[18] Matthew E Ritchie, Belinda Phipson, Di Wu, Yifang Hu, Charity W Law, Wei Shi, and Gordon K Smyth. limma powers differential expression analyses for RNA-sequencing and microarray studies. *Nucleic Acids Research*, 43(7):e47, April 2015. doi: 10.1093/nar/gkv007.

[19] Gabriele Sales, Enrica Calura, Duccio Cavalieri, and Chiara Romualdi. graphite - a Bioconductor package to convert pathway topology to gene network. *BMC Bioinformatics*, 13(1):20, December 2012. doi: 10.1186/1471-2105-13-20.

[20] Kaiyang Zhang, Erdogan Pekcan Erkan, Sanaz Jamalzadeh, Jun Dai, Noora Andersson, Katja Kaipio, Tarja Lamminen, Naziha Mansuri, Kaisa Huhtinen, Olli Carpén, Sakari Hietanen, Jaana Oikkonen, Johanna Hynninen, Anni Virtanen, Antti Häkkinen, Sampsa Hautaniemi, and Anna Vähärautio. Longitudinal single-cell RNA-seq analysis reveals stress-promoted chemoresistance in metastatic ovarian cancer. *Sci. Adv.*, 8(8):eabm1831, February 2022. doi: 10.1126/sciadv.abm1831.

[21] K. D. Hansen, R. A. Irizarry, and Z. Wu. Removing technical variability in RNA-seq data using conditional quantile normalization. *Biostatistics*, 13(2):204–216, April 2012. doi: 10.1093/biostatistics/kxr054.

[22] Marc Gillespie, Bijay Jassal, Ralf Stephan, Marija Milacic, Karen Rothfels, Andrea Senff-Ribeiro, Johannes Griss, Cristoffer Sevilla, Lisa Matthews, Chuqiao Gong, Chuan Deng, Thawfeek Varusai, Eliot Ragueneau, Yusra Haider, Bruce May, Veronica Shamovsky, Joel Weiser, Timothy Brunson, Nasim Sanati, Liam Beckman, Xiang Shao, Antonio Fabregat, Konstantinos Sidiropoulos, Julieth Murillo, Guilherme Viteri, Justin Cook, Solomon Shorser, Gary Bader, Emek Demir, Chris Sander, Robin Haw, Guanming Wu, Lincoln Stein, Henning Hermjakob, and Peter D'Eustachio. The reactome pathway knowledgebase 2022. *Nucleic Acids Research*, 50(D1):D687–D692, November 2021. doi: 10.1093/nar/gkab1028.

[23] Hadley Wickham. *ggplot2: Elegant Graphics for Data Analysis*. Springer New York, New York, NY, October 2009. ISBN 978-0-387-98140-6. doi: 10.1007/978-0-387-98141-3.

[24] M. E. J. Newman and M. Girvan. Finding and evaluating community structure in networks. *Phys. Rev. E*, 69(2): 026113, February 2004. doi: 10.1103/PhysRevE.69.026113.

[25] Fiona Cunningham, James E Allen, Jamie Allen, Jorge Alvarez-Jarreta, M Ridwan Amode, Irina M Armean, Olanrewaju Austine-Orimoloye, Andrey G Azov, If Barnes, Ruth Bennett, Andrew Berry, Jyothish Bhai, Alexandra Bignell, Konstantinos Billis, Sanjay Boddu, Lucy Brooks, Mehrnaz Charkhchi, Carla Cummins, Luca Da Rin Fioretto, Claire Davidson, Kamalkumar Dodiya, Sarah Donaldson, Bilal El Houdaigui, Tamara El Naboulsi, Reham Fatima, Carlos Garcia Giron, Thiago Genez, Jose Gonzalez Martinez, Cristina Guijarro-Clarke, Arthur Gymer, Matthew Hardy, Zoe Hollis, Thibaut Hourlier, Toby Hunt, Thomas Juettemann, Vinay Kaikala, Mike Kay, Ilias Lavidas, Tuan Le, Diana Lemos, José Carlos Marugán, Shamika Mohanan, Aleena Mushtaq, Marc Naven, Denye N Ogeh, Anne Parker, Andrew Parton, Malcolm Perry, Ivana Piližota, Irina Prosovetskaia, Manoj Pandian Sakthivel, Ahamed Imran Abdul Salam, Bianca M Schmitt, Helen Schuilenburg, Dan Sheppard, José G Pérez-Silva, William Stark, Emily Steed, Kyösti Sutinen, Ranjit Sukumaran, Dulika Sumathipala, Marie-Marthe Suner, Michal Szpak, Anja Thormann, Francesca Floriana Tricomi, David Urbina-Gómez, Andres Veidenberg, Thomas A Walsh, Brandon Walts, Natalie Willhoft, Andrea Winterbottom, Elizabeth Wass, Marc Chakiachvili, Bethany Flint, Adam Frankish, Stefano Giorgetti, Leanne Haggerty, Sarah E Hunt, Garth R IIsley, Jane E Loveland, Fergal J Martin, Benjamin Moore, Jonathan M Mudge, Matthieu Muffato, Emily Perry, Magali Ruffier, John Tate, David Thybert, Stephen J Trevanion, Sarah Dyer, Peter W Harrison, Kevin L Howe, Andrew D Yates, Daniel R Zerbino, and Paul Flicek. Ensembl 2022. *Nucleic Acids Research*, 50(D1):D988–D995, November 2021. doi: 10.1093/nar/gkab1049.

[26] Raivo Kolde. *pheatmap: Pretty Heatmaps*, 2019. URL `https://CRAN.R-project.org/package=pheatmap`. R package version 1.0.12.

[27] Mei Dong, Xiaoyan Pang, Yang Xu, Fang Wen, and Yi Zhang. Ubiquitin-Conjugating Enzyme 9 Promotes Epithelial Ovarian Cancer Cell Proliferation in Vitro. *International Journal of Molecular Sciences*, 14(6):11061–11071, May 2013. doi: 10.3390/ijms140611061.

[28] Gonghua Qi, Chenyi Zhang, Hanlin Ma, Yingwei Li, Jiali Peng, Jingying Chen, and Beihua Kong. Cdca8, targeted by mybl2, promotes malignant progression and olaparib insensitivity in ovarian cancer. *American journal of cancer research*, 11(2):389, 2021.

[29] Ruoyao Zou, Haoya Xu, Feifei Li, Shengke Wang, and Liancheng Zhu. Increased expression of UBE2T predicting poor survival of ovarian cancer: based on bioinformatics analysis of UBE2s, clinical samples and the GEO database. preprint, In Review, September 2020.

[30] Beata Gąsowska-Bajger, Agnieszka Gąsowska-Bodnar, Paweł Knapp, and Lubomir Bodnar. Prognostic Significance of Survivin Expression in Patients with Ovarian Carcinoma: A Meta-Analysis. *Journal of Clinical Medicine*, 10(4):879, February 2021. doi: 10.3390/jcm10040879.

[31] Yan Gao, Hongyu Zhao, Meng Ren, Qi Chen, Jie Li, Zhefeng Li, Chenghong Yin, and Wentao Yue. TOP2A Promotes Tumorigenesis of High-grade Serous Ovarian Cancer by Regulating the TGF-β/Smad Pathway. *Journal of Cancer*, 11 (14):4181–4192, 2020. doi: 10.7150/jca.42736.

[32] Radoslav Chekerov, Irina Klaman, Menelaos Zafrakas, Dominique Könsgen, Alexander Mustea, Beate Petschke, Werner Lichtenegger, Jalid Sehouli, and Edgar Dahl. Altered Expression Pattern of Topoisomerase IIα, in Ovarian Tumor Epithelial and Stromal Cells after Platinum-Based Chemotherapy. *Neoplasia*, 8(1):38–45, January 2006. doi: 10.1593/neo.05580.

List of Figures

1 Schematic illustration of the differences betweeen conventional pathway analysis methods and sSNAPPY. Instead of being limited to treatment-level analyses, *sSNAPPY* allows the detection of pathway perturbation in individual samples by using sample-specific estimates of fold-change instead of experiment-wide estimates. (Created with BioRender.com). 16

2 Gene-wise standard deviations are plotted against the mean logCPM values with mean-variance trend modelled by a loess fit. Whilst standard deviations are shown here for the purposes of visualisation, gene-level weights are calculated using variances at this stage of the sSNAPPY algorithm. 17

3 (A) Q-Q plot and (B) distributions of permuted perturbation scores of six randomly selected pathways. All sampled empirical distributions are approximately normally distributed with a mean of zero. 18

4 Significantly perturbed Reactome pathways identified among post-chemotherapy samples using sSNAPPY, colored by (A) predicted direction of changes and (B) -log10(p-values). Only the 10 most significantly inhibited and 10 most significantly activated pathways are shown. 19

5 Significantly perturbed Reactome pathways identified among post-chemotherapy samples using sSNAPPY, colored by community structures detected through the louvain algorithm. The main biological processes associated with the top 20 pathways that wer most perturbed by the chemotherapy were shown. 20

6 Significantly perturbed Reactome pathways identified among post-chemotherapy samples using sSNAPPY, showing any genes in the top 500 ranked by magnitude of change in expression, and which pathways they are likely contributing to. Only the 10 most significantly inhibited and 10 most significantly activated pathways are shown. 21

7 Gene-level perturbation scores for the top 15 genes in the "SUMOylation of DNA replication proteins" pathway ranked by average contribution to the perturbation score. Samples were annotated by patient chemotherapy response score (CRS), along with the range for sample-level Z-scores as a guide to sample-specific pathway perturbation. The genes CDCA8, TOP2A, UBE2I, BIRC5 were identified as possible key drivers of the inhibition of of this pathway. 22

8 Single-sample logFC (ssFC) of potential key genes driving the inhibition of "SUMOylation of DNA replication proteins" pathway as a response to chemotherapy in HGSOC tumours. 23

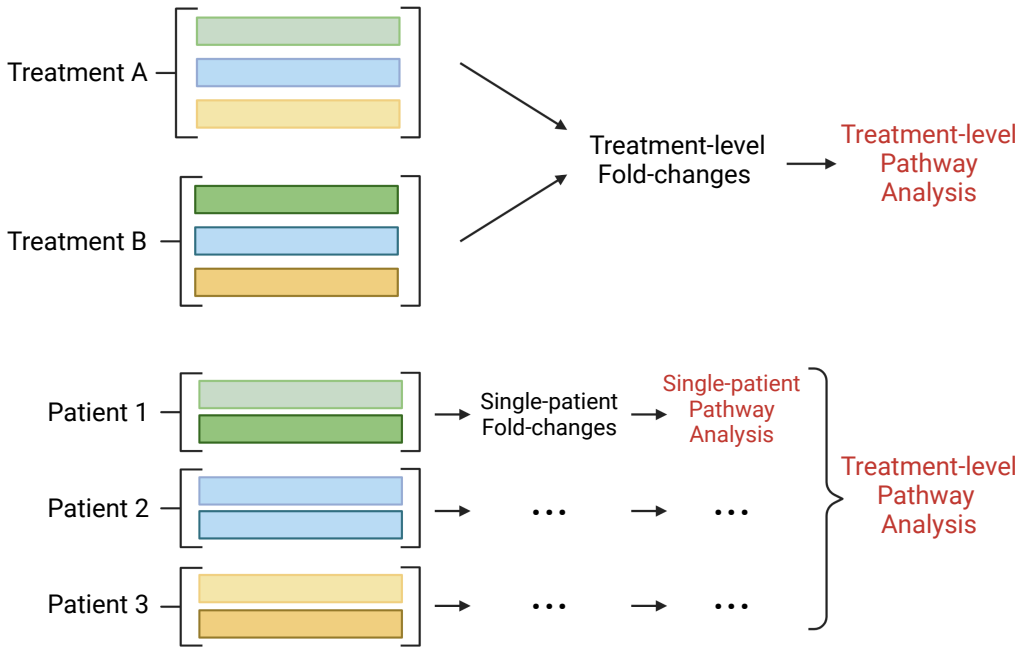

Figure 1. Schematic illustration of the differences betweeen conventional pathway analysis methods and sSNAPPY. Instead of being limited to treatment-level analyses, *sSNAPPY* allows the detection of pathway perturbation in individual samples by using sample-specific estimates of fold-change instead of experiment-wide estimates. (Created with BioRender.com).

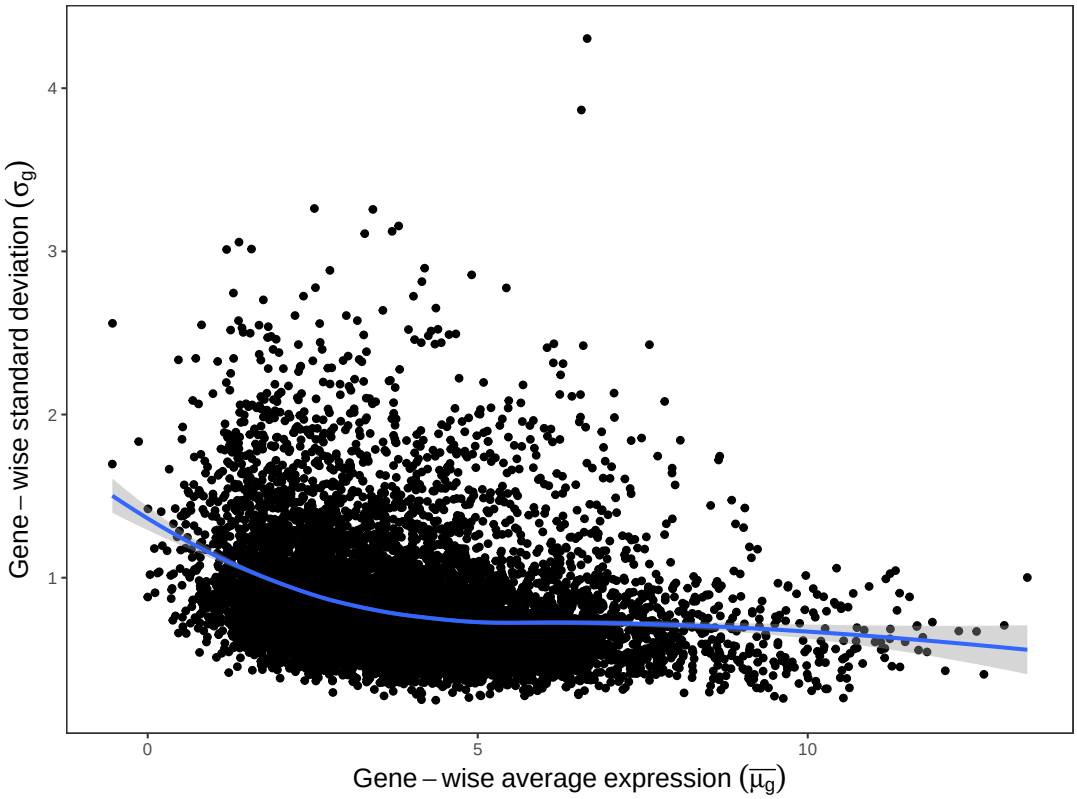

Figure 2. Gene-wise standard deviations are plotted against the mean logCPM values with mean-variance trend modelled by a loess fit. Whilst standard deviations are shown here for the purposes of visualisation, gene-level weights are calculated using variances at this stage of the sSNAPPY algorithm.

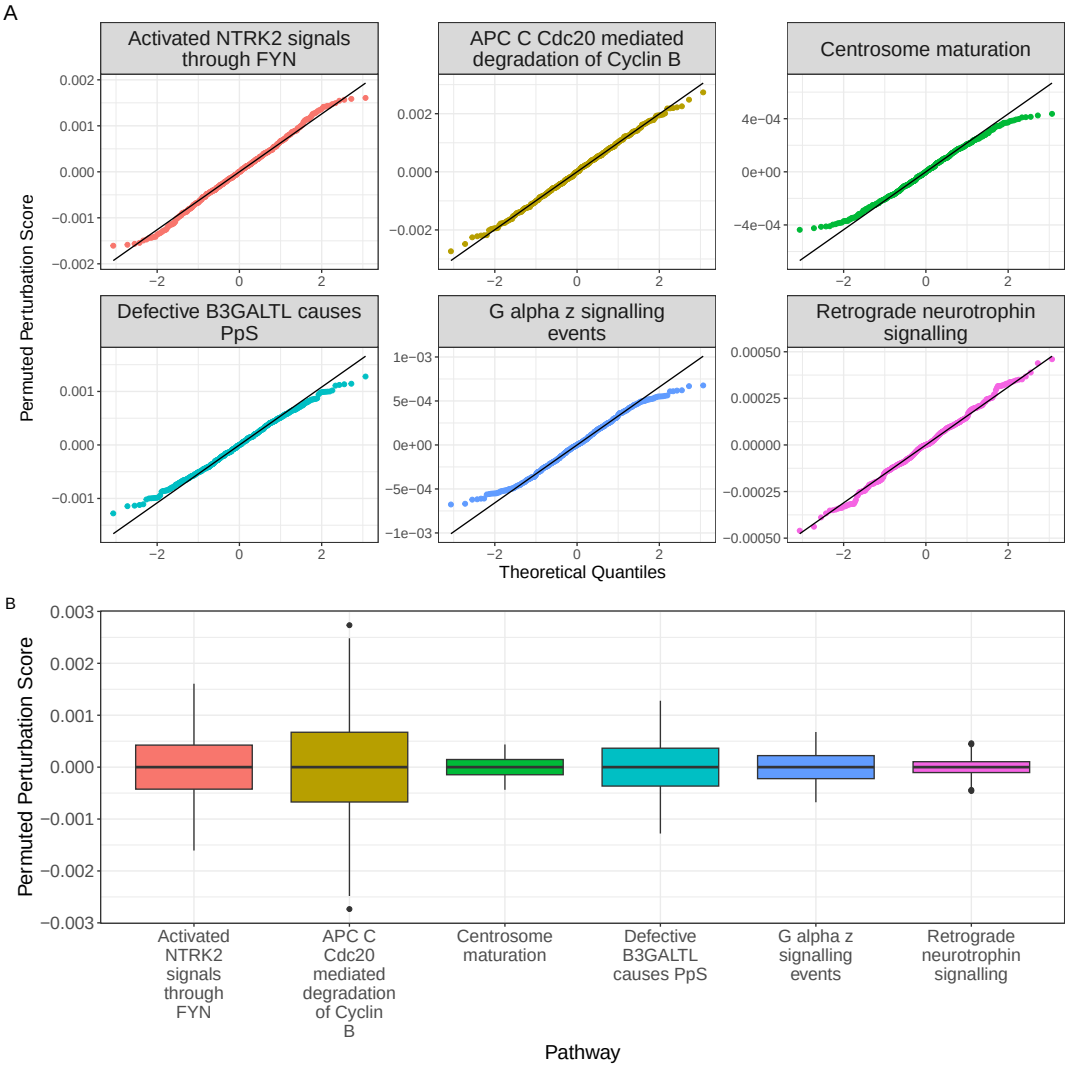

Figure 3. (A) Q-Q plot and (B) distributions of permuted perturbation scores of six randomly selected pathways. All sampled empirical distributions are approximately normally distributed with a mean of zero.

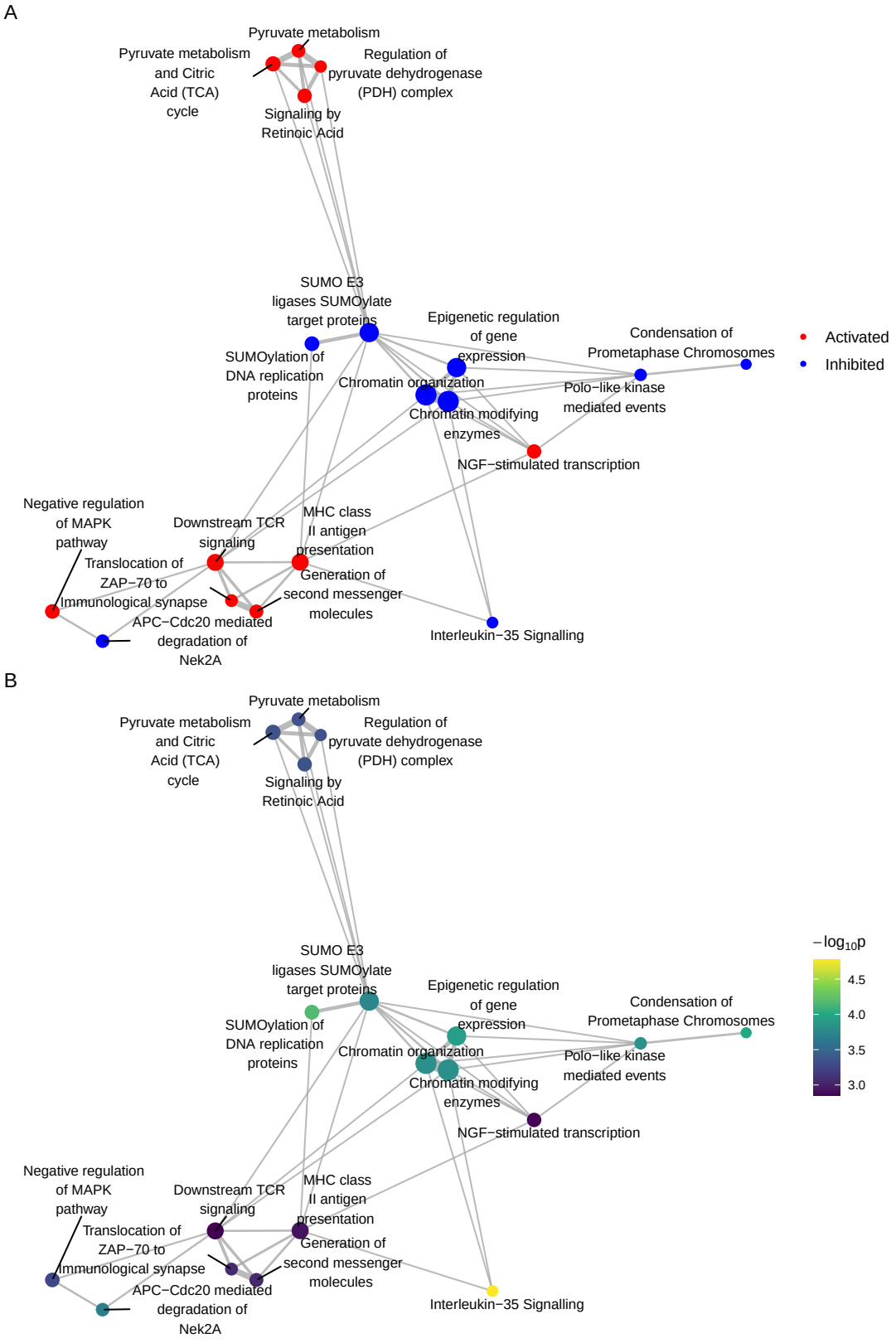

Figure 4. Significantly perturbed Reactome pathways identified among post-chemotherapy samples using sSNAPPY, colored by (A) predicted direction of changes and (B) -log10(p-values). Only the 10 most significantly inhibited and 10 most significantly activated pathways are shown.

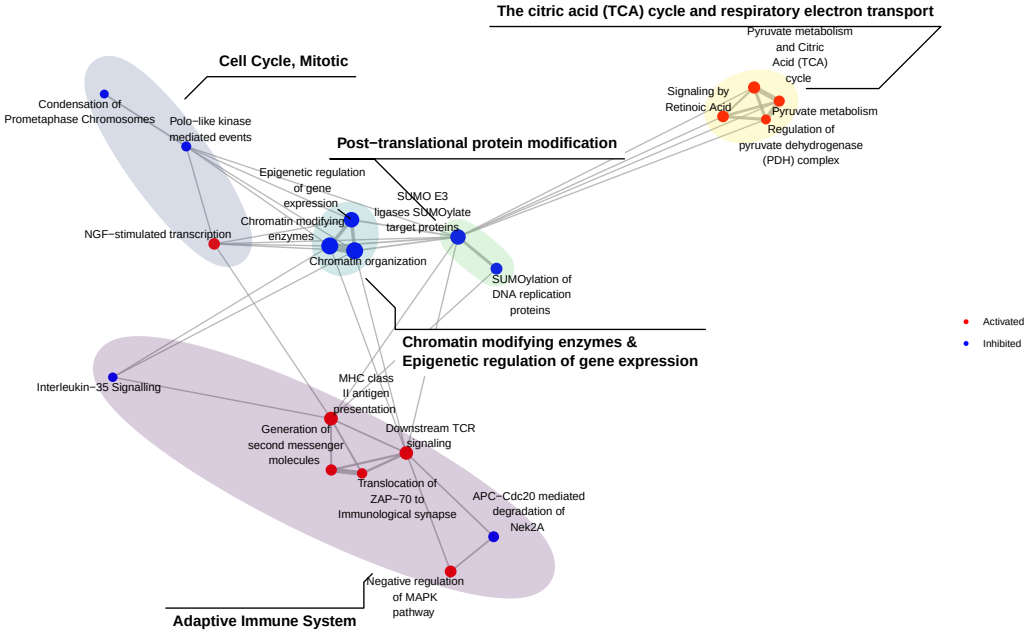

Figure 5. Significantly perturbed Reactome pathways identified among post-chemotherapy samples using sSNAPPY, colored by community structures detected through the louvain algorithm. The main biological processes associated with the top 20 pathways that wer most perturbed by the chemotherapy were shown.

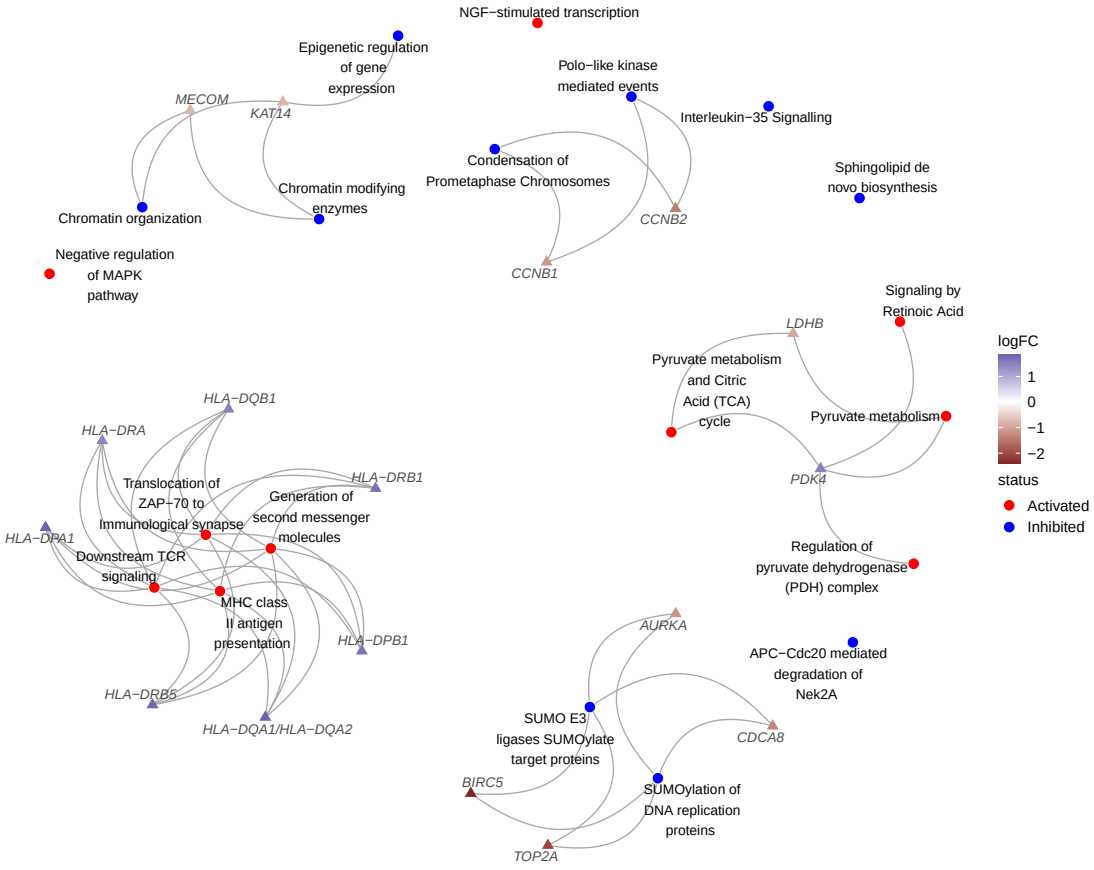

Figure 6. Significantly perturbed Reactome pathways identified among post-chemotherapy samples using sSNAPPY, showing any genes in the top 500 ranked by magnitude of change in expression, and which pathways they are likely contributing to. Only the 10 most significantly inhibited and 10 most significantly activated pathways are shown.

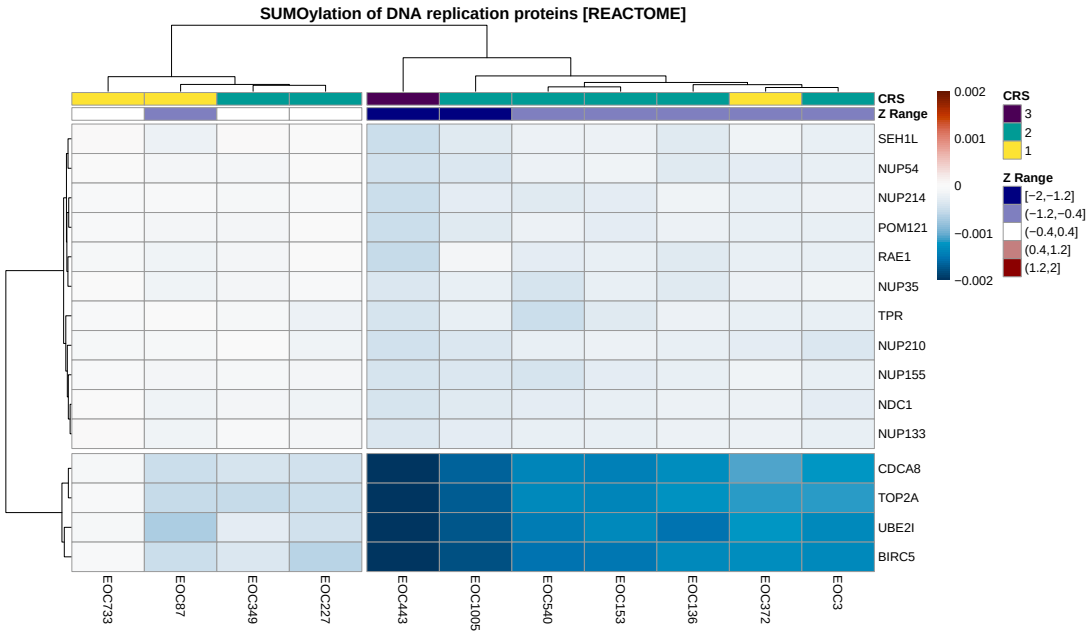

Figure 7. Gene-level perturbation scores for the top 15 genes in the "SUMOylation of DNA replication proteins" pathway ranked by average contribution to the perturbation score. Samples were annotated by patient chemotherapy response score (CRS), along with the range for sample-level Z-scores as a guide to sample-specific pathway perturbation. The genes CDCA8, TOP2A, UBE2I, BIRC5 were identified as possible key drivers of the inhibition of of this pathway.

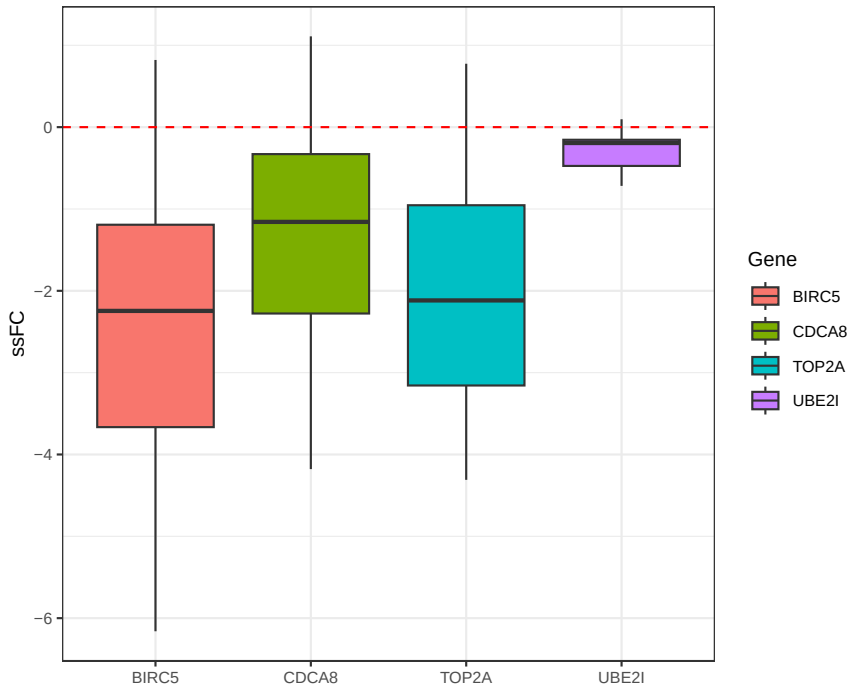

Figure 8. Single-sample logFC (ssFC) of potential key genes driving the inhibition of "SUMOylation of DNA replication proteins" pathway as a response to chemotherapy in HGSOC tumours.

List of Tables

1 Significantly impacted Reactome pathways identified among post-chemotherapy samples using sSNAPPY. Only the 10 most significantly inhibited and 10 most significantly activated pathways are shown. 25

Table 1. Significantly impacted Reactome pathways identified among post-chemotherapy samples using sSNAPPY. Only the 10 most significantly inhibited and 10 most significantly activated pathways are shown.

Pathway	Change	PValue	FDR	Direction
Signaling by Retinoic Acid	0.601	4.45e-04	0.0152	Activated
Regulation of pyruvate dehydrogenase (PDH) complex	0.598	4.59e-04	0.0152	Activated
Pyruvate metabolism	0.598	4.59e-04	0.0152	Activated
Pyruvate metabolism and Citric Acid (TCA) cycle	0.598	4.59e-04	0.0152	Activated
Negative regulation of MAPK pathway	0.627	5.40e-04	0.0174	Activated
Translocation of ZAP-70 to Immunological synapse	0.624	8.75e-04	0.0218	Activated
Generation of second messenger molecules	0.624	8.75e-04	0.0218	Activated
MHC class II antigen presentation	0.628	1.17e-03	0.0241	Activated
NGF-stimulated transcription	0.582	1.36e-03	0.0247	Activated
Downstream TCR signaling	0.633	1.39e-03	0.0247	Activated
Interleukin-35 Signalling	-0.902	1.69e-05	0.0151	Inhibited
Sphingolipid de novo biosynthesis	-0.896	2.75e-05	0.0151	Inhibited
SUMOylation of DNA replication proteins	-0.819	6.86e-05	0.0152	Inhibited
Condensation of Prometaphase Chromosomes	-0.904	9.91e-05	0.0152	Inhibited
Epigenetic regulation of gene expression	-0.790	1.21e-04	0.0152	Inhibited
Polo-like kinase mediated events	-0.880	1.43e-04	0.0152	Inhibited
Chromatin modifying enzymes	-0.791	1.52e-04	0.0152	Inhibited
Chromatin organization	-0.791	1.52e-04	0.0152	Inhibited
SUMO E3 ligases SUMOylate target proteins	-0.767	1.83e-04	0.0152	Inhibited
APC-Cdc20 mediated degradation of Nek2A	-0.838	2.13e-04	0.0152	Inhibited