The Journal of Open Source Software

LTRpred: *de novo* annotation of intact retrotransposons

Hajk-Georg Drost[1, 2]

1 The Sainsbury Laboratory, University of Cambridge, Bateman Street, Cambridge CB2 1LR, UK **2** Department of Molecular Biology, Max Planck Institute for Developmental Biology, 72076 Tuebingen, Germany

DOI: 10.21105/joss.02170

Software

- Review
- Repository
- Archive

Editor: Lorena Pantano

Reviewers:

- @dcassol
- @mdozmorov

Submitted: 28 February 2020
Published: 18 June 2020

Summary

Transposable elements (TEs) play a crucial role in altering the genomic landscape of all organisms and thereby massively influence the genetic information passed on to succeeding generations (Sundaram & Wysocka, 2020). In the past, TEs were seen as selfish mobile elements populating host genomes to increase their chances for transgenerational transmission over long evolutionary time scales. This notion of selfish elements is slowly changing (Drost & Sanchez, 2019) and a new picture drawing a complex genetic landscape benefitting both, host and TE, emerges whereby novel forms can arise through random shuffling of genetic material. For example, the tomato fruit shape (Benoit et al., 2019), moth adaptive cryptic coloration that occurred during the industrial revolution (Chuong, Elde, & Feschotte, 2017), and inner cell mass development in human embryonic stem cells (Chuong et al., 2017) were all shown to be driven by TE activity. Thus, the impact of these elements on altering morphological traits is imminent and requires new attention in the light of evolvability. However, TEs tend to degenerate their sequence leaving their fragmented copies considered as *junk* DNA in host genomes, which hamper assembly and annotation of new genomes.

Nowadays, the *de novo* detection of transposable elements is performed by annotation tools specifically designed to capture any type of repeated sequence, TE family, or remnant DNA loci that can be associated with known transposable elements within a genome assembly. The main goal of such efforts is to retrieve a maximum number of loci that can be associated with known TEs. If successful, such annotation can then be used to mask host genomes from TE remnants to simplify genomics studies focusing on host genes. Therefore, there is no automatically performed distinction between complete and potentially active TE and their mutated copies.

Here, we introduce the `LTRpred` pipeline which allows to *de novo* annotate functional and thus potentially mobile retrotransposons in any given genome assembly. Different from other annotation tools, `LTRpred` focuses on retrieving structurally intact elements within sequences of genomes rather than characterizing all traces of historic TE activity.

Such functional annotation is most useful when trying to spot retrotransposons responsible for recent reshuffling of genetic material in the tree of life. Detecting and further characterization of those active retrotransposons yields the potential to harness them as mutagenesis agents by inducing transposition bursts in a controlled fashion to stimulate genomic reshaping processes towards novel traits.

`LTRpred` emerged as valuable tool for diverse TE mobilization studies

`LTRpred` was successfully used in previous studies to annotate functional retrotransposons for various applications. In detail, `LTRpred` was used to annotate the retrotransposon family

RIDER within the plant kingdom, shown to be involved in tomato fruit shape elongation. Together with experimental evidence, our analyses revealed that RIDER elements can be activated via drought stress and may help plants rich in RIDER activity to better adapt to drought stress conditions (Benoit et al., 2019). In a complementary study, LTRpred was used to generate a candidate list of potentially mobile retrotransposon families in rice and tomato, which were then confirmed to produce extrachromosomal DNA using the ALE-Seq methodology (Cho et al., 2019). Finally, LTRpred supported efforts to annotate and date functional retrotransposons in tomato and *Arabidopsis* which led to the finding that chromodomain DNA methyltransferases (CMTs) silence young and intact retrotransposons in distal chromatin whereas older non-functional retrotransposons are affected by small RNA-directed DNA methylation (Wang & Baulcombe, 2020).

Together, potentially functional retrotransposons annotated *de novo* with LTRpred were subsequently shown to be active and mobile in diverse molecular studies. This approach may stimulate a new wave of research towards understanding the physiological role of functional retrotransposons and to reveal the mechanistic principles of transposon associated evolvability.

Pipeline dependencies

The LTRpred pipeline depends on the R packages biomartr (Drost & Paszkowski, 2017), dplyr (Wickham et al., 2019), Biostrings, stringr (Wickham et al., 2019), IRanges (Lawrence et al., 2013), RColorBrewer, ggplot2 (Wickham, 2016), readr (Wickham et al., 2019), magrittr, downloader, BSDA, ggrepel, ggbio (Yin, Cook, & Lawrence, 2012), and gridExtra.

In detail, the LTRpred pipeline calls the command line tools suffixerator, LTRharvest (Ellinghaus, Kurtz, & Willhoeft, 2008), and LTRdigest (Steinbiss, Willhoeft, Gremme, & Kurtz, 2009), which are part of the GenomeTools library (Gremme, Steinbiss, & Kurtz, 2013) using customized parameter settings to screen for repeated LTRs, specific sequence motifs such as primer binding sites (PBS), polypurine tract motifs (PPT), and target site duplications (TSD) and for conserved protein domains such as reverse transcriptase (gag), integrase DNA binding domain, integrase Zinc binding domain, RNase H, and the integrase core domain. The LTRharvest and LTRdigest outputs are efficiently parsed by LTRpred and transformed into a tidy data format (Wickham et al., 2019) which subsequently enables automation of false positive curation. Next, open reading frame (ORF) prediction is performed by a customized wrapper function that runs the command line tool usearch (Edgar, 2010). This step allows to automatically filter out retrotransposons that might have conserved protein domains such as an integrase or reverse transcriptase, but fail to have any ORFs and thus might not be expressed. In a third step, retrotransposon family clustering is performed using sequence clustering with vsearch (Rognes, Flouri, Nichols, Quince, & Mahé, 2016) which defines family members by $>$90% sequence homology of the full element to each other. In a fourth step, an automated hmmer search (Finn, Clements, & Eddy, 2011) against the Dfam database (Hubley et al., 2016) is performed to assign super-family associations such as Copia or Gypsy by comparing the protein domains of de novo predicted retrotransposons with already annotated TEs in the Dfam (https://dfam.org/home) database. In the last step, the *de novo* annotated 5 prime and 3 prime LTR sequences are used to estimate the evolutionary age of the retrotransposon which should be treated with caution since retrotransposons can undergo reverse-transcriptase mediated recombination (Sanchez, Gaubert, Drost, Zabet, & Paszkowski, 2017).

Example workflow

After installing all prerequisite command line tools (https://hajkd.github.io/LTRpred/articles/Introduction.html#installation) users can run the LTRpred() pipeline using the

default parameter configuration. In the following example, an LTR transposon prediction is performed for parts of the Human Y chromosome.

```
# load LTRpred package
library(LTRpred)
# de novo LTR transposon prediction for the Human Y chromosome
LTRpred(
    genome.file = system.file("Hsapiens_ChrY.fa", package = "LTRpred"),
    cores = 4
)
```

The LTRpred() output table `*_LTRpred_DataSheet.tsv` is in tidy format and can then be imported using `read.ltrpred()`. The tidy output format is designed to work seamlessly with the tidyverse and R data science framework.

```
# import LTRpred prediction output
Hsapiens_chrY <- read.ltrpred("Hsapiens_ChrY_ltrpred/
Hsapiens_ChrY_LTRpred_DataSheet.tsv")
# look at some results
dplyr::glimpse(Hsapiens_chrY)
```

```
Observations: 21
Variables: 92
$ species                <chr> "Hsapiens_ChrY", "Hsapien...
$ ID                     <chr> "Hsapiens_ChrY_LTR_retrot...
$ dfam_target_name       <chr> NA, NA, NA, NA, NA, NA, N...
$ ltr_similarity         <dbl> 80.73, 89.85, 79.71, 83.2...
$ ltr_age_mya            <dbl> 0.7936246, 0.2831139, 0.7...
$ similarity             <chr> "(80,82]", "(88,90]", "(7...
$ protein_domain         <chr> "RVT_1", "RVT_1", NA, NA,...
$ orfs                   <int> 1, 1, 0, 0, 0, 0, 0, 1, 0...
$ chromosome             <chr> "NC000024.10Homosa", "NC0...
$ start                  <int> 3143582, 3275798, 3313536...
$ end                    <int> 3162877, 3299928, 3318551...
$ strand                 <chr> "-", "-", "+", "+", "-", ...
$ width                  <int> 19296, 24131, 5016, 12952...
$ annotation             <chr> "LTR_retrotransposon", "L...
$ pred_tool              <chr> "LTRpred", "LTRpred", "LT...
$ frame                  <chr> ".", ".", ".", ".", ".", ...
$ score                  <chr> ".", ".", ".", ".", ".", ...
$ lLTR_start             <int> 3143582, 3275798, 3313536...
$ lLTR_end               <int> 3143687, 3276408, 3313665...
$ lLTR_length            <int> 106, 611, 130, 126, 218, ...
$ rLTR_start             <int> 3162769, 3299338, 3318414...
$ rLTR_end               <int> 3162877, 3299928, 3318551...
$ rLTR_length            <int> 109, 591, 138, 137, 219, ...
$ lTSD_start             <int> 3143578, 3275794, 3313532...
$ lTSD_end               <int> 3143581, 3275797, 3313535...
$ lTSD_motif             <chr> "acag", "ttgt", "ttag", "...
$ rTSD_start             <int> 3162878, 3299929, 3318552...
$ rTSD_end               <int> 3162881, 3299932, 3318555...
$ rTSD_motif             <chr> "acag", "ttgt", "ttag", "...
$ PPT_start              <int> NA, NA, NA, NA, NA, 34660...
$ PPT_end                <int> NA, NA, NA, NA, NA, 34660...
```

```
$ PPT_motif               <chr> NA, NA, NA, NA, NA, "agag...
$ PPT_strand              <chr> NA, NA, NA, NA, NA, "+", ...
$ PPT_offset              <int> NA, NA, NA, NA, NA, 23, N...
$ PBS_start               <int> NA, NA, 3313667, 3372512,...
$ PBS_end                 <int> NA, NA, 3313677, 3372522,...
$ PBS_strand              <chr> NA, NA, "+", "+", "-", "+...
$ tRNA                    <chr> NA, NA, "Homo_sapiens_tRN...
$ tRNA_motif              <chr> NA, NA, "aattagctgga", "c...
$ PBS_offset              <int> NA, NA, 1, 3, 0, 5, 2, 5,...
$ tRNA_offset             <int> NA, NA, 1, 0, 2, 5, 1, 5,...
$ `PBS/tRNA_edist`        <int> NA, NA, 1, 1, 1, 1, 1, 1,...
$ orf.id                  <chr> "NC000024.10Homosa_314358...
$ repeat_region_length    <int> 19304, 24139, 5024, 12960...
$ PPT_length              <int> NA, NA, NA, NA, NA, 27, N...
$ PBS_length              <int> NA, NA, 11, 11, 11, 11, 1...
$ dfam_acc                <chr> NA, NA, NA, NA, NA, NA, N...
$ dfam_bits               <dbl> NA, NA, NA, NA, NA, NA, N...
$ dfam_e_value            <dbl> NA, NA, NA, NA, NA, NA, N...
$ dfam_bias               <dbl> NA, NA, NA, NA, NA, NA, N...
$ `dfam_hmm-st`           <dbl> NA, NA, NA, NA, NA, NA, N...
$ `dfam_hmm-en`           <dbl> NA, NA, NA, NA, NA, NA, N...
$ dfam_strand             <chr> NA, NA, NA, NA, NA, NA, N...
$ `dfam_ali-st`           <dbl> NA, NA, NA, NA, NA, NA, N...
$ `dfam_ali-en`           <dbl> NA, NA, NA, NA, NA, NA, N...
$ `dfam_env-st`           <dbl> NA, NA, NA, NA, NA, NA, N...
$ `dfam_env-en`           <dbl> NA, NA, NA, NA, NA, NA, N...
$ dfam_modlen             <dbl> NA, NA, NA, NA, NA, NA, N...
$ dfam_target_description <chr> NA, NA, NA, NA, NA, NA, N...
$ Clust_Cluster           <chr> NA, NA, NA, NA, NA, NA, N...
$ Clust_Target            <chr> NA, NA, NA, NA, NA, NA, N...
$ Clust_Perc_Ident        <dbl> NA, NA, NA, NA, NA, NA, N...
$ Clust_cn                <int> NA, NA, NA, NA, NA, NA, N...
$ TE_CG_abs               <dbl> 62, 125, 35, 70, 139, 83,...
$ TE_CG_rel               <dbl> 0.003213101, 0.005180059,...
$ TE_CHG_abs              <dbl> 659, 830, 150, 396, 742, ...
$ TE_CHG_rel              <dbl> 0.03415216, 0.03439559, 0...
$ TE_CHH_abs              <dbl> 2571, 3454, 748, 1743, 31...
$ TE_CHH_rel              <dbl> 0.1332400, 0.1431354, 0.1...
$ TE_CCG_abs              <dbl> 13, 24, 6, 15, 33, 16, 4,...
$ TE_CCG_rel              <dbl> 0.0006737148, 0.000994571...
$ TE_N_abs                <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0...
$ CG_3ltr_abs             <dbl> 1, 0, 1, 4, 8, 3, 1, 11, ...
$ CG_3ltr_rel             <dbl> 0.009433962, 0.000000000,...
$ CHG_3ltr_abs            <dbl> 2, 24, 9, 8, 14, 14, 9, 9...
$ CHG_3ltr_rel            <dbl> 0.01886792, 0.03927987, 0...
$ CHH_3ltr_abs            <dbl> 18, 69, 18, 26, 43, 70, 2...
$ CHH_3ltr_rel            <dbl> 0.16981132, 0.11292962, 0...
$ CCG_3ltr_abs            <dbl> 0, 0, 0, 2, 2, 0, 1, 4, 5...
$ CCG_3ltr_rel            <dbl> 0.000000000, 0.000000000,...
$ N_3ltr_abs              <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0...
$ CG_5ltr_abs             <dbl> 1, 0, 1, 4, 8, 3, 1, 11, ...
$ CG_5ltr_rel             <dbl> 0.009433962, 0.000000000,...
$ CHG_5ltr_abs            <dbl> 2, 24, 9, 8, 14, 14, 9, 9...
$ CHG_5ltr_rel            <dbl> 0.01886792, 0.03927987, 0...
$ CHH_5ltr_abs            <dbl> 18, 69, 18, 26, 43, 70, 2...
```

```
$ CHH_5ltr_rel          <dbl> 0.16981132, 0.11292962, 0...
$ CCG_5ltr_abs          <dbl> 0, 0, 0, 2, 2, 0, 1, 4, 5...
$ CCG_5ltr_rel          <dbl> 0.000000000, 0.000000000,...
$ N_5ltr_abs            <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0...
$ cn_3ltr               <dbl> NA, NA, NA, NA, NA, NA, N...
$ cn_5ltr               <dbl> NA, NA, NA, NA, NA, NA, N..
```

LTRpred output

The `LTRpred()` function internally generates a folder named `*_ltrpred` which stores all output annotation and sequence files.

In detail, the following files and folders are generated by the `LTRpred()` function:

- **Folder `*_ltrpred`**
  - `*_ORF_prediction_nt.fsa` : Stores the predicted open reading frames within the predicted LTR transposons as DNA sequence.
  - `*_ORF_prediction_aa.fsa` : Stores the predicted open reading frames within the predicted LTR transposons as protein sequence.
  - `*_LTRpred.gff` : Stores the LTRpred predicted LTR transposons in GFF format.
  - `*_LTRpred.bed` : Stores the LTRpred predicted LTR transposons in BED format.
  - `*_LTRpred_DataSheet.tsv` : Stores the output table as data sheet.
  - **Folder `*_ltrharvest`**
    - `*_ltrharvest/*_BetweenLTRSeqs.fsa` : DNA sequences of the region between the LTRs in fasta format.
    - `*_ltrharvest/*_Details.tsv` : A spread sheet containing detailed information about the predicted LTRs.
    - `*_ltrharvest/*_FullLTRRetrotransposonSeqs.fsa` : DNA sequences of the entire predicted LTR retrotransposon.
    - `*_ltrharvest/*_index.fsa` : The suffixarray index file used to predict putative LTR retrotransposons.
    - `*_ltrharvest/*_Prediction.gff` : A spread sheet containing detailed additional information about the predicted LTRs (partially redundant with the *_Details.tsv file).
  - **Folder `*_ltrdigest`**
    - `*_ltrdigest/*_LTRdigestPrediction.gff` : A spread sheet containing detailed information about the predicted LTRs.
    - `*_ltrdigest/*_index_ltrdigest.fsa` : The suffixarray index file used to predict putative LTR retrotransposons with LTRdigest.
    - `*_ltrdigest/*-ltrdigest_tabout.csv` : A spread sheet containing additional detailed information about the predicted LTRs.
    - `*_ltrdigest/*-ltrdigest_complete.fas` : The full length DNA sequences of all predicted LTR transposons.
    - `*_ltrdigest/*-ltrdigest_conditions.csv` : Contains information about the parameters used for a given LTRdigest run.
    - `*_ltrdigest/*-ltrdigest_pbs.fas` : Stores the predicted PBS sequences for the putative LTR retrotransposons.
    - `*_ltrdigest/*-ltrdigest_ppt.fas` : Stores the predicted PPT sequences for the putative LTR retrotransposons.

* `*_ltrdigest/*-ltrdigest_5ltr.fas` and `*-ltrdigest_3ltr.fas`: Stores the predicted 5' and 3' LTR sequences. Note: If the direction of the putative retrotransposon could be predicted, these files will contain the corresponding 3' and 5' LTR sequences. If no direction could be predicted, forward direction with regard to the original sequence will be assumed by LTRdigest, i.e. the 'left' LTR will be considered the 5' LTR.
* `*_ltrdigest/*-ltrdigest_pdom_<domainname>.fas` : Stores the DNA sequences of the HMM matches to the LTR retrotransposon candidates.
* `*_ltrdigest/*-ltrdigest_pdom_<domainname>_aa.fas` : Stores the concatenated protein sequences of the HMM matches to the LTR retrotransposon candidates.
* `*_ltrdigest/*-ltrdigest_pdom_<domainname>_ali.fas` : Stores the alignment information for all matches of the given protein domain model to the translations of all candidates.

Visualising functional retrotransposons annotated with `LTRpred`

Finally, users can visualise the positioning of *de novo* annotated retrotransposons along the chromosomes. Here, we choose an example based on the yeast genome.

```
# install.packages("biomartr")
# download the yeast genome from ENSEMBL
sc_genome_path <- biomartr::getGenome(db = "ensembl",
                                      organism = "Saccharomyces cerevisiae",
                                      path = "yeast_genome",
                                      gunzip = TRUE)
# run LTRpred on the yeast genome (-> this may take a few minutes)
LTRpred::LTRpred(
    genome.file = sc_genome_path,
    cores = 4
)
# import functional retrotransposon annotation
sc_ltrpred <- LTRpred::read.ltrpred(
 file.path("Saccharomyces_cerevisiae_ltrpred",
                       "Saccharomyces_cerevisiae_LTRpred_DataSheet.tsv"))
# filter for potentially active candidates
sc_ltrpred_filtered <- LTRpred::quality.filter(sc_ltrpred,
                                               sim = 95,
                                               strategy = "stringent",
                                               n.orfs = 1)
# visualise the position of potentially active retrotransposons
# along the yeast chromosomes
LTRpred::plot_element_distr_along_chromosome(sc_ltrpred_filtered,
                                             sc_genome_path)
```

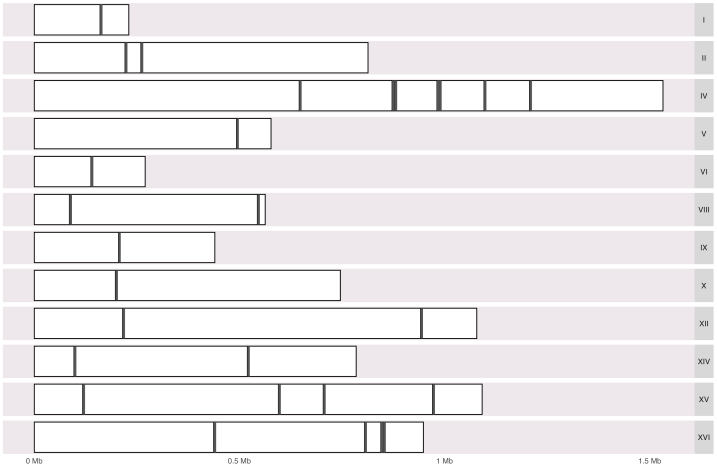

Figure 1: Positions of functional retrotransposons annotated by LTRpred along the yeast chromosomes.

Metagenome scale annotations

`LTRpred` allows users to generate annotations not only for single genomes but for multiple genomes (metagenomes) using only one pipeline function named `LTRpred.meta()`.

Users can download the `biomartr` package (Drost & Paszkowski, 2017) to automatically retrieve genome assembly files for the species of interest.

```
# specify the scientific names of the species of interest
# that shall first be downloaded and then be used
# to generate LTRpred annotations
species <- c("Arabidopsis thaliana", "Arabidopsis lyrata", "Capsella rubella")
# download the genome assembly files for the species of interest
#
# install.packages("biomartr")
biomartr::getGenomeSet(db = "refseq", organisms = species,
                       path = "store_genome_set")
# run LTRpred.meta() on the 3 species with 3 cores
LTRpred::LTRpred.meta(genome.folder = "store_genome_set",
                      output.folder = "LTRpred_meta_results",
                      cores = 3)
```

Acknowledgements

This work was supported by a European Research Council grant named EVOBREED [grant number 322621] and a Gatsby Fellowship [grant number AT3273/GLE]. The author also thanks Jerzy Paszkowski for supporting the development and application of `LTRpred` and for providing valuable suggestions to improve the manuscript.

References

Benoit, M., Drost, H.-G., Catoni, M., Gouil, Q., Lopez-Gomollon, S., Baulcombe, D., & Paszkowski, J. (2019). Environmental and epigenetic regulation of rider retrotransposons in tomato. *PLoS Genet.*, *15*(9), e1008370.

Cho, J., Benoit, M., Catoni, M., Drost, H.-G., Brestovitsky, A., Oosterbeek, M., & Paszkowski, J. (2019). Sensitive detection of pre-integration intermediates of long terminal repeat retrotransposons in crop plants. *Nature Plants*, *5*(1), 26–33.

Chuong, E. B., Elde, N. C., & Feschotte, C. (2017). Regulatory activities of transposable elements: From conflicts to benefits. *Nat. Rev. Genet.*, *18*(2), 71–86.

Drost, H.-G., & Paszkowski, J. (2017). Biomartr: Genomic data retrieval with R. *Bioinformatics*, *33*(8), 1216–1217.

Drost, H.-G., & Sanchez, D. H. (2019). Becoming a selfish clan: Recombination associated to Reverse-Transcription in LTR retrotransposons. *Genome Biol. Evol.*, *11*(12), 3382–3392.

Edgar, R. C. (2010). Search and clustering orders of magnitude faster than BLAST. *Bioinformatics*, *26*(19), 2460–2461.

Ellinghaus, D., Kurtz, S., & Willhoeft, U. (2008). LTRharvest, an efficient and flexible software for de novo detection of LTR retrotransposons. *BMC Bioinformatics*, *9*, 18.

Finn, R. D., Clements, J., & Eddy, S. R. (2011). HMMER web server: Interactive sequence similarity searching. *Nucleic Acids Res.*, *39*(Web Server issue), W29–37.

Gremme, G., Steinbiss, S., & Kurtz, S. (2013). GenomeTools: A comprehensive software library for efficient processing of structured genome annotations. *IEEE/ACM Trans. Comput. Biol. Bioinform.*, *10*(3), 645–656.

Lawrence, M., Huber, W., Pagès, H., Aboyoun, P., Carlson, M., Gentleman, R., Morgan, M. T., et al. (2013). Software for computing and annotating genomic ranges. *PLoS Comput. Biol.*, *9*(8), e1003118.

Rognes, T., Flouri, T., Nichols, B., Quince, C., & Mahé, F. (2016). VSEARCH: A versatile open source tool for metagenomics. *PeerJ*, *4*, e2584.

Sanchez, D. H., Gaubert, H., Drost, H.-G., Zabet, N. R., & Paszkowski, J. (2017). High-frequency recombination between members of an LTR retrotransposon family during transposition bursts. *Nature Communications*, *8*(1), 1283.

Steinbiss, S., Willhoeft, U., Gremme, G., & Kurtz, S. (2009). Fine-grained annotation and classification of de novo predicted LTR retrotransposons. *Nucleic Acids Res.*, *37*(21), 7002–7013.

Sundaram, V., & Wysocka, J. (2020). Transposable elements as a potent source of diverse cis-regulatory sequences in mammalian genomes. *Philos. Trans. R. Soc. Lond. B Biol. Sci.*, *375*(1795), 20190347.

Wang, Z., & Baulcombe, D. (2020). Transposon age and non-cg methylation. *Nature Communications*, *In press*.

Wickham, H. (2016). *Ggplot2: Elegant graphics for data analysis*. Springer.

Wickham, H., Averick, M., Bryan, J., Chang, W., McGowan, L., François, R., Grolemund, G., et al. (2019). Welcome to the tidyverse. *Journal of Open Source Software*, *4*(43), 1686.

Yin, T., Cook, D., & Lawrence, M. (2012). Ggbio: An R package for extending the grammar of graphics for genomic data. *Genome Biol.*, *13*(8), R77.