Guide to the class cjs-rcs-article for *The Canadian Journal of Statistics*

Statistical Society of Canada
Class and documentation developed by Vincent Goulet

Abstract Authors submitting an article to *The Canadian Journal of Statistics* should use the document class cjs-rcs-article and its companion bibliographic styles to typeset their manuscript with LaTeX. This guide provides a complete description of the characteristics and features of the class, as well as the details of its implementation and the version history. A French version of this guide is also available, but without these last two elements of documentation.

Résumé Les auteurs qui soumettent un article dans *La revue canadienne de statistique* doivent utiliser la classe cjs-rcs-article et les styles bibliographiques qui l'accompagnent pour préparer leur manuscript avec LaTeX. Ce guide fournit une description complète des caractéristiques et fonctionnalités de la classe, ainsi que les détails de sa mise en œuvre et l'historique des versions. Une version française de ce guide est également disponible, mais celle-ci ne contient pas ces deux derniers éléments de documentation.

Version This is the documentation for the class cjs-rcs-article v1.2, dated 2025/07/10.

1 Introduction

The LaTeX document class cjs-rcs-article and its companion bibliographic styles `cjs-rcs-en` and `cjs-rcs-fr` typeset manuscripts immediately in accordance with the presentation rules of *The Canadian Journal of Statistics*. In particular, the dimensions of the typeblock, the page layout, the fonts and the list of references are identical to the final version of the article. But for a few modifications, the present guide uses the page layout of the class.

Typesetting a manuscript with cjs-rcs-article requires a recent and up-to-date TeX distribution. We recommmend the distributions TeX Live—or its variant for macOS MacTeX—and MiKTeX. You may also use an online LaTeX editor such as Overleaf; their TeX distributions usually satisfy the conditions above.

The class cjs-rcs-article is part of the standard TeX distributions via the package **cjs-rcs-article**. If your distribution is up-to-date, you should be able to use the class without any special intervention.

This guide is organized as follows. Section 2 explains how to get started quickly with the class, and Section 3 describes how to install the class if it is not already available on your system. Section 4 gives all the details of how to use the package for the various components of your article. Section 5 discusses hyperlinks and the creation of the table of contents of the PDF. Finally, Section 6 explains how you can use the class to prepare supplementary material. For TeX experts, we include two appendices. Appendix A discusses features that are primarily intended for editors rather than authors, and Appendix B gives the details of the implementation.

2 Quick start

The class ships with templates for articles in English and in French. In TeX Live, these are located with the documentation of the class. You may also obtain the templates from CTAN or in the project website. To quickly start writing your article, copy the appropriate template to your working directory and follow the brief instructions and the examples therein. You should also review the author guidelines that ship along the present documentation.

If your manuscript contains non-ASCII characters (such as accented letters), you have to save your file using the UTF-8 character encoding. To take full advantage of the class, you should compile your manuscript using LuaLaTeX.

The remainder of this guide provides detailed information on the characteristics and features of the class. We recommend that you read at least Section 4.7 on mathematical typesetting.

3 Installation

The class cjs-rcs-article is distributed as the package **cjs-rcs-article** through the Comprehensive TeX Archive Network (CTAN). It is part of the standard TeX distributions such as TeX Live, MacTeX and MiKTeX. We strongly recommend that you install or update the package using the package manager of your TeX distribution. For example, with TeX Live, you may install or update from the graphical package manager, or using either of these commands from the command line:

```
$ tlmgr install cjs-rcs-article
```

or

```
$ tlmgr update cjs-rcs-article
```

If you are not in position to install packages or update the TeX distribution—notably when using an online editor—you may instead opt for a simplified installation of the class, but that will be limited in scope to your project. Download the archive `cjs-rcs-article-project-install.zip` from the project website, uncompress it and place its contents in the directory of your project. You will then have access to all the essential files of the class, including the templates, within this project.

Manual installation with full access to the source code is possible for TeX experts. Please follow the instructions provided in the `README.md` file included in the package.

4 Usage

The class cjs-rcs-article relies on modern and advanced tools, all of which are available in recent TeX distributions. In particular:

- the class memoir on which cjs-rcs-article is based (therefore, cjs-rcs-article inherits all the features of memoir);
- the fonts STIX Two for text and mathematics;
- the fonts Fira for decorative elements (notably Fira Sans for section titles and `Fira Mono` for computer code);
- the package **babel** to handle multiple languages in a single document.

The class is largely compatible with the engine pdfLaTeX. However, to take full advantage of its characteristics[1] we recommend that you compile your manuscript using the more modern engine LuaLaTeX.

4.1 Templates

The class is distributed with templates for articles in English and French. If you are using the version of the class included with your TeX distribution, the templates may be located with the documentation (this is the case in TeX Live). You may also obtain them from CTAN or in the project website. If you installed the class manually by following the instructions in Section 3, the templates reside in your working directory.

We strongly encourage to start from a template to write your article. It may also help to follow a template while reading this section.

[1] In particular when it comes to fonts. The engine pdfLaTeX uses the Type 1 versions of the fonts STIX Two and Fira. Since the PostScript Type 1 technology has been deprecated, the fonts are no longer updated. The engine LuaLaTeX uses the more modern—and complete—OpenType versions of the fonts.

Table 1: Summary of the commands and environments of the class cjs-rcs-article to provide title-page information. Names starting with the symbol "\" are commands, and the others are environments. Items marked with ★ are required.

Name	Usage
`\title`★	main title of the article (with an abbreviated version as an option)
`\author`★	name of an author (and metadata as an option)
`\surname`★	surname of an author (used inside `\author`)
`\affil`★	affiliation of an author
`\runningauthor`	author identification for the running heads
`englishabstract`★	English abstract
`frenchabstract`★	French abstract
`keywords`	list of keywords
`classification`★	mathematics subject classification MSC2020

4.2 Text encoding

When the text contains accented letters or any other non-ASCII symbol, you must save your manuscript using the UTF-8 character encoding.[2]

4.3 Class declaration and options

The class is loaded with the command:

```
\documentclass[⟨options⟩]{cjs-rcs-article}
```

Some ⟨*options*⟩ of the class are intended for the authors, whereas others are instead intended for the editors of *The Journal*. Presentation of the editor ⟨*options*⟩ is deferred to Appendix A. Here, we concentrate on the author ⟨*options*⟩, that is: the languages of the article, `review`, `supplement`, and `nocjs`.

`english` (*option*)
`french` (*option*)

The Canadian Journal of Statistics asks for abstracts in English and French. Therefore, `english` and `french` are required in the ⟨*options*⟩ of the class. You need to declare these options in a specific order, because the *last* one becomes the main language of the article.

`review` (*option*)

The option `review` produces a partially anonymized version of the manuscript suitable for peer review. With this option, the names of the authors and their affiliations are removed from the title page; the acknowledgements, funding information and ORCID iDs are removed from the back matter; the lines are numbered; and the line spacing is increased.

`supplement` (*option*)

The class option `supplement` allows you to use the class cjs-rcs-article to prepare your supplementary material. See Section 6 for additional details.

`nocjs` (*option*)

If you wish to use the class cjs-rcs-article to typeset documents other than an article for *The Journal*, use the option `nocjs` to hide all journal-specific publication information.[3] The option also allows you to specify a licence agreement in a statement that will be displayed in the footer of the title page; see Section 4.6 for additional details.

4.4 Titling information

The class cjs-rcs-article defines a number of commands and environments to provide information for the title page. See Table 1 for a summary; the detailed descriptions follow.

4.4.1 Main title

`\title`

The command `\title` defines the title of the article. An optional argument specifies an abbreviated version of the title that will be used for the running head. Therefore, the syntax of the command is:

```
\title[⟨Short title⟩]{⟨Full title⟩}
```

[2] Since UTF-8 is usually not the default encoding on Windows, be particularly careful if you are using this operating system.

[3] This guide uses the option `nocjs`.

Table 2: Keys and values defined for the options of the command `\author`

Key	Value	Description
`orcid`	⟨*id*⟩	ORCID iD of the author
`email`	⟨*email*⟩	email address of the author
`corresponding`	`true/false`	corresponding author

Write the title in sentence style, that is, with only an initial capital. In English, also capitalize the first word after a colon. Break a long title needing more than one line using `\\` or `\newline`.

4.4.2 Authors and affiliations

You must provide the names of all the authors and their persistent digital identifiers ORCID iD; the affiliation for each author; the email address of the corresponding author; and the list of authors for the running head. The class cjs-rcs-article uses an information entry system inspired by the package **authblk** consisting of a series of pairs of commands `\author` and `\affil`, one per author.

`\author` `\surname`

The command `\author` is the main component. Its general syntax is the following:

```
\author[⟨options⟩]{⟨Forename⟩ \surname{⟨Surname⟩}}
```

The ⟨*Surname*⟩ must be entered using the command `\surname`. Write the ⟨*Forename*⟩ and ⟨*Surname*⟩ in lowercase (except for the initial capital and initials). If needed, the ⟨*Forename*⟩ may include one or more initials, each followed by a period.

The ⟨*options*⟩ are used to supply an author's metadata using a ⟨*key*⟩ = ⟨*value*⟩ interface. Table 2 shows the currently available keys and the expected value for each. The ⟨*key*⟩–⟨*value*⟩ pairs are separated by commas.

The key `orcid` provides the ORCID iD of the author, that is, a 16-digit alphanumeric code of the form `0000-0000-0000-0000`.

The key `email` provides the email address of the author. Currently, only the email address of the corresponding author appears in the manuscript. Nevertheless, we recommend providing this address for all authors.

The key `corresponding` identifies the corresponding author. The value of the key is implicitly `true` when it appears in the options, and `false` when not present. In other words, instead of specifying `corresponding=true` and `corresponding=false` for every author, you may simply indicate `corresponding` for the corresponding author. This key has no effect if the `email` key is missing for the author.

`\affil`

Second component of the author entry system, the command `\affil` declares the affiliation of the author mentioned in the immediately preceding `\author` command. The general form of an affiliation is:

```
\affil{⟨Department, University, City, Country⟩}
```

`\runningauthor`

The running head contains the surnames of the authors as automatically collected by the command `\surname`. If the list is too long, you may use the command `\runningauthor` to provide a shorter version: the first author's surname followed by *et al.* (in English) or *et collab.* (in French):

```
\runningauthor{⟨Surname⟩ et al.}
\runningauthor{⟨Surname⟩ et collab.}
```

4.4.3 Abstracts

`englishabstract` (*env.*) `frenchabstract` (*env.*)

Articles in *The Canadian Journal of Statistics* have both English and French abstracts. Use the environments `englishabstract` and `frenchabstract` to enter the abstracts. Their position on the title page is set automatically by the class.

The standard environment `abstract` has no effect in the class cjs-rcs-article.

4.4.4 Keywords

`keywords` (*env.*) The environment `keywords` is used to provide the list of keywords for the article. Its usage is similar to an `itemize` or `enumerate` list:

```
\begin{keywords}
\item Keyword 1
\item keyword 2
...
\end{keywords}
```

As shown above, an initial capital is needed for the first keyword only. Punctuation is added automatically.

4.4.5 Mathematics subject classification

`classification` (*env.*) The environment `classification` is used to enter the mathematics subject classification of the article. Its interface is the following:

```
\begin{classification}[⟨type⟩]
\item[⟨level⟩] code 1, code 2, ...
\item[⟨level⟩] code 1, code 2, ...
...
\end{classification}
```

The default classification system used is MSC2020. Should you need to use a different system for your manuscript that would be published elsewhere than in *The Journal*, you may provide its name in the optional argument ⟨*type*⟩. The required entries ⟨*level*⟩ identify the classification levels, for example *Primary* and *Secondary*. Insert commas between the classification codes of a level, but no other punctuation.

4.4.6 Creation of the title page

`\maketitle` The aforementioned titling information may be provided in the preamble or at the very beginning of the document body. The standard command `\maketitle` typesets the title page at the place where it appears. Therefore, you should issue this command after `\begin{document}`, but before any content.

The class reserves a space at the bottom of the title page for the licence statement that will be added by the publisher (Wiley) after acceptance of the paper. For the options `review`, `supplement` and `nocjs`, the space is replaced by a footer with the page number.

4.4.7 ADNI data use agreement

`\ADNIacknowledgement` The Alzheimer's Disease Neuroimaging Initiative (ADNI) Data Sharing and Publication Policy requires authors using their data to acknowledge ADNI in the list of authors and give a specific statement on the title page. The command `\ADNIacknowledgement` adds these two requirements. The command has no arguments; simply insert it after the declarations of the authors.

`\ADMCacknowledgement` Similarly, the command `\ADMCacknowledgement` acknowledges the Alzheimer's Disease Metabolomics Consortium (ADMC).

4.5 Back matter information

Articles in *The Canadian Journal of Statistics* end with elements forming the so-called back matter information: supplementary material available online, data sharing statement, acknowledgements, etc. The class cjs-rcs-article provides specialized environments to enter this information. See Table 1 for a summary; the detailed descriptions follow.

The environments described in this section are optional.

Although it is displayed at the end, we recommend that you enter the back matter information with the titling information, in the preamble or at the beginning of the main body of the document.

Table 3: Summary of the environments of cjs-rcs-article to provide the back matter information

Name	Usage
`supplement`	supplementary material
`sharing`	data sharing conditions
`acknowledgements`	acknowledgements
`funding`	funding information

Table 4: Commands to typeset the icons for the main Creative Commons licences. All commands feature a starred variant that add a matching copyright icon.

Command	Output	Starred variant
`\ccby`		
`\ccbysa`		
`\ccbync`		
`\ccbyncsa`		
`\ccbynd`		
`\ccbyncnd`		

`supplement` (*env.*) Use the environment `supplement` to provide, as free form text, information on any supplementary material: figures, tables, computer code, long mathematical proofs, etc. This material must be cited in the main article.

`sharing` (*env.*) *The Canadian Journal of Statistics* adheres to the Expects Data Sharing editorial policy of Wiley. Therefore, you must use the environment `sharing` to make a statement about the availability or absence of shared data. You may also use the environment to describe how to access shared computer code.

`acknowledgements` (*env.*)
`acknowledgments` (*env.*) Use the environment `acknowledgements` (or its alias `acknowledgments`) to write your acknowledgements in free form text. Avoid funding information here. The environment accepts an optional argument to replace its default name ("Acknowledgements" in English; "*Remerciements*" in French) by a name of your choosing.

`funding` (*env.*) The environment `funding` lets you provide, in free form text, all pertinent funding information. The environment accepts an optional argument to replace its default name ("Funding information" in English; "*Financement*" in French) by a name of your choosing.

`\makebackmatter` Similar to `\maketitle`, the command `\makebackmatter` typesets all the back matter information at the place where it appears. The information is listed in the following order, when available: supplementary material; data sharing; acknowledgements; funding information; full ORCID iD of the authors. The latter is built automatically by the class from the information provided in the commands `\author`.

4.6 Licensing

`\licence`
`\license` You may specify a licence for the manuscript with the command `\licence` (or its alias `\license`). We recommend that you include this command with the title and back matter information.

For articles accepted for publication in *The Canadian Journal of Statistics*, the licence statement is added by the publisher. Therefore, any licensing information provided with either the default, `review` or `supplement` options is ignored by the class.

On the other hand, for the option `nocjs`, the licence statement is displayed in the footer of the title page, right-aligned on an odd page, left-aligned on an even page. Make sure to keep the statement short, as the space is limited to a single line.

`\ccby`
`\ccbysa`
`\ccbync`
`\ccbyncsa`
`\ccbynd`
`\ccbyncnd` You may use the commands from Table 4 to identify Creative Commons licences in your licence statement. All commands have a starred variant (name followed by "`*`") that follows the licence icons by a matching copyright icon. See *About CC Licenses* for more information on Creative Commons licences and how to apply them.

Table 5: Additional commands for mathematical symbols and operators defined by the class cjs-rcs-article

Command	Description	Example	Output
`\Pr`	probability	`\Pr[X = x]`	$\mathrm{P}[X = x]$
`\E`	expected value	`\E[X]`	$\mathrm{E}[X]$
`\Var`	variance	`\Var[X]`	$\mathrm{var}[X]$
`\Cov`	covariance	`\Cov(X, Y)`	$\mathrm{cov}(X, Y)$
`\corr`	correlation	`\corr(X, Y)`	$\mathrm{corr}(X, Y)$
`\prdist`	probability distribution	`\prdist{N}`	$\mathcal{N}$
`\mat`	matrix or vector	`\mat{A}`	$\mathbf{A}$
`\matit`	matrix or vector (italic)	`\matit{x}`	$\boldsymbol{x}$
`\trsp`	transposition	`\mat{A}\trsp`	$\mathbf{A}^{\mathsf{T}}$
`\tr`	trace	`\tr(\mat{A})`	$\mathrm{tr}(\mathbf{A})$
`\diag`	diagonal	`\diag(\mat{A})`	$\mathrm{diag}(\mathbf{A})$
`\Nset`	natural numbers	`\Nset`	$\mathbb{N}$
`\Zset`	integers	`\Zset`	$\mathbb{Z}$
`\Qset`	rational numbers	`\Qset`	$\mathbb{Q}$
`\Rset`	real numbers	`\Rset`	$\mathbb{R}$
`\Cset`	complex numbers	`\Cset`	$\mathbb{C}$

For example, the licence statement that appears on the title page of this document was created with the following command:

```
\licence{{\ccbysa*} 2025, Statistical Society of Canada {\textbar}
  Société statistique du Canada}
```

4.7 Mathematics

The class automatically loads the packages **amsmath**, **amsthm** and, when the document is compiled with LuaLaTeX, **unicode-math**. Note that it is not compatible with the package **amssymb**. The class also defines a number of commands and environments to standardize the typesetting of mathematical content in *The Canadian Journal of Statistics*.

Table 5 provides a summary of the additional commands defined by the class.

`\Pr` `\E` `\Var` `\Cov` `\corr`

LaTeX provides a large number of commands to typeset the most common (and many not so common) mathematical operators. In addition, use the commands `\Pr`, `\E`, `\Var`, `\Cov` and `\corr` to uniformly typeset the operators for probability[4], expected value, variance, covariance, and correlation. These commands have no arguments, so it is your responsibility to provide delimiters around the operands.

`\prdist`

The command `\prdist` typesets the symbol for a probability distribution. For example:

`\prdist{N}(\mu, \sigma^2)`	→	$\mathcal{N}(\mu, \sigma^2)$	(Normal)
`\prdist{B}(n, p)`	→	$\mathcal{B}(n, p)$	(Binomial)
`\prdist{P}(\lambda)`	→	$\mathcal{P}(\lambda)$	(Poisson)
`\prdist{E}(\lambda)`	→	$\mathcal{E}(\lambda)$	(Exponential)
`\prdist{G}(\alpha, \beta)`	→	$\mathcal{G}(\alpha, \beta)$	(Gamma)

`\mat` `\matit`

Bold in mathematics is a tricky issue. The commands `\mat` and `\matit` generate bold symbols reliably across TeX engines. The first typesets a variable in roman bold, a style often used for matrices and vectors (hence the name of the command). With LuaLaTeX, `\mat` also works with Greek letters, but their shape is not modified: $\boldsymbol{\mu}$, $\mathbf{\Gamma}$. The command `\matit` explicitly typesets a variable in bold italic.

`\trsp`

To indicate matrix transposition, use only the command `\trsp`.

`\tr` `\diag`

The commands `\tr` and `\diag` define two other common linear algebra operators, the trace and

[4]The class redefines the standard command `\Pr`.

Table 6: Theorem-like environments defined by the class cjs-rcs-article. The last column refers to the styles defined by **amsthm**.

Environment	English title	French title	Style
`theorem`	Theorem	Théorème	`plain`
`lemma`	Lemma	Lemme	`plain`
`proposition`	Proposition	Proposition	`plain`
`corollary`	Corollary	Corollaire	`plain`
`definition`	Definition	Définition	`definition`
`algorithm`	Algorithm	Algorithme	`definition`
`remark`	Remark	Remarque	`remark`

the diagonal.

`\Nset` `\Zset` `\Qset` `\Rset` `\Cset`

To ensure a uniform representation of number sets, use the class commands `\Nset`, `\Zset`, `\Qset`, `\Rset`, and `\Cset` to typeset the sets of natural numbers, integers, rational numbers, real numbers and complex numbers, respectively.

You may define new mathematical operators in the preamble of your document. To do so, use the command `\DeclareMathOperator` of **amsmath**; see Section 5 of the package documentation for more details.

Table 6 lists the theorem-like environments defined by the class. The environment `proof` defined by **amsthm** is also available. All these environments accept an optional argument to provide additional information in the heading, such as the name of the theorem or its author. Here is a brief overview of the style of the environments:

Theorem 1. *Text.*

Lemma 1 (Famous lemma)**.** *Text.*

Proposition 1. *Text.*

Corollary 1. *Text.*

Definition 1. Text.

Algorithm 1. Text.

Remark 1. Text.

Proof. Text. □

If you need to define additional theorem-like environments, use the features of **amsthm**, in particular the commands `\theoremstyle` and `\newtheorem`.

4.8 Computer code and software

The class does not offer any special features to typeset listings of computer code. If you need to display code, we recommend that you take advantage of specialized packages such as **fancyvrb** or **listings**. If you use R for your statistical analysis, we suggest that you prepare the file using a literate programming system such as Sweave or knitr.

That said, the class defines a few commands to ease and standardize the display of software names and code chunks.

`\proglang` The command `\proglang` typesets the names of programming languages and software. The class also provides shortcuts for the most common cases; see Table 7.

`\pkg` The command `\pkg` typesets the names of software packages, extensions or modules.

`\code` The command `\code` typesets code chunks within the text. The command escapes the LaTeX special characters `_`, `~` and `$`. Therefore, you may write, for example, `\code{var_name}` to get `var_name`.

4.9 Appendices

`\appendix` If needed, use the standard command `\appendix` to indicate the start of the appendices. The command changes the numbering of the following sections to an alphabetic form.

Table 7: List of shortcuts for the names of programming languages and software.

Software	Command
R	`\Rlang`
SAS	`\SASlang`
SPSS	`\SPSSlang`
Stata	`\Statalang`
Python	`\Pylang`
Julia	`\Julialang`
C	`\Clang`
C++	`\Cpplang`

4.10 Citations and list of references

Use BibTeX for citations and to generate the list of references. The class relies on the package **natbib** to manage and typeset citations.

The class uses its own bibliography styles to typeset the list of references: `cjs-rcs-en` for articles in English, and `cjs-rcs-fr` for articles in French. These styles support all the standard BibTeX entry types (`article`, `book`, `proceedings`, etc.) and all the standard fields (`author`, `title`, `journal`, etc.). Furthermore, the styles introduce support for the following fields:

`doi` The Digital Object Identifier (DOI) of a resource. Enter the value only as ⟨*prefix*⟩/⟨*suffix*⟩ (for example: `10.1000/182`). The styles automatically create a hyperlink to the resource.

`isbn` The International Standard Book Number (ISBN) of a book. When present, the ISBN is displayed in the list of references.

`issn` The International Standard Serial Number (ISSN) of a serial publication, such as a magazine. Although allowed in the bibliographic database, the number is not displayed in the list of references.

`language`
The language of the reference. Including this information in an entry allows to hyphenate the words in the title according to the rules of its language.

`url` A URL for the reference. If both a URL and a DOI are present in the database, the latter takes precedence.

`\bibliography` The command `\bibliography` inserts the list of references at the point where it appears. Its arguments are the root names, separated by commas, of the bibliographic databases. Use this command to generate your reference list.

4.11 Writing in French

The class is also carefully configured to prepare manuscripts in French. If you are writing in this language, please read the corresponding section in the French version of the documentation for additional details.

5 Hyperlinks

The class loads the package **hyperref** to insert internal and external hyperlinks into the PDF. You should use the command `\autoref` to create references to labeled text elements. The class defines the English and French description labels for all sectioning levels, for figures and tables, and for all the theorem-like environments introduced in Section 4.7. If you need to define a description label of a new environment ⟨*env*⟩, use the following command in the preamble of your document:

```
\addto\extrasenglish{\def\⟨env⟩autorefname{⟨description⟩}}
```

(If the main language of the article is French, replace `\extrasenglish` by `\extrasfrench`.[5])

[5] For further details, see Section 6 of the documentation of **hyperref**.

`\texorpdfstring` The package **hyperref** builds the table of contents of the PDF. If you use mathematical notation or symbols in a section title, compilation of your manuscript may either result in warnings of the form

> Token not allowed in a PDF string

or halt with the error message

```
Improper alphabetic constant
```

If you positively need the notation or symbol in the title, wrap it in `\texorpdfstring`. This command takes two arguments: the notation or symbol that should appear in the document, and the replacement text for the table of contents of the PDF. For example, you would replace

```
\section{Algorithm for $U_n$}
```

by something along the lines of

```
\section{Algorithm for \texorpdfstring{$U_n$}{U[n]}}
```

You would also replace

```
\subsection{Contribution of $\beta$}
```

by

```
\subsection{Contribution of \texorpdfstring{$\beta$}{beta}}
```

or even

```
\subsection{Contribution of \texorpdfstring{$\beta$}{β}}
```

if you can enter Unicode characters on the keyboard.

6 Supplementary material

The supplementary material is useful for additional results or information, such as tables and figures, computer code, long mathematical proofs, etc. This material must be cited in the main article, and it is peer reviewed.

You can use cjs-rcs-article to prepare the supplementary material by specifying the class option `supplement`. The differences are as follows:

- a statement identifying the document as supplementary material is added before the title;
- the abstracts, keywords and mathematics subject classification are hidden;
- the space for the licence statement is replaced by a footer with the page number;
- the back matter is also hidden;
- the sections, equations, figures, etc., are numbered with a prefix “S”.

The title and back matter material may still be present in the source code. In other words, you may create the supplementary material from the same template as your main article; just add `supplement` to the class options.

The option `supplement` is compatible with the option `review` to create an anonymized version of the supplementary material suitable for peer review.

Have fun writing!

A Features for the editors

The options and commands presented in this appendix should mostly be of use to the editors of *The Canadian Journal of Statistics* during the peer-reviewing phase, and for the production of the final version before publication.

`review` (*option*) The option `review`, already mentioned in Section 4.3, typesets the anonymized version of the manuscript for peer review. Line spacing is also increased and the lines are numbered. This option is compatible with `supplement`.

If the document was successfully compiled with the option `review` and the option is later removed, you will need to either delete the `.aux` file before compiling again or compile the document at least twice.

`final` (*option*) The option `final`—which overrides `review`—activates checks that the publication information and article history are correctly defined. Furthermore, if licensing information is provided using the commands of Section 4.6, the licence statement is displayed at the bottom of the first page. The option has no other effect on the rendering of the article.

`\jvol` `\jissue` `\jyear` `\firstpage` The commands `\jvol`, `\jissue` and `\jyear` provide the publication information: volume, number and year of the issue in which the article appears. The command `\firstpage` specifies the number of the first page; the number of the last page is determined automatically. This information appears on the title page.

`\received` `\accepted` The commands `\received` and `\accepted` specify the revision history that is displayed at the very end of the article. Their argument is a date in the ISO 8601 format YYYY-MM-DD. For example:

```
\received{2025-07-10}
```

`\specialack` The command `\specialack` provides a generic interface to give special acknowledgement on the title page, such as the one required by ADNI (Section 4.4.7). The command takes two arguments:

`\specialack{`⟨*by-line*⟩`}{`⟨*statement*⟩`}`

The ⟨*by-line*⟩ is inserted in the list of authors, whereas the ⟨*statement*⟩ is typeset at the bottom of the title page, with the licensing information. Each argument may be empty, in which case the corresponding information is not printed. For example, the command `\ADNIacknowledgement` is effectively defined as:

```
\specialack{\CJS@adnibyline}{\CJS@adnistatement}
```

(The internal commands `\CJS@adnibyline` and `\CJS@adnistatement` typeset the official ADNI by-line and statement. The analogous commands `\CJS@admcbyline` and `\CJS@admcstatement` serve the same purpose for the Alzheimer's Disease Metabolomics Consortium.)

`\specialackmark` The command `\specialackmark` sets the footnote marker used by `\specialack`. The default definition of the marker is:

```
\textsuperscript{\ensuremath{*}}
```

It may happen that the title material occupies too much space (or, rarely, too little). When this happens, the editors can tweak the typesetting of the title page using the following elements.

`\titlingskip` (ℓ) The distance between the elements of the title page (from the abstracts to the corresponding author address) is `\titlingskip`. This value may be changed as usual via `\setlength`. The default value is `2.0pt plus 2.0pt minus 0.8pt`.

`\fancybreaksep` (ℓ) The decorative element between the title information and the body of the article is surrounded by a vertical skip of length `\fancybreaksep`.[6] This value may be changed as usual via `\setlength`. The default value is `6.0pt plus 3.0pt minus 1.0pt`.

`\suppressfancybreak` `\showfancybreak` The command `\suppressfancybreak` suppresses the decorative element between the title information and the body of the article. This will be useful if it appears at the bottom of the title

[6] The terminology *fancy break* is taken from memoir.

page or the top of the subsequent page. The command must appear *before* `\maketitle`. For the sake of symmetry, the class also provides the command `\showfancybreak` to force the display of the decorative element, although this is the default behaviour.

B Implementation

This appendix describes the TEX and LATEX code of the class. It is likely to be of interest only to experts, bug hunters or casual readers curious to know how the class is implemented.

B.1 Class setup

B.1.1 Boolean values

We start by defining the boolean values needed throughout the class. In the sequel, the booleans `\ifCJS@nocjs` and `\ifCJS@journalinfo` will be used as negation of one another.

```
%<*class>
\newif\ifCJS@review        \CJS@reviewfalse        % review version?
\newif\ifCJS@supplement    \CJS@supplementfalse    % supplementary material?
\newif\ifCJS@nocjs         \CJS@nocjsfalse         % not a CJS article?
\newif\ifCJS@final         \CJS@finalfalse         % final version?
\newif\ifCJS@titlingfooter \CJS@titlingfooterfalse % footer on title page?
\newif\ifCJS@journalinfo   \CJS@journalinfotrue    % display journal info?
\newif\ifCJS@specialack    \CJS@specialackfalse    % special acknowledgement?
\newif\ifCJS@fancybreak    \CJS@fancybreaktrue     % display fancy break?
\newif\ifCJS@history       \CJS@historyfalse       % history present?
\newif\ifCJS@corrauth      \CJS@corrauthfalse      % corresponding author?
\newif\ifCJS@newauth       \CJS@newauthtrue        % internal; author blocks
```

B.1.2 Options

The class features options for authors and some others targeted more specifically to editors.

`review` (*option*) The class option `review` composes an anonymized version of the article (or the supplementary material) suitable for revision by external referees. It uses a standard footer on the title page.

```
\DeclareOption{review}{%
  \CJS@reviewtrue
  \CJS@titlingfootertrue}
```

`supplement` (*option*) The class option `supplement` enables authors to use the class to prepare the supplementary material to the article. It uses a standard footer on the title page.

```
\DeclareOption{supplement}{%
  \CJS@supplementtrue
  \CJS@titlingfootertrue}
```

`nocjs` (*option*) The class option `nocjs` is a convenience feature for authors who wish to use the class for something else than an article in *The Canadian Journal of Statistics*. It hides any journal-specific information from the manuscript. It also uses a standard footer on the title page.

```
\DeclareOption{nocjs}{%
  \CJS@nocjstrue
  \CJS@titlingfootertrue
  \CJS@journalinfofalse}
```

`final` (*option*) The class option `final` activates sanity checks for the preparation of the final copy of the article, and the display of the licence statement on the title page (when available). The option takes precedence over `review`.

```
\DeclareOption{final}{%
  \CJS@finaltrue
  \CJS@reviewfalse}
```

B.1.3 Underlying document class

The class cjs-rcs-article is based on memoir. We force the options `letterpaper`, `10pt`, `article` and `twoside`.

```
\DeclareOption*{\PassOptionsToClass{\CurrentOption}{memoir}}
\ProcessOptions\relax
\LoadClass[letterpaper,10pt,article,twoside]{memoir}
```

B.1.4 Required packages

The class tries to load as few packages as possible to avoid conflicts. Yet, some are required for standard features of a mathematics oriented journal, or else for internal processing. Useful packages like **iftex** and **etoolbox** are already loaded by memoir.

We rely on **natbib** for author-year citations. This package must be loaded before **babel**. The option `sectionbib` ensures that the bibliography (or list of references) is typeset as a section, rather than as a chapter.

```
\RequirePackage[round,semicolon,authoryear,sectionbib]{natbib}
```

Articles in *The Canadian Journal of Statistics* contain at least two languages, English and French, for the abstracts. Multilingual support is provided by **babel**. The languages are declared with `\documentclass` in the article source file. The package **numprint** prints numbers with a separator every three digits and convert numbers in scientific notation. The package alters its behaviour with the active language declared to **babel**. The option `np` activates the convenient shortcut `\np` for the command `\numprint`.

```
\RequirePackage{babel}
\RequirePackage[autolanguage,np]{numprint}
```

The required packages **bookmark** and **hyperref** must usually be loaded very last. To achieve this within the class, the packages are loaded with the **etoolbox** command `\AtEndPreamble`. Since they require **hyperref** to be loaded, it is also at this point that we define the English and French contextual labels for `\autoref` for all elements defined by the class.

```
\AtEndPreamble{%
  \RequirePackage{bookmark}     % loads hyperref
  \hypersetup{%
    colorlinks = true,
    urlcolor = {CJSurl},
    linkcolor = {CJSlink},
    citecolor = {CJScitation}}
  \bookmarksetup{%
    open = true,
    depth = 3,
    numbered = true}
  \addto\extrasenglish{%
    \def\sectionautorefname{Section}%
    \def\subsectionautorefname{Section}%
    \def\subsubsectionautorefname{Section}%
    \def\figureautorefname{Figure}%
    \def\tableautorefname{Table}%
    \def\theoremautorefname{Theorem}%
    \def\lemmaautorefname{Lemma}%
    \def\propositionautorefname{Proposition}%
    \def\corollaryautorefname{Corollary}%
    \def\definitionautorefname{Definition}%
    \def\algorithmautorefname{Algorithm}%
    \def\remarkautorefname{Remark}}
  \addto\extrasfrench{%
    \def\sectionautorefname{section}%
    \def\subsectionautorefname{section}%
    \def\subsubsectionautorefname{section}%
    \def\figureautorefname{figure}%
```

```
    \def\tableautorefname{tableau}%
    \def\theoremautorefname{théorème}%
    \def\lemmaautorefname{lemme}%
    \def\propositionautorefname{proposition}%
    \def\corollaryautorefname{corollaire}%
    \def\definitionautorefname{définition}%
    \def\algorithmautorefname{algorithme}%
    \def\remarkautorefname{remarque}}}
```

The next set of packages are mostly for internal purposes: **keyval** provides the key-value interface used for `\author`; **enumitem** and **environ** help to define new environments for the titling information; **datetime2** is used to store and format dates from the article history; **xcolor** provides LaTeX extensions to colourize text; **paracol** and **ragged2e** are used to typeset the abstracts in columns, ragged right and with proper hyphenation; **url** is a fairly standard package to enter an URL or email address, used at least here for the corresponding author email; **relsize** provides commands to set font sizes relatives to the current size, like `\smaller` or `\larger`; **fontawesome5** provides a number of symbols, including the ORCID logo and the Creative Commons licence icons.

```
\RequirePackage{keyval}
\RequirePackage[shortlabels]{enumitem}
\RequirePackage{environ}
\RequirePackage[useregional,french,en-GB]{datetime2}
\RequirePackage{xcolor}
\RequirePackage{paracol}
\RequirePackage{ragged2e}
\RequirePackage{url}
\RequirePackage{relsize}
\RequirePackage{fontawesome5}
```

The **graphicx** package is not directly used by the class, but many authors may need it to display graphics and images, so we load it by default.

```
\RequirePackage{graphicx}
```

The **amsmath** and **amsthm** packages are standard requirements for mathematics papers. They are used in the class to define new commands for mathematical symbols and theorem-like environments.

```
\RequirePackage{amsmath}
\RequirePackage{amsthm}
```

Finally, **lineno** is loaded to number lines with option `review`.

```
\ifCJS@review
  \RequirePackage{lineno}
  \linenumbers
\fi
```

Note that some additional packages related to font management are loaded in Section B.2.1.

B.1.5 Incompatible package

The package **geometry** is incompatible with memoir and should not be used in the first place to change the document layout. We issue an error message if users load it in their article.

```
\AtBeginDocument{%
  \@ifpackageloaded{geometry}{%
    \ClassError{cjs-rcs-article}%
      {Package geometry is incompatible with this class}%
      {You should not change the page layout.\MessageBreak%
       If absolutely needed, use the memoir class facilities.}}{\relax}}
```

B.2 Document layout

The document layout of the class is inspired by the layout of the newspaper *Le Devoir* designed by Lucie Lacava in 2018.

B.2.1 Fonts and character encoding

The class uses special fonts for the main text and mathematics, for decorative text (section titles, headers and such), and for text set in monospace (like computer code). The class also imposes Unicode UTF-8 character encoding for the source files.

We need two sets of commands to load the main document fonts depending on the engine used. For the modern engines LuaTeX and XeTeX, commands from **fontspec** and **unicode-math** load the OpenType variants of the fonts. The encoding is UTF-8 by default for these engines.

```
\iftutex
  \RequirePackage[babel=true]{microtype}
  \RequirePackage{fontspec}
  \RequirePackage{unicode-math}
  \setmainfont{STIXTwoText}
  [
    Extension = .otf,
    UprightFont = *-Regular,
    BoldFont = *-SemiBold,
    ItalicFont = *-Italic,
    BoldItalicFont = *-SemiBoldItalic,
    Scale = 1,
    Ligatures = TeX
  ]
  \setmathfont{STIXTwoMath-Regular}
  [
    Extension = .otf,
    Scale = 1,
    bold-style = TeX
  ]
  \setmathfont{FiraMath-Regular}
  [
    version = titles,
    Extension = .otf,
    Scale = 0.92
  ]
```

For the traditional engine pdfTeX, we need to rely on ancillary packages and the Type 1 variants of the fonts. These are no longer updated, in contrast to their OpenType counterparts. Furthermore, the Type 1 variant of STIX Two does not contain the semibold weight.

```
\else
  \RequirePackage[babel=true]{microtype}
  \RequirePackage[utf8]{inputenc}
  \RequirePackage[T1]{fontenc}
  \RequirePackage{stix2}
\fi
```

For the sans serif font, the packages **FiraSans** and **FiraMono** provide a unified interface for the OpenType and Type 1 versions of the fonts.

```
\RequirePackage[book,medium,proportional,lining,scale=0.92]{FiraSans}
\RequirePackage[medium,lining,scale=0.90]{FiraMono}
```

With option `review`, we redefine the font for the line numbers and position them closer to the left margin (default is 10 pt) to avoid overlap in abstracts.

```
\ifCJS@review
  \renewcommand*\linenumberfont{\normalfont\firalining\tiny}
  \setlength\linenumbersep{5pt}
\fi
```

B.2.2 Colours

We define some colours used by the class. First, the three main colours of *The Canadian Journal of Statistics* logo, hand-picked from the SVG code of the latter: .

```
\definecolor{CJSpink}{HTML}{E3038B}
\definecolor{CJSred}{HTML}{D10000}
\definecolor{CJSyellow}{HTML}{FEC70D}
```

Next, the colours for the hyperlinks (inspired by **classicthesis**): for internal links; for external links; for citations.

```
\definecolor{CJSlink}{rgb}{0,0.4,0.6}   % internal links
\definecolor{CJSurl}{rgb}{0.6,0,0}      % external links
\definecolor{CJScitation}{rgb}{0,0.5,0} % citations
```

Finally, the colour of the ORCID icon: .

```
\definecolor{CJSorcid}{HTML}{A6CE39}
```

B.2.3 Page size and margins

The page size was set to letter paper in the options passed to memoir. We use the very nice commands of memoir to set the margins. Their size is determined such that the typeblock has a height to width aspect ratio of approximately $\sqrt{2}:1$, the same as ISO A series paper.

```
\setlrmarginsandblock{24mm}{30mm}{*}
\setulmarginsandblock{24mm}{26mm}{*}
\setheadfoot{13.5pt}{9mm}
\checkandfixthelayout[nearest]
```

B.2.4 Journal name and publication information

The title page starts with the journal name and publication information set next to a decorative element (called “ornament” in the code).

First, define macros containing the English and French names of the journal, and another one that displays both.

```
\def\CJS@journalname@en{The Canadian Journal of Statistics}
\def\CJS@journalname@fr{La revue canadienne de statistique}
\def\CJS@journalname{\CJS@journalname@en~{\textbar}~\CJS@journalname@fr}
```

The decorative element is created using plain TEX rules. Its height and depth are identical to the journal information.

```
\newsavebox\CJS@journalornament
\setbox\CJS@journalornament=\vbox{%
  \setbox\z@=\vbox{\sffamily\hbox{\CJS@journalname}\hbox{\bfseries V,}}
  \setbox\tw@\hbox{\vrule \@width4pt\@height\ht\z@\@depth\dp\z@}
  \hbox{\textcolor{CJSpink}{\copy\tw@}\kern2pt
        \textcolor{CJSred}{\copy\tw@}\kern2pt
        \textcolor{CJSyellow}{\copy\tw@}\kern6pt}}
```

`\makejournalinfo` The command `\makejournalinfo` puts the journal information together. The actual information is provided through the editor commands defined in Section B.11. Although a visible command, `\makejournalinfo` is actually called by `\maketitle`.

```
\newcommand*\makejournalinfo{%
  \mbox{%
    \box\CJS@journalornament
    \vbox{%
      \sffamily
      \hbox{\CJS@journalname}
      \hbox{\bfseries Vol.~\CJS@vol, %
                      No.~\CJS@issue, %
                      \CJS@year, %
                      \CJS@firstpage--\thelastpage}}}}
```

B.2.5 Headers and footers

We need to define headers and footers for the internal pages, and specific ones for the title page. They are built from scratch using `tabular` environments.

First, a command to set the font.

```
\def\CJS@setheadfootfont{%
  \firalining\fontsize{9}{13}\selectfont}
```

Next, a command to draw the vertical rule that will be used as a separator between the header or footer content and the page folio.

```
\newlength\CJS@headdivheight
\settoheight\CJS@headdivheight{\CJS@setheadfootfont 1}
\setlength\CJS@headdivheight{0.9\CJS@headdivheight}
\def\CJS@headdivrule{%
  \rule[0.05\CJS@headdivheight]{0.75\normalrulethickness}{\CJS@headdivheight}}
```

Now, the page style `cjs` for the internal pages. The even header contains the names of the authors flush left. The odd header contains the article title (short version, if appropriate) flush right.

```
\newlength\CJS@pagenumwidth
\settowidth\CJS@pagenumwidth{9\qquad}
\makepagestyle{cjs}
\makeevenhead{cjs}{%
  \CJS@setheadfootfont
  \begin{tabular*}{\textwidth}%
    {@{}p{\CJS@pagenumwidth}@{\CJS@headdivrule\qquad}l@{\extracolsep\fill}}
    \thepage & \ifCJS@review\else\MakeTextUppercase{\CJS@runningauthor}\fi \\ \hline
  \end{tabular*}}{}{}
\makeoddhead{cjs}{}{}{%
  \CJS@setheadfootfont
  \begin{tabular*}{\textwidth}%
    {l@{\extracolsep\fill}r@{\extracolsep{0pt}\qquad\CJS@headdivrule}p{\CJS@pagenumwidth}@{}}
    & \MakeTextUppercase{\CJS@shorttitle} & \hfill\thepage \\ \hline
  \end{tabular*}}
```

The footer for internal pages is empty.

```
\makeevenfoot{cjs}{}{}{} \makeoddfoot{cjs}{}{}{}
```

Use the `cjs` page style.

```
\pagestyle{cjs}
```

We need a special page style for the title page. The header of the title page is empty.

```
\makepagestyle{cjsfirstpage}
\makeevenhead{cjsfirstpage}{}{}{} \makeoddhead{cjsfirstpage}{}{}{}
```

With the `review`, `statement` and `nocjs` options, the title page features a footer with the page folio. With option `nocjs`, the footer may also contain a licence statement specified with `\licence`; see Section B.5.

```
\ifCJS@titlingfooter
  \makeevenfoot{cjsfirstpage}{%
    \CJS@setheadfootfont
    \begin{tabular*}{\textwidth}%
      {@{}p{\CJS@pagenumwidth}@{\CJS@headdivrule\qquad}l@{\extracolsep\fill}}
      \hline \thepage &
      \ifCJS@nocjs\CJS@licence\fi
    \end{tabular*}}{}{}
  \makeoddfoot{cjsfirstpage}{}{}{%
    \CJS@setheadfootfont
    \begin{tabular*}{\textwidth}%
      {l@{\extracolsep\fill}r@{\extracolsep{0pt}\qquad\CJS@headdivrule}p{\CJS@pagenumwidth}@{}}
      \hline &
      \ifCJS@nocjs\CJS@licence\fi &
      \hfill\thepage
    \end{tabular*}}
\fi
```

B.2.6 Section titles

The memoir class assumes that sectioning starts at the chapter level—even with the `article` option. To get around this requirement, we disable the effect of `\chapter`. A desirable side effect is to make `\autoref` work correctly for appendices.

```
\let\chapter\relax
\counterwithout{section}{chapter}
```

The class uses a numbering system of at most three levels.

```
\setsecnumdepth{subsubsection}
\maxsecnumdepth{subsubsection}
```

We then define the style of section titles from level `\section` down to `\subparagraph`. With the LuaTEX and XETEX engines, we make sure to use a matching font for mathematics in case some appear in the titles.

```
\iftutex
  \setsecheadstyle{\normalfont\Large\sffamily\mathversion{titles}\bfseries\raggedright}
  \setsubsecheadstyle{\normalfont\large\sffamily\mathversion{titles}\bfseries\raggedright}
  \setsubsubsecheadstyle{\normalfont\sffamily\mathversion{titles}\bfseries\raggedright}
  \setparaheadstyle{\normalfont\sffamily\mathversion{titles}\bfseries}
  \setsubparaheadstyle{\normalfont\mathversion{titles}\bfseries}
\else
  \setsecheadstyle{\normalfont\Large\sffamily\bfseries\raggedright}
  \setsubsecheadstyle{\normalfont\large\sffamily\bfseries\raggedright}
  \setsubsubsecheadstyle{\normalfont\sffamily\bfseries\raggedright}
  \setparaheadstyle{\normalfont\sffamily\bfseries}
  \setsubparaheadstyle{\normalfont\bfseries}
\fi
\setbeforesecskip{-1.75ex plus -0.5ex minus -0.1ex}
\setaftersecskip{1.2ex plus 0.1ex}
\setbeforesubsecskip{-1.5ex plus -0.5ex minus -0.1ex}
\setaftersubsecskip{0.75ex plus 0.1ex}
\setbeforesubsubsecskip{-1.5ex plus -0.5ex minus -0.1ex}
\setaftersubsubsecskip{0.75ex plus 0.1ex}
\setbeforeparaskip{0pt}
\setafterparaskip{-0.5em}
\setbeforesubparaskip{0pt}
\setaftersubparaskip{-0.25em}
```

B.2.7 Line spacing

With option `review`, line spacing is increased using the command `\OnehalfSpacing` of memoir.

```
\ifCJS@review
  \OnehalfSpacing
\fi
```

B.2.8 Captions

The caption name for tables and figures is typeset in bold.

```
\captionnamefont{\bfseries}
```

B.2.9 Configuration for French

We change some of the defaults of **babel-french** and allow French opening and closing quotes (« ») to be entered directly with the keyboard.

```
\frenchsetup{%
  SmallCapsFigTabCaptions=false,
  ThinSpaceInFrenchNumbers=true,
  og=«, fg=»}
\renewcommand*\frenchtablename{Tableau}
\renewcommand*\frenchfigurename{Figure}
```

B.3 Titling material

Before we dive into the definition of the commands to enter titling information, we first define the various lengths that will be needed to position the elements on the title page.

\journalinfoskip (ℓ) The length \journalinfoskip holds the distance between the top of the typeblock and the journal information.

```
\newlength\journalinfoskip
\setlength\journalinfoskip{-4ex}
```

\maintitleskip (ℓ) The length \maintitleskip holds the distance between the journal information and the start of the main title area.

```
\newlength\maintitleskip
\setlength\maintitleskip{6ex}
```

\supptitleskip (ℓ) The length \supptitleskip holds the distance between the supplementary material statement and the main title of the article.

```
\newlength\supptitleskip
\setlength\supptitleskip{2ex}
```

\authorskip (ℓ) The length \authorskip holds the distance between the main title and the author block.

```
\newlength\authorskip
\setlength\authorskip{5ex}
```

\titlingskip (ℓ) The length \titlingskip holds the distance between the other elements of the title page, that is, from the abstract to the corresponding author.

```
\newlength\titlingskip
\setlength\titlingskip{2pt plus2pt minus0.8pt}
```

\fancybreaksep (ℓ) The length \fancybreaksep holds the distance between the fancy break and the surrounding text.

```
\newlength\fancybreaksep
\setlength\fancybreaksep{6pt plus3pt minus1pt}
```

B.3.1 Article title

The main title area is composed of two elements: the statement that the document is supplementary material to an article when the option supplement is specified, and the main title of the article.

\supplementfont The public command \supplementfont defines the font of the supplementary material title.

```
\def\CJS@supptitlename@en{supplementary material for the article}
\def\CJS@supptitlename@fr{matériel additionnel pour l'article}
\newcommand*\supplementfont{\normalfont\Large\scshape}
```

\maintitlefont The public command \maintitlefont defines the font of the main title.

```
\newcommand*\maintitlefont{\raggedright\normalfont\huge\bfseries}
```

\title The implementation of \title, copied from amsart.cls, allows an optional argument to provide the short title that is displayed in the running head.

```
\renewcommand*\title[2][]{\gdef\CJS@shorttitle{#1}\gdef\@title{#2}}
\edef\title{\@nx\@dblarg
  \@xp\@nx\csname\string\title\endcsname}
```

B.3.2 Author and affiliation information

The mechanism to enter the author and affiliation information is inspired from **authblk**, but the implementation is rewritten more legibly using the tools from **etoolbox**.

`\authfont` `\affilfont` First, we define two public commands to control the visual style of author names and affiliations, respectively.

```
\newcommand*\authfont{\normalfont\sffamily\bfseries\color{black}}
\newcommand*\affilfont{\normalfont\sffamily\color{gray}}
```

`\affilsep` (ℓ) `\authsep` (ℓ) Next, we define (and initialize) two lengths holding the space between the author identification and the affiliation, and between author blocks.

```
\newlength\affilsep \setlength\affilsep{\z@}
\newlength\authsep  \setlength\authsep{1ex}
```

`\author` We can now define a new version of `\author` that will inject the author names into `\@author`, along with styling instructions and the appropriate separator between the author blocks (none before the first author, a space `\authsep` between the following ones).

The command also allows collection of the ORCID iD, email address and corresponding author status through a now fairly standard key-value interface.

```
\define@key{author}{orcid}{\def\CJS@orcid{#1}}
\define@key{author}{email}{\def\CJS@email{#1}}
\define@key{author}{corresponding}[true]{%
  \csname if#1\endcsname \CJS@corrauthtrue\fi}
\def\CJS@corrauthname@en{Corresponding author}
\def\CJS@corrauthname@fr{Auteur correspondant}
\newcommand*\CJS@corrauth{}
\newcommand*\CJS@orcidlist{}
\newcommand*\CJS@orcidname{ORCID}
\newcommand*\CJS@authsep{, }
\newcommand*\CJS@orcid@and{}
\newcommand*\CJS@blk@and{\protect\authfont}
\newcommand*\CJS@pand{\protect\\[\authsep] \protect\authfont}
\renewcommand\author{%
  \new@ifnextchar[{\author@}%
                  {\author@[]}}
\newcommand*\author@{}
\def\author@[#1]#2{%
  \begingroup
    \setkeys{author}{#1}
    \let\protect\@unexpandable@protect
    \ifCJS@newauth
      \xappto\@author{\CJS@blk@and#2}
    \else
      \xappto\@author{\CJS@authsep#2}
    \fi
    \ifCJS@corrauth
      \ifundef{\CJS@email}{\relax}{%
        \protected@xdef\CJS@corrauth{\protect\url{\CJS@email}}
        \xappto\@author{\;%
          \protect\href{mailto:\CJS@email}{\corrauthmark}}}
    \fi
    \ifundef{\CJS@orcid}{\relax}{%
      \xappto\@author{\;%
        \protect\href{https://orcid.org/\CJS@orcid}%
                     {\textcolor{CJSorcid}{\faOrcid}}}
      \xappto\CJS@orcidlist{\CJS@orcid@and#2: %
        \protect\href{https://orcid.org/\CJS@orcid}%
                     {\textcolor{CJSorcid}{\faOrcid}\, \protect\url{https://orcid.org/\CJS@orcid}}}
```

```
      \gdef\CJS@orcid@and{\protect\\}}
  \endgroup
  \CJS@newauthfalse
}
```

\affil We immediately define the command \affil since it works in tandem with \author. The command provides an affiliation for the author previously mentioned in \author. Note how the separator \CJS@blk@and is redefined here.

```
\newcommand*\affil[1]{%
  \CJS@newauthtrue
  \let\CJS@blk@and\CJS@pand
  \begingroup
    \let\protect\@unexpandable@protect
    \gappto\@author{\protect\\[\affilsep]\protect\affilfont #1}
  \endgroup
}
```

\surname The command \surname defines the styling of the surname (using \CJS@printsurname), but also collects the surnames in the \CJS@runningauthor macro with the appropriate separators. This will be printed in the running head.

The command \surname is expanded by \maketitle. We need to take into account that \runningauthors (see below) may be used in the document, in which case the command should not collect the surnames in \CJS@runningauthor. When \runningauthors is used, \CJS@runningauthor is not empty when \maketitle is expanded and, therefore, \CJS@runningauthor@and is never defined by \surname. Hence the test with \ifvoiddef in the "false" branch of \ifdefempty.

```
\newcommand*\CJS@runningauthor{}
\newcommand*\CJS@printsurname[1]{\MakeTextUppercase{#1}}
\newrobustcmd{\surname}[1]{%
  \ifdefempty{\CJS@runningauthor}{%
    \def\CJS@runningauthor@{#1}%
    \edef\CJS@runningauthor@and{\iflanguage{french}{ et }{ and }}%
    \gappto\CJS@runningauthor{#1}}{%
      \ifdefvoid{\CJS@runningauthor@and}{\relax}{%
        \global\let\CJS@runningauthor\CJS@runningauthor@
        \appto\CJS@runningauthor@{, #1}%
        \xappto\CJS@runningauthor{\CJS@runningauthor@and#1}%
        \edef\CJS@runningauthor@and{\iflanguage{french}{ et }{, and }}}}%
  \CJS@printsurname{#1}%
}
```

\corrauthmark The command \corrauthmark contains the marker that is used as a hyperlink next to the name of the corresponding author, and as the mark next to the titling information.

```
\newcommand*\corrauthmark{{\small\faAt}}
```

\runningauthor The command \runningauthors may be used to provide an alternative (shorter) version of the author names for the running head.

```
\newcommand*\runningauthor[1]{\renewcommand*\CJS@runningauthor{#1}}
```

Finally, we need to reset \@author to avoid `No \author given` warnings.

```
\def\@author{}
```

B.3.3 Abstracts

englishabstract (*env.*) frenchabstract (*env.*) The environments for the English and French abstracts are defined using the facilities of the package **environ**. They each merely collect their body into a macro whose content is later typeset by \maketitle.

```
\def\CJS@englishabstract{}
\def\CJS@frenchabstract{}
\NewEnviron{englishabstract}{\global\let\CJS@englishabstract\BODY}
\NewEnviron{frenchabstract}{\global\let\CJS@frenchabstract\BODY}
```

B.3.4 Keywords

keywords (*env.*) The treatment of keywords involves two environments. The first, `keywords`, is the visible one that collects the information into a macro. The second, `CJS@keywordslist`, is the (inline) environment based on `itemize` that actually typesets the information inside `\maketitle`. (The command `\newlist` comes from **enumitem**.)

```
\def\CJS@keywords{}
\def\CJS@keywordsname@en{Keywords}
\def\CJS@keywordsname@fr{Mots-clés}
\NewEnviron{keywords}{\global\let\CJS@keywords\BODY}
\newlist{CJS@keywordslist}{itemize*}{1}
\setlist[CJS@keywordslist]{%
  mode=unboxed,
  label={},
  afterlabel={},
  itemjoin={{; }},
  after={.}
}
```

B.3.5 Mathematics subject classification

classification (*env.*) The implementation of the environment `environment` is similar to `keywords`, the main difference being that this environment has an optional argument with a default value stored in the macro `\CJS@defclassifname`.

```
\def\CJS@classif{}
\def\CJS@classifname{}
\def\CJS@defclassifname{MSC2020}
\NewEnviron{classification}[1][%
  \CJS@defclassifname]{%
  \global\def\CJS@classifname{#1}
  \global\let\CJS@classif\BODY}
\newlist{CJS@classiflist}{itemize*}{1}
\setlist[CJS@classiflist]{%
  mode=unboxed,
  itemjoin={{; }},
  after={.}
}
```

B.3.6 Fancy break

\pfbreakdisplay The fancy break is the three-rule separator between the titling information and the body of the paper. Using the facilities of memoir, we simply redefine the command `\pfbreakdisplay`.

```
\renewcommand\pfbreakdisplay{%
  \vskip\fancybreaksep\par%
  \rule[\z@]{\linewidth}{\normalrulethickness}\hskip-\linewidth
  \rule[4\normalrulethickness]{\linewidth}{\normalrulethickness}\hskip-\linewidth
  \rule[8\normalrulethickness]{\linewidth}{\normalrulethickness}\par%
  \vskip\fancybreaksep}
```

\suppressfancybreak \showfancybreak The command `\suppressfancybreak` allows suppression of the fancy break should it appear at the very bottom of the title page or at the very top of the second page. The converse `\showfancybreak` forces display, although it should not be needed as it is the default behaviour of the class.

```
\newcommand*\suppressfancybreak{\CJS@fancybreakfalse}
\newcommand*\showfancybreak{\CJS@fancybreaktrue}
```

B.3.7 Typesetting of the titling material

\maketitle The command `\maketitle` typesets the titling material. Single spacing is turned on, top floats are suppressed from the title page (we still need bottom floats for the licence statement; see below), and paragraph spacing and indentation is set to zero.

```
\renewcommand\maketitle{{%
  \SingleSpacing
  \suppressfloats[t]
  \setlength\parskip{\z@}
  \setlength\parindent{\z@}
  \setcounter{page}{\CJS@firstpage}
  \thispagestyle{cjsfirstpage}
```

The decorative element and journal information is hidden with option `nocjs`.

```
  \ifCJS@journalinfo
    \vspace*{\journalinfoskip}
    \makejournalinfo\par
  \fi
  \vspace*{\maintitleskip}
```

We need to know the height of the supplementary material statement to replace it by a strut for regular articles.

```
  \setbox\z@=\vbox{\supplementfont%
    \iflanguage{french}{\CJS@supptitlename@fr}{\CJS@supptitlename@en}}
  \ifCJS@supplement
    \box\z@\par
  \else
    \rule{\z@}{\dimexpr\ht\z@}\par % strut
  \fi
  \vskip\supptitleskip
  \begin{minipage}{0.9\textwidth}
    \maintitlefont\@title\par
  \end{minipage}
  \vskip\authorskip
```

The author block is displayed when the option `review` is not specified.

```
  \ifCJS@review\else
    \hrulefill
    \vskip\belowrulesep
    \begin{minipage}{0.9\textwidth}
      \@author
    \end{minipage}\par
    \vskip\aboverulesep
    \hrulefill\par
  \fi
```

The rest of the titling material, except the corresponding author address, is hidden with option `supplement`.

```
  \ifCJS@supplement\else
    \vskip2\titlingskip
```

The abstracts are positioned automatically depending on the main language of the article.

```
    \begin{paracol}{2}
      \RaggedRight
      \switchcolumn[\iflanguage{english}{0}{1}]
      \begingroup
        \selectlanguage{english}\paragraph{\abstractname}\CJS@englishabstract
      \endgroup
      \switchcolumn[\iflanguage{french}{0}{1}]
      \begingroup
        \selectlanguage{french}\paragraph{\abstractname}\CJS@frenchabstract
      \endgroup
    \end{paracol}\par
```

Start a group for the keywords and subject classification that are typeset ragged right.

```
    \begingroup
      \sloppy\RaggedRight
```

```
      \ifdefempty{\CJS@keywords}{\relax}{%
        \vskip\titlingskip
        \paragraph{\iflanguage{french}{\CJS@keywordsname@fr}{\CJS@keywordsname@en}}
        \begin{CJS@keywordslist}
          \CJS@keywords
        \end{CJS@keywordslist}\par}
      \vskip\titlingskip
```

The subject classification block is always shown, just with an empty content if the information is not provided.

```
      \paragraph{{\CJS@classifname}}
      \ifdefempty{\CJS@classif}{\mbox{}}{%
        \begin{CJS@classiflist}
          \CJS@classif
        \end{CJS@classiflist}}\par
    \endgroup
  \fi
```

The corresponding author address is only hidden with option `review`.

```
  \unless\ifCJS@review
    \ifdefempty{\CJS@corrauth}{\relax}{%
      \vskip\titlingskip
      \paragraph{\corrauthmark\;%
        \iflanguage{french}{\CJS@corrauthname@fr}{\CJS@corrauthname@en}}
      \CJS@corrauth\par}
  \fi
  \ifCJS@fancybreak
    \fancybreak{\pfbreakdisplay}
  \fi
```

Finally, unless overridden by a class option, allocate space for the licence statement at the bottom of the title page. By default, a notice indicates that the licence will appear here. With option `final`, the licence statement is displayed, if provided.

```
  \unless\ifCJS@titlingfooter
    \CJS@measurelicence
    \setlength\textfloatsep{0pt}
    \def\CJS@notice{%
      \iflanguage{french}{Futur emplacement de la déclaration de licence.}%
                         {Licence statement to appear here.}}
    \begin{figure}[!b]
      \raisebox{0pt}[0pt][\dimexpr\CJS@licenceht-\footskip\relax]{%
        \vbox{%
          \CJS@setlicencefont
          \rule{\textwidth}{\normalrulethickness}
          \parbox[t][\CJS@licenceht][t]{\textwidth}{%
            \ifCJS@specialack\CJS@specialackstatement\fi
            \ifCJS@final\CJS@licence \else\textit\CJS@notice \fi}}}
    \end{figure}
  \fi
}}
```

B.3.8 ADNI data use agreement

`\ADNIacknowledgement` `\ADMCacknowledgement`

The commands `\ADNIacknowledgement` and `\ADMCacknowledgement` automatically create special acknowledgements for the Alzheimer's Disease Neuroimaging Initiative (ADNI) and the Alzheimer's Disease Metabolomics Consortium (ADMC), respectively.

```
\newcommand*\CJS@adnibyline{for the Alzheimer's Disease Neuroimaging Initiative}
\newcommand*\CJS@adnistatement{%
  Data used in preparation of this article were obtained
  from the \href{https://adni.loni.usc.edu}{Alzheimer's Disease Neuroimaging Initiative}
  (ADNI) database. As such, the investigators within the ADNI
```

```
  contributed to the design and implementation of ADNI and/or provided
  data but did not participate in the analysis or writing of this
  report. A complete listing of ADNI investigators can be found at:
  \url{https://adni.loni.usc.edu/wp-content/uploads/how_to_apply/ADNI_Acknowledgement_List.pdf}.}
\newcommand*\CJS@admcbyline{for the Alzheimer's Disease Metabolomics Consortium}
\newcommand*\CJS@admcstatement{%
  Data used in preparation of this article were generated by the
  Alzheimer's Disease Metabolomics Consortium (ADMC). As such, the
  investigators within the ADMC provided data but did not participate
  in analysis or writing of this report. A complete listing of ADMC
  investigators can be found at:
  \url{https://sites.duke.edu/adnimetab/team/}.}
\newcommand*\ADNIacknowledgement{%
  \specialack{\CJS@adnibyline}{\CJS@adnistatement}}
\newcommand*\ADMCacknowledgement{%
  \renewcommand*\CJS@specialackmark{\textsuperscript{\ensuremath{**}}}
  \specialack{\CJS@admcbyline}{\CJS@admcstatement}}
```

B.4 Back matter

Perhaps a word is in order to explain the presence of the back matter environments in the class. The funding information started its life on the title page. It was therefore natural to provide an environment `funding` along with the environments `keywords` and `classification`.

It was later decided to move this information at the end of the article, where space is less at a premium. We could have just included the appropriate sectioning commands in the templates, but it turned out that providing a command `\makebackmatter` proved useful to impose the order of the back matter and, furthermore, to display the full ORCID iDs, when available. The other environments of this section followed.

`supplement` (*env.*)
`sharing` (*env.*)

The environments `supplement` and `sharing` only collect their contents into macros.

```
\def\CJS@supplement{}
\def\CJS@supplementname@en{Supplementary material}
\def\CJS@supplementname@fr{Matériel additionnel}
\NewEnviron{supplement}{\global\let\CJS@supplement\BODY}

\def\CJS@sharing{}
\def\CJS@sharingname@en{Data sharing}
\def\CJS@sharingname@fr{Partage des données}
\NewEnviron{sharing}{\global\let\CJS@sharing\BODY}
```

`acknowledgements` (*env.*)
`acknowledgments` (*env.*)
`funding` (*env.*)

The environments `acknowledgements` (or its alias `acknowledgments`) and `funding` are very similar, except that they accept an optional argument to change the name of the information provided.

```
\def\CJS@ack{}
\def\CJS@ackname@en{Acknowledgements}
\def\CJS@ackname@fr{Remerciements}
\NewEnviron{acknowledgements}{\global\let\CJS@ack\BODY}
\let\acknowledgments\acknowledgements
\let\endacknowledgments\endacknowledgements

\def\CJS@funding{}
\def\CJS@fundingname{}
\def\CJS@fundingname@en{Funding information}
\def\CJS@fundingname@fr{Financement}
\NewEnviron{funding}[1][%
  \iflanguage{french}{\CJS@fundingname@fr}{\CJS@fundingname@en}]{%
    \global\def\CJS@fundingname{#1}
    \global\let\CJS@funding\BODY}
```

`\makebackmatter` Similar to `\maketitle`, the command `\makebackmatter` typesets all elements of the back matter that are available at the point where it appears.

```
\newcommand\makebackmatter{%
  \ifCJS@supplement\else
    \ifdefempty{\CJS@supplement}{\relax}{%
      \section*{\iflanguage{french}{\CJS@supplementname@fr}{\CJS@supplementname@en}}
      \CJS@supplement}
    \ifdefempty{\CJS@sharing}{\relax}{%
      \section*{\iflanguage{french}{\CJS@sharingname@fr}{\CJS@sharingname@en}}
      \CJS@sharing}
    \ifCJS@review\else
      \ifdefempty{\CJS@ack}{\relax}{%
        \section*{\iflanguage{french}{\CJS@ackname@fr}{\CJS@ackname@en}}
        \CJS@ack}
      \ifdefempty{\CJS@funding}{\relax}{%
        \section*{\CJS@fundingname}
        \CJS@funding}
      \ifdefempty{\CJS@orcidlist}{\relax}{%
        \section*{\CJS@orcidname}
        \begingroup
          \let\surname\relax
          \let\corrauth\@gobble
          \CJS@orcidlist
        \endgroup}
    \fi
  \fi}
```

B.5 Licensing

The licensing information is used in two main ways by the class: either the licence statement provided by the publisher is inserted at the bottom of the title page with option `final`, or else the user provided licence statement is displayed in the footer of the title page with option `nocjs`. In all other cases the licensing information is ignored.

In order to maintain the page layout as much as possible, we leave space by default at the bottom of the title page for the future licence statement.

First, set the font for the licence statement.

```
\def\CJS@setlicencefont{%
  \firalining\fontsize{8pt}{9pt}\selectfont}
```

`\CJS@measurelicence` We need to compute the required space for a licence statement and, possibly, the ADNI statement. The internal command `\CJS@measurelicence` measures the total height required using either the licence statement provided with `\licence`, or the seemingly longest version of the license statement used by the publisher. The command is used inside `\maketitle`.

```
\newlength\CJS@licenceht
\def\CJS@licencesample{%
  {\ccbyncnd*} 2023 The Authors. The Canadian Journal of
  Statistics~{\textbar}~La revue canadienne de statistique published
  by Wiley Periodicals LLC on behalf of Statistical Society of
  Canada~{\textbar}~Société statistique du Canada. This is an open
  access article under the terms of the Creative Commons
  Attribution-NonCommercial-NoDerivs License, which permits use and
  distribution in any medium, provided the original work is properly
  cited, the use is non-commercial and no modifications or adaptations
  are made.}
\newsavebox\CJS@licencebox
\newcommand\CJS@measurelicence{%
  \setbox\CJS@licencebox=\vbox{%
    \CJS@setlicencefont
    \ifCJS@specialack\CJS@specialackstatement\fi
    \ifdefempty{\CJS@licence}{\CJS@licencesample}{\CJS@licence}}%
  \setlength\CJS@licenceht{\dimexpr\ht\CJS@licencebox+\dp\CJS@licencebox}}
```

\licence
\license

The command \licence (alias \license) specifies the licence statement for the manuscript. It may be used by the editors or the publisher to insert the statement at the bottom of the title page with option final. Otherwise, the statement is only displayed in the footer with option nocjs.

```
\def\CJS@licence{}
\newcommand\licence[1]{\renewcommand*\CJS@licence{#1}}
\let\license\licence
```

For convenience, we define commands to display the icons of the most common Creative Commons licences. The starred versions add the Font Awesome copyright icon © at the end. Inspired by the package **ccicons** (not used by the class since **fontawesome5** contains all the Creative Commons icons).

\ccby
\ccby*

The command \ccby typesets the icons for the Attribution licence: .

```
\newcommand*\ccby{\ccby@\@ifstar{~\faCopyright[regular]}\relax}
\def\CJS@fakern{\kern0.1em}
\def\ccby@{\mbox{%
    \faCreativeCommons\CJS@fakern
    \faCreativeCommonsBy\CJS@fakern}}
```

\ccbysa
\ccbysa*

The command \ccbysa typesets the icons for the Attribution-ShareAlike licence: .

```
\newcommand*\ccbysa{\ccbysa@\@ifstar{~\faCopyright[regular]}\relax}
\def\ccbysa@{\mbox{%
    \faCreativeCommons\CJS@fakern
    \faCreativeCommonsBy\CJS@fakern
    \faCreativeCommonsSa}}
```

\ccbync
\ccbync*

The command \ccbync typesets the icons for the Attribution-NonCommercial licence: .

```
\newcommand*\ccbync{\ccbync@\@ifstar{~\faCopyright[regular]}\relax}
\def\ccbync@{\mbox{%
    \faCreativeCommons\CJS@fakern
    \faCreativeCommonsBy\CJS@fakern
    \faCreativeCommonsNc}}
```

\ccbyncsa
\ccbyncsa*

The command \ccbyncsa typesets the icons for the Attribution-NonCommercial-ShareAlike licence: .

```
\newcommand*\ccbyncsa{\ccbyncsa@\@ifstar{~\faCopyright[regular]}\relax}
\def\ccbyncsa@{\mbox{%
    \faCreativeCommons\CJS@fakern
    \faCreativeCommonsBy\CJS@fakern
    \faCreativeCommonsNc\CJS@fakern
    \faCreativeCommonsSa}}
```

\ccbynd
\ccbynd*

The command \ccbynd typesets the icons for the Attribution-NoDerivatives licence: .

```
\newcommand*\ccbynd{\ccbynd@\@ifstar{~\faCopyright[regular]}\relax}
\def\ccbynd@{\mbox{%
    \faCreativeCommons\CJS@fakern
    \faCreativeCommonsBy\CJS@fakern
    \faCreativeCommonsNd}}
```

\ccbyncnd
\ccbyncnd*

The command \ccbyncnd typesets the icons for the Attribution-NonCommercial-NoDerivatives licence: .

```
\newcommand*\ccbyncnd{\ccbyncnd@\@ifstar{~\faCopyright[regular]}\relax}
\def\ccbyncnd@{\mbox{%
    \faCreativeCommons\CJS@fakern
    \faCreativeCommonsBy\CJS@fakern
    \faCreativeCommonsNc\CJS@fakern
    \faCreativeCommonsNd}}
```

B.6 Mathematics

We provide a number of shortcuts for the more common mathematical operators.

```
\let\Pr\relax
\DeclareMathOperator{\Pr}{P}
\DeclareMathOperator{\E}{E}
\DeclareMathOperator{\Var}{var}
\DeclareMathOperator{\Cov}{cov}
\DeclareMathOperator{\corr}{corr}
```

The command `\prdist` typesets the symbol or the name of a probability distribution. With the newer TEX engines that load **unicode-math**, `\symcal` is recommended over `\mathcal`.

```
\iftutex
  \let\prdist\symcal
\else
  \let\prdist\mathcal
\fi
```

The class provides special commands to typeset matrix and vector names, a carefully designed transpose symbol, and the most common linear algebra operators.

```
\iftutex
  \let\mat\symbf
  \let\matit\symbfit
  \newcommand*\trsp{^{\mkern-1.5mu\symsfup{T}}}
\else
  \let\mat\mathbf
  \let\matit\mathbfit
  \newcommand*\trsp{^{\mkern-1.5mu\mathsf{T}}}
\fi
\DeclareMathOperator{\tr}{tr}
\DeclareMathOperator{\diag}{diag}
```

To ensure uniformity, the class provides commands to typeset the sets of natural numbers, integers, rational numbers, real numbers, and complex numbers.

```
\iftutex
  \newcommand*\Nset{\symbb{N}}
  \newcommand*\Zset{\symbb{Z}}
  \newcommand*\Qset{\symbb{Q}}
  \newcommand*\Rset{\symbb{R}}
  \newcommand*\Cset{\symbb{C}}
\else
  \newcommand*\Nset{\mathbb{N}}
  \newcommand*\Zset{\mathbb{Z}}
  \newcommand*\Qset{\mathbb{Q}}
  \newcommand*\Rset{\mathbb{R}}
  \newcommand*\Cset{\mathbb{C}}
\fi
```

The class defines a number of theorem-like environments using the **amsthm** styles.

```
\def\CJS@theoremname@en{Theorem}
\def\CJS@theoremname@fr{Théorème}
\def\CJS@lemmaname@en{Lemma}
\def\CJS@lemmaname@fr{Lemme}
\def\CJS@propositionname@en{Proposition}
\def\CJS@propositionname@fr{Proposition}
\def\CJS@corollaryname@en{Corollary}
\def\CJS@corollaryname@fr{Corollaire}
\def\CJS@definitionname@en{Definition}
\def\CJS@definitionname@fr{Définition}
```

```
\def\CJS@algorithmname@en{Algorithm}
\def\CJS@algorithmname@fr{Algorithme}
\def\CJS@remarkname@en{Remark}
\def\CJS@remarkname@fr{Remarque}
\theoremstyle{plain}
\newtheorem{theorem}{\iflanguage{french}{\CJS@theoremname@fr}{\CJS@theoremname@en}}
\newtheorem{lemma}{\iflanguage{french}{\CJS@lemmaname@fr}{\CJS@lemmaname@en}}
\newtheorem{proposition}{\iflanguage{french}{\CJS@propositionname@fr}{\CJS@propositionname@en}}
\newtheorem{corollary}{\iflanguage{french}{\CJS@corollaryname@fr}{\CJS@corollaryname@en}}
\theoremstyle{definition}
\newtheorem{definition}{\iflanguage{french}{\CJS@definitionname@fr}{\CJS@definitionname@en}}
\newtheorem{algorithm}{\iflanguage{french}{\CJS@algorithmname@fr}{\CJS@algorithmname@en}}
\theoremstyle{remark}
\newtheorem{remark}{\iflanguage{french}{\CJS@remarkname@fr}{\CJS@remarkname@en}}
```

`\eqref` The command `\eqref` from **amsmath** should be used to refer to equations. We redefine it such that the parentheses are part of the hyperlink, and that the command inherits font changes, notably in titles.[7]

```
\renewcommand*\eqref[1]{\hyperref[{#1}]{\textup{(\ref*{#1})}}}
```

B.7 Computer code and software

`\proglang` The command `\proglang` provides a means to uniformly typeset the name of a software or programming language. The command also prevents hyphenation within the name. The class also provides shortcuts to the most common languages. Notice that the `\Cpplang` case is specially crafted.

```
\newrobustcmd*\proglang[1]{\mbox{#1}}
\newcommand*\Rlang{\proglang{R}}
\newcommand*\SASlang{\proglang{SAS}}
\newcommand*\SPSSlang{\proglang{SPSS}}
\newcommand*\Statalang{\proglang{Stata}}
\newcommand*\Pylang{\proglang{Python}}
\newcommand*\Julialang{\proglang{Julia}}
\newcommand*\Clang{\proglang{C}}
\newcommand*\Cpplang{\proglang{C\CJS@plus\CJS@plus}}
\def\CJS@plus{%
  \ifx\f@family\sfdefault%
    \protect\hspace{-.04em}\protect\raisebox{-.125ex}{\larger+}%
  \else%
    \protect\hspace{-.03em}\protect\raisebox{.25ex}{\smaller+}%
  \fi}
```

`\pkg` The command `\pkg` typesets the name of software packages, extensions or modules.

```
\let\pkg\textbf
```

`\code` The following pretty smart implementation of `\code` allows use of the active characters _, ~ and $ as is within the command. It is lifted from the class file of the *Journal of Statistical Software*.

```
\newcommand\code{\bgroup\@makeother\_\@makeother\~\@makeother\$\code@}
\def\code@#1{{\normalfont\ttfamily\hyphenchar\font=-1 #1}\egroup}
```

B.8 Appendices

`\appendix` Given that appendices are sections in the class (instead of chapters as with memoir), we redefine `\appendix` based on the standard article class definition.

```
\renewcommand*\appendix{\par
  \setcounter{section}{0}
  \gdef\thesection{\@Alph\c@section}}
```

[7] With thanks to Heiko Oberdiek (https://tex.stackexchange.com/a/192754) and David Carlisle (https://tex.stackexchange.com/a/631254).

B.9 Citations and references

Very little to do here, as most of the work on the citations front is handled by **natbib**. We loaded the package with the option `sectionbib` to obtain the list of references as a section, but memoir will use `\section` in such instances. We want an unnumbered section (yet showing in the virtual table of contents of the PDF). To achieve this efficiently, we redefine the internal command of memoir.

```
\renewcommand{\@memb@bsec}{%
  \section*{\bibname}%
  \addcontentsline{toc}{section}{\bibname}%
  \prebibhook}
```

Furthermore, we need to redefine the title of the section, because memoir uses “Bibliography” by default. Due to the presence of **babel**, we have to use `\setlocalecaption` rather than redefine `\bibname` directly.

```
\setlocalecaption{english}{bib}{References}
\setlocalecaption{french}{bib}{Références}
```

The class uses its own bibliographic styles: `cjs-rcs-en` for articles in English, and `cjs-rcs-fr` for articles in French. For additional information on these styles, please refer to the documentation of the master bibliographic style in `cjs-rcs-merlin.pdf`.

```
\bibliographystyle{\iflanguage{french}{cjs-rcs-fr}{cjs-rcs-en}}
```

Finally, condense the bibliography. The default spacing between the entries defined by **natbib** is `\itemsep` + `\parsep`, which amounts to 2 × `\itemsep`. Reduce this to `\itemsep`.

```
\setlength{\bibsep}{\itemsep}
```

B.10 Supplementary material

In order to easily distinguish them from their counterparts of the main article, sections, equations, theorems, figure, etc., are numbered with “S”in prefix. For document divisions, we only need to change `\thesection` as the lower level printing commands are based on the latter.

```
\def\CJS@suppprefix{S}          % required for the documentation
\ifCJS@supplement
  \renewcommand*\thesection{\CJS@suppprefix\arabic{section}}
  \renewcommand*\thefigure{\CJS@suppprefix\arabic{figure}}
  \renewcommand*\thetable{\CJS@suppprefix\arabic{table}}
  \renewcommand*\theequation{\CJS@suppprefix\arabic{equation}}
  \renewcommand*\thetheorem{\CJS@suppprefix\arabic{theorem}}
  \renewcommand*\thelemma{\CJS@suppprefix\arabic{lemma}}
  \renewcommand*\theproposition{\CJS@suppprefix\arabic{proposition}}
  \renewcommand*\thecorollary{\CJS@suppprefix\arabic{corollary}}
  \renewcommand*\thedefinition{\CJS@suppprefix\arabic{definition}}
  \renewcommand*\thealgorithm{\CJS@suppprefix\arabic{algorithm}}
  \renewcommand*\theremark{\CJS@suppprefix\arabic{remark}}
\fi
```

B.11 Commands for the editors

`final` (*option*) The class option `final` activates a number of sanity checks on the document: the journal publication information (volume, issue, year) is provided; the folio of the first page is set; the article history (date received, date accepted) is provided. These checks are conducted at the end of the document.

```
\AtEndDocument{%
  \ifCJS@final
    \ifdefempty{\CJS@vol}{%
      \ClassError{cjs-rcs-article}%
        {Unknown Journal volume}%
        {Use \string\jvol\space to specify the volume.}}{\relax}
    \ifdefempty{\CJS@issue}{%
      \ClassError{cjs-rcs-article}%
        {Unknown Journal issue}%
        {Use \string\jissue\space to specify the issue.}}{\relax}
```

```
    \ifdefempty{\CJS@year}{%
      \ClassError{cjs-rcs-article}%
        {Unknown Journal publication year}%
        {Use \string\jyear\space to specify the year.}}{\relax}
    \ifdefstring{\CJS@firstpage}{1}{%
      \ClassWarningNoLine{cjs-rcs-article}%
        {The first page number is not set.\MessageBreak%
         Using the default folio 1}}{\relax}
    \ifCJS@history\else
      \ClassError{cjs-rcs-article}%
        {Unknown article history}%
        {Use \string\received\space and \string\accepted\space to provide the article history.}
    \fi
  \fi}
```

\jvol
\jissue
\jyear
\firstpage

The editors use the commands `\jvol`, `\jissue`, `\jyear` and `\firstpage` to enter the publication information of the journal.

```
\def\CJS@vol{}
\def\CJS@issue{}
\def\CJS@year{}
\def\CJS@firstpage{1}
\newcommand*\jvol[1]{\renewcommand*\CJS@vol{#1}}
\newcommand*\jissue[1]{\renewcommand*\CJS@issue{#1}}
\newcommand*\jyear[1]{\renewcommand*\CJS@year{#1}}
\newcommand*\firstpage[1]{\renewcommand*\CJS@firstpage{#1}}
```

\received
\accepted

The article history—dates when the article was received and accepted—is printed at the very end of the article when present. Its treatment involves a number of steps.

First, the dates themselves are input in ISO format YYYY-MM-DD and converted in language dependent strings using the package **datetime2**. In English, we want to use the British format without the ordinal for the day: 1 January 1970. In French, we can simply rely on the default format.

```
\DTMlangsetup[en-GB]{ord=omit}
```

Next, we need interface commands, a special kind of list to display the information, and a macro to actually typeset the history. Providing a *received* date marks the article history as present. The dates are saved and printed using commands from **datetime2**.

```
\def\CJS@receivedname@en{Received}
\def\CJS@receivedname@fr{Reçu le}
\def\CJS@acceptedname@en{Accepted}
\def\CJS@acceptedname@fr{Accepté le}
\newcommand*\received[1]{\CJS@historytrue\DTMsavedate{received}{#1}}
\newcommand*\accepted[1]{\DTMsavedate{accepted}{#1}}
\newlist{CJS@historylist}{itemize*}{1}
\setlist[CJS@historylist]{%
  mode=unboxed,
  font=\sffamily,
  itemjoin={{ --- }},
  after={}
}
\newcommand*\CJS@history{%
  \small\sffamily%
  \begin{CJS@historylist}
  \item[\iflanguage{french}{\CJS@receivedname@fr}{\CJS@receivedname@en}]
    \DTMusedate{received}%
  \DTMifsaveddate{accepted}{%
  \item[\iflanguage{french}{\CJS@acceptedname@fr}{\CJS@acceptedname@en}]
    \DTMusedate{accepted}}{\relax}%
  \end{CJS@historylist}\par}
```

We also define a decorative element similar to the one used on the title page, only adapted for a single line content. To create a balanced element, we use the height of a letter with an ascender and the depth of the old-style number 4.

```
\newsavebox\CJS@historyornament
\setbox\CJS@historyornament=\vbox{%
  \setbox\z@=\vbox{\hbox{\firaoldstyle\small l4}}
  \setbox\tw@\hbox{\vrule \@width1.5pt\@height\ht\z@\@depth\dp\z@}
  \hbox{\textcolor{CJSpink}{\copy\tw@}\kern.75pt
        \textcolor{CJSred}{\copy\tw@}\kern.75pt
        \textcolor{CJSyellow}{\copy\tw@}\kern3pt}}
```

Finally, the history information, when present, is printed at the end of the article.

```
\def\CJS@historybox{
  \par\addvspace{2\baselineskip}\noindent%
  \smash{\box\CJS@historyornament \CJS@history}\par%
  \vspace*{40pt}}
\AtEndDocument{\ifCJS@journalinfo\ifCJS@history\CJS@historybox\fi\fi}
```

`\specialack` The command `\specialack` is the generic interface to give special acknowledgement on the title page. The command inserts its first argument in the list of authors, and its second argument at the bottom of the title page, with the licensing information. Each argument may be empty, in which case the corresponding information is not displayed.

```
\def\CJS@specialackstatement{}
\newcommand*\specialack[2]{%
  \ifstrempty{#1}{}{%
      \author{#1\CJS@specialackmark}}
  \ifstrempty{#2}{}{%
      \CJS@specialacktrue
      \renewcommand*\CJS@specialackstatement{\CJS@specialackmark\,#2\medskip\newline}}}
```

`\specialackmark` The command `\specialackmark` changes the footnote marker used by `\specialack` from its default definition.

```
\def\CJS@specialackmark{\textsuperscript{\ensuremath{*}}}
\newcommand*\specialackmark[1]{%
  \renewcommand*\CJS@specialackmark{#1}}
```

</class>

Version history

0.1

First test version for review by CJS editorial team. 1

0.1a

Change the paper size from 7in x 10in to letter paper. 1

Command `\surname` to enter the surname of an author and automatically populate the running authors list. 1

Funding information moved to the end of the paper with other back matter. 1

Support for `\firstpage` and automatic display of the last page number in the journal information. 1

Support for ORCID iD. 1

0.1b

Documentation on the commands for editors moved in appendix. 1

Option `final` for the editors. 1

Support for hyperlinks. 1

The ornament in the journal information now uses the CJS colours. 1

0.1c

The corresponding author mark * is replaced by an envelope. The marker positioned next to the corresponding author name is now a `mailto:` hyperlink. 1

Use a key-value interface in `\author` to specify not only the ORCID iD, but also the email and corresponding author status. 1

Use Font Awesome for all symbols, instead of ccicons for the Creative Commons symbols and a solution based on TikZ to