



Percepciones de conflicto entre trabajadores y directivos

Seminario de Grado - Carrera de Sociología

Andreas Lafferte Tamayo

Profesor/a guía: Giorgio Boccardo

Fecha

Índice general

Resumen	1
Agradecimientos	3
1. Introducción: tesisfacso down	5
1.1. Requisitos	6
1.2. Pasos	6
1.3. Algunos detalles adicionales:	8
1.4. Publicación web y versionamiento	11
2. Antecedentes conceptuales y empíricos	13
2.1. (concepto principal - explanandum - objeto de estudio)	13
2.2. Factores asociados a objeto de estudio	13
3. Metodología	15
3.1. Datos	15
3.2. Variables	15
3.3. Métodos	15
4. Análisis	17
4.1. Análisis descriptivo	17
4.2. Modelos	17
5. Conclusiones	19
Bibliografía	21

Índice de tablas

1.1. Un ejemplo de tabla	11
------------------------------------	----

Índice de figuras

1.1. Flujo tesisfacsodown	5
1.2. Generando los documentos en pdf y html	7
1.3. Imagen de ejemplo	10
1.4. Gráfico de ejemplo	10
1.5. Bookdown en Github Pages	12

Resumen

Resumen aquí

Agradecimientos

Agradecimientos aquí.

Capítulo 1

Introducción: tesisfacsodown

Tesisfacsodown es básicamente una propuesta o plantilla para poder escribir un trabajo final formateado en alta calidad, con opción tanto para pdf como para html. Para ello, basta con escribir el trabajo en un lenguaje de estructura simple: **markdown** (detalles más adelante), el que luego es transformado a un documento formateado en pdf tal como se puede ver [aquí](#), y si se desea también en html de manera simultanea. Es decir, se trata de concentrarse en escribir y el trabajo de formateo se lo dejamos a un tercero, que en este caso es la librería **bookdown** de R ([Xie, 2017](#)) .

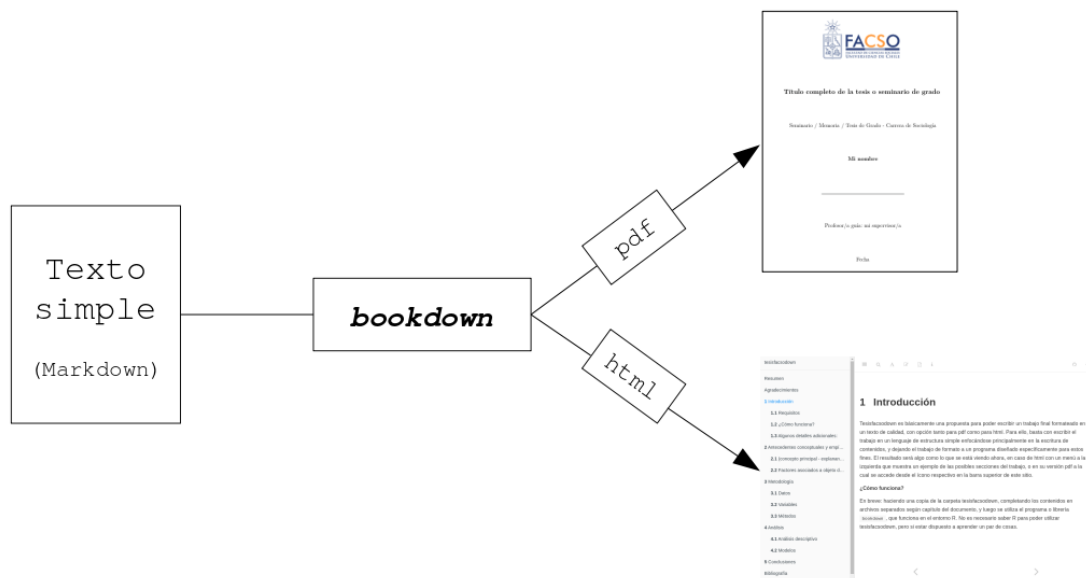


Figura 1.1: Flujo tesisfacsodown

Junto con realizar el trabajo de formato, quienes realizan análisis de datos con R tienen la posibilidad de incluir también los resultados (tablas, gráficos) de manera

dinámica. Esto quiere decir que los análisis se realizan en el mismo documento, que se actualiza automáticamente en caso de modificaciones evitando así la ineficiencia de estar cortando y pegando cada vez que se realiza un cambio.

Tesisfacsodown está basada en otros proyectos similares como [tesisdown](#) y también [oxforddown](#).

¿Cómo funciona?

En breve: haciendo una copia de la carpeta *tesisfacsodown*, completando / reemplazando los contenidos en los archivos separados según secciones del documento (introducción, antecedentes, etc), y luego se utiliza el programa o librería **bookdown**, que funciona en el entorno R y que produce los documentos formateados en pdf y/o html. No es necesario saber R para poder utilizar tesisfacsodown, pero si estar dispuesto a aprender un par de cosas.

1.1. Requisitos

- Software: R/RStudio (ambos gratuitos).
- librerías de R: **bookdown**, se instala desde RStudio en el menú Tools > Install packages > bookdown. Si se maneja R es más directo ejecutar desde la consola `install.packages("bookdown")`
- Escritura: conocimiento de *Markdown*, que es básicamente escribir simplemente texto con muy pocas marcas de edición (como encabezados, cursivas, negritas). Estas marcas de edición luego son interpretadas para poder generar documentos formateados en pdf, html (web) y también word. No requiere gran conocimiento previo, una introducción breve se puede encontrar [aquí](#).
- No se requiere conocimiento de análisis de datos con R. Si bien este formato facilita la inclusión de resultados de análisis cuantitativos, puede ser perfectamente usado para trabajos con otras metodologías.
- Si además de pdf se quiere publicar en formato web (es opcional) se necesitan conocimientos básicos de Git y Github. Una breve introducción se puede ver [aquí](#), y con mucho mayor detalle en <https://happygitwithr.com/>.

1.2. Pasos

El primer paso consiste es obtener la carpeta de ejemplo **tesisfacsodown**, luego reemplazar algunos contenidos y obtener el documento formateado mediante un simple click en RStudio. Existen dos formas:

- simplemente bajar la carpeta desde aquí
- para quienes tengan una cuenta en Github, lo más facil es hacer un fork del repositorio <https://github.com/juancarloscastillo/tesisfacsodown>, luego clonarlo localmente y comenzar a trabajar desde ahí. Esta opción es la más conveniente si se quiere luego publicar la tesis online (detalles abajo en sección de publicación web).

Una vez que se tiene la carpeta localmente, hacer doble click en el archivo `tesis.Rproj`, con lo que se abrirá RStudio (tiene que estar instalado previamente, tutorial [aquí](#)). En RStudio hacer click sobre el botón “build book”:

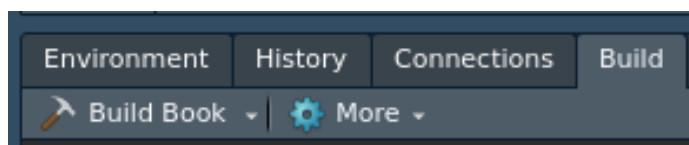


Figura 1.2: Generando los documentos en pdf y html

Con esto aparecerá en el visor de RStudio (pestaña Viewer) el documento en formato (html) y también se generará el documento `tesis.pdf` en la carpeta `docs`, y al que se puede acceder desde el ícono pdf en la barra superior del documento en html. Lo que aparece en los documentos html y pdf es este mismo documento pdf/html que están viendo ahora, pero que ahora se encuentra guardado localmente en sus computadores y que pueden modificar con los contenidos de sus trabajos.

Detalles

La carpeta tiene una serie de archivos y subcarpetas. Lo que hace el botón “build book” es activar la función `bookdown` de R que permite convertir el contenido escrito en Markdown hacia html y pdf. Para ello, los contenidos de la carpeta se organizan de la siguiente manera:

- **Archivo `index.Rmd`:** este archivo contiene
 - una serie de campos para poder personalizar e identificar la tesis, como título, autor, etc.
 - de manera opcional se puede incluir un archivo con referencias bibliográficas `.bib` en caso de utilizar formato `bibtex` (que puede ser generado con gestores de referencias como Zotero o Mendeley, mayores detalles revisar [aquí](#)).
 - también aparecen algunas opciones que tienen que ver con la generación del documento pdf y del sitio que no hay que modificar.
 - finalmente, en este archivo también se puede incluir el Resumen / abstract del trabajo.

- **Secciones del trabajo:** es una propuesta de archivos markdown con extensión `.Rmd` que comienzan con un número, y es donde se separan las distintas secciones o capítulos. Podría también ser un solo gran archivo con todos los contenidos, pero esta división permite un mayor orden en el trabajo y también separa en distintas páginas web en el caso de publicación en este formato. El orden en que aparecen las secciones en el documento final se indica con un número al inicio del archivo (ej: 01-intro.Rmd). Los archivos de las secciones que aparecen en esta carpeta son solo de ejemplo y pueden ser eliminados / reemplazados de acuerdo a la estructura preferida.

Adicionalmente se encuentran:

- *Archivos Readme* (`.md` y `.html`): contienen la información que aparece aquí, pueden ser eliminados posteriormente.
- *Archivos .yaml*: son `_output` y `_bookdown`, y contienen información para generar los documentos pdf y html, no modificar
- *tesis.Rproj*: este es archivo que permite acceder a RStudio desde esta carpeta y activar las funciones necesarias para generar el documento formateado. Por lo tanto, luego de bajar la carpeta hacer doble click en este archivo.
- *tesis.bib*: es el archivo que contiene las referencias bibliográficas en formato bibtex, en caso de utilizar esta forma de referenciar. El archivo debe mantenerse con este nombre, de lo contrario no funcionará en el pdf.
- *Carpetas*:
 - `images`: para guardar imágenes que luego se puedan llamar desde los documentos. Para poder insertarlas desde los documentos Rmd se requiere dar la ruta a la imagen, por ejemplo: `images/mi_imagen.jpg`. Que las imágenes se encuentren en esta carpeta es opcional, solo ayuda al orden de los documentos.
 - `templates`: aquí se encuentran archivos que permiten dar formato a los documentos pdf y html (no modificar ni eliminar)
 - `docs`: carpeta producida automáticamente que contiene los documentos generados y formateados en pdf y/o html.

1.3. Algunos detalles adicionales:

- para más opciones de bookdown y detalles en general consultar https://rubenfcasal.github.io/bookdown_intro/ (en español).

- el formato del pdf es para ser impreso por los dos lados, por eso incluye algunas páginas en blanco (por ejemplo, la página de atrás de la portada), y también los números de página se van alternando entre izquierda y derecha
- Es posible generar una referenciación automática de tablas, imágenes y gráficos; además, se generan los índices respectivos en el formato pdf. Si estos provienen simplemente de archivos externos, se pueden almacenar en la carpeta “imágenes” e insertar así:

```

'''{r nombre, echo=FALSE, fig.cap="Imagen de ejemplo", fig.align = 'center', out.wid
knitr::include_graphics(path = "images/build_book.png")
'''

```

Lo que hay aquí es un trozo o “chunk” de código R, que en (R)Markdown se inserta simplemente entre tres acentos invertidos (“““”). Luego hay dos partes: las especificaciones de formato que van entre llaves {}, y la instrucción de lo que hay que hacer, en este caso llamar una imagen. Vamos por parte:

- especificaciones de formato:
 - **r** solamente informa que viene código R
 - **nombre**: es el nombre que identifica al chunk, puede ser cualquiera
 - **echo=FALSE**: esto se utiliza para que solo se muestre el resultado sin el código. Si fuera **TRUE**, entonces aparecería la función que llama la imagen además que la `mi_imagen`
 - **fig.cap**: el título de la imagen
 - **fig.align**: alineación de la figura
 - **out.width**: ancho en relación a la página. Hay diferentes formas de especificar las dimensiones, la ventaja de esta es que funciona bien tanto para pdf como para html, por lo que se recomienda
- instrucción o función: aquí se da la ruta a la imagen dentro del paréntesis con **path=**, y lo que antecede es la función para incluir gráficos de la librería `knitr` de R. Pero en simple: solo reemplazar el nombre de la imagen

Al incluir este trozo de código el resultado será el siguiente:



Figura 1.3: Imagen de ejemplo

Donde la referenciación (caption) tiene el número 1.2, que quiere decir que es la segunda imagen del capítulo 1. En el pdf ahora esta figura aparece en el índice de figuras.

- en el caso de gráficos generados por R, también se insertan dentro de un chunk, por ejemplo

```

{r echo=FALSE, fig.cap="Gráfico de ejemplo", fig.align = 'center', out.width = '50%'}
boxplot(Sepal.Length~Species,data=iris,
        xlab="Species", ylab="Sepal Length", main="Iris Boxplot")

```

Y el resultado es este:

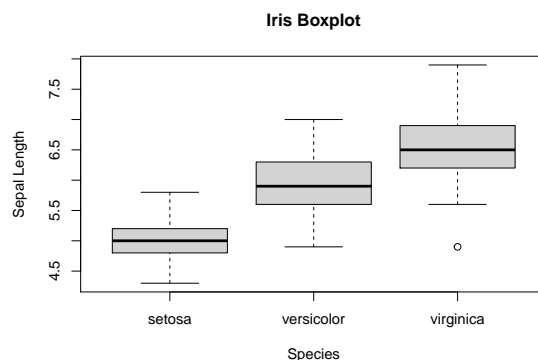


Figura 1.4: Gráfico de ejemplo

- finalmente para las tablas producidas por R, es un poco más difícil si se quiere obtener el documento tanto en html como en pdf (que se genera vía Latex). En

Tabla 1.1: Un ejemplo de tabla

Sepal.Length	Sepal.Width	Petal.Length	Petal.Width	Species
5.1	3.5	1.4	0.2	setosa
4.9	3.0	1.4	0.2	setosa
4.7	3.2	1.3	0.2	setosa
4.6	3.1	1.5	0.2	setosa
5.0	3.6	1.4	0.2	setosa
5.4	3.9	1.7	0.4	setosa

este caso, las tablas deben ser producidas en formato markdown, lo que no es muy común todavía en las distintas librerías de R que en general privilegian o html o latex. La función para producir tablas markdown en R es principalmente `kable`, de la librería `knitr`, por ejemplo:

```
““{r tabla, echo=FALSE, caption="tabla de ejemplo", align = 'center'}
knitr::kable(head(iris), caption = "An example table caption.")
““
```

```
knitr::kable(head(iris), caption = "Un ejemplo de tabla")
```

- nota sobre ubicación de imágenes y tablas en documento pdf (latex): en general se producen problemas en la ubicación, ya que automáticamente se ajustan al lugar disponible más cercano, sea atrás o delante. Si se quiere fijar la posición en el pdf, agregar antes y después del texto `\FloatBarrier`, que es un comando latex que no será visible ni en pdf ni en html.

1.4. Publicación web y versionamiento

Una de las ventajas del formato html es poder publicarlo en línea para una fácil lectura y seguimiento. Esto puede ocurrir desde las fases iniciales o para el documento final, aumentando las posibilidades de difusión, colaboración y también de obtener revisiones y comentarios. De todas maneras, requiere algunos conocimientos más específicos que se introducen aquí.

Existen diferentes maneras de generar un sitio web con el documento, pero el más directo es a través de **Github** y de su opción *githubpages*. Git/Github es una plataforma de versionamiento y colaboración cuyos detalles y posibilidades claramente no alcanzan a ser cubiertos en este espacio, pero si se desea mayor información se puede revisar [aquí](#).

1. Generación de repositorio: para quienes obtuvieron la carpeta partiendo desde Github con un *fork* / *clone* de tesisfacsodown el repositorio se encuentra listo. En caso de haber bajado/copiado la carpeta manera simple, entonces a) crear un repositorio en Github, b) clonarlo localmente, c) copiar/mover los archivos de la carpeta tesisfacsodown a la carpeta clonada
2. Subir los cambios al repositorio (push)
3. Una vez que los archivos ya están en el repositorio Github ir a Settings > Github Pages y dejar de la siguiente manera:

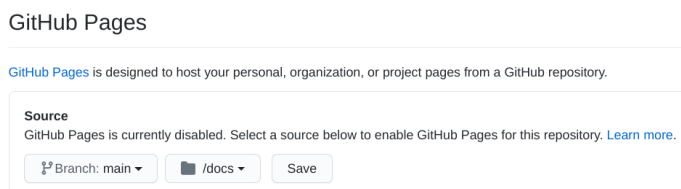


Figura 1.5: Bookdown en Github Pages

Luego “Save” y se actualiza la página. Al bajar nuevamente a la sección de Github Pages debería aparecer “Your site is ready to be published at [dirección web]”. Esa dirección es donde se puede acceder al sitio una vez que pasen algunos minutos.

Capítulo 2

Antecedentes conceptuales y empíricos

2.1. (concepto principal - explanandum - objeto de estudio)

...

2.2. Factores asociados a objeto de estudio

2.2.1. Factor 1

Al final de esta sección, enunciar la hipótesis correspondiente

2.2.2. Factor N

Al final de esta sección, enunciar la hipótesis correspondiente

Capítulo 3

Metodología

3.1. Datos

3.2. Variables

3.3. Métodos

Capítulo 4

Análisis

4.1. Análisis descriptivo

4.2. Modelos

Capítulo 5

Conclusiones

Bibliografía

Bibliografía

Xie, Y. (2017). *Bookdown: Authoring Books and Technical Publications with R Markdown*. CRC Press, Boca Raton, FL.