



**Data for
Development Impact**

Análisis de Datos

Construcción de Datos

Rony Rodriguez-Ramírez

July 17, 2020

LAMBDA

Introducción

Inputs

Constructed dataset:

- Incluir solo las variables necesarias para el análisis
- Libro de códigos de acompañamiento con descripción y definición de variables.
- Hecho a medida para responder sus preguntas de análisis
 - Muestra
 - Unidad de observación

Outputs

- Los resultados son exportados a archivos que pueden ser vistos como inputs para papers o reportes.
- Tablas independientes y gráficos.
- Formatos accesibles.

Outputs

- Los resultados finales, como documentos, informes breves e incluso creados para analizar los resultados, deben actualizarse automáticamente cuando se actualizan los resultados sin procesar.
- \LaTeX es una herramienta extremadamente útil para hacer esto.
- Si no sabe cómo usarlo, consulte nuestra capacitación en \LaTeX de DIME.

Documentación

- Otro resultado importante del análisis es un mapa de cómo se crearon los resultados
- El script maestro (master do file) es la mejor manera de hacer esto: debe hacer un seguimiento de lo que son las entradas y salidas de cada script que ejecuta.
- Un archivo README también es una buena forma de hacerlo, especialmente cuando se usan idiomas de programación y software específicos.

Análisis

El proceso de Análisis

- El análisis de datos se puede dividir en dos etapas.

El proceso de Análisis

- El análisis de datos se puede dividir en dos etapas.
- Durante el análisis **exploratorio de datos**, el equipo de investigación generalmente buscará patrones en los datos, de manera más descriptiva.

El proceso de Análisis

- El análisis de datos se puede dividir en dos etapas.
- Durante el análisis **exploratorio de datos**, el equipo de investigación generalmente buscará patrones en los datos, de manera más descriptiva.
- El proceso luego avanza hacia el **análisis final** cuando el equipo comienza a decidir cuáles son los resultados principales, que formarán parte del resultado de la investigación.

El proceso de Análisis

- El análisis de datos se puede dividir en dos etapas.
- Durante el análisis **exploratorio de datos**, el equipo de investigación generalmente buscará patrones en los datos, de manera más descriptiva.
- El proceso luego avanza hacia el **análisis final** cuando el equipo comienza a decidir cuáles son los resultados principales, que formarán parte del resultado de la investigación.
- Para los proyectos que tienen planes de preanálisis, las especificaciones principales se predefinirán, por lo que la fase exploratoria tiene menos implicaciones para los resultados finales.

Trabajo de datos durante el análisis

- La forma en que maneja el código y los resultados para el análisis exploratorio y final es diferente.

Trabajo de datos durante el análisis

- La forma en que maneja el código y los resultados para el análisis exploratorio y final es diferente.
- Durante el análisis exploratorio de datos, estará tentado a escribir muchos análisis en un gran script, o incluso directamente en la consola.

Trabajo de datos durante el análisis

- La forma en que maneja el código y los resultados para el análisis exploratorio y final es diferente.
- Durante el análisis exploratorio de datos, estará tentado a escribir muchos análisis en un gran script, o incluso directamente en la consola.
- Esto fomenta sutilmente prácticas riesgosas como no limpiar el espacio de trabajo y no recargar los datos relevantes.

Trabajo de datos durante el análisis

- La forma en que maneja el código y los resultados para el análisis exploratorio y final es diferente.
- Durante el análisis exploratorio de datos, estará tentado a escribir muchos análisis en un gran script, o incluso directamente en la consola.
- Esto fomenta sutilmente prácticas riesgosas como no limpiar el espacio de trabajo y no recargar los datos relevantes.
- Para evitar errores, es importante tomarse el tiempo para organizar el código que desea usar de nuevo de manera limpia.

Documentos dinámicos durante el análisis exploratorio

- Una forma de evitar caer en malas prácticas durante el análisis exploratorio de datos es crear documentos dinámicos
- Le permiten escribir código, tomar notas sobre sus observaciones y visualizar resultados en un solo documento
- Las opciones de Stata incluyen **markstat**, que usa una sintaxis similar a markdown, y **texdoc**, que combina código \LaTeX y Stata
- En R, **RMarkdown** es ampliamente adoptado
- La principal limitación de este tipo de documentos dinámicos son las opciones de formato limitadas que se ofrecen y la dificultad de manejar código y texto al mismo tiempo.

Documentos dinámicos durante el análisis final

- Dadas las limitaciones de crear documentos dinámicos en software estadístico, el equipo tiende a preferir pasar al editor de texto o a los sistemas de preparación de documentos para escribir los resultados finales de la investigación.
- Al configurar este flujo de trabajo, es importante pensar en la integración entre las salidas de código y el texto.

Documentos dinámicos durante el análisis final

- El código generalmente sigue evolucionando a medida que se redactan los documentos e informes, y es importante mantener los resultados del código actualizados en los documentos finales
- \LaTeX es la forma más popular de hacer esto
- Le permite escribir referencias a los archivos que contienen resultados de análisis, para que se actualicen cada vez que se compila el documento \LaTeX .

Un workflow automatizado

Exportando outputs

- Está bien no exportar todas y cada una de las tablas y gráficos creados durante el análisis exploratorio.
- Los resultados finales deben exportarse para que estén listos para ser incluidos en un documento o informe.
- No deberían ser necesarias ediciones manuales, incluido el formateo, después de exportar los resultados finales.
- No cree un flujo de trabajo que implique copiar y pegar en diferentes programas.

Automatizando outputs

- Las ediciones manuales son difíciles de replicar e inevitablemente necesitará realizar cambios en los resultados.
- La cantidad de trabajo necesaria en un flujo de trabajo de copiar y pegar aumenta rápidamente con el número de resultados, y también lo hacen las posibilidades de que la versión incorrecta resulte en su documento o informe.
- Automatizar la creación de resultados le ahorrará tiempo al final del proceso.
- Los resultados finales de pulido pueden llevar mucho tiempo
- No pase demasiado tiempo formateando hasta que su equipo haya acordado los resultados finales.

En caso que no sea claro...

Nunca configure un flujo de trabajo que requiera copiar y pegar resultados.

Automatizando outputs

- Copiar resultados de Excel a Word es propenso a errores e ineficiente.

Automatizando outputs

- Copiar resultados de Excel a Word es propenso a errores e ineficiente.
- La copia de los resultados de una consola de software es aún más ineficiente y es completamente innecesaria.

Automatizando outputs

- Copiar resultados de Excel a Word es propenso a errores e ineficiente.
- La copia de los resultados de una consola de software es aún más ineficiente y es completamente innecesaria.
- Existen numerosos comandos para exportar salidas de R y Stata a una miríada de formatos.

Automatizando outputs

- Copiar resultados de Excel a Word es propenso a errores e ineficiente.
- La copia de los resultados de una consola de software es aún más ineficiente y es completamente innecesaria.
- Existen numerosos comandos para exportar salidas de R y Stata a una miríada de formatos.
- Nuestro comando Stata preferido para exportar tablas son **esttab**, **outreg2** y **outwrite**.

Automatizando outputs

- Copiar resultados de Excel a Word es propenso a errores e ineficiente.
- La copia de los resultados de una consola de software es aún más ineficiente y es completamente innecesaria.
- Existen numerosos comandos para exportar salidas de R y Stata a una miríada de formatos.
- Nuestro comando Stata preferido para exportar tablas son **esttab**, **outreg2** y **outwrite**.
- Nuestro paquete R preferido para exportar tablas es **stargazer**.

Automatizando outputs

- Copiar resultados de Excel a Word es propenso a errores e ineficiente.
- La copia de los resultados de una consola de software es aún más ineficiente y es completamente innecesaria.
- Existen numerosos comandos para exportar salidas de R y Stata a una miríada de formatos.
- Nuestro comando Stata preferido para exportar tablas son **esttab**, **outreg2** y **outwrite**.
- Nuestro paquete R preferido para exportar tablas es **stargazer**.
- ¡Hay muchos más por ahí!

Automatizando outputs

- **estout** puede resolver la mayoría de los problemas.
- Puede exportar tanto summary statistics y tablas de regresión de una manera sencilla.
- También suportar bastante tipos de customización, y exporta tanto a Excel como a \LaTeX .

Automatizando outputs

Puede encontrar más ejemplo en los siguientes repositorios:

1. <https://github.com/bbdaniels/stata-tables>
2. <https://github.com/RRMaximiliano/stata-latex-tables>

Automatizando outputs

- También puede editar el conjunto de datos directamente y exportar los datos a Excel con exportación de Excel, a csv con exportación delimitada o a \LaTeX con salida de datos
- Si te apetece, puedes crear matrices y exportarlas usando `mat2txt` o sobrescribir.
- Finalmente, puede exportar **tabulaciones** de una y dos vías usando `tabout`.

Escribiendo scripts de análisis

Organización del script (do file)

Un scripts (do file) muy bien ordenado:

- Comienza con un espacio de trabajo completamente nuevo.

Organización del script (do file)

Un scripts (do file) muy bien ordenado:

- Comienza con un espacio de trabajo completamente nuevo.
- Carga el conjunto de datos construido.

Organización del script (do file)

Un scripts (do file) muy bien ordenado:

- Comienza con un espacio de trabajo completamente nuevo.
- Carga el conjunto de datos construido.
- Toma decisiones de investigación explícitamente (muestreo, agrupamiento, inclusión de controles).

Organización del script (do file)

Un scripts (do file) muy bien ordenado:

- Comienza con un espacio de trabajo completamente nuevo.
- Carga el conjunto de datos construido.
- Toma decisiones de investigación explícitamente (muestreo, agrupamiento, inclusión de controles).
- Tiene un código simple que permite al usuario enfocarse en la econometría.

Organización del script (do file)

Un scripts (do file) muy bien ordenado:

- Comienza con un espacio de trabajo completamente nuevo.
- Carga el conjunto de datos construido.
- Toma decisiones de investigación explícitamente (muestreo, agrupamiento, inclusión de controles).
- Tiene un código simple que permite al usuario enfocarse en la econometría.
- Exporta los resultados obtenidos.

Organización del script (do file)

Un scripts (do file) muy bien ordenado:

- Comienza con un espacio de trabajo completamente nuevo.
- Carga el conjunto de datos construido.
- Toma decisiones de investigación explícitamente (muestreo, agrupamiento, inclusión de controles).
- Tiene un código simple que permite al usuario enfocarse en la econometría.
- Exporta los resultados obtenidos.
- Se ejecuta de forma completamente independiente de todos los demás códigos, excepto el script maestro.

Organización del script (do file)

Un scripts (do file) muy bien ordenado:

- Comienza con un espacio de trabajo completamente nuevo.
- Carga el conjunto de datos construido.
- Toma decisiones de investigación explícitamente (muestreo, agrupamiento, inclusión de controles).
- Tiene un código simple que permite al usuario enfocarse en la econometría.
- Exporta los resultados obtenidos.
- Se ejecuta de forma completamente independiente de todos los demás códigos, excepto el script maestro.
- Se puede vincular a su salida por nombre.

Organización del script (do file)

- El código de análisis debe ser limpio y simple: incluso puede crear un script para cada salida

Organización del script (do file)

- El código de análisis debe ser limpio y simple: incluso puede crear un script para cada salida
- Si tiene múltiples conjuntos de datos de análisis, cada uno de ellos debe tener un nombre descriptivo sobre su muestra y unidad de observación, por lo que está claro qué conjunto de datos se debe utilizar para cada análisis.

Organización del script (do file)

- El código de análisis debe ser limpio y simple: incluso puede crear un script para cada salida
- Si tiene múltiples conjuntos de datos de análisis, cada uno de ellos debe tener un nombre descriptivo sobre su muestra y unidad de observación, por lo que está claro qué conjunto de datos se debe utilizar para cada análisis.
- En ambos casos, la denominación debe ser intuitiva para que pueda rastrear las entradas y salidas de cada script.

Organización del script (do file)

- Cuando su equipo toma decisiones sobre la especificación del modelo, puede crear globales u objetos en la secuencia de comandos maestra para usar en las secuencias de comandos.

Organización del script (do file)

- Cuando su equipo toma decisiones sobre la especificación del modelo, puede crear globales u objetos en la secuencia de comandos maestra para usar en las secuencias de comandos.
- Esto asegurará que las especificaciones sean consistentes durante todo el análisis.

Organización del script (do file)

- Cuando su equipo toma decisiones sobre la especificación del modelo, puede crear globales u objetos en la secuencia de comandos maestra para usar en las secuencias de comandos.
- Esto asegurará que las especificaciones sean consistentes durante todo el análisis.
- También hará que su código sea más dinámico, por lo que es fácil actualizar las especificaciones y los resultados sin cambiar cada script.

Organización del script (do file)

- Cuando su equipo toma decisiones sobre la especificación del modelo, puede crear globales u objetos en la secuencia de comandos maestra para usar en las secuencias de comandos.
- Esto asegurará que las especificaciones sean consistentes durante todo el análisis.
- También hará que su código sea más dinámico, por lo que es fácil actualizar las especificaciones y los resultados sin cambiar cada script.
- Utilice comandos preexistentes siempre que sea posible: evite abarrotar su código con comandos complicados para crear y agregar matrices intermedias.

Outputs finales

¡Mira tu outputs!

- ¿El aspecto es decente?
- ¿Alguien más puede entenderlo?
- Verifique el número de observaciones
- Pregúntese si los resultados tienen sentido
- Verifique nuevamente el número de observaciones
- Intenta interpretar el resultado
- Verifique las escalas

¡Mira tu outputs!

Al escribir un artículo pregúntese: ¿sus tablas y figuras despiertan alegría?

¿Ahora qué?

- Si sigue los pasos descritos en este capítulo, la mayoría del trabajo de datos involucrado en el último paso del proceso de investigación - publicación - ya estará hecho.

¿Ahora qué?

- Si sigue los pasos descritos en este capítulo, la mayoría del trabajo de datos involucrado en el último paso del proceso de investigación - publicación - ya estará hecho.
- Su código de análisis se organizará de manera reproducible, por lo que todo lo que necesitará para lanzar un paquete de replicación es una última ronda de revisión de código.

¿Ahora qué?

- Si sigue los pasos descritos en este capítulo, la mayoría del trabajo de datos involucrado en el último paso del proceso de investigación - publicación - ya estará hecho.
- Su código de análisis se organizará de manera reproducible, por lo que todo lo que necesitará para lanzar un paquete de replicación es una última ronda de revisión de código.
- Esto te permitirá concentrarte en lo que importa: escribir tus resultados en una historia convincente

STATA TIME



Nos vemos mañana.