Reunión Nacional de Ramas IEEE RNR2015

Programación Literaria Investigación Reproducible y Software Libre

Ing. Milton Labanda, Mg.

Carrera de Ingeniería Informática y Multimedia
Universidad Internacional del Ecuador

11 de agosto de 2015

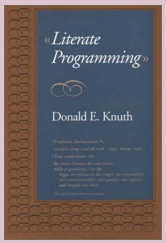
Contenido

Historia

Como empieza todo?

Çomputer programs should be written in a combination of the programming language (the usual source code) and the natural language, which explains the logic of the program". **Donald Knuth**

El objetivo



I believe that the time is ripe for significantly better documentation of programs, and that we can best achieve this by considering programs to be works of literature. Hence, my title: "Literate Programming" **Donald Knuth**

Historia

WEB

Primera herramienta de implementación de lo que se conoció como Programación Literaria. Producía código PACAL compilable y la documentación formateada usando Tex

CWEB

Descendiente del entorno WEB usa en cambio C como lenguaje de programación pero el mismo Tex para la generación de la documentación

¿Qué es la Programación Literaria ?

- Paradigma cotrario a la programación tradicional: En vez de escribir código que contiene documentación el programador literario escribe documentación que contiene código.
- Los Programas literarios pueden ser tejidos (WEAVED) para producir documentos legibles para los humanos y enredados (TANGLED) para producir documentos legibles para las máquinas"

Elementos de la Programación Literaria

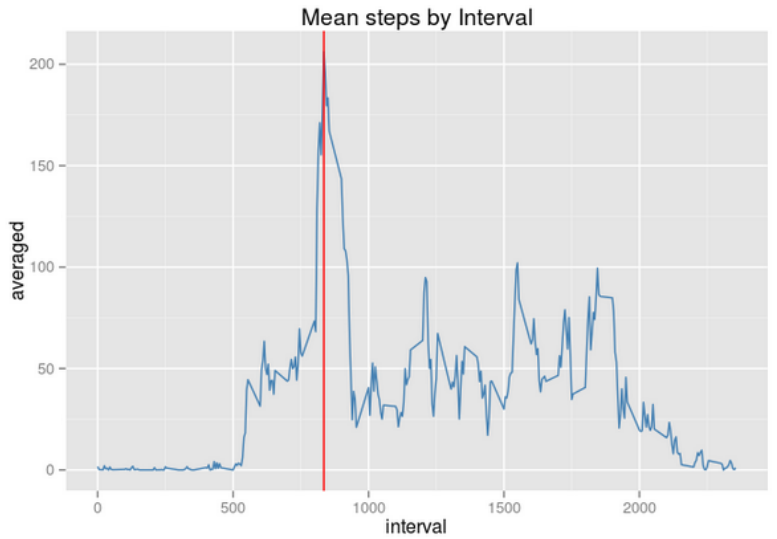
- Cada "trozo" de código fuente carga datos y calcula resultados mientras que el código de presentación formatea la salida
- Consecuentemente el paradigma de la programación literaria requiere:
  1. Un lenguaje de documentación (human readable)
  2. Un lenguaje de programación (machine readable)

Ejemplos de implementaciones o entornos

- Sweave = LaTeX+ R
- knitr = HTML + R
- Jupyter = HTML + Python
- Rambutan = TeX+ Java
- CoffeScript (Markdown + CoffeScript)

Ejemplo knitr

```
47
48 ```{r}
49 byInterval <- ddply(df, ~interval, summarize, averaged=mean(steps,
   na.rm=T))
50 max_averaged = max(byInterval$averaged)
51 max_interval = byInterval[byInterval$averaged==max_averaged,]$inte
   rval
52 ggplot(byInterval, aes(interval, averaged)) +
   geom_line(colour="steelblue") + labs(title="Mean steps by
   Interval") + geom_vline(xintercept=max_interval, colour="red")
53 ```
54
55 2. Which 5-minute interval, on average across all the days in the
   dataset, contains the maximum number of steps?
56
57 The interval that contain the maximun number of steps by day is
   **`r max_interval`** with an average of `r max_averaged` steps.
58
59 ## Imputing missing values
```



2. Which 5-minute interval, on average across all the days in the dataset, contains the maximum number of steps?

The interval that contain the maximun number of steps by day is **835** with an average of 206.1698113 steps.

Imputing missing values

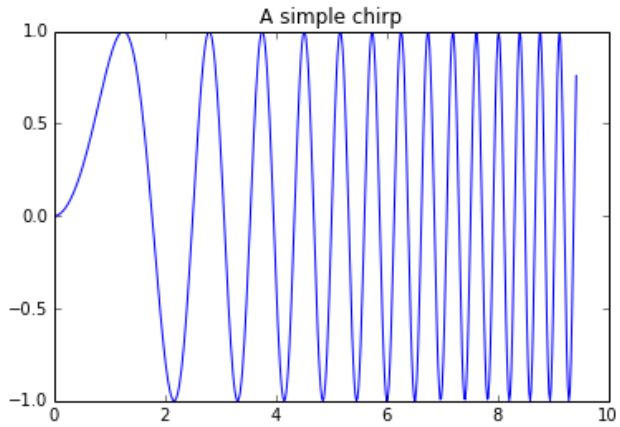
Ejemplo Jupyter (IPython Notebook)

Making a simple plot

With matplotlib enabled, plotting should just work.

In [2]:
```
import matplotlib.pyplot as plt
import numpy as np
```

In [3]:
```
x = np.linspace(0, 3*np.pi, 500)
plt.plot(x, np.sin(x**2))
plt.title('A simple chirp');
```



These images can be resized by dragging the handle in the lower right corner. Double clicking will return them to their original size.

Contenido

Investigación Reproducible

¿Que persigue?

Hacer los **datos analíticos** y el **código** disponibles para que otros puedan reproducir los descubrimientos

¿Por qué necesitamos Investigación Reproducible?

... hoy más que nunca

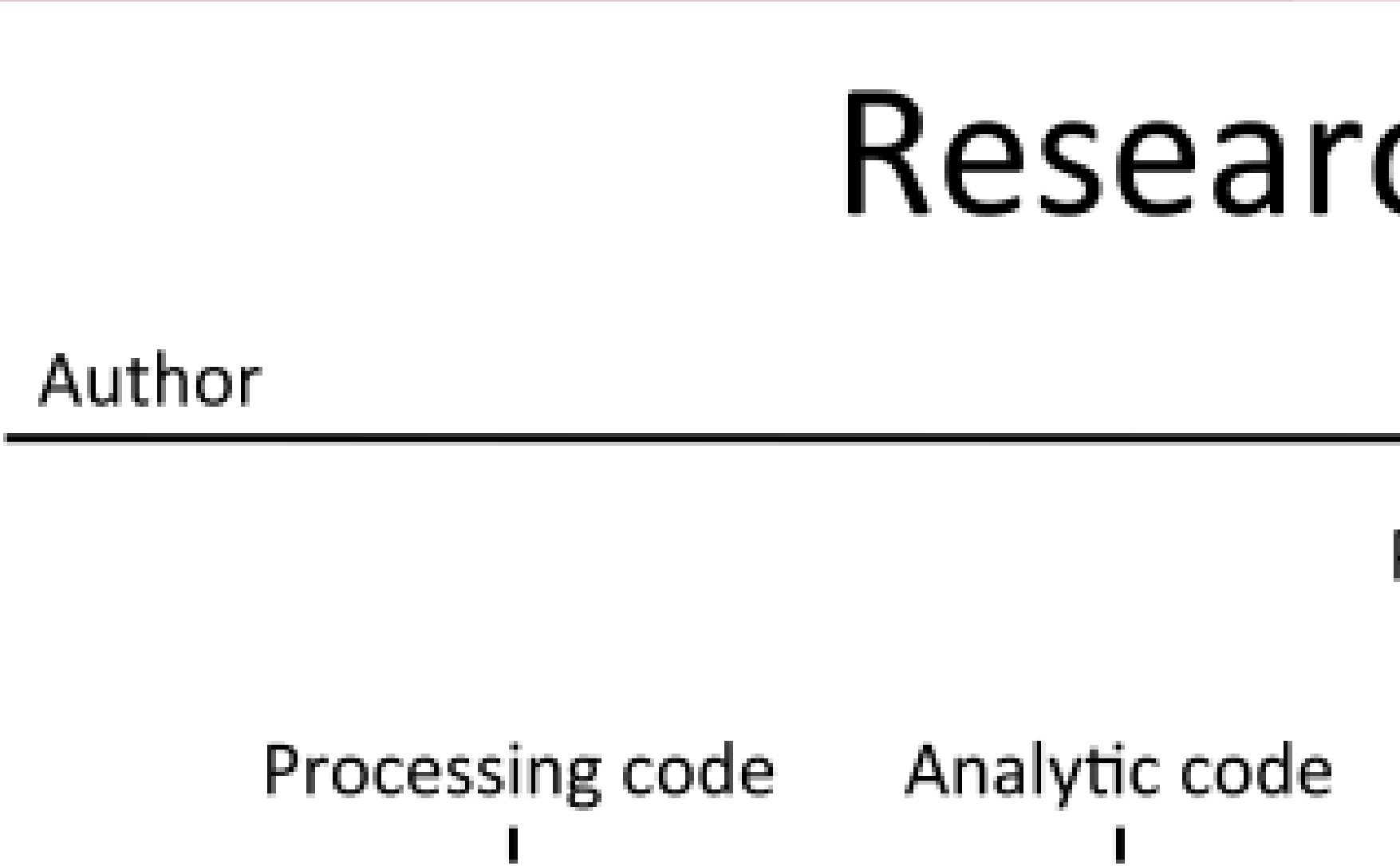
- Las nuevas tecnologías incrementan las colecciones de datos cada vez más
- Los datos son más complejos y extremadamente multidimensionales
- Las bases de datos existentes pueden ser mezcladas dentro de "mega bases de datos"
- La capacidad de cómputo es altamente incrementable permitiendo análisis mas sofisticados
- Para cada campo 'X' existe un campo Çomputacional 'X'

El Análisis de Datos

Su estructura

- Definir la pregunta
- Definir el dataset ideal
- Determinar que datos se pueden acceder
- Obtener los datos
- Limpiar los datos
- Análisis de Datos exploratorio
- Modelamiento/predicción estadístico
- Interpretar los resultados
- Escribir/sintetizar los resultados
- Crear código reproducible

El flujo tipico de la investigación



Los actores en el Análisis de Datos

Autores

- Desean hacer su investigación reproducible
- Quieren herramientas de IR que hagan su vida más facil ("no muy dura")
- Relizan considerables esfuerzos para publicar su datos. E: servidor web
- Entonces =¿ Con IR colocan solo lo mínimo en la Web así como materiales suplementarios en las revistas o en bases de datos centrales

Lectores

- Quieren reproducir y posiblemente expandir los descubrimientos de interés
- Quieren herramientas de IR para hacer su vida más fácil ("no muy dura")
- Deben descargar datos y juntar las piezas: que datos va con qué codigo?
- Entonces =¿ Descargan los datos y analizan, juntan el software y ejecutan

Elementos del Análisis de Datos

- Datos
  - Datos crudos (raw)
  - Datos procesados
- Figuras
  - Figuras exploratorias
  - Figuras finales
- Código
  - Scripts borrador/no usados
  - Scripts finales (R, Julia, Python ...)
  - Scripts Literarios
    - Archivos .rmd (R markdown)
    - Archivos .ipynb (Jupyter/IPython notebooks)
- Texto:
  - Archivo de ayuda (README)
  - Texto del análisis/reporte (pdf, html) generado comunmente por herramientas de programación literaria

Estructura mínima de un reporte de Análisis de Datos

- Titulo
- Introducción/motivación
- Metodos (estadísticos)
- Resultados
- Conclusiones

Contenido

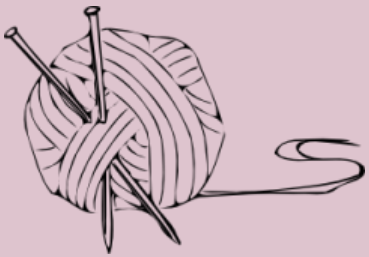
Entronos y herramientas modernas

Jupyter(IPython Notebook)

- Creado por Fernando Perez (estudiante de ingeniería aeronáutica)
- Entorno completo de computación interactiva en el browser
- HTML o Markdown + Python o Julia
- Permite binding hacia otros lenguajes como: R, ruby, shell, ...
- Exporta los documentos resultantes hacia: PDF, HTML, Latex o reveal.js
- **nbviewer** [`http://nbviewer.ipython.org/`] el servicio de publicación gratuito de notebooks en la Web

Entornos y herramientas modernas

knitr + RStudio



- **knitr** desarrollado por Yihui Xie al realizar su trabajo de graduación
- Es una librería escrita en R y para R
- HTML(o LaTeXo Markdown) + R
- Orientado a mitigar las limitaciones de SWEAVE.
- Puede exportar a PDF, HTML o Word
- **knitr** es integrable con el IDE **RStudio**
- **Rpubs** [`http://rpubs.com`] su servicio de publicación gratuito de RStudio en la Web

Entornos y herramientas modernas

git

- **D**esarrollado por Linus Torvalds en el 2005
- Herramienta de control de versiones distribuida
- Rápido y pequeño
- Abundantes comandos que permiten el trabajo colaborativo
- Cuarda el rastro de los cambios muy detalladamente
- Git LFS: extensión para manegar archivos de gran tamaño

Entornos y herramientas modernas

github

- Originalmente conocida como Logical Awsome
- Plataforma de desarrollo de software colaborativo, pero, con millares de repositorios que contienen codigo y datos de experimentos reproducibles
- El código se alamacena de forma pública con la posibilidad de hacerlo privado
- Utilizado en academia e investigación: escuela tradicional, MOOCS, ciencias,..
- Github alcanzó **1 millón** de repositorios en el **2010**, **2 millones** en el **2011**, **10 millones** en el **2013** y llegando a **21 millones** de repositorios en Abril del **2015** y **9 millones de usuarios**

Contenido

Demos

Créditos

Expositor
Ing. Milton Leonardo Labanda Jaramillo, Mg.

@miltonlab



UÍDE

Universidad Internacional del Ecuador sede Loja
Escuela de Informática y Multimedia
2015