

Project Plan

Ricardo R. Palma

2023-08-15

Paquete plans R-Cran

autor: **daniel kelley** 2022-04-06

1. Introducción
2. gráficos de quemado
3. diagramas de Gantt

Abstract.

El paquete del plan proporciona funciones para planificar y describir proyectos y monitorear el progreso hacia la finalización de tareas individuales dentro de los proyectos.

1. Introducción

Este paquete proporciona herramientas (principalmente gráficas) para la planificación de proyectos. Esta es una versión anterior del paquete, que brinda soporte para diagramas de trabajo y diagramas de Gantt.

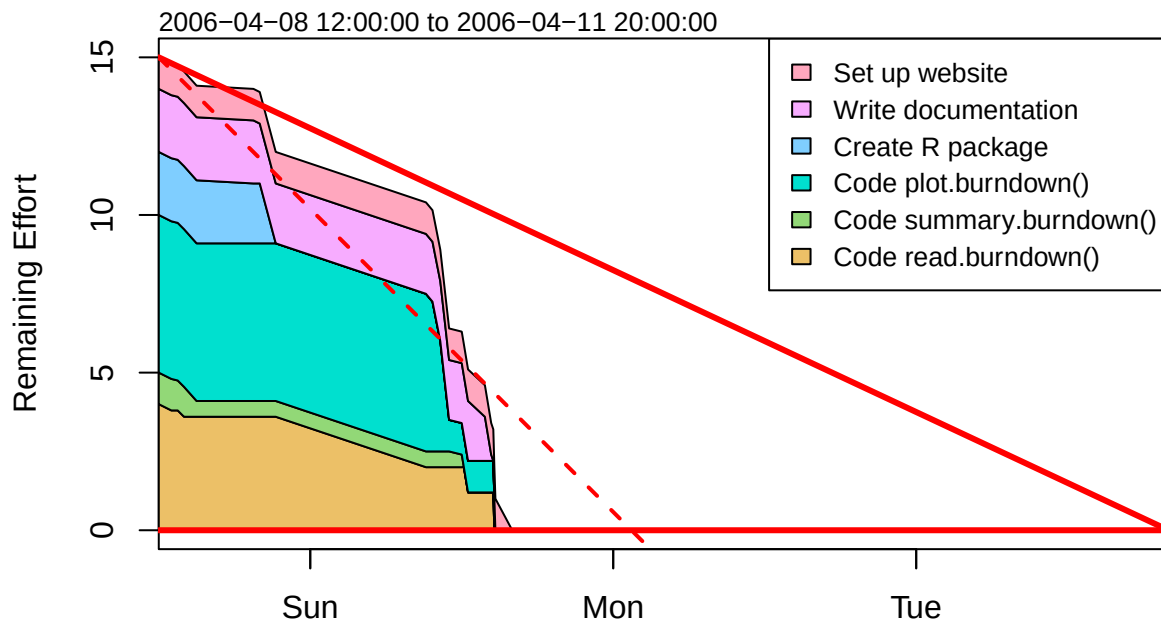
2. gráficos de Burndown

Los gráficos Burndown se utilizan para mostrar una serie temporal del progreso hacia los objetivos de un proyecto. El formato es sencillo. El eje x representa el tiempo, que va desde el inicio del proyecto hasta la fecha límite para su finalización. El eje y representa el esfuerzo restante que se requiere para realizar el trabajo. A medida que se lleva a cabo el trabajo, el gráfico proporciona un resumen continuo del progreso hacia la fecha límite. Si el trabajo avanza sin problemas de acuerdo con el cronograma, el gráfico toma la forma de un triángulo, y el esfuerzo restante cae desde su valor inicial a cero en la fecha límite.

A modo de orientación, se dibuja una línea de trazos para indicar esta situación ideal. Si el trabajo se realiza más rápido de lo esperado, los datos estarán debajo de esta línea gris. Sin embargo, si el proyecto se está retrasando, la línea de datos estará por encima de la línea ideal. Por lo tanto, un vistazo al gráfico indica si se puede cumplir con el plazo. En la mayoría de los casos, el trabajo se divide en subtareas y el esfuerzo restante en cada tarea se muestra con un color diferente en el gráfico. Esto es útil para identificar tareas que pueden necesitar más atención.

A continuación, se traza un gráfico de trabajo pendiente de muestra; reemplace la llamada `data()` con una llamada `read.burndown()`, para crear su propio gráfico.

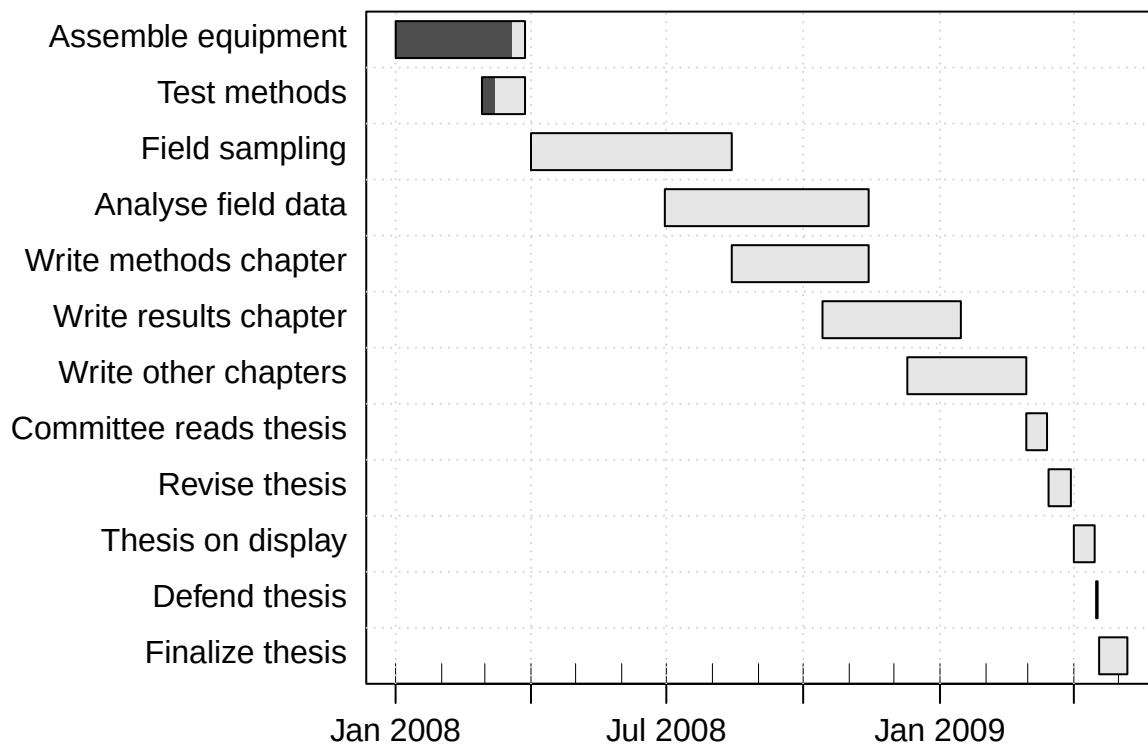
```
library("plan")
data(burndown)
plot(burndown)
```



3 diagramas de Gantt

Los diagramas de Gantt indican un cronograma para completar los componentes de un proyecto. El Gantt del conjunto de datos de muestra es un plan de investigación hipotético para un proyecto de maestría; puede ver cómo se grafica con ejemplo (`plot.gantt`), o ingresando el siguiente código.

```
library("plan")
data(gantt)
plot(gantt)
```



Yendo más allá de este conjunto de datos incorporado, tenga en cuenta que hay 3 formas principales de crear objetos de clase Gantt: (a) escribir información en un archivo de texto y usar `read.gantt()`, (b) usar `as.gantt()` para ensamblar el objeto en un paso (algo complicado), o (c) use `new()` para crear un objeto y luego `ganttAddTask()` para agregar tareas una por una. El tercer enfoque puede ser el más fácil para los principiantes, por lo que se ilustra a continuación, con un esquema de un programa típico de maestría en oceanografía. Tenga en cuenta que se agrega una leyenda, junto con una indicación del momento en que se preparó el gráfico.

```
library("plan")
g <- new("gantt")
g <- ganttAddTask(g, "Módulo TyHM II") # no times, so a heading
g <- ganttAddTask(g, "Physical Oceanography", "2023-09-03", "2023-12-05", done=100)
g <- ganttAddTask(g, "Chemistry Oceanography", "2023-09-03", "2023-12-05", done=100)
g <- ganttAddTask(g, "Fluid Dynamics", "2023-09-03", "2023-12-05", done=100)
g <- ganttAddTask(g, "Biological Oceanography", "2024-01-03", "2024-04-05")
g <- ganttAddTask(g, "Geological Oceanography", "2024-01-03", "2024-04-05")
g <- ganttAddTask(g, "Time-series Analysis", "2024-01-03", "2024-04-05")
g <- ganttAddTask(g, "Redacción Tesina") # no times, so a heading
g <- ganttAddTask(g, "Literature review", "2023-09-03", "2024-02-01", done=20)
g <- ganttAddTask(g, "Develop analysis skills", "2023-09-03", "2024-08-01", done=30)
g <- ganttAddTask(g, "Thesis work", "2023-10-01", "2024-09-01")
g <- ganttAddTask(g, "Defend thesis proposal", "2024-05-01", "2024-06-01")
g <- ganttAddTask(g, "Write papers & thesis", "2024-03-01", "2024-04-01")
g <- ganttAddTask(g, "Defend thesis", "2024-09-01", "2024-10-15")
font <- ifelse(is.na(g[["start"]]), 2, 1)
plot(g, ylabel=list(font=font),
     event.time="2024-01-01", event.label="Report Date")
par(lend="square") # default is round
legend("topright", pch=22, pt.cex=2, pt.bg=gray(c(0.3, 0.9)),
     border="black", xpd=NA,
     legend=c("Completado", "Por realizar"), title="Diploma Ingeniero", bg="white")
```

