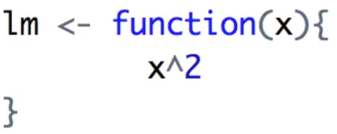
**Semana 2**

**11. Reglas de Alcance**

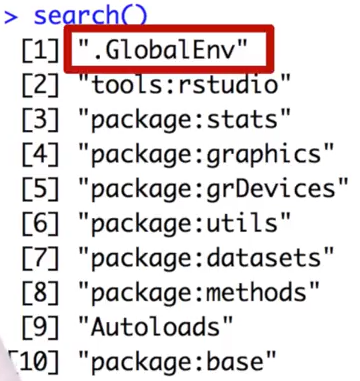
En esta lección aprenderemos las regla de alcance, que son la principal diferencia entre el lenguaje R y el lenguaje S, y determinan cómo se asocia un valor con una variable libre en una función en R. Las variables libre son variables que no aparecen en la lista de argumentos formales de una función. R utiliza el alcance léxico.

¿Cómo sabe R que valor asignar a un símbolo? Hay una función *lm()* dentro del paquete stats que recibe un parámetro y regresa el valor según la operación que hayamos definido.

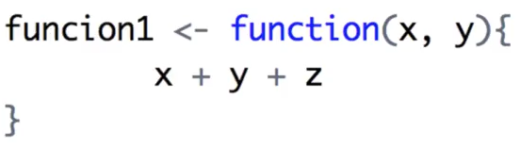


Cuando R trata de ligar un valor a un símbolo hace una búsqueda en una serie de ambientes.

Cuando queremos traer el valor de un objeto primero busca en el ambiente global un símbolo con un nombre igual al que le solicitan, luego busca en los espacios de nombres de la lista de búsqueda (que se puede buscar con el comando *search()*). El ambiente global es siempre el primero que se busca.



El paquete base es siempre el último en la lista de búsqueda. El orden de los paquetes en la lista es el orden en el que se hará la búsqueda. Cuando cargamos un paquete con la función *library()* se pone por omisión segundo en la lista y los demás se desplazan una posición hacia abajo.



En este caso, la función z es un valor libre, y si lo ejecutamos, sale un error porque z no está definido, pero si lo definimos antes de ejecutar el comando, entonces ya no nos da dicho error. EL valor de z dentro de la función es el valor que encuentra primero en el ambiente global. Esto es justamente de lo que trata el alcance léxico, que indica que los valores de las variables libres se buscan en el ambiente en que la función fue definida. Un ambiente no es más que un conjunto de parejas de símbolos y valores, teniendo cada ambiente un ambiente padre y contiene múltiples ambientes hijos. El único ambiente sin padre es el ambiente vacío.

Si el valor de un símbolo no se encuentra en el ambiente, se busca en el padre, y así hasta el más alto, el ambiente global. Tras llegar al ambiente más alto llega hasta el ambiente vacío. Esto es lo que pasó en el caso anterior, que buscó hasta llegar al ambiente vacío.

*¿Qué sucede cuando se encuentra, en una función, una variable libre? En su forma más básica, R toma su valor del ambiente en el que fue definida la función.*

Esto típicamente no tendría sentido, pero en R podemos definir funciones dentro de otras funciones. Podemos ver de manera más clara lo que sucede dentro de un ambiente con la función *environment()*. Si usamos *ls(environment(“función”))*, nos indica los valores que hay dentro. Si llamamos con la función *get()* nos indica los valores dentro de una función.

Es importante la manera en la que R lleva la evaluación y las reglas de alcance. Es importante que no se repitan los nombres de variables y funciones. Finalmente siempre es importante tener en cuenta esto cuando se va a modificar un código.