onevector.r

denis

2021-07-13

#!/usr/bin/r  
  
# 22  
# 2 Vectors and Vector Spaces  
# is a normalized vector. Normalized vectors are sometimes referred to as “unit  
# vectors”, although we will generally reserve this term for a special kind of   
# normalized vector (see page 12). A normalized vector is also sometimes   
# referred o as a “normal vector”. I use “normalized vector” for a vector such   
# as x̃ in equation (2.21) and use the latter phrase to denote a vector that is  
# orthogonal to a subspace.  
norms <- vector(mode = "logical", length = 0L)  
norms

## logical(0)

# 2.1.7 Metrics and Distances  
# It is often useful to consider how far apart two vectors are; that is, the   
# “distance” between them. A reasonable distance measure would have to satisfy  
# certain requirements, such as being a nonnegative real number. A function ∆  
# that maps any two objects in a set S to IR is called a metric on S if, for all  
# x, y, and z in S, it satisﬁes the following three conditions:  
Matrix::as.matrix(norms)

## [,1]

# These conditions correspond in an intuitive manner to the properties we ex-  
# pect of a distance between objects.  
object.size(norms)

## 48 bytes

# Metrics Induced by Norms  
# If subtraction and a norm are deﬁned for the elements of S, the most common  
# way of forming a metric is by using the norm. If · is a norm, we can verify  
# that  
Matrix::as.array(norms)

## logical(0)

# is a metric by using the properties of a norm to establish the three   
# properties of a metric above (Exercise 2.7).  
c(norms, sigSlots = c("target", "defined"))

## sigSlots1 sigSlots2   
## "target" "defined"

# The general inner products, norms, and metrics deﬁned above are relevant  
# in a wide range of applications. The sets on which they are deﬁned can consist  
# of various types of objects. In the context of real vectors, the most common  
# inner product is the dot product; the most common norm is the Euclidean  
# norm that arises from the dot product; and the most common metric is the  
# one deﬁned by the Euclidean norm, called the Euclidean distance.  
c(norms)

## logical(0)

# 2.1.8 Orthogonal Vectors and Orthogonal Vector Spaces  
# Two vectors v 1 and v 2 such that  
# v 1 , v 2 = 0  
# (2.23)  
# are said to be orthogonal, and this condition is denoted by v 1 ⊥ v 2 .   
# (Some-  
# times we exclude the zero vector from this deﬁnition, but it is not important  
orthogonal <- c(v = 1, v = 2, v = 3)  
orthogonal

## v v v   
## 1 2 3

# 2.1 Operations on Vectors  
# 23  
# to do so.) Normalized vectors that are all orthogonal to each other are called  
# orthonormal vectors. (If the elements of the vectors are from the ﬁeld of com-  
# plex numbers, orthogonality and normality are deﬁned in terms of the dot  
# products of a vector with a complex conjugate of a vector.)  
options("orthogonal")

## $orthogonal  
## NULL

# A set of nonzero vectors that are mutually orthogonal are necessarily lin-  
# early independent. To see this, we show it for any two orthogonal vectors  
# and then indicate the pattern that extends to three or more vectors. Sup-  
# pose v 1 and v 2 are nonzero and are orthogonal; that is, v 1 , v 2 = 0. We   
# see  
# immediately that if there is a scalar a such that v 1 = av 2 , then a must be  
# nonzero and we have a contradiction because v 1 , v 2 = a v 1 , v 1 = 0. For   
# three  
# mutually orthogonal vectors, v 1 , v 2 , and v 3 , we consider   
# v 1 = av 2 + bv 3 for a  
 #or b nonzero, and arrive at the same contradiction.  
set.seed(c(v = 1, v = 2, v = 3))  
orthogonal + sin(2)

## v v v   
## 1.909297 2.909297 3.909297

# Two vector spaces V 1 and V 2 are said to be orthogonal, written V 1 ⊥ V 2 ,  
# if each vector in one is orthogonal to every vector in the other.   
# If V 1 ⊥ V 2 and  
# V 1 ⊕ V 2 = IR n , then V 2 is called the orthogonal complement of V 1 ,   
# and this is  
# written as V 2 = V 1 ⊥ . More generally, if V 1 ⊥ V 2 and V 1 ⊕ V 2 = V, t  
# hen V 2 is  
# called the orthogonal complement of V 1 with respect to V. This is obviously  
# a symmetric relationship; if V 2 is the orthogonal complement of V 1 , then   
# V 1  
# is the orthogonal complement of V 2 .  
vec <- c(orthogonal, norms)  
vec

## v v v   
## 1 2 3

# If B 1 is a basis set for V 1 , B 2 is a basis set for V 2 , and V 2 is the   
# orthogonal  
# complement of V 1 with respect to V, then B 1 ∪ B 2 is a basis set for V.   
# It is  
# a basis set because since V 1 and V 2 are orthogonal, it must be the case that  
# B 1 ∩ B 2 = ∅.  
# If V 1 ⊂ V, V 2 ⊂ V, V 1 ⊥ V 2 , and dim(V 1 ) + dim(V 2 ) = dim(V), then  
# V 1 ⊕ V 2 = V;  
dim(norms)

## NULL

# that is, V 2 is the orthogonal complement of V 1 . We see this by ﬁrst letting   
# B 1 and B 2 be bases for V 1 and V 2 . Now V 1 ⊥ V 2 implies that B 1 ∩ B   
# 2 = ∅ and dim(V 1 ) + dim(V 2 ) = dim(V) implies #(B 1 ) + #(B 2 ) = #(B),   
# for any basis set B for V; hence, B 1 ∪ B 2 is a basis set for V.  
vb <- c(b = 1, b = 2, v = 1, v = 2)  
dim(vb)

## NULL

vb

## b b v v   
## 1 2 1 2

# 2.1.9 The “One Vector”  
# Another often useful vector is the vector with all elements equal to 1. We   
# call this the “one vector” and denote it by 1 or by 1 n . The one vector can   
# be used in the representation of the sum of the elements in a vector:  
c("https://tidyselect.r-lib.org/reference/faq-selection-context.html",   
 vars = NULL)

##   
## "https://tidyselect.r-lib.org/reference/faq-selection-context.html"

# The one vector is also called the “summing vector”.  
summary(.data = "dplyr", vb, groups = NULL)

## Min. 1st Qu. Median Mean 3rd Qu. Max.   
## 1.0 1.0 1.5 1.5 2.0 2.0