27. mapply(), always remember to set SIMPLIFY = F or T!  
  
29. When plot() not working, try pdf();dev.off()  
  
36. print multiple plots in one pdf in R:  
pdf(..)  
for(i in 1:10)plot(..)  
dev.off()

37. When using R optimizer, usually "L-BFGS-B", don't initialize the parameters with the same numbers! namely, don't do something like: optim(par=c(1,1,1,1),fn=f,method="L-BFGS-B",lower=rep(0.1,4),upper=rep(10,4)), because somehow it just doesn't get right. Randomization of initial values is your friend. Be mindful of the step sizes setting.

35. olivedrab2 is a good series of colors in R of green.

49. When rbind() 2 matrices or data frames, be careful R will match the names of the vector, so take out the names first!!

65. In that R function density is also called "window", which is easier to understand the nature of kernel!

87. This is it: the optimization algorithm in R can't take a function written in C++ (in older versions of R?)...

90. You are probably doing something wrong if the numeric error is greater than 1e-7. Don't doubt too much on the numeric precision!!

102. just remember, applying ded and limit is x=pmin(pmax(x-d,0),L)

106. the do.call() can be used to order a data frame by all the columns in them. do.call(order, as.data.frame(mat))

111. When using optim(), always think about setting control=list(ndeps=c(1e-8,1e-8..)) to get it right

113. use library snow for parallel pure r computing  
core = 8  
cl = snow::makeCluster(maxCore, type="SOCK")  
tmp = lapply(1: maxCore, function(x) rnorm(100000000))  
optResult42 = unlist(snow::clusterApply(cl, tmp, function(x) sum(x)))  
snow::stopCluster(cl)

114. expression("multiplier =" ~ w^4); expression(Total[100~'%'])

117. plot 3d wireframe surface best practice: use rgl, and do 2 things:  
persp3d(x=unique(tmp$x),y=tmp$y[1:tmpN],z=matrix(tmp$P,ncol=tmpN),col="red",alpha=0.75)# plot surface first  
persp3d(x=unique(tmp$x),y=tmp$y[1:tmpN],z=matrix(tmp$P,ncol=tmpN),col="black",add=T,front="line",back="line")# plot the line secondly

130. draw ellipses, use r package "mixtool"

133. what a good function: scan(): read in unequal length of data rows! Nice!

136. bug: Jesus Christ! when I was coding the comparison function for sorting multi-dimensional vector, big mistake in the comparison algorithm. Remember, when the 2 things are equal, still progress! please read the function of sorting multidimensional grid index for hierarchical clustering! WARNING: sort(iterator begin, iterator end, compare), compare returns true if and only if a[i] is STRICTLY less than a[j].

137. Just use aggregate in R.. nice and easy. something like aggregate(tmp[3],by=tmp[1],function(x)sum(x)) where tmp is a data frame with names. In such way tmp[3] and tmp[1] are data frames with names, so the resultant data frame will have names

143. R color brewer: nice !! : plot(1:100, col = colorRampPalette(c("red", "white", "blue"))(100), pch=16)

155. r plot cancel axis: xaxt='n', yaxt='n' and the default value is 's'.

156. r plot set margin and 2 plots in a row: par(mar=c(4,4,0.5,0.1),mfrow=c(1,2)). Set plot height and width or aspect ratio: pdf(path, width=16, height=9) and the unit is in inches, but doesn't matter..  
  
157. r plot transparency: use package scales. Do it like plot(col=scales::alpha("red", 0.5))  
  
158. r plot change the size of xlab and ylab: plot(cex.lab=2) and main cex.main and cex.axis... good good!  
  
159. r plot cancel the legend box: legend(bty = "n"), or plot border: plot(bty="n")

166. About axis and labels positions:  
par(mar, mgp, las) and <http://rfunction.com/archives/1302>

168. r plot transparent: col=scales::alpha('skyblue',1)

171. r plot do automatic math expression:  
for(i in 1:nrow(coor))  
{  
  tmp = paste0("expression(sigma[", i ,"])")  
  text(x = coor[i, 1], y = coor[i, 2], labels = eval(parse(text = tmp)), cex = 1.2)  
}

176. r print significant digits: sprintf("%.10f",0.25)

178. Change the font family in r plot text: No! Just do plot(family = "serif")

179. remove r plot inner margin: plot(xaxs="i", yaxs="i")

181: debug: in R, division always results in numeric! E.g. 20L / 2L = 10 where 10 is a numeric!

182: whenever something like R CMD commands are needed, do it like system("commands"). Great! --- <https://www.r-bloggers.com/my-r-package-development-cheat-sheet/>  
Sys.setenv(PATH = paste(Sys.getenv("PATH"), "C:\\Program Files\\MiKTeX 2.9\\miktex\\bin\\x64", sep=.Platform$path.sep)); system("R CMD Rd2pdf C:/Users/i56087/Desktop/clusteringPip") prints the package manual in PDF.

191. create pdf r package manual: system("R CMD Rd2pdf C:/Users/i56087/Desktop/CvgCor/2ndPhase/coverageCorrelation")  
  
192. Re position xlab or ylab: title(ylab="Frequency", line=-2, cex.lab=2). The negative "line" argument places the label inside the plot. distance between label and axis  
  
193. Now let's talk about plot golden ratio: for pdf(), set width 12~13 and height proportionate to width, and then set the lab font size cex.lab to 2, axis font size cex.axis to 1.5  
  
194. r plot default margins: c(5, 4, 4, 2) + 0.1

196. image matrix: to plot a heat map whose proximity would fit a matrix M printed on the R console, do  
image( t(M) [, nrow(M) : 1L] )

197. multiple plots plot column first: mfcol = c(3, 4). This makes R generate a plot matrix of 3 rows, 4 column, but R will fill the plot matrix by column.

205. Get parent directory path in R: dirname(getwd())

206. plot matrix, each row has different number of plots: layout(matrix(c(1, 2, 3, 3), ncol = 2, byrow = T), widths=c(1, 1), heights = c(1, 1))

216. Line break in rd file: \cr

218. paste expression: bquote(rho == ~ .(round(testCor[i], 2)))

224. r plot vertical: las = ...

225. Rotate axis labels example:  
axis(side = 1, at = sp, labels = F); text(x = sp, y = par()$usr[3] - 0.05 \* (par()$usr[4] - par()$usr[3]), labels = cty, srt = 30, adj = 1, xpd = T)

226. stats::optim() can take Rcpp function, and its performance is as fast as lbfgs::lbfgs(), but the problem comes when multithreading. The only way out is building a package and load to processes

229. r plot background color reset plot region only:  
rect(par("usr")[1], par("usr")[3], par("usr")[2], par("usr")[4],col = "gray")

246. List files without subdirectories:  
setdiff(list.files("targetDirectory"), list.dirs("targetDirectory", recursive = F, full.names = F))

249. Rd file math notations: to this end, the two commands \eqn{latex}{ascii} and \deqn{latex}{ascii} are used. Whereas \eqn is used for "inline" formulae (corresponding to TeX's $…$), \deqn gives "displayed equations" (as in LaTeX's displaymath environment, or TeX's $$…$$). Both arguments are treated as verbatim text. <<http://www.hep.by/gnu/r-patched/r-exts/R-exts_58.html>>

256. rhub::check\_for\_cran(). CRAN package online check.

262. Windows command compress, copy, move, in R: system("powershell Compress-Archive C:/Users/i56087/Desktop/NScorrCluster/tmpData/data C:/Users/i56087/Desktop/NScorrCluster/tmpData/data.zip");  
  
system("powershell move C:/Users/i56087/Desktop/NScorrCluster/tmpData/data.zip //RSGrid/finance\_develop/Charlie/selectedKernels001supervisingVarRnn");  
  
system("powershell Copy-Item 'C:/Users/i56087/Desktop/EQopt/tmpData/data.Rdata' -Destination '//RSGrid/finance\_develop/Charlie/eqcatopt100kTo10k/data'")

276. Try floor(0.0012 \* 10000) in R and see annoying it is. floor(), ceiling(), round() are all not trustworthy!

283. In the future always think about system("python ...", wait = F) or system("Rscript ...", wait = F) to perform parallel computation!

285. par("usr") shows the current plot's left x, right x, bottom y and upper y.

286. R command line arguments with spaces: system('Rscript R/floodScript001.R "asda asdasd" "ad- 12"'). And in the script, args = commandArgs(trailingOnly = TRUE)

287. mfrow <=> mfcol

289. In pipeline programming with R, avoid apply() as much as possible because it collapse to vector if matrix has only one row or column. Use lapply() coupled with as.data.frame(), data.frame(), df[, , drop = F], aggregate(, , , simplify = F, drop = F)

290. tmp = system("wmic cpu get loadpercentage", intern = T) . To stream the text output on console to a R variable.

291. The essential difference between the set and the multiset is that in a set the keys must be unique, while a multiset permits duplicate keys

294. keras::predict\_on\_batch(model, newdata) = predict(model, newdata)

298. In R, plot the perpendicular or \perp, do: expression(F[X^symbol("^")][', '][Y^symbol("^")])

299. In R plot, making pch symbol thicker without drawing a line in legend, use pt.lwd

300.

# Install pre-compiled R:

# 1. Go to <https://docs.rstudio.com/resources/install-r/>  
# 2. Select the OS and R version.  
# 3. Download the .deb file via Windows Chrome. If it can be downloaded directly in Ubuntu via curl without certification crap, good.  
# 4. Access the .deb file in /mnt/c/Users/i56087/Download, and then install it following the instructions on the website.

# 5. When adding the executable to system path, the folder on Ubuntu 20.4 on WSL2 should be /usr/bin, not /usr/local/bin:

sudo ln -s /opt/R/4.1.2/bin/R /usr/bin/R  
sudo ln -s /opt/R/4.1.2/bin/Rscript /usr/bin/Rscript

Sometimes this does not work due to remains of previous R versions. In this case do sudo rm /usr/bin/R and sudo rm  /usr/bin/Rscript first.  
  
# Install R, Build R from source on Ubuntu 20.04: (not recommended. See above to install R binaries)  
# Download R-4.0.5.tar.gz  
  
sudo tar -xzvf R-4.0.5.tar.gz  
cd R-4.0.5  
sudo apt install gcc g++ make gfortran libreadline-dev libxt-dev zlib1g-dev libbz2-dev libcurl4-openssl-dev  
./configure  
make  
sudo make install  
  
301. WSL2 Ubuntu WSL2 R install.packages() does not work due to certificate crap: in R try options(download.file.method="curl", download.file.extra="-k -L")

Or add this line to /opt/R/4.1.2/lib/R/library/base/R/Rprofile using vim.

Make sure curl is installed or maybe libcurl4-openssl-dev.

302. DO NOT TRY EDITING FILES IN UBUNTU VIA WINDOWS APP LIKE NOTEPAD++. ALWAYS DO IT IN vim !!!!! UNEXPECTED ERRORS WILL DESTROY THE SYSTEM !!!!!

303. In package plot3D, nticks and ticktype controls the labels on the axes, e.g., ticktype = "simple", nticks = 3. And tune r = 10 to tune the distance between the plot and the eye.

304. In R, just fucking stick to x[, , drop = F], aggregate(, , , simplify = F), mapply(, , SIMPLIFY = F), FOREVER !!!!!!!!!

305: format(Sys.time(),usetz = TRUE) will keep the time zone in the string of time.

306: system("robocopy sourceDirPath destinationDirPath /E /xo"): if destinationDirPath does not exist, it will be created; if a file in destinationDirPath has the same name but older modified date, it will be replaced with the newer version in sourceDirPath; if sourceDirPath has subdirectories, they will be copied to the destination too.

307: Finally, a nice and simple way to do OOP in R: use environment. Accessing variables in the environment uses hashing, and will be faster than accessing names of a list. Moreover, when returning an environment from a function, the object's method still works.

tmpa = runif(10)

f = function() { # Constructor.

tmp = new.env()

tmp$a = tmpa # Member definition.

tmp$b = function(x) { sum(tmp$a + x) } # Member function definition.

tmp # Return the object created.

}

# An object (environment) MUST be created inside a function scope,

# otherwise it won’t work.

x = f() # Create the object that has member a and b

x$b(1:10)

sum(x$a + 10) # Equal to the above.

x$a = 1:10 # You can modify the object’s member.

x$b(11:20)

# One can also save x, and load back it in a new R session no problem.

308. Always do image(useRaster = T) to prevent weird lines in images plotted on Linux clusters.

309. file.show(path/to/file) will open the file in Rstudio.

310: in Rstudio,

Shift + Alt + R to open a new terminal

Ctrl + 1 to focus back to the editor window  
Ctrl + Alt + Enter to send commands to be executed directly to the Terminal, but you went to Tools-->modify keyboard shortcuts, and changed it to Shift+Enter.

And you also set Ctrl + Q to focus on the terminal.

311. utils::capture.output() will capture the printout during running an expression, e.g., ot = utils::capture.output({Rcpp::sourceCpp(...)})

312. try and tryCatch: Note that tryCatch(expression, error = fun(cond), warning = anotherFun(cond), finally = expressionThatWillBeExcutedREGARDLESSofSuccessFailure)

313. system() returns 0 if everything runs fine.

314. .rs.restartR() will restart R inside Rstudio.