Cyberseminar Transcript

Date: September 10, 2020

Series: VA Informatics and Computing Infrastructure

Session: Efficiently Accessing VINCI Data in R

Presenter: Andrew Redd, PhD

*This is an unedited transcript of this session. As such, it may contain omissions or errors due to sound quality or misinterpretation. For clarification or verification of any points in the transcript, please refer to the audio version posted at* [http://www.hsrd.research.va.gov/cyberseminars/catalog-archive.cfm](file:///C:\Users\VHASLCMyersK\AppData\Local\Microsoft\Windows\Temporary%20Internet%20Files\Content.IE5\73RXHSNF\l)

Dr. Andrew Redd: Okay, so I’m Andrew Redd to let you know. I have going on around 15 years of experience with R. And I have been with the VA, and I'm housed at the University of Utah. Here I'm working with the VA and VA data for about the last 10 years. So I've got a fair amount of experience. I've recently joined the VINCI helpdesk team to help with situations regarding R and R data. And I've already gotten several similar problems and complaints. Tickets that I've helped field. So I'd like to go over using R efficiently on VINCI. I’ll focus primarily on getting and working with data but there are some other tips and tricks that I will go over.

So let's talk about some of the common complaints that I have seen. Some of these are unavoidable issues. There is connection issues, internet issues, getting to VINCI is oftentimes difficult. For me I have to go through the CAG and go through layer after layer after layer. This compounds problems of connection, and if you're disconnected getting back into things can be frustrating and aggravating. And I certainly hear those complaints because I've got those complaints as well. And when VINCI isn't working well after all of the trouble to get into VINCI it can be frustrating. And when you are disconnected from VINCI, and if it's over night or things servers can get restarted, which means getting back into work can get complicated. So I'm going to give you some tools today that will help alleviate a lot of that frustration. Or at least it has for me.

So using R on VINCI. I hope everyone in the audience today has worked with R on VINCI. If you are a new R user welcome. R is in my opinion definitely the place to be. And VINCI has some good infrastructure for dealing with R on VINCI. Now I recommend using RStudio, and particularly I recommend using RStudio with Projects. If you are not familiar with the Projects on RStudio, I will cover those a little bit. But I definitely encourage you to read up on that. I will be talking about some specific packages that we use, that using them on VINCI will make your life much better. And just using them in R is good practice and will make your life better, even if you’re not using them on VINCI itself. So the first one that I will talk about is a specific VINCI package. This is going to be kind of a preview as it’s not quite on global release yet, but we are working towards that goal. The second is, Tidyverse. Dplyr, dbplyr, odbc, DBI connections we’ll go into details of all that. So.

Okay. Accessing RStudio on VINCI. It is accessible either through standard applications directly, or through the remote desktop applications once you are logged into what I would call the terminal node—I’m not sure that’s the correct name for it. The kind of remote desktop inside of VINCI. If you’re using the standard workspace, it doesn’t matter which direction you access RStudio from, because both are running on the same servers. There are two servers that are BP for running RStudio or R instances on it. And when you reconnect to a session, you’ll be reconnected to the same R server.

Okay, so let’s talk about Projects. So leveraging Projects in RStudio will allow you to organize all of your work. It will also allow you to properly leverage that information about the Project to your advantage. To create a new Project you’re going to use either the new Projects button, which is on the toolbar. It looks like this with the plus with the R in a box. That’s the Project icon for RStudio. You can also do it from the file new Project menu. Or you can use the Project menu, which is on the top, on the righthand side on the far right of the toolbar. It has a little drop-down menu. It shows the current projects here. So this would be for the Project, this would be for the VINCI Project, it is the VINCI package that I’ll show a little bit later. Then there’s the new Project, open Project in a new session so you can have multiple sessions open. And I do this all the time, so it can be working on, so I’m working on multiple projects for multiple different investigators at the same time. And then it has a list of recent projects. You can clear the project list and then change the options for that project, and you can get to the project options a couple of different ways as well.

I’m going to jump back to a slide for just one really quick second to answer one of the questions. You can download today’s PowerPoint slides from this Tinyurl. I’ll give you just a second to look at that. Hopefully everyone can copy that down and get to the slides. I don’t want anybody to be left behind if you need the slides to follow along.

Moderator: Dr. Redd, I’ll send out the direct link right now.

Dr. Andrew Redd: That would be fabulous, thank you. Okay so there are a lot of advantages to using Projects. One is that it encapsulates and organizes all of your files into one. So very rarely do I use a project where I’m only using one file. I very often have files that will create an analysis dataset, one that performs an analysis, one that shows plots for it, or does summarizations. Like table 1 or the demographics table. Each file does its job. And it’s nice to keep them separate. But also, I have data files that go in, I have output files that are tables or graphics. All of those things can be kept into once place. And working in that it also manages your working directory. So that all the file input and output is relative to that root directory that you’re working from. It just makes things easier all around. It makes it easier to run scripts. It makes it easier to rebuild reports. All of that is an advantage. So if you’re not using Projects, use them. Because it makes your life a lot easier.

Okay. Now a quick comment about packages on VINCI. Now I’m talking about R packages of course. When R is updated on VINCI, all the packages on CRAN at that time that R is updated are installed with R. VINCI IT manages that for us. But that’s not always enough for individuals needs for research. Very often times you need updated packages that were not available on CRAN at the time. Or you’re writing your own packages. It is acceptable to upload those packages to VINCI and install them manually. Now to do that, to get an updated package you’re going to want to go to CRAN, download the Windows binary version of that package and any of the dependencies that it may need. So oftentimes an updated package will depend on updated dependencies. You’re going to download all those, upload them into VINCI. Then install them into an R user or a project library as a binary package. I’m not going to get into the details of how to manage your project libraries. I might be able to, I might have time for that later. If there’s time and there’s interest in looking at how to manage multiple libraries, I’ll be happy to go into it. If not, read up on the dot libPaths and the dot Rprofile files to be able to learn how to manage those more appropriately.

Okay, now VINCI is, the VINCI R package is package that is context aware specifically for VINCI. A lot of the frustrations that I have come across when I’ve been doing my analysis. I have packaged into this package, and I’m working with VINCI IT to try and get this into globally installed. Because it doesn’t have any purpose to be on CRAN, there’s no purpose in putting it on CRAN. So it’s not going to be automatically installed. But we’re working on getting a global installation put into that. But this is going to act as a preview for how it can make your life easier in research.

Now the Tidyverse. If you’re not versed in the Tidyverse, you don’t know about the Tidyverse, read up on it. Tidyverse dot org, or R for Data Science. And you know if you have the slides, those are clickable links. If not search for R for Data Science by Hadley Wickham. He’s the chief data science officer for RStudio. He is the grand architect of the Tidyverse, writing all the dplyr and plyr and everything else that is wonderful in the Tidyverse. He’s the guy who designed all of that. Now the Tidyverse is big. I’m going to focus on data flow portions of it for today.

Okay. So you need to know dplyr to understand how to appropriately leverage the tools that are available to easily access VINCI data. So the dplyr, it provides and abstraction layer to data processes and data manipulation. So like the filtering or a sub-setting operation, selecting variables or features of data. Joining, merging, transforming data. Renaming data. All of that is what dplyr abstracts so that you can manipulate all of your data in the same way. Whether it’s local data, remote data, cloud data, whatever. All you have to do is have an appropriate adapter to implement those methods for the data that you’re looking for. Dplyr does implement it for the traditional in-memory data.

Now, when I say in-memory data, R works in a manner where it assumes that you’re working with data in-memory. If you can’t fit it in, if you have more data than you can fit in your memory, you can’t work on it in R. That’s the restriction of R, that’s the way it was originally designed. So we have to make tools to be able to work with data. It’s either too big to work in memory, or not local to work in memory. Not local on the files. And that’s where dbplyr comes in. So dbplyr is the database plyr so the database toolset, and all it does it extends dplyr operations to work on remote SQL relational database files and tables. DBI is a common interface that unifies all of the database platforms. So you’ve got Microsoft SQL server which VA is using. There are Oracle databases. There are

PostgreSQL databases. You’ve got local databases like MySQL, or SQLite is what I was thinking, and there’s MySQL which is a remote database as well. And they all have different flavors. DBI is a unifying interface across all of them. And then odbc is the specific one that implements DBI’s methods for odbc connections, which SQL server is an odbc compliant database server. So that is the one that is recommended. Now a quick comment about to try and fend off questions that may come about this. There is an R odbc package that predates the odbc package. And oftentimes when you’re looking for a package an old package is a good indicator that it is mature and there are a lot of users of R odbc. However that R odbc is not compliant with the DBI interface. So the odbc package is supported by RStudio, it has contributions from Google. It’s very well implemented in most cases. There are some improvements that can be made to it, and it’s being improved all the time. So if you are in the habit of using the R odbc package, you should change to the odbc package. It’s where the active development and bug fixes are coming through.

Which brings me to, use the right tool for the job. R is great at complicated algorithms like fitting models, doing plots, all that wonderful stuff that we love R for. However, it’s not always the best for managing lots of big data. Doing things like merges. That stuff is not what R was designed for. But SQL is great at that. It’s great at data management tasks, because that’s what it was for. So but it sucks if you’re trying to do complicated things. You can’t make plots in SQL, at least I’ve never seen it. If somebody has done it, that’s fantastic. But use the tool for the job. And the wonderful thing, SQL server is actually getting some R implementation in it. So some of the latest versions of SQL server have ways to run R code on the database, on the server. R is the same way in where you can leverage dbplyr to execute SQL statements on a server. Which is just fantastic. So all of your data management can be done in R. Now the question came in, how big is big? Well if I’m running things on my laptop, I think my laptop has 32 gigabytes of RAM. That’s big. I can certainly handle most data that I’m processing. But if I am trying to do something like look at examining terabytes and petabytes of data, not going to happen. You’re not going to be able to do that. The VA has access to billions of records. You’re not going to want to pull all of that data into R. Specifically you’re not going to want to transfer all of that data between where it’s housed and R in order to analyze it. You want to filter the data as close as you can and then bring it in. So don’t bring it into R until you can really, until you really need to bring it into R. And I think something may have screwed up on me.

There. Okay. So and when I’m saying that, don’t bring it in, R is actually officially a lazy language. Which actually does have a computer science definition for it. I’m not just calling it, I’m not just being disparaging calling it lazy. It is a lazy language in that it does not evaluate things until it really, really has to. Dbplyr takes this to a bit of an extreme. So when you are creating a table that is a dbplyr remote table what it does is it creates a pointer to that table, or a definition for that table. And then as you perform operations to it, it creates a longer and longer definition for that. But it doesn’t actually evaluate any data until you really need it to. For example, it will need to know what kind of data you’re working with. Whether you’re dealing with integers, or double precision floating point, I just call those doubles or numerics [sic], whether you’re dealing with character variables. That kind of stuff. So it will kind of evaluate a little bit to say, okay, what is this. And as soon as it knows that it will go on, and work with it appropriately. And as you combine definitions, so if you join two tables, it creates a new description of the table that says, oh this object is a join of table A and table B.

So let’s do a tangent. So we talked about tables, how do we actually access data on the VINCI database servers? So there’s three ways that you can connect to the databases. You can connect to them through RStudio data connections, you can connect to it through connection strings, or you can use the VINCI DB function. Now you don’t have to have the double colon operator. It is tradition when presenting things, or instruction to put the package name and then a double colon in the function when you’re talking about a function like that. So that’s just to let you know that it comes from the VINCI packages. If you load the library of VINCI, you don’t need to know that. I hope everybody is aware of R to that point to know those.

So let’s talk about RStudio data connections. If you’re not aware of this, this is a good thing to know about. When you connect to a database, you have options for connections. And RStudio facilitates making connections through odbc data source names when using the connections tab in the top right. I’m going to switch to a live screen and show a little bit of this. I think this will probably be the only one that I share live until we get to the end.

So hopefully everyone can see my screen right now. Here we have the RStudio window, and then in the top right there are four panes. In the top right, or the upper righthand pane you have environment, history, connections and build. So if I click on the connections, I get the new connection button and connections. If I see this, this is a connection that I’ve created previously. And it will remember it. And it has the symbol for SQL server to let me know that that’s a SQL server database. If I had a Access database it would be something different, if I had a Oracle server it would be a different thing. And I can look and see how this connection is created. There is a problem in this. So if I create a connection, I’m going to go ahead and create a new connection. And see it says connects to existing data sources. So I have my data sources from an internal database, an SQL server native client, and then there are these defined ones. VHACDW A01, A06, RB01, RB02, and RB03. These are the ones that I usually, that usually have the data. I’m not sure exactly what A01 and A06 are. What these are, are these are defined odbc connections. If we connect to one of them, we’ll go ahead and connect to RB02, connect from the R console. Going to go ahead and click okay. It creates a connection. And we get a whole ton of databases in here. I don’t have permission to read most or any of these, but I would have to keep going through show more, show more, show more until I see the database that I actually want. This is kind of tedious. But this is one way that you could connect to the data. One that I would actually have access to, and that everyone I believe has access to is the like the CDWWork. I can see okay; inside the schema I have source and it will actually take a while because or I think it’s in one of them. So this is how you would actually look at these and you can view them and stuff. It’s not always the easiest.

Now, let me go back. Okay. But it is there. And when you’re using it, it does work, but because of the amount of databases that are on the SQL servers, it is much better to actually specify the database that you want to connect to.

And so you can connect it through connection strings. I don't know why anybody would do this, but you can do this. There is a website that is actually kind of cool. Connection strings dot com slash SQL server to see a bunch of examples for how to connect, how to properly create a connection string.

I just thought I’d throw that in there is anyone is a masochist. So. The VINCI DB is the easiest way when you’re on VINCI. Because it does a lot of things for you. If you’re working in a project it examines the context of the RStudio project. VINCI has the databases match the name of the project, so VINCI is already working in a project context. So if you’re just saying, I need to connect to the database that matches this project. It examines the RStudio project context, and then attempts to find the matching database for that project. It will look through the RB01, RB02, and RB03, find the appropriate database, construct the appropriate connection, and establish and odbc connection. It sets that as an active database, so you don’t have to continually re-specify that for functions inside of the VINCI package. But you can also just say get the active database connection and it will return that back so that you’re not reestablishing the connection every time. Just a helpful thing to say I know that I’m working in a project, I need to connect to the database that corresponds to this project, just connect it to it. And it does that automatically.

Okay. Now once you have a database connection, you then have to create tables, your table definitions or the table objects that point to the tables in the SQL server. So you use the tbl, table, or tibble, is actually the way it’s pronounced is T-I-B-B-L-E, tibble. To define a remote table, and you say, use the in schema to say, it’s in this specific schema. So here’s some examples. I’m taking VINCI DB assigning that to the DB object and then I’m saying, my crosswalk object points to my cohort crosswalk table, which is in DB. The S patient is in the source schema, and this is the way it comes through in most projects. As far as ever project that I’ve seen, this is how it comes across, is you have a source dot S patient underscore S patient which would be the S patient dot S patient table from CDW filtered down to your specific project’s cohort. Okay to be able to get to another database inside of that database, that’s on the same server, but not in the database that you’re connected to, you would next in schema. So you’re saying, this is dim ICD9. So it’s the ICD9 descriptive version, which is in the dim schema, which is in the CDWWork database on RB01 or RB03 is usually what I’m connected to.

So this is how you would want to connect to data. Once you are, you do all your operations, you filter it, you subset it, you do some summarizations on it, you’re going to want to get that data to go into R to be able to work with it. Maybe you’re going to fit models or something. You want to collect that data. So let me answer some questions. When you are speaking of database, are you speaking of true databases or just a dataset? I am talking about a true SQL server database. These are the actual SQL servers that are behind VINCI. So this is the mirrored version of the CDW data. And when you create a new project and go through the data, through the DART you get a project created for you that has access to data for your project. And it has views. And if you want to actually do analysis on that data you need to get that into R. This is the easiest way to get that into R. I apologize if I did not make that clear enough. Once you, so you get it back into R you’re going to use collect. So that’s collecting the data from the SQL server, from the database server into R. If you want to do the computations but store it as a temporary, or even as a permanent table, you use compute. It’s temporary by default. If you want it to make a permanent table, you just say temporary equals false. But you’re going to name that table and say, that’s a select into table all of your computations. If you need to pull something that’s not a table, you’re going to use pull. So pull it will perform a computation and bring back the resulting non-table result. Whether that’s a vector of values or it’s a, like if you’re going to put do like a count star, so count the number of rows you could say, count and then pull the count back.

Okay. So caveat emptor, let the buyer beware. Dbplyr is fantastic in what it does. What it does not do is it does not perform any optimization on the SQL that it produces. This means that it can create some very long, very complicated queries. Technically they are correct. But complicated operations can easily overwhelm the built-in SQL server optimizer. And we’ve encountered this a couple of times where what should be a simple query runs for a long time, bogs down the server and makes it unfunctional for multiple people and the query has to be killed by the database admins. There are a couple ways that you can make your life easier and VINCI admins lives easier. That is by using a judicious use of compute. If you’re going to create a table and you’re going to reuse that in multiple places like you’re going to compute a result and then compare against it multiple times, you’re going to want to compute that into a temporary table and then reuse that temporary table. You’re going to want to reuse that temporary table so that you’re kind of, if you know that you can optimize it, it makes it all that much easier for SQL’s server to run the query. And you can rethink how you’re going to do the query. Order of operations in SQL can matter a lot. So if you’re filtering everything down and then doing the computations it’s a lot better than doing all the computations and then filtering down. Because you don’t need to compute things for data that you’re not going to end up needing. And last strategy, cache tables and datasets between sessions. It’s good practice to create an analysis dataset and work only off of that analysis dataset. And don’t tweak it. Do all of your tweaking, then do an analysis. That’s just good research practice.

It doesn’t always happen like that in practice. But it’s good. But the other advantage of caching your results, or saving your intermediate results is if you get interrupted, I know most people don’t ever get interrupted when they’re doing research. But I do get interrupted occasionally, and when I do, I get disconnected from VINCI and it may be a day or two or a week before I get back to it. So if my tables are saved, and I can come back to it, that’s good practice. And it just makes it a lot easier. Collect, compute, and pull are all functions of dplyr, and dbplyr. They are not part of VINCI DB. I will specify specifically what is in the VINCI package. If it’s not in the VINCI package it’s part of the Tidyverse.

Speaking of that, this one is a function that is in VINCI. It’s the need table function. Which is helpful and very, very useful. What it does is, it checks if—you give it what you’re looking for. You’re looking for a table like S patient underscore S patient. It checks, do you have S patient in your workspace? If it does have workspace, if it does have that object in your workspace, great. As long as it’s the right thing, as long as it’s a tbl pointing to the table that you want. If it’s not a tbl, say it’s just an integer, it will say, that’s not what I want, it will give you an error to tell you that it’s not an appropriate type. If it doesn’t find it in your active workspace, it will check your active database to see is there a table already existing that matches this. So if you’re saying, I need S patient, S patient, it will say, okay, there’s an S patient in source, this is what I’m going to give back to you. And it will tell you what it’s giving back. Otherwise, if you provide it code to generate that table it will run the code, save the results, in a table, and give you the pointer to that table. So this is a way to recreate the table if you need to recreate it. Or once you create the table you save it and say okay, I need all my code to regenerate this if I have to, but if I’ve already gone through this step, just give me the pointer back to it. Now there are some worries. If you change data that that table depends on, so say you, I’ve got a project right now that I’m working on. I created a bunch of tables and then we had to expand our cohort by four years. And so I have to go through and regenerate all of those table, but I have the code, I just have to tweak it slightly for the additional years. Part of what is wonderful when you’re programming in R and designing things with reproducibility in mind.

So here’s an example. I will run need table Dim ICD9 and it will produce something like, okay I’m using the table source dot Dim ICD9 found in the database. That’s not an actual database, so don’t go looking for it. But it’s our imaginary database for today. The second time I evaluated it, so I highlighted all of my code and said, run everything again. The second time I run it, it will say Dim ICD9 is already a Microsoft SQL server table, and those are all the classes. So it’s a tbl Microsoft SQL server, tbl DBI, tbl SQL, tbl lazy, tbl object. So those are all the classes that it has. And that is what it’s supposed to be.

Okay, now there was a question about whether this is applicable to working just with databases or working with datasets themselves. So sometimes projects just bring in data, they have to bring in data from the outside. If you want to work with just data, there is a need data analog to the need table function. It just stores things in files in a data directory by default. So here I would say, I need the data, ICD labels. And this would do is say, okay if I don’t have the ICD labels in my workspace, do I have a saved copy of it? If I don’t have a saved copy of it, go to the code. Then I’ll take, for the code I’ll take the Dim ICD9 table which I’m assuming already exists and join it to the ICD descriptive version. Select my ICD code in my description, just take the unique rows so within get the code in the description. And then I’m going to pull that down so it’s local dataset and not a remote tbl, and it becomes a local dataset. It will save it in the object and save it in a file, so I have a copy of it, so I don’t have to rerun that. It’s a toy example, but it is something that I actually do.

Autosave. Autosave is a simple utility that just saves things. And this should be a VINCI autosave. It just saves an object. You give it a file and say, save this. And it says, okay. I’m going to take that, save it in file, with that same name, just put the dot R data on it and save it in the data directories. And you can specify the data directory. They’re explicitly when you’re using autosave or the data directory option by set assign options. Or if there is a data subdirectory from the current working directory, it assumes that that’s your data directory and then it’s okay to cache things there. Or you can set a data directory environment variable if you’re running things in batch mode, that can be handy as well.

Okay. So here’s a few helpful extras in the VINCI package that I have used. That when we get it installed for everyone that I hope people find helpful. N row SQL just gives the number of rows in a SQL table. It’s basically just a count the number of rows, and then pull the result. Get data directory lists the data directories that you have specified. And the first is where new data gets saved to. List data lists all of the files that are in your data directory, or directories. Is key is a cool one. It will test if a set of variables on a table form a key. Which means that it is a unique identifier. It will uniquely identify the rows. So if there’s a primary key, a lot of tables I have found don’t explicitly say what their primary or if they have secondary unique keys on them. So is key is a way to check in the data if I am correctly identifying what uniquely identifies those rows. Or if I’ve got duplicates. In database is a shortcut for replacing the nested as schema calls. Or in schema, that should be in schema, not as schema. SQL case is an improvement over the case win that comes with dplyr. But it works with Microsoft SQL server with a default value. Find tables with is cool because it searches for the database tables that have the specified columns. So if I wanted to say, find me tables with STA3N and PatientSID it will give me all of the tables, which would actually be a ton of tables, that have both STA3N and PatientSID.

Okay, there’s a bunch of schema specializations that are not built into dplyr that have been added with VINCI. Info schema is just a way to create information schema tbl views. And db has schema, add schema and remove schema all self-evident of what they do. Get the default schema, when you connect with database there is a default schema used if no schema is specified. And that usually is the username or it’s the DFLT schema. And then db get database. Get the name of the database that you are connected to. So if you go through the RStudio connections and create a connection to that server. Like if you created a connection directly to RB03 or RB01 or RB02 and you said, db get database. And you hadn’t specified the database, it will connect to the master database. Which I’m not even sure has anything in it. But it certainly doesn’t have your project data. So no point in doing that. Db get user gets the logged in user which should be you if you have any doubts. It’s kind of redundant.

But there are some improvements that since odbc uses very generic odbc syntax, there are couple of slight improvements or changes that need to be made for SQL server. And I’ve implemented those, and I think some of these have already been fixed in the odbc package. But before it gets published for everyone it will come out, but those are the has table, save query, and the create table, those are all the behind the scene things that no one will really see unless they come across it.

Okay, so I do have, we’re at 12:45. I do have a bonus about how to kill a frozen R session. Because this actually comes to the helpdesk quite a bit. Of help, I can’t fix, RStudio is not responding. My R session is stuck. There are actually ways that you can do this yourself without having to ask for a Windows admin to kill the process for you. Before we go that route, we can ask, we can answer some of the questions. And this is a good time to get any of the questions that you would like answered. Put it into chat. So I’ll go through a couple questions that I haven’t been able to answer yet. And then I’ll go over the bonus of killing a frozen R session. So let’s go down if I’ve missed any. Okay, do RStudio projects autosave as you are working? Yes, they do. In the files, so the files are saved. One problem is that you’re, so your files do autosave usually. Save often, save frequently. All of those usual caveats apply because I have found where RStudio has lost a significant amount of work. Or it goes back to a, it didn’t autosave recently. So yes, it will autosave the files that you are working on. It will not necessarily save data objects and things like that. It especially will not maintain database connections across sessions. So if you close out a session and bring it back up you have to reestablish that database connection. You’re not able to keep the database connection. Now a funny tangent on that is, if you are using something like Box, or Dropbox for syncing files on your personal machine or your work machine. Like I have university machine that I have that we use Box on and oftentimes it will be syncing files in the background, and then it will try to autosave and it will say, it can’t save this because it’s, the file is locked for syncing. Just a tangent, so it does try to autosave. So what Coursera, VINCI produced, or other online training courses would you recommend for someone trying to develop skills necessary to work efficient with VINCI data in R? Assume a bit of experience with working with R on a local machine. What would I recommend for someone trying to build skills necessary to work effectively with VINCI data in R? I am on the helpdesk; you’re welcome to ask questions to the helpdesk and get answers. What Coursera? I’m not familiar with Coursera’s offerings. I do know that DataCamp has some projects. I do know that there are lots of resources out there. What you’re going to want, I would say anyone to start with R for Data Science. That’s where I will point anyone. If you haven’t read R for Data Science, read R for Data Science. It is a free book. You can buy a print version of it from O’Reilly press, but there is a free version online, and that is updated between printings. And if you look at the Bookdown dot org there are several other books specifically related to R that are fantastic reading as well. Okay. Did Dr. Redd say that we can bring in new legitimate R packages ourselves now? I believe in the past we couldn’t. Yes, I did say that. That is what we have been saying from the helpdesk. If that’s not correct, I’m in big trouble. Because I’ve been telling people to do it. So I hope yeah, so I actually, that was on an email about that today just saying that there are no problems installing specific R packages. You just have to install it in a project specific directory. There are occasionally issues if compiled code. If it includes C++ or other types of compiled code, or Java. Just avoid any packages with Java. That’s just a pain. Can I currently load VINCI db? I do not have the VINCI package put in anywhere to get it out yet. We are working with VINCI IT to get that installed globally for all R users. And it would just be a package that you would load up like any other. That hasn’t come through yet. This is just kind of a, the VINCI stuff is preview. But a lot of the principles of just okay save your table as compute, save the table that you can all do without VINCI. Are there any equivalent methods for accessing large SAS et cetera tables? That is a fantastic question. Let me take a tangent for that. Because I love that question so much.

That I realize I should have talked about that. To get to, I’m going to come to another package as where I actually used several SAS files. The package that you’re looking for is called haven, H-A-V-E-N. It is part of the Tidyverse. So it’s supported by RStudio. So there is actual commercial money in it. Here’s some data that I am working with for the Veteran Choice Act. So it’s a project examining impacts of the Choice Act for Veterans. So we have State data that has been uploaded and put into a VINCI project. And I say haven, read SAS. And then I point it to the file, the SAS 7 B dot file, and it reads it in. When it reads those in, it will also read in labels and things like that. There are a couple quirks with SAS. Specifically, SAS assumes an empty string is a missing string. They’re the same thing in SAS. So I do several fixes to make sure that the data is the way that it should be. So I say, if it’s a, so this is a mutate if, which is part of dplyr and saying change the variable if it meets these criteria. So if it is a character variable, change all of the, so it’s missing if it is this value. So if it’s an empty string, change it to a missing value which are distinct things in R. A missing string is different than an empty string. And I’m saying, if it should be a flag where it’s an indicator variable of zero or one, change that to a logical true false. If it’s integer-ish, which means it’s a double, but it’s storing integers, so usually counts, change everything that should be integer-ish, to being an actual integer. And then I say, here mutate at the specific homeless indicator that that should be an integer because that was missed for some reason. And then I can say, okay, these variables specifically I want to be integers, or I’m changing variables with regular expressions.

But the specific thing, haven is for managing foreign data. There is also a foreign package. Haven is better supported. You can write SAS. I have come across a couple of problems with writing SAS files. Reading them I’ve never encountered a problem. But I have had a problem writing them. Okay, it seems that exploring databases and writing queries to assemble an analysis ready sat file are best done with SQL manager. Besides reproducibility, what is the advantage of using dbplyr? That is a good question. There are advantages to using dbplyr because it’s a more natural interface. If you are comfortable with SQL management studio, I think is what it’s called, that’s fine. That’s great. I’m not going to tell anybody what they need to use or not. I’m saying that this is a tool that if you don’t know SQL could be very helpful to be able to pull data directly even though you don’t know SQL but you know the operations that you want to do. And Then there’s a natural synergy with what you’re doing on local data is the same with what you’re doing on SQL data. And there is actually, and you can actually push data from R into SQL server. That’s a copy to, copy underscore to, and then you say what you’re copying. Where you’re copying it to and what you’re copying and the name of the object that you’re going to do. Is there a version control available for working in R projects? There is actually a version control available in RStudio because it has built-in GIT tools. So when you create a new project. Let’s go back.

So if I create a new project, I don’t ever want to save my environment, and I say I want to create a new directory, new project, oh. You know what, I may be incorrect on that. I apologize. I was under the impression that we had GIT tools available to us. And it does not appear that those are enabled. I will have to get back to you on that, that is a very good question. And I think that we should have GIT available.

I will have to take that and talk to VINCI management about getting GIT installed on there. So thank you, I hope I’m saying it, Andrea Kovall [phonetic 0:58:27], thank you. Can we library VINCI right now in R within VINCI or will it be available? You can’t do it right now. You can, you will, hopefully soon. And I don't know what soon is. VA timelines I’ve never been good at estimating. So I may have better luck trying to put the code on there. Thank you for the link for R for Data Science. Is it ever necessary to actually pass SQL query strings to the VINCI library functions you touched upon? There is a way to do that. I didn’t want to get into that. You can actually either do full SQL queries, and that is just a db execute a query. What dbplyr does in the background, is it takes all of the operations definition and translates those into SQL statements. And there is a way that you can translate snippets of SQL code. So you can actually, when you are creating a new variable and you know the exact SQL code you can wrap it in a SQL, SQL function. And say this is SQL, don’t try and interpret it, just pass this in as raw SQL and it can do that.

Moderator: Dr. Redd, we can stay a few minutes late if you want to answer all of these questions. I think five, ten minutes is fine.

Dr. Andrew Redd: Okay, that would be good. Can VINCI R experts like you be paid as consultants on a grant in order to get your help with instruction and manipulation? I don't know how that would work. That question right now is above my pay grade. When will the latest Tidyverse dplyr be available? I’m unsure of when the update schedule is. R 4.0 is causing havoc in the packages systems. And so I’m not sure when that is going to be updated. But when the new, when R is next updated, the next version of dplyr will be installed. And I believe that will be installed for the newest R 4.0 release. Thank you. In RStudio there’s a tool version control dropdown. I think that’s if you have version control enabled. And I’m not sure exactly what you have to do for that. So, oh, is there a single RMD file which illustrates minimal completion roundtrip CDW SQL example? I think I may have something like that. It’s not simple. Oh, no, I don’t want to do that, I want to switch my project. Oh. I didn’t mean to do that. That’s going to take a second. So while we’re talking about aborted sessions. Let me go this route and say, so killing a frozen R session. The steps are pretty simple. Basically what you’re going to do is you’re going to use the wingui package, which is one that I wrote. So I am tooting my own horn. Again. You’re going to get what the processes on the machine are. Find the process, the problem that has the process. So it’s stuck. So once you get the wingui win processes it takes actually a few minutes to run. And I’m not sure why. But it does pull out the process information. And then you can, it returns it as a data frame and then you can filter that data frame. So what you’re looking for is the rsession dot exe. It will show you all processes, but it won’t tell you who they belong to. It will only tell you, your processes and the system processes. It will tell you everybody’s process, but it will only have the username populated for you and the system. So you should see a few processes.

Then go through once you’ve figured out which one it is that you want to kill, you can use the wingui win kill. And yes those are three colons, not two colons. Because we’re internal and I haven’t released the version that has them as exposed. But basically you say, okay I’ve to the process ID, the PID of the process that I want to kill. Then I go in and get the, pass it to wingui and say PID kill this PID. And if successful the frozen RStudio session should show a message saying, R session aborted. R encountered a fatal error. The session was terminated. And start a new session. One caveat with this is you will, when you start the new session, it will start RStudio without a project open, so you have to reopen that project.

Just a couple notes. If you want, you may actually need to kill RStudio itself. When you do that you replace the RStudio dot exe or the rsession dot exe. And you can go through and just kill all of your RStudio sessions. Or all of your session dot exe’s. This is helpful when you’re running, if you open up the base R GUI, not the RStudio interface. If you open up the base R GUI and say wingui win kill and then just kill everything that’s in RStudio dot exe. It will give you warnings that you cannot kill processes from other users, which is good. You should not be able to kill other users’ processes. But you will, it will kill off all of your RStudio sessions. But doing that will kill everything, not just the one that is a problem. Okay. Let me, so getting an RMD file. A code that you shared, example haven, included wonderful tips and tricks for working with VA data, could such code be shared? That wasn’t exactly code. But I could certainly write something up. I’m not sure exactly where I would put that. Is there pseudo code for pulling in a SQL server query into R? Marva [phonetic 1:05:47] I’m not sure exactly what you’re asking. However, I’m not aware of anything that really translates SQL code into R code. It works great the other way. I’m not sure exactly what the code would be to the other way around. Because R and SQL are very different languages. R is a procedural language, whereas SQL is a declarative language or an imperative language where you tell it what you want, and you let it figure out what it brings in. Whereas R is very much more step-wise oriented of saying okay, take this, drop these variables off, transform this variable from here to here. And in that way, it translates to SQL, but it translates it procedurally. And it does handle things like window functions and group by and all that very naturally. But I’m not sure of any way that, or anyone that has worked on pulling SQL server query into R. You can just take the SQL query and say, run this query, and I believe there is actually a way to run SQL code on data that is not on a database server, but is a database. And there’s like a mini parser or something that will do that. Where you’re running SQL code on a local dataset. I’ve never played with that, so I can’t talk about it in detail. Now coming back to the RMD sharing. I think this will be the last thing that I share. And I think that we’ll quit at this because we’re already over. And one thing that I failed to mention is that if you need, as the viewer you can zoom into this. This is a dataset for doing an amyloidosis analysis. And I have a setup section here and say, I’m using all of these packages and you guys don’t have access to this using function. It’s something I wrote for myself, but I’m just saying, this just replaces a bunch of library statements. So I define my database, I say I need my info tables, which my information schema tables. The views columns, and view column usage. Then I go info columns for CDWWork to be able to find, this is all stuff to find the tables that I need. And so I say, okay, tbl in schema source cohort patient SID. I could actually do this as need table cohort patient SID and that could replace all of these. I did not do that in this, this is kind of an older statement. But let me, come in here and so here is an example of, I take my S patient, oh here’s a good need table. So DDMS, so DD is the data definition. I want the marital status table, and so this is just something that I’m kind of pushing into. So DDMS is I’m taking this vector of unknown, married, never married, separated, divorced, or widowed, pushing that into a table and then I’m copying that to my database with the name DD marital status. I’m committing it, and then I’m returning that so that I get it back as the pointer. I think that that’s actually unnecessary.

Moderator: Dr. Redd, we really do need to start wrapping up now.

Dr. Andrew Redd: Okay. I think that this is about it, though. So patient selections, join it to something else, selections. Change some variables. Then I compute to save it as a temporary table that I’m going to end up using later. That’s all there is to it. It’s a lot of small things that build up to a big thing in R. And that’s the what that R designed to work. So with that I think I will wrap up. Thank you everyone for your attention and your wonderful questions.

[ END OF AUDIO ]