DDAHS 2026 summer school Workshop 6 CAMELS-GB v2 using an open dataset and machine learning to explore British hydrological diversity-20260716_130420-Meeting Recording

16 July 2026, 12:04pm

2h 0m 48s

Gianni Vesuviano started transcription

Gianni Vesuviano 0:19

All right, I've started recording. Welcome to Workshop 6, CAMELS-GBV2, using an open data set and machine learning to explore British hydrological diversity. And with us we have Molly Asher and we have Matt Fry and Tom Keel.

And Amulya.

So, whenever you're ready to start, please do.

Molly Asher 0:46

Can you hear me?

Yeah, okay, cool. Well, yeah, thanks everyone for coming back this afternoon. I realise it's the last workshop of the week, so hopefully people have still got a bit of energy left. My name is Molly, I'll be leading the session and I work

as a research associate at the University of Bristol, doing research related to future flood risk. And I also work as part of the FDRI project for UKCEH. I think someone just posted in the chat that maybe, Matt, you could also introduce yourself.

So if you want to do that now.

Matthew Fry 1:32

Yeah, afternoon everybody. Yeah, it's great to be here. Thanks everyone for coming. My name is Matt Fry. I'm also at UK Centre for Ecology & Hydrology and I work on kind of hydrological data and data systems. So in particular, trying to collate data sets to support the research community and others.

on doing hydrology. So the kind of underpinning data to a lot of the stuff that Molly is talking about today, things like CAMELS, river flow, rainfall data sets. And at the moment, that's we're doing that through this Floods and Droughts Research

Infrastructure project, which where you've heard of had an intro to, and I'm leading the sort of data and digital
part of that work leading up to delivery of a kind of set of data sets and the digital platform will be launching in September. So please keep your eyes out for that. Thanks.

Molly Asher 2:22
Okay, cool. So yeah, as with previous workshops this week, we've got lots of people on the call who are able to help you in the breakout rooms or in the chat or whatever. So if you've got issues with anything that we're doing, then please just let us know and we'll try and get them sorted as soon as possible. So just before we go any further,
Amulya is going to post a link in the chat, which if you could all just click on that now, then that will open up this kind of material that we're going to be working through. I'll just say some more stuff, but it can take a bit of time to load up. So if you can just kind of click on that link as soon as possible,
But hopefully, by the time we get to needing it, it will have loaded. Okay, so the workshop today is going to be based around a data set called CAMELS-GB, which Matt just mentioned. And I think it might have been mentioned a bit in this morning's workshop maybe, or one of the previous workshops.
Um, and, ohh, God.
Sorry. And yeah, I know that there was also the workshop yesterday with Tom on hydrological research and kind of thinking about what is research. So today's workshop is going to be a bit of like another example of the kind of hydrological research that you can do with data sets like CAMELS, which are openly available. So it's like a nice example of the kinds of things you can do with
with data that you would just be able to access yourself. So the way the workshop will work is that I'm going to do like a fair bit of talking through and explaining code that's already written. And you're going to be kind of following along with that. And then there's going to be several different exercises where you get a chance to go and spend a little bit.
bit of time working on some code yourself. And hopefully, like, it would be nice if that was a bit interactive. And if you're happy to share some of the outputs of what you've done in the chat, then that would be really nice to see. As well on that point, if anyone is like happy to switch on their camera,

then that's like a really nice way to just make the workshop feel a bit more communal, to give us as like the people running it, a sense of kind of who's present. It's also nice for you guys to feel like you're in like a collective of people. So if anyone's happy to do that, then that would be nice. Okay.
So I am going to share my screen at this point.
And.
Okay, so can everyone see my screen? You should be able to see like a notebook.
Um...
I'm further on than I want it to be, I just realized. So you should have seen like a kind of launching Binder circle thing going round in circles when you clicked on the link. And then it will take you to a page that looks a little bit different to this because I already clicked on the link. But you should be able to see a file explorer
at the site where there's a folder called Workshop 6. So if you navigate into the Workshop 6 folder and then if you click on the Workshop6.ipynb, then it should open up so that you see a screen that is the same as the one that I've currently got on my screen.

Tom Keel 5:53
Hi, Molly. Sorry, do you mind zooming in a little bit?

MA

Molly Asher 5:56
Yeah, sure. Sorry, I'm really bad for having my screen zoomed out.

Tom Keel 6:02
Maybe one or two more.

MA

Molly Asher 6:02
Is that is that?
One or two more.

Tom Keel 6:06
Yeah, I think that's perfect, yeah.

MA

Molly Asher 6:07
Is that good?

Okay, great. Thanks, Tom. Okay, so just to check that we're all kind of at the same stage, if you've got this loaded up so that you can see the same thing as I'm showing on my screen, could you do like a thumbs up reaction using the kind of Teams React thing?
And if you've not got that, if you can kind of post the link, post a comment in the chat or something to let us know that you're having problems with that.
So.
It looks like most people that's working for. Again, if it's not working, let us know. Just to say that this is a slightly different setup to what you've been using before. So this is using something called Binder. It's very similar to Google Colab. It's just slightly different in the way it presents the notebook.
You don't need to make a copy of it in the way that you're doing with Google Colab. You can just work directly into this. But just to say like anything that you do type into here will not be saved. So it will just be wiped at the end of the session. So if there's anything that like you had written yourself in terms of the code,
then maybe you could save it somewhere else.
Okay, so I'm going to presume that everyone is with me on this and I'm just going to start with like a bit of an introduction about what the workshop is going to be about. So we're going to be looking at how we can use data to better understand, explain and predict how catchments are likely to respond to rainfall events.
So this is really important because there's lots of diversity in Great Britain. Same thing applies to other countries in terms of the kinds of rainfall events which happen. So some catchments might get more kind of very heavy convective rainfall events. Other catchments might get more
kind of persistent frontal rainfall events. So different catchments get different kinds of rainfall events. And there's also lots of differences in the way the catchments kind of store and release rainfall once it's happened. So you might get a catchment with really steep sides or impermeable soils, which means that the water kind of moves through those catchments really quickly.
And in comparison, if you've got a catchment which is much more shallow, it's got more permeable soils, then that means that rainfall takes a lot longer to kind of move through the catchment. So these are much slower to respond to rainfall events. And kind of understanding these differences between catchments is really important if you want to be able to kind of understand and predict
the kinds of risks you're going to get in a catchment in terms of things like floods

and droughts.
So the way the workshop is structured is going to be in terms of answering three different research questions. So these questions are written here. So the first one is, how does the hydrological behaviour of catchments vary spatially across Great Britain? The second one is then looking at what the controls on that spatial variability and hydrological behaviour might be. And then the third one is looking at whether we can predict the hydrological behaviour of catchments from the landscape and climate attributes of the catchment alone. So that's going to be the part where we're looking at building some models. So in terms of time scales,
for the workshop. We will be spending 10 minutes on the workshop introduction and setup. That's now, that's finished. And then do a short introduction to the data set. We're then going to spend about 15 minutes on the first two research questions, including a bit of time of you working by yourself.
And then there's like a 10-minute break where you can leave your computer. And then we've got an hour to spend on the final section, which is that third research question where we're looking at making some predictions. And a good chunk of that time will also be for you to be writing your own code and trying some stuff out yourself.
Okay, so what is CAMELS is the first thing to talk about. It's got nothing to do with camels, unfortunately. It is just another acronym, which stands for Catchment Attributes and Meteorology for Large Sample Studies.
The GB is telling you that it's for Great Britain. I think there's some other versions of a similar data set for other places. And the V2 is just saying that this is like the second version of this data set that's been produced. So CAMELS is a large sample hydrological data set, which means that it covers
lots and lots of catchments, so just less than 700 in this case. And this is really useful because it lets us kind of study the behaviour of catchments across lots of different places simultaneously, which is really helpful because by looking at all those kind of catchments at the same time, we can start to understand
whether there's patterns in the way that catchments behave, to try and understand the reasons for those patterns existing, like things to do with the landscape or the climate, can test whether general relationships hold across all those places, and we can then start to kind of build predictive models, as I've mentioned.
So CAMELS is like a really, really big data set. It's got lots of stuff within it. So you can use it for lots of different things. Summarised the data in this table. So you've got

catchment boundaries. So these are like shape files showing you where the catchments are.
We've got lots of catchment attributes. So these are like static summaries of things which describe the catchment, so stuff to do with the topography, soils, land cover. There's also like mean descriptors of climate and hydrology. So for instance, like the mean annual rainfall in a catchment.
We've also got time series data for both the meteorological variables and river flow. So for like the last 50 years, roughly, we've got daily and hourly time series of different meteorological and hydrological variables. The data from CAMELS is all available on the CEH data catalogue, which I've given a link to there.
You don't need to use that link today because everything that you need data-wise is already saved within this project in this data folder, which you can see in the top left. But just for your own reference, if you ever want to use this data set in the future, you can access it at that link. The second link is to a paper.
which is about the CAMELS data set. If you're interested, this might be a nice thing to have like a proper read of later or whenever. We're also going to be using this paper throughout the workshop because it contains some descriptions of some of the variables that we're going to be looking at. So we're going to be referring back to this paper. So it might be useful to click on that and have it open. But I'll let you know when we need to refer to it and kind of where in the paper you need to look at.
Okay, so I'm just going to start walking through some code now. As I said before, This, the way it will work is that I'm going to keep my screen shared and I'm going to be running code.
I would like everyone to have their own version of it open and to also be running the code. And if at any point it's not running for you to let us know. When we get to the exercises, which are at the end of each section, they will assume that you've executed all of the cells up until that point.
So that's why it's important that you're also running the code as well as just as well as just watching along.
And.
I think you've been working on Jupyter Notebooks all this week, but just to double check that everyone knows what they're doing in terms of running the code. And I think it is slightly different in Binder compared to Google Colab. If you want to run a cell, you need to put your cursor within it. There's no like run button, which I think Google Colab has at the side. There is a run button at the top which you can press or

you can press control and enter. So for instance if I type in my name here.
and press control and enter. Then it prints out and tells you my name is Molly. So just to check, you can all run the code if you want to just have a go at typing in your name in that gap and pressing control and enter to run it and just check there's not like something wrong with your notebook or something.
And...
You've got a hand up. Is that an issue with the code running?

Jones, Robyn (MECP) 15:38
I just want to make sure when we did the Google collab, we had to make a copy. We can just change right in this version or do we have to copy?

Molly Asher 15:47
Yeah, so for this, you don't need to make a copy. I don't know if you maybe missed when I said this at the start, but also just to know, it won't save your changes.

Jones, Robyn (MECP) 15:57
I did hear that part, yeah.

Molly Asher 15:59
Yeah.
Yeah, Amulya.

Amulya Chevuturi 16:03
Just in case you want to make any changes and then you want to save it, once the workshop is done, if you could just hit download, go to file and download the notebook. So you will have a notebook which just has your edits in it.

Molly Asher 16:18
Yeah, that's a good idea.
And.
Okay, so yeah, the other thing I just wanted to check, people knew about was if you wanted to comment anything out, so...
make it so it doesn't run. So if I run this cell that doesn't appear anymore, then you can use control and slash on the keyboard to do that.

Okay, if anyone has had an issue with that running, then just let us know, of course, but I will assume that it's all good and move on.
So yeah, the first thing that I'm doing in this script is just setting it up with the packages that I need. So the majority of the packages that I'm using here are like pretty standard Python packages, which I think you have already used so far this week.
There's a few in there which are more like specific geospatial analysis packages. I think you've also used some of these as well to be fair. So like GeoPandas, Shapely, Contextly are some really useful ones for working with geospatial data. Put some links here to more information about those packages, which again, like if you want to read more about in your own time, then that would be a nice thing to do. But for now, I'm just going to run that cell. We should import the packages. This second line is just telling the script where the data is stored. So as I mentioned before, that's in this folder in the top left.
And you could go in there and kind of click on some of these data sets and look at them as like CSV files. But this line is just telling the script that it's in, that's where the data is. So we need to run that as well, otherwise it won't know what's going on with that.
Okay, so going to move on. Oh, sorry, is that zoom level okay? I keep zooming out by accident. Okay, so yeah, we'll move on to the first kind of chunk of the workshop, which is looking at this first research question, which is how does the hydrological behaviour of catchments
vary spatially across Great Britain. So we're basically going to just spend a bit of time exploring the data that's available within CAMELS-GB and see what we can learn from it about catchment behaviour. So we're going to load up
some of the different catchment attributes. And then we're going to do some map making to visualise how the catchment attributes kind of change across the country. And then maybe think a bit about what are the reasons that could be driving the spatial patterns that we see in the catchment attributes.
So the first thing is to load in the catchment boundaries file. So as I mentioned before, these are shapefiles which just show like the boundaries of the different catchments. So I'm going to run this first cell and read those in.
And then underneath, I'm using this stop plot command, which again, I think you were using in Matt's workshop, which because this is a spatial data file, it's going to plot spatially. So

you can see that this plots out something which is fairly similar to the outline of Great Britain. If you zoom in a bit more, there should be kind of boundaries. Actually, no, there shouldn't because I didn't specify that. Anyway, this is just showing all of the catchments which are included within CAMELS.
If you're familiar with the UK, then you might notice that there's some areas that are missing from this. So around this area, there should be other catchments, but that's just telling us that CAMELS is not like a complete data set. There's some of the catchments which it doesn't have data from. So that's something to bear in mind.
So the next thing I'm going to do is to use Pandas to load in the catchment attribute data. So within the data folder, there is 6 or so different sets of catchment attribute data kind of organised by different themes.
So we've got climate data, topography, hydrogeology, land cover, soil, and hydrological signatures. And we're going to load these in using the read_csv command in Pandas. And then for this hydrologic attributes, I'm then going to plot the first
5 rows of the data set using this .head command. So if I run that, I'm just going to print it out again. And at this point, it's useful to just have a little bit of a look at the data set, try and understand what is being shown here. So
The way it's structured is that it's got a gauge_id column, which is like an identifier for the gauge in each catchment. And these are just strings of numbers. And then each of the other columns contains a different hydrological variable.
and its value for each catchment. So this is the point at which I was mentioning earlier, we're going to have a look at this paper, which is giving extra information about CAMELS, because some of these variable names, you won't necessarily know what they mean. So for instance,
When I'm looking at this, maybe I don't know what q_mean means. So I can go to this paper and you can either go to table 5. So table 5 is where it says summary table of attributes available in CAMELS-GB. And then you can kind of just manually look for the variable that you're interested in. So I'm looking at the hydrological signatures, so I can see that there's q_mean, and that tells me that that's referring to the mean daily discharge in that catchment. Similarly, you've got an explanation for the runoff_ratio, the stream_elas.
And etcetera, etcetera, so.
If I went back and forth between those two things, then I would get a proper understanding of what all these different variables mean.

Yeah, you can also, if you just want to quickly find something, do like a ctrl F and type it in, q_mean, and it will take you straight to the kind of variable definition.
Okay, so I've read in these hydrological attributes, and then I'm going to read in the rest of these attribute data sets. I'm not going to look at these ones. I'm just going to read them in for now using the kind of the read_csv command from Pandas.
And then the next thing that I'm going to do is to get all of these catchment attributes together into one kind of master data set. And I'm going to do that by using this merge command. This is possible for me to do this because all of these data sets are the same length, so they
all have the same structure where they've got.
673 or whatever it was, number of catchments, rows, and they've all got this gauge_id column, which is the same in each of the data sets. So because of that, I'm able to just join them all together. So if I run that cell and then again use this head command to just look at the top five rows, I can see that now
Whereas, before I had...
a handful of columns. I've now got 162 columns. So that's containing the information from all of those different data sets. So now I've also got the name of the catchment, which is quite nice. And I've got location data as well.
Okay, so the location data is going to be important because we're going to be plotting the catchments spatially.
So now that we've got all this data together in one place, the next thing that we're going to do as just kind of like a data exploration technique is to make some maps. So this is like a really quick and easy and useful thing to do if you just want to get a bit of an idea of what kind of structures exist in your data set.
Because if you just get a table like this and look at it, it's pretty meaningless to me anyway, as a human being. Just lots of numbers. You've got really no idea what to make of any of that.
Whereas as soon as you put it on a map, it's usually immediately obvious if there's some kind of spatial patterns or something interesting going on. So that's what we're going to do here. So the first thing that we need to do is to use GeoPandas, which converts our data frame. So this data frame here into something called
a GeoDataFrame, which is basically the same thing, but it's just got shapes attached to it. So to do this, we use that geometry column, which was originally part of the catchment boundaries file.
So I'm just going to run this cell, which will do that conversion.

And then using this GeoDataFrame, we can then quite easily produce a map using the matplotlib plotting functionalities. So for this, I'm just going to start off by plotting the q_mean. It says daily precipitation above it, but that's wrong.
It's gonna be.
Mean daily discharge. Sorry, that's just a mistake I forgot to update. So there's a few different things you can specify in your map production. So the first thing is you can say which variable it is you want to plot. So I've specified here q_mean, but you could include
Various other variables, and...
The ax command is just saying that you want it to put it on this figure object. You can then specify the edge colour and the edge width. You can see if you want a legend. And you can specify the colour scale you want to use. You can also specify how to scale the colour bar. So this is useful if perhaps there's like an uneven distribution of catchments across the range of values that you're looking at. You can also set a title and put on a base map, which is really useful.
to kind of orientate yourself in space.
So, if I run this, and...
takes a second, I think, because of the base map line, which is like pulling information from like a map provider, which just makes it a little bit slower. So we can have a look at my map. I'm just going to zoom out a little bit for this. So this is a map of the mean daily discharge.
As I mentioned before, now we've got the base map on here, we can see even more clearly that there's some places where catchment information is missing. So around the east coast of England and some places in the northwest of Scotland, we've got some catchments missing.
You can also see that there are definitely some spatial patterns in this variable, the mean daily discharge. So we've got generally higher values around the west coast and going up to the north of Scotland and then lower values around the east of England. It's not like a super
prominent pattern, but there's definitely some kind of spatial variability and kind of consistent spatial variability going on there.
So just to demonstrate some of the things you could change, I could say I wanted to give these red edges and I want to make them really thick, although I think that will look horrible. And then you could say you want to have greens instead of blues as the colour scale. We could

ask to plot the p_mean instead, so that was the mean precipitation, and then I change this to mean.
daily precipitation and you can rerun it and it will update according to the things that I've changed.
And it will just take a second to load.
Okay, so as I said, that's a horrible map. But you get the idea that there's various things that you can kind of customise in the plotting of your map, which will make the data kind of easier or harder to understand. And obviously, you want it to be easier to understand, so you can kind of play around with various different parameters to make your map as kind of intuitive and easy to understand as possible.
So at this point, we've got our first kind of task for you to work on by yourself. So the task is basically to just take this code in the cell that I've just been talking about as a template and to go and make your own
map. So to make a map for a different hydrological variable and to kind of change all the parameters to kind of best suit the thing that you're plotting.
maybe have a go at looking at some of the different variables and try and find one which shows a pattern that feels like it's meaningful and maybe interpretable so that you could then maybe have a bit of a think about why it might be showing the spatial patterns that it is.
There's a few kind of hints in here as to how to go about doing this. And
You can do it in these cells below where I've just put these comments. So you can also add new cells using this plus button here if you want to add new cells, or you can kind of just delete these comments and or type code below them.
And...
Has anyone got any questions about what we've talked about in this section or any questions about how to go about doing this task?
And.
Can see someone's got their hand up. Do you want to shout or comment what your question is?

MW **Moa'az Wagdy** 31:41
Yeah, that was me. I just misclicked. All good on my side. Thanks, Molly.

MA **Molly Asher** 31:45

Okay, good. Anyone else got any questions?
Okay, if not, then I'm going to say that if we can spend 5 minutes on this, I will set this timer. And if you want to just give this a go, we're not going to go into breakout rooms or anything. So if you just have a go at doing this by yourself, if you've got any questions,
let us know. If you have completed the task and you've got a nice plot that you'd be happy to share. If you want to post that in the chat, that would be cool to see.
If you do this really quickly, then there's also a bonus challenge below, which you can just have a look at if you've finished the task and want to move on to something else.
Okay, I'm going to start the timer. So yeah, just let us know if you've got any problems you need help with.

Tom Keel 32:48
Molly, just to say, it seems like there's been two issues with name errors so far. I think it's because you can see all the outputs already, like it looks like the cells have already been run. So I think the issue that Tunbi has, Tunbi Kelly has there,
They just need to run all the cells before, basically. But there's a slight issue because all the, you can already see the outputs of the cells, so it's not clear which ones have been run, basically.

Amulya Chevuturi 33:18
Maybe you can share and show them how to remove like all outputs, like restart kernel, remove all outputs, and then they can start from the beginning.

Molly Asher 33:31
And.
I'm actually not sure how you would remove all the outputs. Does anyone else know how you do that?

Amulya Chevuturi 33:38
So yeah, so if you go back, go to the kernel, kernel on top, yeah, there it says restart kernel and clear out, clear outputs of all cells. So whoever might be having the issue, either please rerun the cells.

Tom Keel 33:39

Yeah, I don't know.

MA **Molly Asher** 33:50

Okay.

Yeah.

Amulya Chevuturi 33:58

Above you.

or do this and rerun the cells so that you will know which ones you have run and which ones you haven't run very clearly. And if you're still having the problem, then you get back to us.

MA **Molly Asher** 34:15

Yeah, that sounds good.

MF **Matthew Fry** 35:17

Yeah, just posted in the chat that.

Yeah, well, you can click on the cell and then click play, but it can be quite useful as a way to step through the notebook just to use the play button at the top, because each time you press it, it steps to the next cell. And you know that wherever you are, it's run all the cells in the notebook. So as long as you start on the first cell, click that play button each time. It will make sure it's all run. And also worth keeping an eye on the little numbers in the square brackets next to the cells, because you could see whether a cell, if you've got, you know, if you're confused about whether a cell's been run or not, you can sort of see if it's in the order of the cells above that you've run already.

MA **Molly Asher** 37:00

It looks like that error, name error name plot is not defined is also coming from the same problem that you must have not run the initial, like the very first cell in the notebook where you load in the packages.

Okay, so we're almost at the end of the 5 minutes. Has anyone like got close to making? Oh, God, sorry. Anyone got close to making a map or anything that they are

able to share? Or do people want to have a bit more bit more time working on this? You can see one map from Avintha that looks good.
In terms of your question, Robyn, how did you use the list thing that I've posted here? You can literally just copy and paste that into the cell. And the idea of that is to just give yourself a list of all the different variables that you might want to plot.

Molly Asher 38:42
So, you can see there's the q_mean that we plotted before, and that's the baseflow_index that Avintha has plotted, for instance. So, that was just, you don't really need to use it per se, it's just to get an idea of what like the menu of options is in order to plot from.

Jones, Robyn (MECP) 38:59
So if it didn't work, I probably need to clear the kernels.

Molly Asher 39:04
As in like when you tried to run this cell it just didn't work.

Jones, Robyn (MECP) 39:07
Yeah, it said list, hydrologic, attributes, columns, indentation error?

Molly Asher 39:14
So that's probably because you've got, I'd imagine it's because you've got a space at the side.
Oh, that's still working for me. That's weird. So if you try to just make sure you clear.

Jones, Robyn (MECP) 39:22
Okay.

Matthew Fry 39:23
Could be, yeah, could have accidentally, yeah, so.

Jones, Robyn (MECP) 39:26
Oh, that was it. That was it, yeah.

Molly Asher 39:28
OK.

Jones, Robyn (MECP) 39:30
Thank you.

Molly Asher 39:33
Okay, so just having a look at the things people are posted in the chat. There's some nice maps here. You can see the one that Adi has posted has got a bit of a different kind of colour scheme on it. So instead of kind of going through one colour from light to dark, their colour scheme goes from one colour
red through white to a completely different color, which is got pros and cons, I guess, but it's like quite nice and kind of really highlighting areas of difference.
Okay, looks like lots of people have managed to do this.
And...
Okay, I think in the interest of time, I'm just going to move on to the next section. If anyone...
has had like a problem with that and is interested in coming back to it later, then we can always pick it up later on. But yeah, for now, I am going to start in the second kind of section of the work. So this is looking at the research question of
what are the controls on the variability in the hydrological behaviour of the catchments? So this is basically saying that in that first part, we were exploring how catchment attributes vary across the country, building up a spatial pattern, building up a picture of the spatial patterns in hydrology. So on your maps, you're kind of showing that for certain hydrological variables, there's spatial patterns. They might be higher in the east, lower in the west, or different things like that. And that's useful. But as well as just kind of describing where catchments behave differently, we also really want to explain why that might be. So to try and to explain
to try to identify the catchment properties that might kind of explain these hydrological patterns. So that's what we're going to look at next. And for this, we're going to focus on a variable called the baseflow_index, which I think a couple of you plotted in your map.
So this is a variable that is provided within CAMELS, which kind of is quite important for describing catchment behaviour. So what the baseflow_index is, is the proportion

of the total stream flow, so of the total water that's in streams, which comes from groundwater rather than direct surface runoff.
So what that means is if there's water coming from groundwater, then it's coming from below the ground, so from water stored within soils or rocks, rather than just coming from rainfall falling from the sky and then kind of washing over the surface and directly into the rivers.
And...
The extent to which there is groundwater is quite variable in different places in Great Britain. So in this table, I've just kind of laid out two quite different examples of catchment behaviour in terms of the baseflow_index to help you kind of understand what this means. So in the first example, if we think about an upland catchment in Wales. So in these kind of areas, you might have a geology which is comprised of really impermeable rock. So water isn't really able to get into the rock and quite thin soils. And what this means is that when it rains, the water kind of rushes very quickly over the surface towards the rivers.
and that means that response to rainfall is very fast. So there's a short amount of time between it raining and the river responding.
And like additionally to that, the flow is also likely to drop away quite quickly after a rainfall event. And A catchment like this, we would say, has got a low baseflow_index because there's not very much rainfall coming from groundwater. It's mostly just coming from the inputs from the sky.
In contrast, if we think about chalk catchments, which are quite prevalent in the south of England, the underlying geology here is very, very different. So it's this kind of permeable chalk aquifer. So chalk basically has got lots of gaps in it, essentially, which means that it can store lots of water within it.
So when you get a rainfall event on this kind of catchment, then the response to that rainfall is much lower, because as well as some of the water kind of travelling across the surface, a fair amount of it also goes into the ground and becomes stored in the rocks. And that water is then kind of released much more slowly over time.
And so this means that the response to rainfall in these kind of catchments is much slower, but it continues for a much longer time. So the flow rates after a storm might be elevated for a much longer time period than you'd expect in this other catchment.
And so this would be an example of a
A high BFI catchment.
So I've kind of already alluded to this a bit, but like why is this useful? Why is the

baseflow_index kind of important to understand? It's important to understand for flood risk. So a low baseflow_index catchment is quite flashy, which means it's kind of likely to experience flash flood events.
Drought resilience, if a catchment's got a higher BFI, it's going to be much more likely to be resilient to drought because it's got this water that's kind of stored underneath the catchment, which can then sustain flow throughout dry periods. And those differences in the flow regimes can also have lots of other knock-on impacts, for instance,
on the kind of river ecology and the habitats and species which can survive in the catchment.
Okay, so...
Before kind of moving on with this, I've just included a plot of the same as we were doing before. So like a map of the baseflow_index, which some of you have already plotted in the previous stage, but I'll just include this here for those of you that focused on something else in the map making exercise.
And.
So, you can see there are some spatial patterns in the baseflow, so...
These catchments around the south and east of England have the highest values and the darkest green, and then the north of England and the west of Scotland have slightly lower values. It's not like super clear spatial patterns, but you can see that there is a bit of structure there.
Okay, so as I said before, we want to try to think about what might explain the reasons for these spatial patterns. So in what I've just been talking about, I've mentioned a few different things which could explain things. So the geology, the soils, the climate, the topography,
or most likely like a whole combination of all of these things together.
So what we're going to do here is to just kind of look into some of those relationships a bit more. So we're going to explore pairwise relationships, so just meaning the relationship between 2 variables at a time, and see if we can find kind of anything about what might be driving the baseflow_index.
So yeah, just to say this is like a kind of step towards prediction, which is what we're going to be doing in the next section. So yeah, if we can understand the factors driving the baseflow_index, then we can also kind of, it's a good suggestion that we might be able to predict it as well. Okay.
So...

What I'm going to do to start with here is to just look at the correlations between the baseflow_index and the climate variables. So I am going to get the climate variable names in a list as the first thing. So that's what this cell is doing.
I'm getting a list of the column names and then I need to delete a couple of them because, well, the gauge_id is not meaningful information about the climate, obviously. But there's also a couple in there which are like strings rather than numbers, so we can't look at correlations with them. The high precipitation timing, I think, is something to do with
The time of year.
Yeah, so the season during which most high precipitation days occur. So that's just like a string telling me the time of year. So we're going to remove those because they just won't work with this analysis. And so if I run this cell, then I get, if I just, this was commented out, so I just commented it back in. And then I can see that these are the climate variables that I'm going to be looking at.
When I, when I compute my correlations.
And then in this next bit, I'm going to calculate the correlations and then I'm going to sort the values and print them out to see what they are. So if I run this, or just to say, so here I've kind of repeated the same line twice more or less.
In the first version, I'm asking for absolute versions of the numbers. So that means like without the positive or negative, just the value itself. So if you want to just see like what the strongest correlation is and you don't care about the sign of the relationship, then that can be useful. But the thing that's printed out at the bottom is the absolute values. So if I run this,
You can see that the strongest relationships with the baseflow_index are the pet_mean, which is the mean evapotranspiration. And the next strongest relationship is this one at the bottom. So that's a negative relationship with the mean.
daily precipitation. So that's suggesting that the higher the rainfall, the lower the BFI value. So catchments with high precipitation tend to have lower baseflow contributions and that the higher the evapotranspiration, the higher the baseflow_index. So catchments where more moisture is lost to evaporation tend to have higher baseflow contributions.
Um...
So one thing to just note on this is that we could have a bit of a think about why these relationships might exist.
But you need to be careful to remember that just because a relationship exists

doesn't mean that one thing is like causing the other thing. So it's not necessarily true that higher rainfall causes a lower baseflow_index, although it might. It could just be the case that
the kinds of conditions that produce higher rainfall are also happen to produce lower baseball index, but the two might not actually be like directly related.
Yeah.
So we can also visualise the relationship between 2 variables using a scatter plot, which is just like a nice quick way to kind of see the relationship. So in this final cell, I have plotted the relationship between the baseflow_index and the mean daily precipitation.
And then these two lines, I'm just setting the label so that the plot is kind of understandable. And you can see that it's like a negative relationship, as we saw before. So
The no, it's not a negative relationship. So at higher baseflow_index values, you've got lower evapotranspiration values. Oh, I see. It's because I'm… the labels wrong. This should say mean precipitation.
So, is a negative relationship, and...
Yeah, so the higher the baseflow_index, the lower the precipitation generally.
Okay, how are we doing for time? Okay, so it's like 5 to 2 now and we were hoping to take a 10-minute break at...
Well, we're slightly behind, but that's okay, because we've got a long time for the final part. So we've got another task here to work on by yourself. I think 5 minutes to do this would probably be fine. So the idea here is, again, to just kind of do a replica. of the analysis that I've done above, but to use one of the other attribute data sets and to look at the relationships between those attributes and the baseflow_index. So here I have used the climatic attributes, but the idea is for you to use one of the other data sets. So we can see at the top
We had an.
these different data sets here. So instead of climatic attributes, you could use topography.
So if I put in topography there, I see a list of the topographical variables. And then from there, you could look at the correlations with those variables and see if you kind of identify any variables which have a strong relationship with the baseflow_index. So the strongest relationship we've seen so far is with this evapotranspiration.
Evapotranspiration, which is not 0.43.

So yeah, I'm going to give 5 minutes to work on this and just get as far as you can with it. Again, there's these cells below where you can type in your code.
And then once the 5 minutes up, 5 minutes is up, we're going to take our 10-minute break.
Does anyone want to ask anything about any of that before we start the five-minute timer?
No, okay. I will start the timer and just give that a go. If you've got any problems with making anything work, then just let us know.
Oh, and also just to say, if you do this and you find a variable with like an interesting relationship with the baseflow_index, like maybe a stronger relationship than the ones we've already seen, or if you've plotted a scatter plot and you're happy to share that in the chat, then please
do so. And again, there's like an extra bonus challenge if anyone finishes the task quickly and wants to have a look at that.
Okay, seeing some plots getting shared in the chat, very fancy one for Matt, which is nice, showing that you can also colour your scatter points by a third variable, which is really nice.
Uh, sure.
Yeah, sorry, that was my timer going off. Let me see what else.
Just give like another 30 seconds or so. If you are near the end, if you want to share something in the chat, then feel free. To be fair, I'm not sure if we're going to get any relationships that are stronger than the ones we already showed. But yeah, if you've got...
Anything which seems interesting then.
Surely.
Okay.
So I'm going to suggest that we start with the break now. So...
One second, there's my window.
Okay, so we're going to go for the break, 10 minutes. I'm going to say if we can get back by...
Uhh...
I don't know why this is not responding.
So if you can come back ready to start in 10 minutes, so at 2:15, then we'll start on this third section, which is doing the kind of building a model stage. I would advise you to take a break from your computer, stand up, get a drink or whatever, and

to come back at quarter past. If anyone wants to ask any questions about what we've just been doing, I can also stick around for a bit. But if not, yeah, please, please leave your computer and get a proper break.
Yeah.

JR **Jones, Robyn (MECP)** 1:01:54
Hi, Molly. I'm just wondering if, so when, if I wanted to change what it was comparing the like baseflow to, to the topography ones, do I still need to include the columns to remove like the gauge_id? Yeah.

MA **Molly Asher** 1:02:04
Yeah, so I realised when I actually went and did this myself that I should have told you this really because topography has got quite a few different strings in it as well. So when you try to
If you try to list the topography columns.

JR **Jones, Robyn (MECP)** 1:02:29
Mm-hmm.

MA **Molly Asher** 1:02:30
and then run this next cell, it doesn't work because you've got...

JR **Jones, Robyn (MECP)** 1:02:35
Yeah, that's what my screen looks like.

MA **Molly Asher** 1:02:37
Yeah, so I think that is because of, so if I just look at what the columns are, then you can see like gauge_name is in there. So if we take out gauge_name,
That might work and then I'm just going to try again. Okay, so then it works. So the problem was that gauge_name.
Being included.
Does that make sense?

JR **Jones, Robyn (MECP)** 1:03:07

That does, and to view those, you just typed in so climatic columns, so the topography columns.

Molly Asher 1:03:13
Yeah, so I'm being a bit lazy. Like if it in an ideal world, I would change this to say. topographic columns. And then once you've run that, if you just type the variable name itself.

Jones, Robyn (MECP) 1:03:21
Okay.

Molly Asher 1:03:27
Um...
and run that, then it prints out what those columns are.

Jones, Robyn (MECP) 1:03:33
Okay.

Molly Asher 1:03:34
And then the kind of like the way I knew that was because when I tried to run this it said.
could not convert string to flow. And then the thing that it printed out, because I've been looking at this data set, I was like, oh, I know that that's the name of the gauge. So I know that it must be because the gauge name's included. But that was a bit of a, I shouldn't have, I should have said that because that's not obvious.

Jones, Robyn (MECP) 1:03:57
Oh, no, no, that's it's good to hear how you walk through it.
OK thank you.

Molly Asher 1:04:05
Okay, no worries.
Yeah.

Molly Asher 1:11:49

Okay, so it's quarter past, so I'm just going to hope that everyone's back from their break and get started again. I thought just before moving on, we'll just quickly explain a little bit about the way Matt's done his plot in the chat.
So as well as kind of just doing the basic plotting of one variable against another, he's also introducing a colouring of the points, which can be really useful because it lets you kind of visualise like a third variable as well as the two that you're plotting on the X and Y axis. So what we were doing before was to just type in the plt.scatter function,
one variable for the x-axis and one variable for the y-axis. And that gives us the kind of plots we were just looking at. So baseflow_index against mean potential evapotranspiration. But you can introduce this colouring by using this third line where you type c.
which I think stands for colour. And then you can specify a third variable that you want to kind of introduce into your plot. So I think, I can't remember what Matt said exactly, but I've coloured it here by the latitude of the gauge. So then when you run that, your cells get a colour, which allows you to kind of
and make some more inferences about what's going on. So that's just, yeah, a little note on how to do that. Okay, I'm going to move on now to the...
Third section of the workshop.
And...
Okay, so we're going to be looking here at our third research question, which is whether we can predict catchment behaviour from landscape and climate attributes alone. So in the previous sections, we've looked at both describing and explaining spatial patterns and catchment behaviour.
So we're focusing on the baseflow_index, which as a little reminder, is the proportion of total stream flow, which is sustained by groundwater rather than by direct surface runoff. And we found that it varies across the country, substantially is maybe a bit of an exaggeration, but it definitely varies across the country. And we saw that there was no
single variable that was able to kind of clearly and clearly explain that variation. So this suggests that it's something which is the product of kind of multiple different factors all working together. And so this leads us on to the question that we're going to be tackling here, which is if we combine
And.
a larger number of these catchment attribute variables that we have as part of

CAMELS together, then would it be possible for us to predict the baseflow_index for a catchment where we've never measured this variable? So before getting started with this, just want to like just spend a minute to think about why we'd actually want to do that.
And so there's a couple of reasons. The first one, I've kind of said a fair bit about already, but like this is a really useful variable to be able to understand. So if we know a catchment's baseflow_index, then we can start to make some kind of assumptions about how that catchment might respond in terms of
both flood events and drought events. So it's a thing that's useful to know for a catchment. And the second thing is that calculating the baseflow_index requires flow measurements from the river. And these are just simply not available in lots of places. So measuring flow requires you to have various infrastructure in places.
And the majority of rivers worldwide have very little or even no flow record. And there's no kind of infrastructure set up to measure that. So that means that in those places, they don't have the data to calculate this and they're having to make decisions about lots of things, flood risk, water resources.
And if there was a way for them to be able to estimate the baseflow_index from catchment attribute data, which is likely to be much more readily available, then that's really useful. So just to kind of say again, if we can learn a relationship between catchment attributes and the baseflow_index from catchments where we do have both of those sets of data,
then theoretically that means that we can then estimate the behaviour of the baseflow_index in other places where we've got the kind of catchment attribute data but not the flow data.
Okay, so in terms of building models, we're going to work through two examples here and kind of compare their performance. So the first one is going to be just like a simple baseline model, which will be a linear regression. So that's just like a very very, very basic model which assumes linear relationships between variables and it assumes that each kind of predictor variable acts independently of all the other ones. We're then going to look at a more powerful machine learning model, which is non-linear and it can capture the interactions between the different predictors.
Okay, so for running this next section of code, we also need to do a bit more setup. So for this, we're importing some more specific packages. So specifically one called scikit-learn or sklearn, which is a free and open source Python package for machine learning. It's a really great package. It makes

Doing.
Lots of quite complicated machine learning operations and pre-processing of data and stuff, very simple. And you can kind of apply these techniques without having really in-depth knowledge of like quite complicated mathematical concepts. So it's very kind of user friendly, which is good. Again, I've
provided some links to more information about scikit-learn, which if you're interested, you can go back and have a look at later.
Okay, so again, just make sure that you're running the cells so that we don't have problems with like the things that you need not being loaded. So I'm just going to run this cell. So that's going to import those packages that I'm going to need for the rest of this modeling.
The next thing I'm going to do is to define the variables to use in the modelling process. So for this, I need to define both the variable that we want to predict, which is called the response variable. So here, that's the baseflow_index. So I'm defining that here. And then on the bottom line, I'm defining the variables that we're going to be basing our predictions upon.
So for this, we're just going to use the climate variables again, which is what we were looking at in the previous section when we were looking at the correlations. So I'm going to run this cell. So I've got both my response variable, thing I want to predict, and the predictor variables, which are the variables we're making our predictions with.
So at this point, I could go and just try and fit a model on all of the data that I have. However, I'm not going to do that because that would be kind of problematic in a few ways. So in order to make like a kind of honest evaluation of
the model performance. Instead of fitting it on all the data that we have, we're going to split it so that we've got the kind of majority of the data, 80% of the data, which we're going to use for training it. And then we're going to keep back 20% of the data that we've got for testing the model's performance. This is important because if we want
like an honest evaluation of the model's performance, we need to test it on data that the model didn't see beforehand, because otherwise it's essentially cheating. It's got like the results before it's having to make its predictions. So we don't want that to happen. And so to do this, we do this
80%, 20% split of the data, 80% for training to fit the model, 20% we keep back to evaluate the model's performance. So this basically means that in theory, the

evaluation we get of the model should be a realistic estimate of how useful that model would be out in the real world.
And.
Luckily for us, scikit-learn has an inbuilt function for splitting the data into testing and training sets. So this is just one line of code which will do this for us. You can edit this test size variable if you want to have more or less of the data in the test set. So this is currently set to give me 20% or
0.2 of the data in the test set. So I'm going to run this. I think I just included this below or just checking that it's doing what I expect. So if I look at the lengths of the test response variables and the train response variables, you can see that that's about the 20-80 that we'd expect.
In this cell, it says bonus materials. This is just something which anyone who's interested in can come back later and have a look at. It's about something called cross validation, which is like an extension of this concept of doing a training and testing split for robust model evaluation. It's something that most applications of machine learning models will
will use. We don't really have time to go into it in detail today, but if you're interested in this kind of thing, then it would be a useful concept to go back and kind of familiarise yourself with. Or if you want to ask any questions about that later, then you also can do.
Okay, so we've got our data ready. So at this point, we're going to have a go at fitting and applying that simple linear model.
So as I mentioned, this is very easy to do. So we specify this linear regression function, which was part of scikit-learn, imported that at the start.
Where is it? It was one of the things I imported was this linear regression function. So we specify that and then we fit the model using the training data. So the predictor variables and the response variable. And then we use the model to predict using this .predict function and we say we want it to make its predictions on our test dataset.
So I can run this code and it will generate predictions. It's basically just an array of numbers, which if you just look at like that is a bit meaningless. But those are basically its predictions for the baseflow_index for our test.
Data set.
So yeah, just looking at that is hard to interpret. So we need to do some proper evaluation of how the model is performing. So there's a few different ways which we can do this. And scikit-learn, again, has got these evaluation metrics included within

it, which can be quite easily applied to evaluate the model's performance.
So for instance, the ones we're going to talk about just now are the R-squared value and the RMSE or the root mean squared error. So the R-squared value tells us what proportion of variation in the baseflow_index across the catchments is explained by the predictors that we're using here. So
just going to run this cell below. So that's applying the R-squared score and printing out the results. So this shows that we get an R-squared value of 0.17. So that means that
17% of the variation in baseflow_index between different catchments can be explained based on those climate variables that we asked the model to base its predictions on.
So, that's not that's not like particularly good, but it's also doing something, which is a good start. And then the RMSE tells us the average magnitude of the model's prediction errors expressed in the same unit as the baseflow_index. So, a lower value indicates predictions that are closer to the observed values.
I personally find this a bit harder to kind of quickly understand than the R-squared value, but a good thing about it is that it's like a kind of concrete.
metric, so it's expressing the same units as the thing that you're trying to predict. So here we've got an RMSE of 0.12, which means that predictions are typically or on average wrong by 0.12 units of the BFI. And that is a
A variable that goes between zero and one, so that's like...
Yeah, again, quite a large error.
I think a nice intuitive way of kind of evaluating how good a model is performing is to plot a scatter plot. Again, so for this, what we are plotting is the
predicted values from the model against the observed values. And then this red line is basically showing you a one-to-one relationship between the two. So that's what we would aim for in an ideal world would be for the predictions to be the same as the observed values. And you can see by plotting this just very quickly,
what's kind of, you can get a general sense of how the model's doing. So you can see that there's a lot of scatter. The model's not doing very well. You can also pick out some specific things. So the observed BFI values go all the way up to 0.9.
We've got quite a lot of values up here. On the predictions, we don't have any values at all which are higher than 0.7.
So something's going on with our model's ability to predict higher baseflow_index values. I'm not sure why, but that's just like the kind of thing that it's useful to have a

look at the scatter plot and to kind of notice.
So...
So yeah, that's the linear model. So as I said before.
The kind of limitations of a linear model is that it assumes that each predictor has like an additive relationship with the baseflow_index and that that is linear. So for instance, doubling the rainfall will always double the effect on the baseflow_index, no matter of other factors. So like the soil type or topography or whatever,
that relationship, it assumes will always be the same. In reality, this is just not true. So the effect of rainfall on runoff is likely to depend on lots of different things. So the soil type, the topography, and it's these kind of interactions and non-linearities that a linear model is just not.
equipped to capture basically.
So it doesn't mean that it's useless. I think it's quite a nice thing to just start with anyway, to get a bit of a feel for what's going on. There's cases where you'll run a linear regression model and it'll do a really good job, in which case there's not really much point necessarily of using a more complicated model. And it's a nice kind of baseline.
against which you can compare something more complicated.
So we're going to try and do better than what the linear model did using a different kind of model called a random forest. So a random forest is like a machine learning model which is able to capture more complex relationships in the data.
And I'm just going to explain a little bit about how a random forest model works.
And to explain that, we need to start by understanding decision trees. So the reason it's called a random forest is basically because it is a collection of decision trees. So lots of trees turn into a forest.
So I guess you're probably all familiar with decision trees. These are, this is an example of like a classification decision tree. So if you want to classify an animal, you could create a decision tree like this. So first question is, does the animal have feathers? If yes, then go to kind of fly.
If yes, it's a hawk. If no, it's a penguin. If you answered no to the first question, you go to a different question. Does it have fins? If yes, it's a dolphin. If no, it's a bear. So that's pretty intuitive to understand. And you can do something similar with regression.
instead of classification. So in this case, the outputs are numbers rather than like classes.

So for predicting a continuous variable like the baseflow_index, on a basic level, you can set up something quite similar. So you can start by asking, for instance, is the mean annual precipitation greater than 1200 millimeters? If yes, then you go to another question, is aridity less than 0.5?
If yes, then you're going to predict a baseflow of around 0.6. If no, a baseflow_index of around 0.4. So basically at each stage, you're kind of segregating the data. So here we start by segregating into kind of low rainfall and high rainfall, and then you split the data again and again and again.
until you end up with an answer.
So decision trees like this are nice because they're quite intuitive. Like you can kind of print out something like this and you can understand what the model's doing, which is nice. But they have a weakness in the fact that they tend to overfit the data. And what this means is that they'll often kind of just
learn the specifics of a data set. And so they're good, very good with that specific data set, the training data set, but then they don't do very well when you kind of move them on to new unseen data.
And so this is where decision trees, no, sorry, this is where random forests come in. So the way a random forest works is basically for one kind of problem, you'll fit hundreds of decision trees, with each decision tree taking a different subset of the catchments and a different subset of the predictor variables at each split.
And then you will get your final answer by taking an average across all of the decision trees. And in the end, this basically ends up with you getting a more robust model.
And...
I don't think you need to understand like the specifics of why that happens, but I guess, yeah, I put here that it's a bit like if you ask one expert a question and they gave you an answer, you might trust their answer. But if you got 500 experts who all had kind of slightly different backgrounds and different starting points,
then and took an average across all of their answers, then you might well end up with a more trustworthy answer.
And.
So these kind of models are great for hydrology because they can handle non-linearities, which I've mentioned before, are very common in hydrology. They cope well with correlated predictors, so something like rainfall and aridity are obviously related.

and a random forest is much better than a linear model at kind of using both of those sensibly. And another good thing about them is that they produce feature importance scores. So these are basically information that we can get back, which tells us which of the variables that we gave the model it found to be the most useful in making its predictions.
This is quite good because in lots of machine learning models, it's much more of like a black box. You give it some data, it gives you an answer. You've got absolutely no idea how it arrived there. So I think it's quite a nice feature of Random Forest that at least you've got some indication of what's going on behind the scenes.
Okay, so we're now just going to fit the model, the random forest model. So this is equally as like quick and easy to implement as the linear model, even though it's way more complicated. But basically it's the same 3 lines of code, but this time
We're using the random forest regressor function rather than the linear regressor function that we used before. And we're using the same response and predictor train and test data as we were using with the linear model. So we just need to run this line. That's just a warning which you can ignore. And then I'm going to print out the same evaluation metrics as we used before. So the R-squared score and the RMSE score and plot a scatter plot to make this easy for us to kind of compare.
Within this cell, I've also included our linear model results from before. I'm going to plot the scatter plot for both the linear model and the random forest model side by side so that we can kind of look at that and easily see which of them is doing better than the other.
So, run that, so...
And okay, so we can see quite quickly that the random forest model is certainly doing better than the linear model. So it's got a R-squared score, which is about double that, even more than double that of the linear model.
So as a reminder of what that means, it's saying that 38% of the variation in baseflow_index amongst these catchments can be explained using the variables that we've given it to base its predictions on. So that's not bad. And it's got an RMSE of 0.1, so that's lower than it was with just the linear model.
In terms of the scatter plot, again, this is quite useful. As I mentioned before, there was the issue with the linear model not making any predictions over 0.7.
and see with the random forest, this has improved A lot. So we've got a good number of points greater than 0.7. And the distribution of predictions around the one-to-one line is kind of more

even at the higher values. So there's quite a lot of scatter around the line still, but it's not got that same kind of issue with high values that it seemed to be having before. And.
Okay, so the last thing is that, as I mentioned before, we can get back these feature importance scores. So this is a measure of how important it found the different features in making its predictions. So this is this feature importance function, which I'm using here in combination with a bar plot.
To just get a quick sense of these, so...
These are the features that it thinks are most important.
And I've just typed out here above what those variables are so that we can quickly check them. So it's saying the most important feature in its predictions was the low precipitation frequency. So that's the frequency of dry days. And then the next one was the mean potential evapotranspiration and then aridity.
One thing to know is that.
So I've just included this below, which is what we did in the second section, which was the pairwise correlations. And one thing to note is just that these are not the same. So it's the same set of variables, but the ordering of them between the pairwise correlations and the feature importance is not the same.
And...
This is because they're kind of, they're different things. So whereas the correlations was just looking at linear pairwise relationships. So it was only ever looking at the kind of target variable that we're predicting and one other variable. And the random forest importance measures are kind of looking at the whole picture at one time.
So A feature's importance is specifically how useful it was given that all of the other variables were present, which is different to just looking at the one by one.
Okay, that was a lot of information. And we've got like 20 minutes left. So there's another task at this point, which is basically to try and go back up to where I fitted to the random forest model.
And, oh, that's not it. Oh no, where is it?
That isn't. To go back to where I fitted the random forest model and to try and use some of the different variables. So
In this example, I was just using the climate variables, but we've also kind of seen before that we think there's lots of other different things which influence the baseflow_index, so the soils, the topography, etc. So this task was to go and to try and include some of the other variables from those catchment attributes and see if

you can improve the performance of the model. There is some hints below about how to do this.
So we could either kind of just get started on that, or if people want to ask any questions about anything that we've just covered,
We can also spend a bit of time on that.
Yeah, if you've got like a question that you'd like to ask, then feel free to pop your hand or shout out or whatever.
Okay, so no one, oh wait a second.
And.
Sorry, I'm just looking at the chat.
Um...
Okay, I'm not sure if that was from previously when I was speaking or not. So I'm just going to, we can get back to that, I guess, in the chat in a bit. But for now, okay, if no one's got any questions, I would say that maybe we just spend 10 minutes giving this task a go. And if and when you come into problems, then just
Ask for help.
I think in these cells that I've included below, I've tried to give you the stages that you need to do.
So yeah, I think just if you give it a go and then just let us know how you get on. And if you're struggling, we can we can kind of try and go through some things.

Tom Keel 1:40:10
Ohh, Molly, you did have a question from Tlakale, like, if you're using one variable, does that mean the random forest model is valid, basically?

MA **Molly Asher** 1:40:13
Um...
I think...
I guess like when I said about...
It's a good place to start with a linear regression model. That I don't think using a random forest model would be bad per se, but if you've only got 1 variable, then it's probably could be quite likely that a linear model would maybe do just as good a job. But I don't think there's any harm in

MA **Molly Asher** 1:40:59

Using a random forest model.
For that, I don't know. Do you have any other thoughts, Tom?

Tom Keel 1:41:07
Well, I mean, like random forests, like the best model even now at all machine learning that discounts overfitting. So you could just throw 100 variables at it and it would be smart enough to sort of just realise that there's aspects of it that aren't. Like, so some variables aren't useful for getting predictions.

Tom Keel 1:41:40
As Sebastian also mentions, the trimming the feature importance as well. Like, that's just another way that you could throw lots of variables at a random forest in the hope that, oh, in combination, they do contribute to better understanding. Then there's a really nice statistic called feature importance, which
you can just use to trim out things which are not important, basically.

MA

Molly Asher 1:42:08
Yeah, which I guess maybe the motivation for doing would be that it would like increase the speed of your model running.

Tom Keel 1:42:15
Exactly.

MA

Molly Asher 1:42:16
Yeah.

Tom Keel 1:46:22
Sebastian, if you don't mind, do you mind clarifying what you mean by your question?

YS

Yelesen, Sebastian 1:46:34
Yeah, I thought ordinary R-squared gets, like, can get distorted when you've got a lot of, like, complex models with a lot of features.

Tom Keel 1:46:36

Mm.
Mm.

Yelesen, Sebastian 1:46:50
That's what I mean by just a R-square, just as a form that kind of tries to correct for the fact that you've got, you know, you've got a lot of explanatory variables.

Tom Keel 1:47:03
Yeah, I forget the term for it for machine learning, but...
Is it collinearity? I think, but basically what you would do, multi-collinearity, is before you fit any sort of model, you would check how correlated they were with each other, like how much they vary at the same.
like together and then you'd set some threshold to remove things that are certain.
of collinear, basically. So I can put a link for, but yeah, I think that's essentially, I mean, I'm more of a, know the word multicollinearity, but I mean, maybe that's what adjusted R squared is trying to do anyway, it's just the same thing with a different name.

Yelesen, Sebastian 1:47:37
Yeah.

Tom Keel 1:47:53
So, I'll put a link for that, so.

Yelesen, Sebastian 1:47:57
Yeah, yeah, okay, I understand. So the ordinary R-squared, as you add more features, will always increase.
Like the score will always go up when you as you increase the amount of features as well, the adjusted one.
tries to correct for that, that effect. So that like, for example, this is how like, you know, for example, when you have like random forest, you know, chuck in 500 variables and then, and then the adjusted one will try to correct for the, for that, for that.

Yelesen, Sebastian 1:48:35

The fact that you've got, you know, very large...
A large, complex model.

Tom Keel 1:48:40
Yeah, OK. Looks at how, like, the cumulative effect of having more variables, basically.

YS

Yelesen, Sebastian 1:48:48
You can actually, if you have any data plot in Excel and then you have like a polynomial regression and you just increase the order of that polynomial, the R-squared will always go up as you get to like order 16 polynomial regression, even though that's obviously too complex, right, for the for the for the data to fit.

Tom Keel 1:49:19
I don't really know, actually.
I don't know if random forests work like that, because they're nonlinear. I feel like more information could actually hurt learning, but I mean, potentially...
It's a good opportunity to try adding more, seeing if there's like some sort of combinations of variables that don't add any more information.

YS

Yelesen, Sebastian 1:49:52
Yeah.
I think the good thing about an R-squared is you can get it out of, you can retrieve an R-squared or an RMSE from a lot of different models.

YS

Yelesen, Sebastian 1:50:05
And what what approach? Yeah, you can get an adjusted R-squared for and for us.

YS

Yelesen, Sebastian 1:50:13
I think, I don't know how you would. Let's have a look and see if we can pass a different argument to it.

Tom Keel 1:50:22
Yeah, I mean, yeah, you raise a really good point there about like benchmarking this. I think Molly mentioned that earlier, comparing to different models together as possibly the way we get the best answers from models. And it also like stops us from

using certain models if we're not getting any more information, linear regression or something simpler.

Yelesen, Sebastian 1:50:58
Have you heard of lazypredict?
It's like a.
It's like a Python library that basically extends scikit-learn so that you can pass it to like a predictor and a set of variables and it will try all of the models available in sklearn automatically like
plug and play thing and then spits out a large table with the diagnostics in. So if you're like starting off with a new study and you don't really know what models to explore, it can like quickly, you know, give you pointers in terms of like, oh,
Does random forest perform well? Does linear regression perform well? I think that's one of the most frustrating parts, is kind of working out what models are appropriate to use for a different application.

Tom Keel 1:52:03
Wow, yeah.
That must take hours to run on some datasets, but do you mind sharing a link to that, or if not, I can as well?

Yelesen, Sebastian 1:52:12
Yeah, I'll share it now.

Tom Keel 1:52:14
Thank you. Back to you, Molly. Sorry.

Molly Asher 1:52:19
And yeah, so that's the that was the end of the 10 minutes, and we've obviously only got 5 minutes left.
So...
I don't know.
How people have got on?
With the.
I don't know, I could just kind of show a quick example of me adjusting it for a

different set of variables, which is maybe helpful if anyone wants to then just go and have a look at it a bit more after the workshop. Conscious we've got very little time left, but I might just quickly
go back to my notebook and just show you what I've done and then hopefully that might be helpful. So what I've done here is to just instead of look at the relationship with the climate variables to look at the soil variables instead. So
The first stage was to get the columns, the soil columns. So instead of the climate columns, I'm just getting the soil columns. So this was using code that was right up at the top of the notebook, which was maybe a bit tricky to find. So those are, there's loads of soil variables, which has some problems as was just being discussed.
And then I've got this set of new predictors which are the soil columns.
And then I'm...
I need to recreate the...
Testing and training data, so here.
The response variable is the same. We're still trying to predict the baseflow_index.
Predictor variables are now going to be my soil columns.
That was not relevant. It's going to be the soil columns from above. I then recreated the test and training data.
Then we run the model using that, and then we can evaluate it.
This shows that using just the soil variables rather than just the climate variables, the model performance is actually quite a lot better. And so it's got a higher R-squared value and it's got less kind of scatter around that predictions versus observed values line.
which suggests that maybe if we kind of took some of the most important soil variables, if we looked at the feature importance plot, and then we also took some of the most important climate variables from looking at the feature importance plot from our original model and joined those together,
then perhaps we'd get like an even better model.
And.
Does anyone have any kind of final questions or comments on anything that we've covered?

Amulya Chevuturi 1:55:31
There is a question by Robyn that I didn't understand.
Um...

If you wanted to take that.
Molly.

Molly Asher 1:55:43
I'm just having a look.

Jones, Robyn (MECP) 1:55:44
I just missed, I think I figured out when you went through, I missed kind of the first step. I didn't remove the columns with the string again, so I don't think it actually brought in any new columns and it just remade the same graphs as before.

Molly Asher 1:55:59
Okay, do you understand how you would do that now?

Jones, Robyn (MECP) 1:56:05
I think so, but I actually have to leave this call. I have a meeting at 10, so.

Molly Asher 1:56:11
Yeah, that's no worries. We're at the time, I guess.

Jones, Robyn (MECP) 1:56:11
Yeah, yeah, sorry, I want to know, but work.

Molly Asher 1:56:17
OK, thanks. Martha, do you want to do you want to ask your question just quickly?

Martha McKee 1:56:23
Yeah, sure. Thank you. Sorry if this had already been answered at a later date, but I was just wondering, when you're making models like this and you're kind of chopping and changing things and trying to get the best fit as you go along, how far do you go until you've overfit the data? You know, like how far do you keep editing it and changing it and adding certain
variables to it before you've kind of made something redundant because you've tampered with it so much.

Molly Asher 1:56:53
I don't think overfitting the model would be a result of you like tampering with the model as you described. Like I think just like kind of playing around with which variables you include and then kind of tailoring it as you see which things improve your model. I don't think that

Molly Asher 1:57:12
would be my understanding of overfitting the model. So I don't think that would be a concern. I don't know, does anyone else disagree?
Yeah, I think I think that would more likely to be that would be more likely to be a problem if you.
if you if there was like an issue with your data, like if you didn't have much like diversity, for instance, in the catchments that you were trying to model, then you might end up with a model that was just like really good for that small section of catchments, which were all quite similar. But then if you went and tried to use it on like a wider section of catchments, which were quite different. It would probably not do very well, even though it seemed like it would do well initially. But yeah, I don't think just like what I've done here. I've said, okay, these climate variables are quite useful and then these soil variables are quite useful and then I'll join them together and make a model. I don't think that is really...

Martha McKee 1:58:22
That doesn't cause, doesn't contribute to overfitting. OK, that makes sense. Thank you.


Tom Keel 1:58:30
Just to add on that, ohh, go on, Gianni.


Gianni Vesuviano 1:58:30
I think.
No, you go ahead Tom first, and then I'll go.


Tom Keel 1:58:35
Oh, very quickly. I think it's also, yeah, Martha, it's down to us as sort of scientists to

really know your data and know what's appropriate to add or not, really. I think that's the real trick, because you can make a conclusion that says, oh, I need to actually have all these things included, and then another scientist come along.
and give a different opinion on maybe don't include this variable because it's not useful, you're just throwing information at it. There's all of these statistics which give you some sort of metric for when I add this, it improves the score, this amount. But then you as the scientist with that domain knowledge, you know what your data is, you know what's appropriate to add.
But yeah, it's completely down to what you're trying to solve, really. Well, as well, you might.

Gianni Vesuviano 1:59:23
Yeah. And I was going to say that I actually think that looking into the model, into the data, playing around with the variables, seeing what they do, adding them, removing them if they don't do much, that could actually act against the risk of overfitting.

MA

Molly Asher 1:59:51
Okay, I guess that we are going to end there. I think just to say, like throughout the notebook, there's been a few like additional bonus materials and bonus tasks. So if you want to go back and have a look at any of those and give them a try, then please feel free to do so.
And...
And yeah, I hope everyone enjoyed the workshop.

YS

Yelesen, Sebastian 2:00:20
Thanks for your time.

VS

Víctor Malagón Santos 2:00:23
Thank you so much.

Gianni Vesuviano 2:00:23
Yeah.

Molly Asher 2:00:24
Thanks.

Gianni Vesuviano 2:00:25
Thank you, everybody. Following this workshop, we're going to have a break for the next 26 minutes, and then we'll be coming back to the final session of the summer school.
And that is going to be a presentation on how to write highly cited papers. So see you in 25 minutes.

Amulya Chevuturi stopped transcription