Programming assignment, R

Ramon Diaz-Uriarte

2024-11-02 (Release: Rev: 8a51715)

A Introduction

The following are the available options. **You have to choose one of them**. I might (might, not necessarily) be persuaded that another option is a great one, but you'll have to explain why. And if you want to do something different you **must** have formed a group and have discussed this with me before Christmas break.

Some of the projects are easier than others (but also possibly more boring). Some of the projects involve more work mapping the biology to the computing, whereas others are more directly computational (little biology involved). Finally, I provide a few references for some projects, but you would of course want to look for additional references (how do I do this? go to PubMed and Google Scholar and look for papers that cite the given references as well as for related papers).

Yes, if any of this ends up in, say, the documentation of OncoSimulR or similar, I give credit where credit is due (see help of OncoSimulR for examples of previous students of this course who are listed as authors of the package and vignette). But, of course, you can **opt out** of this: if you do not want your code to be used in the documentation, examples, etc, just let me know (the copyright is yours).

Oh, yes, and maybe there are too many options[1].

B Groups

We are around 35 (?). Options:

- 9 groups of 4
- 7 groups of 5

We will settle this once we know the exact number of people who are enrolled. If there are groups of different sizes, larger groups will be expected to do more/better work.

Once settled, assign yourselves to the groups in Moodle (Auto-seleccion de grupos, "Groups for practical programming exercise"). If this does not work, I'll create groups randomly. **Groups should have been created by 9-December**.

C What you need to present and send at the end

1. You will give a presentation the day of the final exam (in January). You will need to send me that presentation at the end of that day.
2. You will also need to send me the code and any additional documentation. You will do this via Moodle in "Programming exercise: file upload, grades and feedback".
3. With the previous additional documentation, add, if appropriate, a link to your github repo. That is specially important if you make a pull request.
4. Deadlines: the presentation. (Oh, you should not count on my answering questions, or answering them promptly, during Christmas vacation.)

[1] See the book by Barry Schwartz, *The paradox of choice*.

D References

There are a few PDFs in Moodle with possibly hard to find references for some of the projects. Other references, from journals, are provided below and you might want to search for more on your own.

E Git and github suggestions

You must use version control for this project. git is, as of today, probably the best option.

You should use a repository, possibly a public one, in places like github, bitbucket, gitlab, etc. For some of the projects it will be github, since that is where I keep the OncoSimulR, EvAM-Tools, and other repos.

There are many git tutorials. All I know about git and github can fit in three short sentences. That is probably all you need for this work.

The simplest (in the long term, but also in the medium term) is to

1. (Create a github account if you don't have one. One for each one of you is the simplest)
2. Fork the repository
3. Clone from your fork
4. Create a new branch in your local machine
5. Give access permissions to all your team members to the remote repository
6. Work as you see appropriate, probably submitting to the remote repo
7. Remember to check for updates to the original repo (i.e., to grab the new code I might add during the next few weeks)

The above are just suggestions. Do as you see fit. However, if you want me to possibly incorporate your work it is a lot simpler if you do a pull request from github.

F For projects that involve OncoSimulR: What version of the code

Make sure you use the current master branch from github: https://github.com/rdiaz02/OncoSimul. Fork the github repo.

Before installing it, though, you might want to install the version available from BioConductor, so that all dependencies are satisfied. (As of today, BioC has the most recent version of the code).

Note that I might occasionally add some code to that branch. It is important for you to keep your copy of the code updated!

G For projects that involve EvAM-Tools: What version of the code

Make sure you use the current master branch from github: https://github.com/rdiaz02/EvAM-Tools. Fork the github repo.

EvAM-Tools has some annoying dependencies. That is the way it is, and it is not my fault.

Note that I might occasionally add some code to that branch. It is important for you to keep your copy of the code updated!

G.a EvAM-Tools: Linux, Windows, Docker

Installing the EvAM-Tools R package in Linux is relatively straightforward. Moreover, you do not need the H-ESBCN or MC-CBN or CBN code for these projects. I suspect that similar comments apply to Mac.

Installing it in Windows is much more complicated. Some people have succeeded in the past, but it seems to have been painful. Alternatively, you can use a Docker image that includes the package, all the dependencies, and RStudio. You will need to ensure that Docker can see your local files.

You can use Docker under any other operating system too, though this might not bring any advantage under Linux, and add some disadvantages. But you decide.

H Keep this reasonable

This is not a TFM (but you are not doing this on your own, but in groups). Some of the things that I suggest below are optional and might not be feasible; in other words, a project might be a good one even if not all of the things I say below are done.

At the same time, this uncertainty and the lack of a simple instruction like "There are exactly three cups and two plates in the kitchen sink; please wash and dry them" is something you are very likely to experience in a TFM or PhD.

I Which are the easiest/hardest/most confusing projects

The project that involves the least reading about biology, and probably the least reading overall is *"The perverse procedure: "Normality test and if reject use Wilcoxon; otherwise use a t-test"" (section 1)*. I expect you will do a fair number of simulations and take care that what you are doing makes sense statistically. So little biology but a lot more stats; you will want to look at how these problems are addressed in the statistical literature.

Project *"Simulating frequency dependent fitness in cancer" (section 2)* is, I think, not conceptually difficult but requires reading some papers and playing with OncoSimulR. This is a large package with a manual that is over 200 pages; you do NOT need to read all of it (a lot can be skipped) but you need to know what to skip. You should have (or be willing to acquire) some understanding of fitness and frequency-dependent fitness, and you will need to become familiar with the idea of restrictions in the accumulation of mutations. For some reason, most people do not really get what this project is about (maybe it is poorly explained?).

Project *"Mutual hazard networks in R (evamtools) and Python" (section 3)* requires making yourself familiar with a specific method (MHN) and understanding it might require a passing understanding of continuous-time Markov Chains (which can be gained in a few hours). And then, understanding what the authors are doing to correct for collider bias. It will involve using both Python and R and understanding some parts of one particular package (evamtools). If you use Windows, there might be additional complications with installing evamtools that will require Docker.

The above three are probably the most clearly specified projects (but, again, *"Simulating frequency dependent fitness in cancer" (section 2)* is, consistently, a project that is misunderstood).

The remaining two are much more open and addressing them probably requires the most knowledge of what the phenomena are about.

"Simulating interventions and adaptive therapy" (section 4) requires knowing what adaptive therapy is and becoming familiar with OncoSimulR. The list of references is long. However, there are two or three essential references (those by Hansen) that quickly give an idea of that adaptive therapy is about. However, the key is not adaptive therapy *per se* but how to simulate it with OncoSimulR **and** the role of restrictions in the aquisition of mutations. In the past, the objectives of this project have been misunderstood by some groups, who have just repeated (simply changing values of parameters or labels of things simulated) already existing examples.

"A recipe book for simulations with OncoSimulR" (section 5) requires judiciously reading a large paper and understanding what are interesting scenarios to simulate. And, then, becoming familiar with OncoSimulR and running a bunch of simulations.

---

1 The perverse procedure: "Normality test and if reject use Wilcoxon; otherwise use a t-test"

The objective is to show, via simulations, what are the problems of the perverse procedure discussed in class, in section 14.5 of the stats notes ("A bad (very bad, terrible!) way to choose between nonpara- metric and parametric procedures"). You might want to look at the existing literature (maybe Harrell's "Regresion modeling strategies" has discussed it already? R. Miller in "Beyond Anova" probably has). Some ideas

- Simulate data under different types of distributions and different sample sizes.
- Compare type I error rates of the procedure with just using the t-test and just using Wilcoxon.
- Compare power of the procedure with just using the t-test and just using Wilcoxon.

The key idea here is to emulate the procedure and answer the question: "If people were to use this procedure, what would happen?"

2 Simulating frequency dependent fitness in cancer

The paper by Axelrod et al., 2006 (below) is a classic. However, I know of no examples where there are empirical data that would allow us to simulate processes using, for example, OncoSimulR, for the scenarios in Axelrod. Yes, there are many frequency-dependent examples in OncoSimulR (and many in the literature), but none really respond to Axelrod's scenarios. The objective of this project is:

- Locate biological examples that can naturally be related to Axelrod's examples; some might be available from Mario's TFM. Ideally, they would be about cancer. Other diseases might work too, but cancer has some specific features that make it different so many other examples will not do (e.g., games in the gut microbiota are a qualitatively different phenomenon: can you tell why?).
- Simulate them with OncoSimulR.

2.1 Steps and what to add to the code

1. Understand Axelrod's model. Read the paper, Mario Aguilar's TFM, and go over the examples in OncoSimulR.
2. Locate examples in the literature that can reflect Axelrod's scenarios.
3. Simulate them with OncoSimulR. A key idea is simulating processes with and without frequency dependence: scenarios where the same genotypes are involved, but comparing scenarios where fitness involves cases in which there is frequency dependence with scenarios in which fitness is not frequency dependent.
4. Ideally, simulate sampling from them (using OncoSimulR): do frequencies of genotypes and genes, and "observed genotypes (composite genotypes)" differ between scenarios and with different sampling regimes?
5. If you are interested in this project, talk to me so that I can provide some additional details about what I mean by all this. A key conceptual issue is specifying fitness reasonably.
6. Illustrate the examples via a vignette.
7. **Beware**: this project can be "done" using a ridiculously small amount of work. **Do not try to do that**. The key of doing this well is carefully thinking about the examples, and modelling the fitness relationships between clones and showing how this matters for what one sees when one samples.
8. **Please pay attention to what this project is about**: this is not about just simulating cancer with freq-dep-fitness or replicating simple examples from the literature and adding to the examples: there are already many of those in the vignette. **Focus on Axelrod's ideas** and **think about the acquisition of certain features**, acquisitions that might be under order constraints.
9. **Again: Focus on Axelrod's ideas. Think about the acquisition of certain features, acquisitions that might be under order constraints.**

2.2 What to do: explained in a different way

1. You might want to read p. 41, section "4.7 Frequency-dependent fitness" of Diaz-Uriarte and Johnston, 2024. (And this probably will require reading also pages 1 to 9, section 3, section 4.4, and maybe a few others, judiciously).
2. Read Kuipers et al., 2021 and Luo et al., 2023, cited in Diaz-Uriarte and Johnston. You do not need to read the entire papers, just the comments about exclusivities. "Kuipers et al. (2021, p. 11) discuss biological explanations of different 'exclusivity' patterns, including: (a) mutation pairs that result in complementary phenotypes that cooperate; (b) synthetically lethal pairs of mutations; (c) different mutations with similar phenotypic effects that lead to parallel and convergent evolution.". Luo et al. (2023, p. 2) differentiate between "mutual exclusivity" (defined at the patient level) and "clonal exclusivity" (defined at the clone level) with the consequence that "(...) the consensus genotype [what we have referred to as "mutational profile" or "tumour profile"] of a tumor can contain some or all of the clonally exclusive mutations, but the above CPMs will treat them as evidence of co-occurrence."

3. You might want to read Mario Aguilar Herrador's TFM and QiYao Ye's TFG for inspiration. **Just some inspiration: do not imitate them, since they do not really do what is being asked here**. Why? Because what is asked here is to **focus on Axelrod's ideas. Think about the acquisition of certain features, acquisitions that might be under order constraints. What happens when we consider frequency-dependence?**

4. Doing this work requires playing with OncoSimulR and its frequency dependent-fitness specifications (https://rdiaz02.github.io/OncoSimul/OncoSimulR.html#10_Frequency-dependent_fitness, https://rdiaz02.github.io/OncoSimul/OncoSimulR.html#11_Additional_examples_of_frequency-dependent_fitness). Of course, you might need to read other sections to understand this.

5. You need to define what is meant by "Population DAG" in Figure 20 of Diaz-Uriarte and Johnston, 2024. Is this what Axelrod et al. are referring to?

6. Create scenarios that are comparable for individual-based DAGs of restrictions and population-based DAGs of restrictions.

7. Sample from the two types of scenarios, using both bulk sampling (whole tumor) and single cell sampling. OncoSimulR offers options to do this from the simulated data.

8. Ideally, compare what the different samples show.

9. Even more ideally (though probably not feasible), run some CPMs (CBN, MHN, whatever) and see what they infer from the different models and types of samples. I wrote "even more ideally": unless things go exceptionally well, I am not expecting you to do this!

2.3 References

Axelrod, R., Axelrod, D. E., & Pienta, K. J. (2006). Evolution of cooperation among tumor cells. PNAS, 103(36), 13474–13479. http://dx.doi.org/10.1073/pnas.0606053103

Diaz-Uriarte, R., & Johnston, I. G. (2024). A picture guide to cancer progression and monotonic accumulation models: Evolutionary assumptions, plausible interpretations, and alternative uses (arXiv:2312.06824). arXiv. https://doi.org/10.48550/arXiv.2312.06824.

OncoSimulR's references (https://rdiaz02.github.io/OncoSimul/OncoSimulR.html#10_Frequency-dependent_fitness, https://rdiaz02.github.io/OncoSimul/OncoSimulR.html#11_Additional_examples_of_frequency-dependent_fitness).

Mario Aguilar Herrador's TFM.

QiYao Ye's TFG.

3 Mutual hazard networks in R (evamtools) and Python

There is a method for cancer progression models, or evolutionary accumulation models, called "mutual hazard networks". It is implemented in evamtools (https://github.com/rdiaz02/EvAM-Tools) using original code from the authors. The authors have now developed new code in Python and have extended their model to correct for collider bias.

These are some questions that could be addressed in this project:

- What are the differences/advantages/disadvantages of the two implementations?
  - functionality?
  - speed?
  - different models?
  - input/output?
  - etc
  - Note: not only the MHN code itself but also the bias correction: https://github.com/spang-lab/LearnMHN, https://github.com/cbg-ethz/ObservationMHN
- Would it be possible to call the Python code from evamtools (evamtools the R package, NOT the web app)?
- Would it be possible to modify the existing R code for MHN in evamtools (evamtools the R package, NOT the web app) to use bias correction?

3.1 Steps

1. Fork the two MHN repos
2. Read first 16 pages of picture guide (Diaz-Uriarte and Johnston, 2024) AND section 4.8 (Selection bias) on p. 41.
3. Read the two MHN papers (cited in the review: Schill et al., 2020; Schill et al. 2024; en la p. 59 del review).
4. Play with the MHN Python code.
5. Play with evamtools web app to get an idea of what MHN does in evamtools. Note that there is detailed help, a tutorial, etc.
6. Install evamtools, the R package (or the Docker image for evamtools). Using Docker requires installing Docker and, then, installing the evamtools image with RStudio: https://hub.docker.com/r/rdiaz02/evamrstudio; see also https://github.com/rdiaz02/EvAM-Tools?tab=readme-ov-file#how-to-run-the-r-package-from-the-docker-image)
7. What is available from the Python code that is available or not available from evamtools? Two subquestions: a) from evamtools web app; b) from evamtools package.
8. Speed differences: run several computational experiments. If you have access to machines with CUDA you might want to explore this too. But think critically: is this speed difference relevant for reasonably sized datasets?
9. Would it be possible to implement the bias-correction procedure in our current MHN implementation in evamtools?
10. Would it be possible to call the MHN Python code (which one? bias-correction?) from evamtools?

3.2 References

Diaz-Uriarte, R., & Johnston, I. G. (2024). A picture guide to cancer progression and monotonic accumulation models: Evolutionary assumptions, plausible interpretations, and alternative uses (arXiv:2312.06824). arXiv. https://doi.org/10.48550/arXiv.2312.06824.

Schill, R., Solbrig, S., Wettig, T., & Spang, R. (2020). Modelling cancer progression using mutual hazard networks. Bioinformatics, 36(1), 241–249. https://doi.org/10.1093/bioinformatics/btz513.

Schill, R., Klever, M., Losch Andreas, Hu, Y. L., Vocht, S., Rupp, K., Grasedyck, L., Spang, R., & Beerenwinkel, N. (2024). Overcoming observation bias for cancer progression modeling. In J. Ma (Editor), Research in computational molecular biology (pp. 217–234). Springer Nature Switzerland. https://doi.org/10.1007/978-1-0716-3989-4_14

evamtools: see the web-based app: https://www.iib.uam.es/evamtools/ and the github repo: https://github.com/rdiaz02/EvAM-Tools. There is a short paper too: Diaz-Uriarte and Herrera-Nieto, 2022.

4 Simulating interventions and adaptive therapy

OncoSimulR allows for the simulation of interventions and adaptive therapy (see the vignette, where this is documented in detail, with examples). For example, in https://rdiaz02.github.io/OncoSimul/OncoSimulR.html, see sections "Simulating therapeutic interventions ..." (https://rdiaz02.github.io/OncoSimul/OncoSimulR.html#14_Simulating_therapeutic_interventions_and_adaptive_therapy,_and_using_user-defined_variables) and "Simulating therapeutic interventions that depend on time" (https://rdiaz02.github.io/OncoSimul/OncoSimulR.html#15_Simulating_therapeutic_interventions_that_depend_on_time).

Some context: Sergio Sánchez Carrillo, as part of his TFM for this Masters, added frequency-dependent fitness functionality to OncoSimulR. Frequency-dependent fitness allows us to model evolutionary scenarios where the fitness of a genotype depends on the frequency (or density) of other genotypes. Things such as game theory, or the prisoner's dilemma might be familiar to you. Frequency-dependent fitness is thought to be a possible mechanism that accelerates cancer development, via cooperation between several pre-cancerous cell types. As an added benefit of Sergio's work, we can simulate interventions in the tumor development process, as we can make fitness conditional on population size, or population size of specific clones, or relative population fractions (e.g., relative proportion of one clone relative to another clone). Later, Niklas Endres, a former student of this class, realized we could also easily make fitness depend on time (and he implemented this feature). That means that we can simulate killing genotypes or making certain genes lethal, etc. For example, we can simulate the growth of a tumor or of bacterial populations, and then target some of the subpopulations with antibiotics or chemotherapy. Several examples are available in the vignette, the section called "Simulating therapeutic interventions" . Later, Javier López Cano and Javier Muñoz Haro (TFG students from the EPS) added flexible intervention mechanisms and user variables.

What is the purpose of this exercise? To simulate adaptive therapy regimes using OncoSimulR extending the set of examples available and explore (push to its limits too?) the existing functionality. I said extending, so **please, do not repeat existing examples**. For example, you do not want to simply add an example that is structurally similar to an existing one but only differs on the value of some parameters. **Again: PLEASE, DO NOT REPEAT EXISTING EXAMPLES.**

A possibly very interesting approach to this project is that most examples of adaptive therapy and resistance do not account for restrictions in the order of accumulation of mutations (but see the refereence to Gjini and Wood, below). OncoSimulR is uniquely suited to model these problems. What happens when there are restrictions in how resistance is acquired and we add adaptive therapy? Should adaptive therapy schemes change? (Of course, ideally you would not simply make up fictitious scenarios of restrictions based on existing ones without restrictions, but rather try to find support in the literature for modeling, for example, that a mutation in gene X that confers resistance depends on this previous mutation in gene Z, etc, etc).

Look through literature, and simulate examples. Oh, there is no reason to restrict yourself to cancer. You can instead use examples from bacteria or viruses, etc (asexually reproducing things).

Beware: for some examples you might need to work around limitations of OnconSimulR. Point them out: what would you do differently if these possible limitations did not exist (i.e., if the developers of OncoSimulR had written more flexible software)?

A key objective here is to present one or more examples that clearly show a story, in a well organized way. This includes code that will run (e.g., Rmd or Rnw, but just well commented R file will work) with a sensible example or examples that illustrate something. And remember some key ideas: **adaptive interventions** and **restrictions in how mutations accumulate**.

4.1 Steps and what to add to the code

1. Familiarize yourself with the ideas behind adaptive therapy (see references below).
2. Understand the functionality available in OncoSimulR to simulate interventions.
3. Clone the github repository of OncoSimulR.
4. Try to replicate scenarios where adaptive therapy can be used, possible altering fitness landscapes and intervention regimes. Launch simulations and analyze cases.
5. Identify if there are problems with the current implementation that prevent properly using OncoSimulR to model interventions.

6. Ideally, modify the Rmd/Rnw vignette files to show the examples. When you do this, make sure the examples run in reasonable time. In other words, if needed, differentiate between "full size examples", which might take a few minutes to run, and the minimal example for expository/pedagogical purposes that you would add to the vignette.

7. You might want to add additional tests to the code if you deem it important or relevant (to illustrate existing functionality or to illustrate missing functionality).

8. Ideally, when everything is done you should then do a pull request and I should be able to pull your code into the OncoSimulR code (and list you as authors).

4.2 Anticipated issues and things to be aware of

The OncoSimulR documentation already includes many examples of adaptive therapy. It will, therefore, not be easy to add additional examples. You might want to look in detail in the paper by Hansen and Read, 2020. In addition, a possibly very interesting approach would be simulating scenarios such as those in Gjini and Wood, 2021; this would certainly be new stuff: they discuss constraints in mutational pathways.

As said above, a particularly promising/intriguing/fascinating take would be to explore what happens when there are restrictions in the order of accumulation of mutations, for example resistance mutations, maybe due to epistasis and other similar effects.

So, when you present your work, say what is novel with respect to what is already available in the OncoSimulR documentation.

4.3 References for adaptive therapy/interventions

Start with the two papers by Hansen and the paper by Gjini. If you get into the project, look at the others, or others that cite these, as needed.

Akhmetzhanov, A. R., & Hochberg, M. E. (2015). Dynamics of preventive vs post-diagnostic cancer control using low-impact measures. ELife, 4, e06266. https://doi.org/10.7554/eLife.06266

Enriquez-Navas, P. M. & Gatenby, R. A. Chapter 14 - Applying Tools From Evolutionary Biology to Cancer Research. In: *Ecology and Evolution of Cancer*. Ed. by B. Ujvari, B. Roche, and F. Thomas. Academic Press, 2017, pp. 193–200.

Gatenby, R. A., Silva, A. S., Gillies, R. J., & Frieden, B. R. (2009). Adaptive therapy. Cancer Research, 69(11), 4894–4903. https://doi.org/10.1158/0008-5472.CAN-08-3658

Gjini, E., & Wood, K. B. (2021). Price equation captures the role of drug interactions and collateral effects in the evolution of multidrug resistance. eLife, 10(), 64851. http://dx.doi.org/10.7554/eLife.64851

Gluzman, M., Scott, J. G., & Vladimirsky, A. (2020). Optimizing adaptive cancer therapy: Dynamic programming and evolutionary game theory. Proceedings of the Royal Society B: Biological Sciences, 287(1925), 20192454. https://doi.org/10.1098/rspb.2019.2454

Hansen, E., Woods, R. J., & Read, A. F. (2017). How to Use a Chemotherapeutic Agent When Resistance to It Threatens the Patient. PLOS Biology, 15(2), e2001110. https://doi.org/10.1371/journal.pbio.2001110

Hansen, E., & Read, A. F. (2020). Modifying adaptive therapy to enhance competitive suppression. Cancers, 12(12) https://www.mdpi.com/2072-6694/12/12/3556

Melnikov, S. V., Stevens, D. L., Fu, X., Kwok, H. S., Zhang, J.-T., Shen, Y., Sabina, J., Lee, K., Lee, H., & Söll, D. (2020). Exploiting evolutionary trade-offs for posttreatment management of drug-resistant populations. Proceedings of the National Academy of Sciences, 117(30), 17924–17931. https://doi.org/10.1073/pnas.2003132117

Stankova, K., Brown, J. S., Dalton, W. S., & Gatenby, R. A. (2019). Optimizing Cancer Treatment Using Game Theory: A Review. JAMA Oncology, 5(1), 96–103. https://doi.org/10.1001/jamaoncol.2018.3395

Stearns, S. C. (2012). Evolutionary medicine: Its scope, interest and potential. Proceedings of the Royal Society B: Biological Sciences, 279(1746), 4305–4321. https://doi.org/10.1098/rspb.2012.1326. [In particular, p. 4314, “Evolution-based cancer therapies”]

5 A recipe book for simulations with OncoSimulR

The package OncoSimulR (https://github.com/rdiaz02/OncoSimul) is a package for forward population genetic simulation in asexual populations, with special focus on cancer progression. It allows specifying many different models but it is not clear that some scenarios that might be biologically relevant can be simulated with these models.

The objective of this work is to list a series of biologically interesting scenarios and explain how to simulate them. For example: "If you want to simulate large intra-tumor heterogeneity with small population sizes, you have to set mutation rates to values between x and y; this leads to z levels of intra-tumor heterogeneity for population sizes between v and w".

5.1 Steps

1. Understand the functionality available in OncoSimulR to simulate interventions.

2. Clone the github repository of OncoSimulR.

3. Read "the picture guide": Diaz-Uriarte and Johnston, 2024, focusing on open problems and scenarios to simulate. For example "SSWM", "WSSM" (strong selection, weak mutation, and weak selection, strong mutation), high or low intratumor heterogeneity, tunneling or lack thereof, etc, etc.

   Yes, this is a **looooooooong paper**. Read it judiciously: you are looking for "interesting scenarios to simulate", but you do not care about cancer progression models *per se* (of course, you can read that part with attention, but it is NOT required for this work).

4. Find settings of parameters that produce the scenarios you want. Of course, you do not want one simulation, nor two or three. You ideally run, say, 100 or 1000 replicates of each scenario and you'd be able to say things like "In 90% of our replicates there is this much intra-tumor heterogeneity".

5. Oh, of course, you might need to define carefully what the things you simulate or try to see are. For example, what is intratumor heterogeneity? How do we define and compute it?

   The papers by Diaz-Colunga and Diaz-Uriarte, 2021, and Diaz-Uriarte and Vasallo, 2019, define some of these things. You might want to look at them (including their supplementary material), but the key is the stuff they define (e.g., intra-tumor heterogeneity); there is no need to understand or read the paper in full. And this is just a suggestion, not a requirement.

6. And, in fact, you will need to sample from the simulations. OncoSimulR allows you to sample in different ways. Use the one that matters for your scenario (e.g., single cell or whole-tumor, or bulk sequencing).

7. You might want to avoid the frequency-dependence stuff here, unless you have very specific scenarios you want to model, such as "frequency-dependence and low/high intra-tumor heterogeneity with small mutation rates and small population sizes". Saying "OncoSimulR can be used to simulate frequency-dependence" ... is useless, since everybody (everybody who knows what OncoSimulR is) knows that very well already.

8. Some scenarios already have know parameters in the documentation of OncoSimulR itself. Mentioning them is OK, but you are supposed to go beyond what is already available in the documentation.

5.2 References

Diaz-Uriarte, R., & Johnston, I. G. (2024). A picture guide to cancer progression and monotonic accumulation models: Evolutionary assumptions, plausible interpretations, and alternative uses (arXiv:2312.06824). arXiv. https://doi.org/10.48550/arXiv.2312.06824.

OncoSimulR: https://github.com/rdiaz02/OncoSimul

Diaz-Colunga, J., & Diaz-Uriarte, R. (2021). Conditional prediction of consecutive tumor evolution using cancer progression models: What genotype comes next? Plos Computational Biology, 17(12), e1009055. https://doi.org/10.1371/journal.pcbi.1009055

Diaz-Uriarte, R., & Vasallo, C. (2019). Every which way? on predicting tumor evolution using cancer progression models. Plos Computational Biology, 15(8), e1007246. https://doi.org/10.1371/journal.pcbi.1007246