Mini-Project 4: Fermentation, Due Wednesday, March 16

Important note: Make sure you read the exercises carefully, and do all that's requested. (Don't just supply graphs; also respond to any questions posed. And remember to include your Sage code, in those instances where it is requested.)

1. Complete exercise 8(a)(b), page 195. (Do not do 8(c).)

   NOTES:

   • Ignore references to SIRPLOT. Instead, work from the program Yeast.sws, which is available on "The Sage Page" from our course page.

   • To answer exercise 8(a), you should reflect on what you know about logistic growth. If you're uncertain, you can consult the program Yeast.sws. The differential equation requested in exercise 8(a) here can be found in the code, if you know where to look.

   • To do exercise 8(b), simply run the program Yeast.sws, then print out, and include, a copy of your graph. You don't need to do the part of 8(b) that begins "Indicate on your graph approximately when..."

2. Complete exercise 9, pp. 195-196. You don't need to do the parts of 9(c) beginning with "How close does the yeast get to carrying capacity..."

   Don't forget to print out and include a copy of the graph requested in exercise 9(c). On your graph, make sure you label clearly which curve is Y and which is A. It's OK to label the individual curves by hand, though you're welcome to do it using Sage, if you can figure out how.

   To do this exercise, you'll need to modify Yeast.sws. Specifically, you'll need to:

   • Add a line following, and analogous to, the line that reads Y=0.5. Your new line should be of the form A=.... Here, you're specifying the initial amount of alcohol present. See exercise 9(c).

   • Add a line following, and analogous to, the line that reads b=10. Your new line should be of the form c=.... Here, you're specifying the toxicity coefficient c, meaning the rate at which a pound of alcohol kills yeast, per pound of yeast. See exercise 9(b).

   • Add a line following, and analogous to, the line that reads Yvalues=[ ]. Your new line should be of the form Avalues=[ ]. You're creating a new list to store the values of A to be computed by your program.

   • Add a line following, and analogous to, the line that reads Yvalues.append(Y). Your new line should be of the form Avalues.append(A). You're storing the current value of A into the list Avalues.

- Rewrite the line that reads Yprime=k*Y*(1-Y/b). Hint: your new differential equation for Y should involve the original term k*Y*(1-Y/b) and a new term that involves Y , A, and the toxicity coefficient, c. Read problem 9(b) carefully to understand what this new term should look like. (Keep in mind that the effect of this term on Yprime will be negative, since it enacts a decrease in Y.)

- Add a line following, and analogous to, the line that begins Yprime=.... Your new line should be of the form Aprime=.... Here, you're specifying the differential equation for A. To do so, consider the information given in exercise 9(a).

- Add a line following, and analogous to, the line that reads DeltaY=.... Your new line should be of the form DeltaA=....

- Add a line following, and analogous to, the line that reads Y=Y+DeltaY. Your new line should be of the form A=....

- Add a line following, and analogous to, the line that reads

  Yplot=list_plot(list(zip(tvalues,Yvalues)),plotjoined=True,marker='o',color='blue') .

  Your new line should be of the form

  Aplot=list_plot(list(zip(tvalues,Avalues)),plotjoined=True,marker='o',color='red') .

- Replace the last line, which reads

  show(Yplot,axes_labels=['t (hours)','Y (pounds)'])

  with the line

  show(Yplot+Aplot,axes_labels=['t (hours)','Y,A (pounds)'])

- Modify all the comment lines (the lines starting with #) so that they describe what's going on in your NEW program.

Note: you do not need to print out the code that you end up with upon making the above modifications. But again, do include a copy of the graph generated by that code. (You will be asked to supply some code in exercise 3 below.)

3. Complete exercise 11, pp. 196-197. You don't need to do the parts of 11(c) beginning with "Does all the sugar disappear?" That is, to complete exercise 11(c), just supply the relevant graph and code, as described below.

   To do exercise 11, you'll need to make modifications to the code you ended up with at the end of exercise 2 above. These modifications will be quite similar to the modifications you already made, except: whereas your previous modifications added alcohol A into the mix, your new modifications will regard what happens to sugar S. So the code you get, when you're done here, should plot graphs of yeast, alcohol, AND sugar over time.

   In particular, your last line of code should read

   show(Yplot+Aplot+Splot,axes_labels=['t (hours)','Y,A,S (pounds)'])

Remark: Note that, in this, your carrying capacity will depend on S. Perhaps the best way to code this is:

- Change the value of b that you used in the previous problem to b = 0.4;
- In the differential equation that you had for Yprime in the previous two problems, replace the quantity Y/b by Y/(b*S). [Make sure to include the parentheses, or Sage will think you mean (Y/b)*S.]

Also, to get full credit, your new program should have modified comment lines (lines starting with #) to describe the new features. Don't forget to print out and include a copy of your graph, with the quantities Y, A, and S clearly labelled. Also, please do include a copy of the completed code you used to generate this graph.