Framework for Patient Flow Forecast

Microsimulation model

Raymond Chiasson

January 2020

Sections

Patient decision tree

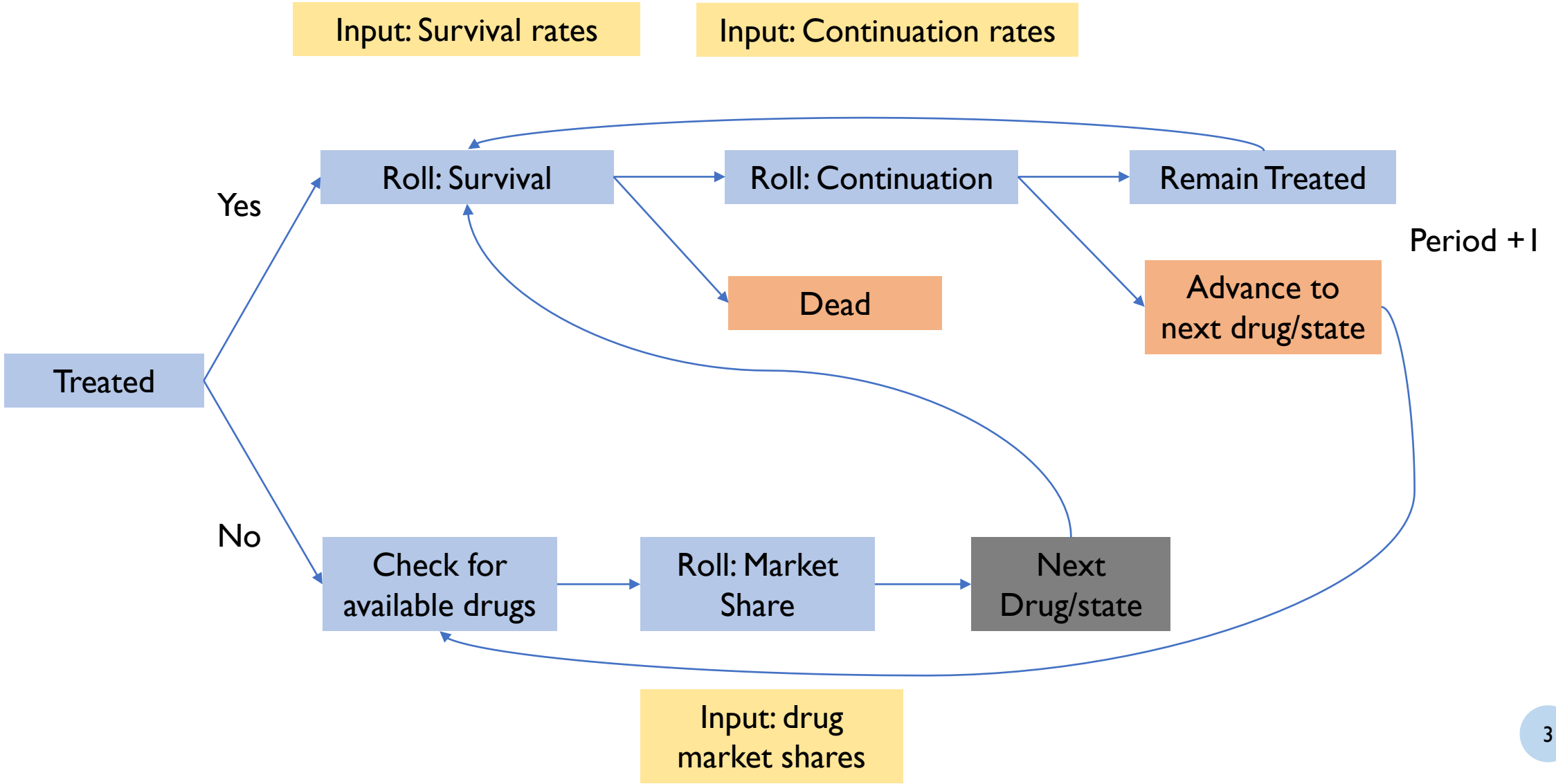
Code documentation

- Classes
- Forecaster inputs
- Simulation code

Sample outcomes

- One patient flow outline
- One patient, one incident period
- Multiple patients, one incident period
- Multiple patients, multiple incident periods

Patient Decision Tree



Code Documentation – Forecaster Inputs

Market Share

- Probability that a patient would choose this drug/state assuming all drugs/states are available
- If a drug is rolled but not available, a different drug is chosen
- Don't worry about assigning market share to impossible scenarios (ex. Drug 1 should ramp up market share all lines post-launch even though there are no available 2L patients until Drug 2 launches)

Line	State	0	1	2	6	9	20
1L	Untreated	100%	100%	90%	10%	10%	10%
1L	Drug 1	0%	0%	10%	90%	60%	45%
1L	Drug 2	0%	0%	0%	0%	30%	45%
1L	Drug 3	0%	0%	0%	0%	0%	0%
2L	Untreated	100%	100%	95%	60%	30%	15%
2L	Drug 1	0%	0%	5%	40%	40%	40%
2L	Drug 2	0%	0%	0%	0%	30%	40%
2L	Drug 3	0%	0%	0%	0%	0%	5%
3L	Untreated	100%	100%	95%	60%	30%	15%
3L	Drug 1	0%	0%	5%	40%	40%	40%
3L	Drug 2	0%	0%	0%	0%	30%	40%
3L	Drug 3	0%	0%	0%	0%	0%	5%
4L	Untreated	100%	100%	95%	60%	30%	15%
4L	Drug 1	0%	0%	0%	0%	0%	0%
4L	Drug 2	0%	0%	0%	0%	0%	0%
4L	Drug 3	0%	0%	0%	0%	0%	0%

Survival Rates

- Probability that a patient would survive this period on given drug/state and line of treatment

Line	State	0	1	2	3	4	5	6	7	8
1L	Untreated	95%	90%	90%	85%	85%	80%	80%	80%	80%
1L	Drug 1	99%	99%	97%	97%	95%	95%	93%	93%	93%
1L	Drug 2	99%	99%	97%	97%	95%	95%	93%	93%	93%
1L	Drug 3	95%	92%	92%	92%	90%	90%	85%	85%	85%
2L	Untreated	90%	90%	85%	85%	80%	80%	75%	75%	65%
2L	Drug 1	99%	97%	95%	95%	93%	93%	91%	91%	
2L	Drug 2	99%	97%	95%	95%	93%	93%	91%	91%	
2L	Drug 3	95%	92%	90%	90%	85%	85%	82%	82%	
3L	Untreated	85%	85%	80%	80%	75%	75%	65%	65%	60%
3L	Drug 1	98%	95%	92%	92%	88%	88%	85%		
3L	Drug 2	98%	95%	92%	92%	88%	88%	85%		
3L	Drug 3	93%	91%	85%	85%	78%	75%	75%		
4L	Untreated	75%	65%	65%	60%	60%	50%	50%	0%	
4L	Drug 1									
4L	Drug 2									
4L	Drug 3									

Continuation Rates

- Probability that a patient would continue through the period on their current drug/state
- Untreated uses only [100%, 0%] so that patients spend one period untreated, then roll a new drug/state in the next period
- Patients therefore remain untreated only when there are no available drugs and are constantly rolling available drugs

Line	State	0	1	2	3	4	5	6	7	8
1L	Untreated	100%	0%							
1L	Drug 1	95%	95%	90%	90%	85%	85%	80%	80%	0%
1L	Drug 2	95%	95%	90%	90%	85%	85%	80%	80%	0%
1L	Drug 3	95%	95%	90%	90%	85%	85%	80%	80%	0%
2L	Untreated	100%	0%							
2L	Drug 1	95%	90%	85%	85%	80%	80%	75%	0%	
2L	Drug 2	95%	90%	85%	85%	80%	80%	75%	0%	
2L	Drug 3	95%	90%	85%	85%	80%	80%	75%	0%	
3L	Untreated	100%	0%							
3L	Drug 1	95%	90%	85%	80%	75%	70%	0%		
3L	Drug 2	95%	90%	85%	80%	75%	70%	0%		
3L	Drug 3	95%	90%	85%	80%	75%	70%	0%		
4L	Untreated	100%	0%							
4L	Drug 1									
4L	Drug 2									
4L	Drug 3									

Code Documentation - Classes

Patient

Function	Key objectives and notes on code
__init__(drug_options, state_options)	• Takes lists of drug and state objects • Sets up lists of drugs, list of states, drugs taken, drugs not taken and current drug. • Sets up an experience counter to track each period survived and continued on each drug/state. • The lists of drugs excludes states using the attribute .is_drug which is the key differentiator between drugs and states • For the experience counter, 'survival' and 'continue' are tracked in separate columns. 'continue count' resets when a patient returns to a previously occupied state. This only matters for states, since drugs are assumed to be taken only once.

.select_drug (current period)	• Selects a new drug or state using these steps:
	  • Builds a list of drugs and states that have launched by the current period
	  • Randomly selects a drug or state according to market share values from each object
	  • If the selected drug is already in the list of drugs taken, a different drug is selected.
	  • If a different drug is not available, a state is selected
	  • Updates the list of drugs taken, drugs not taken, and current drug

.roll_survival()	• Tests if patient survives through the current period. Outcomes: 1 = survival, 0 = death/permanent discontinue • Survival rate is taken from drug object for the current drug • Period is taken from the self.experience_counter dataframe • If the patient survives, the experience counter is incremented for 'survival count'.
.roll_continue()	• Tests if a patient continues on their current treatment. • Outcomes: 1 = continue, 0 = progress to next drug • Continue rate is taken from drug object for the current drug • Period is taken from the self.experience_counter dataframe • If the patient continues, the experience counter is incremented for 'continue count' • If the patient does not continue and the current drug is a state, reset the 'continue count' to 0.

.simulate (start_period, end_period)	• Create two empty results tables, one with each drug/state as an index, one with a double index for each drug/state and each line of treatment.
	• Runs simulation according to the following loop:
	  • Check if patient is on a drug, if not run .select_drug
	  • Roll for survival. Terminate if patient does not survive
	  • Roll to continue. If successful, augment period and record results in table. If unsuccessful, increase line of treatment and clear current drug
	• Returns two dataframes with individual patient results

Drug

Function	Key objectives and notes on code
.__init__(name, market share, continuation_rates, survival_rates, launch_period)	• Market share, continuate rates, and survival rates are all dataframes with a rate for each period and line of treatment. • Name should be a unique string used to reference the object. • Launch period is when the drug becomes available for patients. • Stores all this data as class variables. • .is_drug is set to True to differentiate Drugs from States.
.__repr__	• Handles how the object is referenced in lists and other calls; returns name string

State

Function	Key objectives and notes on code

.__init__(name, market share, continuation_rates, survival_rates, launch_period)	• Identical to Drug class except .is_drug is set to False. Created to allow for non-drug states for patients who are diagnosed before any drugs are available or who have already taken all available drugs. The contemplated state is "untreated".
	• How States are handled differently than Drugs:
	  • States do not appear in the .drugs_taken or .drugs_not_taken lists; patients may return to a state while patients may not take a drug twice
	  • The continuation rate for untreated is 100%, 0% for the first two periods. Patients spend one period in the untreated state, then roll for available drugs the next period. This way they are constantly checking for newly available drugs.
	  • The 'continue count' in the experience counter is reset when a state is discontinued, again to account for the fact that a patient may assume a state more than once.

.__repr__	• Handles how the object is referenced in lists and other calls; returns name string

Code Documentation - Sample Code

Single patient, single period simulation

```
from classes_and_functions import Patient
from drug_and_state_data import untreated, drug_1, drug_2, drug_3
import pandas as pd


# single patient simulation

patient_1 = Patient(drug_options=[drug_1, drug_2, drug_3], state_options=[untreated])

result, result_by_line = patient_1.simulate(start_period=0, end_period=20)
```

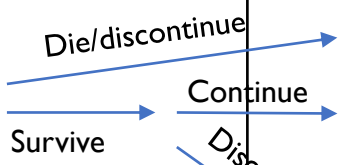
Multiple patient, single period simulation

```
# choose number of patients, start period and end period
n_patients = 100
start_period = 0
end_period = 20

for n in range(n_patients):

    # initialize patient and run simulation
    patient = Patient(drug_options=[drug_1, drug_2, drug_3], state_options=[untreated])
    result, result_by_line = patient.simulate(start_period=start_period, end_period=end_period)

    # add single patient result to results tables
    # surprising that this syntax works - you can simply can add two dfs: df1 + df2
    results += result
    results_by_line += result_by_line
```

Multiple patient, multiple period simulation

```
incident_patients = [100 for x in range(20)] # incident patients in each period
start_period = 0
end_period = 20

for period in range(end_period):

    for n in range(incident_patients[period]):

        # initialize patient and run simulation
        patient = Patient(drug_options=[drug_1, drug_2, drug_3], state_options=[untreated])
        result, result_by_line = patient.simulate(start_period=period,
                                                  end_period=end_period) # use start_period=current_period


        # add single patient result to results tables
        results += result
        results_by_line += result_by_line
```

Sample Outcomes – One Patient Flow Outline

Period	n		n+1
Untreated			
Drug 1	1		0 1
Drug 2			1
Drug 3			

Sample Outcomes – Key Forecaster Inputs

Launch Period	Drug 1	2
	Drug 2	8
	Drug 3	16
Peak Share / Preference	Drug 1	45%
	Drug 2	45%
	Drug 3	5%
Other assumptions	All drugs can be taken in all lines Drugs may only be taken once Order of drugs taken does not matter	

Sample Outcomes – One patient, one incident period

Patient takes drug 1 when it launches in period 2 and survives 7 total periods

Index	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19
untreated	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
drug 1	0	0	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0
drug 2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
drug 3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Results by line of therapy

	Index	Index	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19
0	untreated	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1	untreated	2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
2	untreated	3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
3	untreated	4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
4	drug 1	1	0	0	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0
5	drug 1	2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
6	drug 1	3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
7	drug 1	4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Sample Outcomes – Multiple patients, one incident period

100 incident patients at period 0

Index	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19
untreated	98	86	72	35	12	1	0	1	1	0	0	1	4	2	2	0	0	0	0	1
drug 1	0	0	3	17	27	30	28	27	26	22	17	8	1	0	0	0	0	0	0	0
drug 2	0	0	0	0	0	0	0	0	1	4	7	12	13	14	10	10	7	5	3	0
drug 3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	2	4	6	7

Sample Outcomes – Multiple patients, one incident period

Results by line of therapy

	Index	Index	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19
0	untreated	1	98	86	71	34	10	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1	untreated	2	0	0	1	1	2	1	0	1	1	0	0	1	1	0	0	0	0	0	0	0
2	untreated	3	0	0	0	0	0	0	0	0	0	0	0	0	3	2	2	0	0	0	0	1
3	untreated	4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
4	drug 1	1	0	0	3	17	27	30	28	27	26	22	17	8	1	0	0	0	0	0	0	0
5	drug 1	2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
6	drug 1	3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
7	drug 1	4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
8	drug 2	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
9	drug 2	2	0	0	0	0	0	0	0	0	1	4	7	12	13	14	10	10	7	5	3	0
10	drug 2	3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
11	drug 2	4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
12	drug 3	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
13	drug 3	2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
14	drug 3	3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	2	4	6	7
15	drug 3	4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Sample Outcomes – Multiple patients, multiple incident periods

1000 incident patients in each period

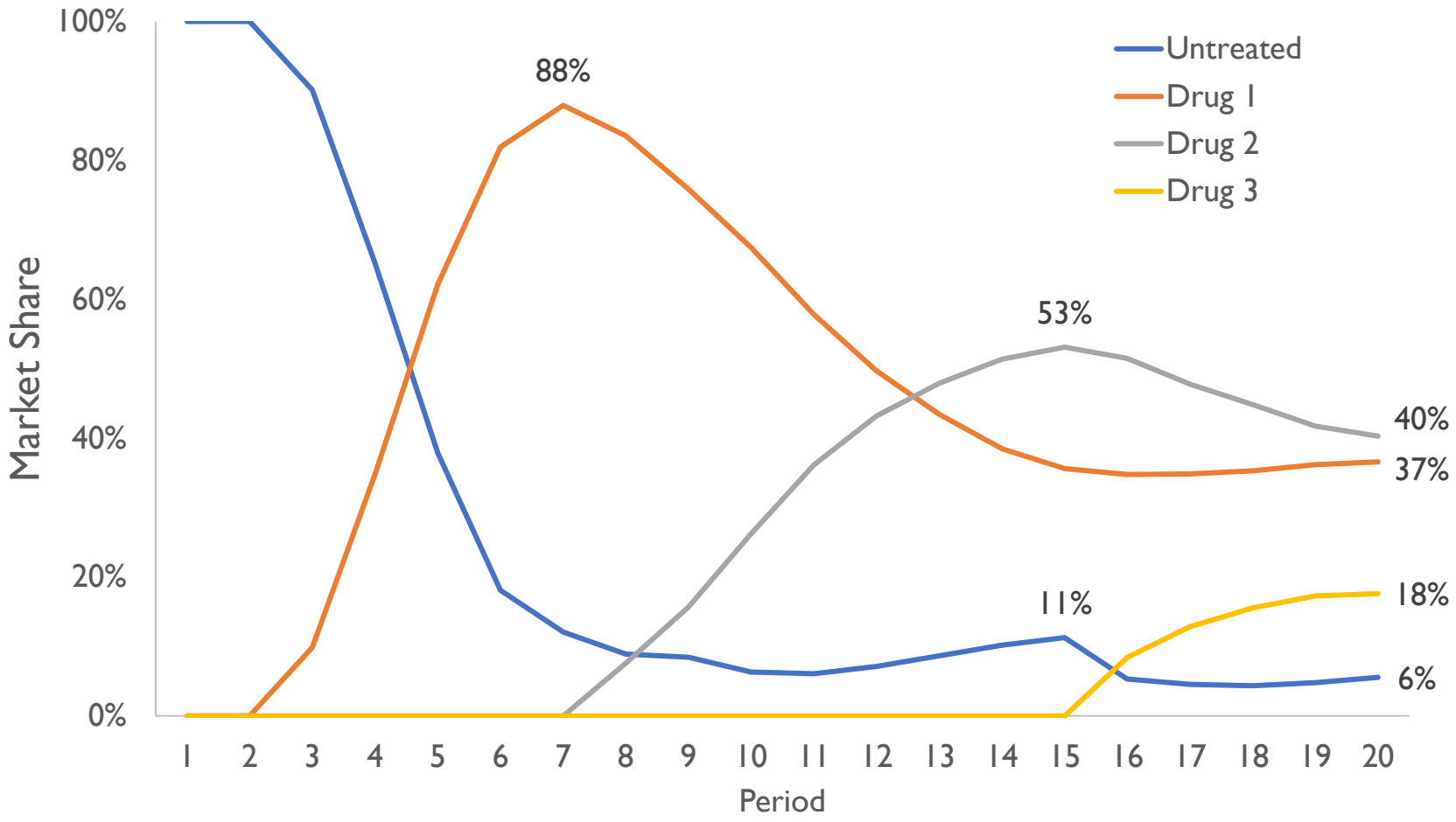
Index	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19
untreated	959	1752	2071	1807	1261	722	564	475	508	418	437	548	705	858	980	477	420	413	465	543
drug 1	0	0	227	964	2069	3269	4108	4462	4550	4469	4164	3824	3532	3244	3094	3119	3237	3361	3507	3580
drug 2	0	0	0	0	0	0	0	406	940	1738	2602	3325	3895	4334	4613	4616	4439	4270	4048	3943
drug 3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	755	1191	1480	1675	1725

Sample Outcomes – Multiple patients, multiple incident periods

Results by line of therapy

	Index	Index	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19
0	untreated	1	959	1752	2061	1760	1124	433	126	108	105	100	106	123	112	117	114	121	97	108	97	110
1	untreated	2	0	0	10	47	137	289	438	353	361	236	159	149	158	189	181	164	157	123	124	111
2	untreated	3	0	0	0	0	0	0	0	14	42	82	172	276	435	552	685	159	106	55	87	83
3	untreated	4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	33	60	127	157	239
4	drug 1	1	0	0	227	964	2069	3269	4108	4462	4541	4436	4090	3661	3288	2867	2589	2443	2391	2358	2377	2373
5	drug 1	2	0	0	0	0	0	0	0	0	9	33	74	163	244	377	505	675	844	1000	1123	1193
6	drug 1	3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	2	3	7	14
7	drug 1	4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
8	drug 2	1	0	0	0	0	0	0	0	85	286	604	1035	1369	1692	1929	2122	2272	2309	2353	2357	2375
9	drug 2	2	0	0	0	0	0	0	0	321	654	1134	1567	1956	2203	2405	2491	2344	2129	1908	1674	1542
10	drug 2	3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	9	17	26
11	drug 2	4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
12	drug 3	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
13	drug 3	2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	14	61	117	154	191
14	drug 3	3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	741	1130	1363	1521	1534
15	drug 3	4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Sample Outcomes – Multiple patients, multiple incident periods



Insights

- Drug 2 overtakes drug 1 despite having equivalent preference. This is driven by the bolus of 2L patients who jump to drug 2 upon launch.
- Drug 3 takes considerable share despite low preference due to lack of other treatment options.

References

The entire project was developed independently and without consulting literature.

After completing this project, I came across these topics when working for Petros Pechlivanoglou.

He and his collaborators have an excellent series of papers on these types of models in R.

- Alarid-Escudero F, Krijkamp EM, Enns EA, Hunink MGM, Pechlivanoglou P, Jalal H. Cohort state-transition models in R: From conceptualization to implementation. arXiv:2001.07824, 2020.
- Krijkamp EM, Alarid-Escudero F, Enns EA, Pechlivanoglou P, Hunink MGM, Jalal H. A Multidimensional Array Representation of State-Transition Model Dynamics. Med Decis Making. 2020 (Online First).
- Alarid-Escudero F, Krijkamp E, Pechlivanoglou P, Jalal H, Kao SYZ, Yang A, Enns EA. A need for change! A coding framework for improving transparency in decision modeling. PharmacoEcon, 2019;37(11):1329–1339.
- Krijkamp E, Alarid-Escudero F, Enns EA, Jalal HJ, Hunink MG, Pechlivanoglou P. Microsimulation Modeling for Health Decision Sciences Using R: A Tutorial. Med Decis Making, 2018;38(3):400-422.
- Jalal HJ, Pechlivanoglou P, Krijkamp E, Alarid-Escudero F, Enns EA, Hunink MG. An overview of R in Health Decision Sciences. Med Decis Making. 2017;37(7): 735-46. Read here.