Modelling the spread of antibiotic resistance

To show that our proposed product would positively benefit the environment where it is proposed to be used, we wrote a computer model of the environment, with and without the product in use, and showed that when it is in use, the model scenario result improved.

In our case, this involved modelling the spread of an antibiotic resistant bacteria in a hospital, with and without our diagnostic tool for quickly identifying it, and showing that the rates of infection, recovery and death improve.

Table of contents

Introduction

Motivation

The purpose of the model is two-fold:

- Demonstrating that our product is beneficial
- Understanding the use cases where it is most and least applicable

Assets

The whole project repository is available on GitHub at: https://github.com/Warwick-iGEM-2021/modelling

The newest model version is also available: Model V3

A interactive toy simulator currently under development for the model is hosted here, but is not fully tested, and may be subject to location change

Model type

Our model is discrete time, stochastic, and compartmental:

- Compartmental means that the model is expressed in terms of the transitions between a set of states. The logic for these transitions forms a fundamental part of the model
- Stochastic means that the model is based on random probabilities, as opposed to a deterministic system of equations
    - A set of constant probabilities define the properties of the model
    - Transitions between states are chosen randomly with these constant probabilities

  Below shows code for a default setting of these probabilities, the meaning of which will be explained further on:

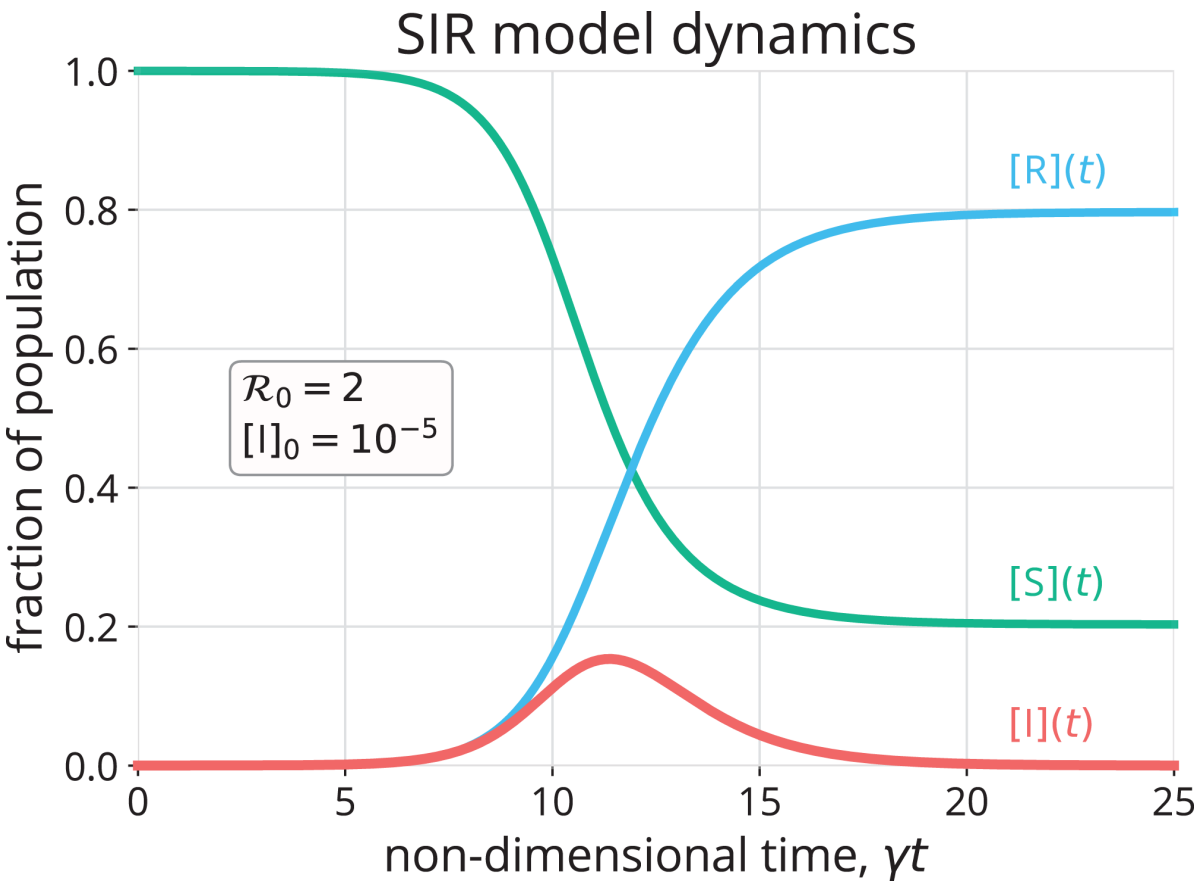
```
# General
NUM_TIMESTEPS = 50
POPULATION_SIZE = 5000
NUM_RESISTANCE_TYPES = 3
# Recovery generally or by treatment
PROBABILITY_GENERAL_RECOVERY = 0.01
PROBABILITY_TREATMENT_RECOVERY = 0.2
# Mutation to higher resistance due to treatment
PROBABILITY_MUTATION = 0.02
PROBABILITY_MOVE_UP_TREATMENT = 0.8
TIMESTEPS_MOVE_UP_LAG_TIME = 5
ISOLATION_THRESHOLD = 2
# Death
PROBABILITY_DEATH = 0.01
# Spreading
PROBABILITY_SPREAD = 1
NUM_SPREAD_TO = 1
# Whether our product is used in the simulation
PRODUCT_IN_USE = True
PROBABILIY_PRODUCT_DETECT = 0.5
PRODUCT_DETECTION_LEVEL = ISOLATION_THRESHOLD
```

- Discrete time means that changes in the model occur at granular timesteps - like turns in a boards game

  Below shows the code for how operations are performed on every person in the population each timestep, and data about them recorded

```
# Make a new data handler for each simulation
self.data_handler.__init__()

# Repeat the simulation for a set number of timesteps
for _ in range(NUM_TIMESTEPS):

    # For each person in the population
    for person in self.population:

        # Record the data throughout the model
        self.data_handler.record_person(person)

```

The model essentially is a modification of the standard SIR model for epidemic disease, adding more “compartments” for additional states people can take, when they are infected with increasingly antibiotic resistant pathogens.



A diagram of the SIR model. Image source: [1]

Implementation

The key features of the model can be split up into five semi-distinct sections:

1. Pathogen and people

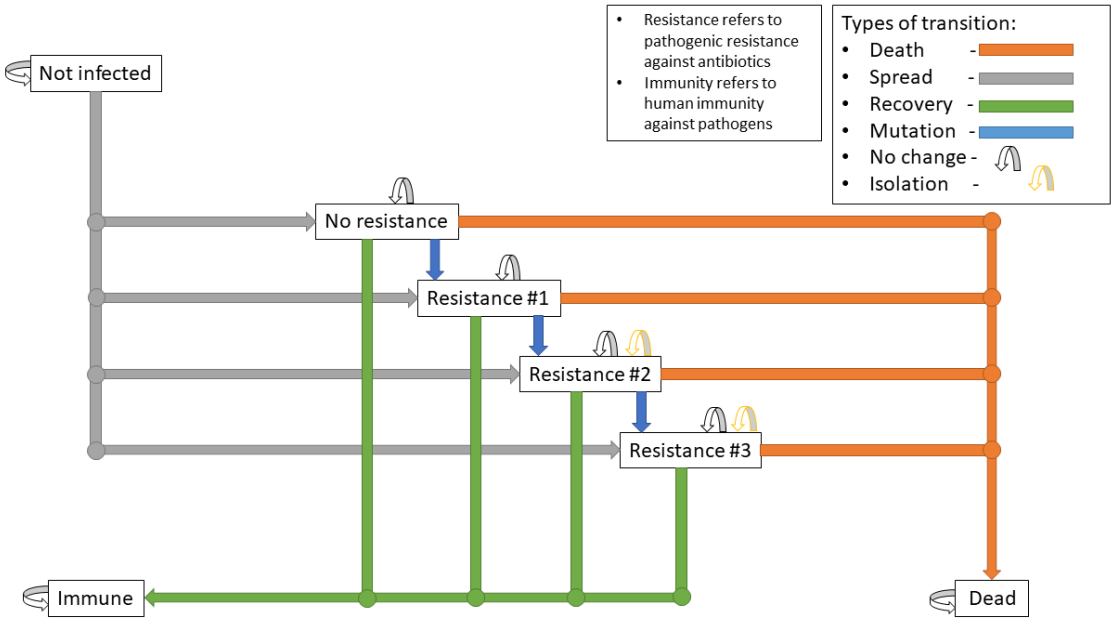
A pathogen with a probability of death and a probability of recovery spreads through the population.

- Patients have a small chance of recovering by themselves, or can be treated with antibiotics, which have a larger chance of curing them
- Different strains of the pathogen exist, which are resistant to different antibiotics
- Pathogens can mutate to more resistant strains in specific circumstances explained in the mutation section
- When they have recovered, they become immune to the all strains of the pathogen irrespective to their resistances
- Patients also have a small chance of dying due to the pathogen

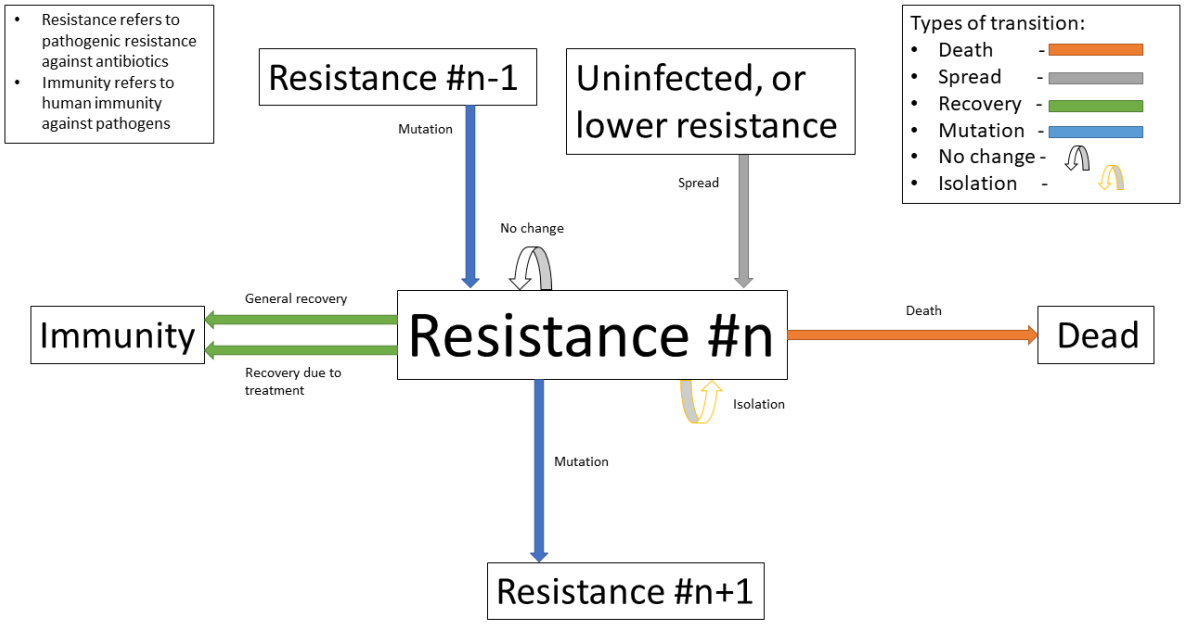
Hence, patients can be in any of the disjoint states: uninfected, infected (possibly with resistance), immune, or dead.

In the limit of time to infinity, all individuals will be either uninfected, immune or dead, as they will all either not be infected in the first place, or recover or die from the pathogen.

Below shows the state transition diagram of every state a person within the population can take (for reasons discussed later in the treatment section, pathogenic resistances to antibiotics will occur in a set order):



Below shows a state transition diagram of a person centred around the state of being infected with a pathogen resistant to antibiotic n in the precedence of antibiotics:



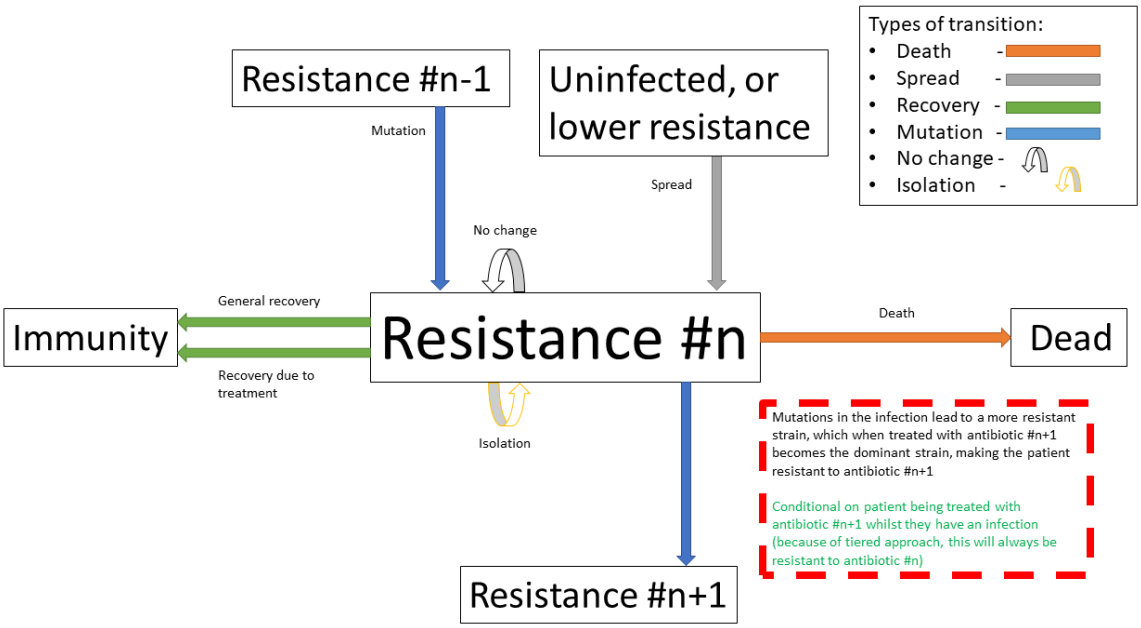
2. Treatment and mutation

Antibiotics are used in a specific order, which are numbered accordingly for clarity (with 1 being the first administered, and n being the last for antibiotics $1..n$). This is to simulate the real-world, where different antibiotics are used in a tiered system, reserving the last for highly dangerous, multi-drug resistant pathogens - and is an important aspect of our model, as our product attempts to identify CRE, which are a type of these resistant pathogens.

Pathogens have a small chance of mutating to develop resistance to antibiotics being used to treat them, as such strains will only become dominant when there is a pressure giving them a survival advantage.

```
# Mutation to higher resistance due to treatment
if decision(PROBABILITY_MUTATION):
    person.mutate_infection()
```

Below shows the same specified diagram used above, with additional information about the mutation step to elucidate it:



The pathogen is modelled as being immediately symptomatic, meaning doctors can immediately identify a patient is infected with it, but they cannot quickly identify whether or not they have a resistant strain if our product is not in use.

Once a person becomes infected, treatment with the lowest tier of antibiotics becomes immediately, as they are immediately symptomatic.

If the pathogen is resistant to the antibiotic, the patient still has the opportunity to make a recovery on their own, but the antibiotic will have no effect, whereas if the pathogen is not, the patient has the opportunity to recover both on their own, and via the antibiotic - increasing their likelihood of recovery each timestep.

Since multiple antibiotics are used in a tiered system, there must be a mechanism to move to a higher antibiotic.

There are a number of days which can be set as a parameter for the model, before which the same antibiotic will be used, then after this is exceeded a probability parameter is used each day to decide whether they will me moved up to a higher treatment tier.

With our product, since it provides a fast testing mechanism for highly resistant strains, patients can be detected as having the resistant strain, and immediately moved up to the required higher treatment

```
# Move up in treatment class if needed
if person.treatment is None:
    # If the person is infected but are not being treated
    # with **anything**, start them on the lowest tier
    # treatment (we can know that the person is infected,
    # but not which tier they are on, without diagnostic
```

```
    # tools, as we can see they are sick)
    person.treatment = Treatment()
else:
    # If the person has been treated for a number of
    # consecutive days with the, a certain probability is
    # exceeded, move them up a treatment tier
    time_cond = person.treatment.time_treated > TIMESTEPS_MOVE_UP_LAG_TIME
    rand_cond = decision(PROBABILITY_MOVE_UP_TREATMENT)
    if time_cond and rand_cond:
        person.increase_treatment()

    if PRODUCT_IN_USE and decision(PROBABILIY_PRODUCT_DETECT):
        # If the product is in use, and it detects the
        # infection (which occurs a certain probability of
        # the time) immediately isolate this with the
        # resistance


        # If the person is known to have a resistance that
        # is higher than their treatment, increase their
        # treatment
        if person.treatment.drug < str(PRODUCT_DETECTION_LEVEL):
            person.treatment.drug = str(PRODUCT_DETECTION_LEVEL)
```

3. Spread

Disease can spread from infected patients to uninfected patients, and patients with a less resistant strain. The likelihood of this occurring, and the number of people spread to each time can be controlled as parameters

```
# Spread the infection strains throughout the population
# We need a deepcopy operation, to prevent someone who has just
# been spread to in this timestep spreading the thing they've
# just received, so technically don't have yet
updated_population = deepcopy(self.population)
for person in self.population:
    if person.infection is not None and decision(PROBABILITY_SPREAD):
        for receiver in sample(updated_population, NUM_SPREAD_TO):
            person.spread_infection(receiver)
self.population = updated_population[:]
```

4. Isolation

Patients can be put into isolation, preventing the spreading the disease. This is the main place where the our product differentiates itself.

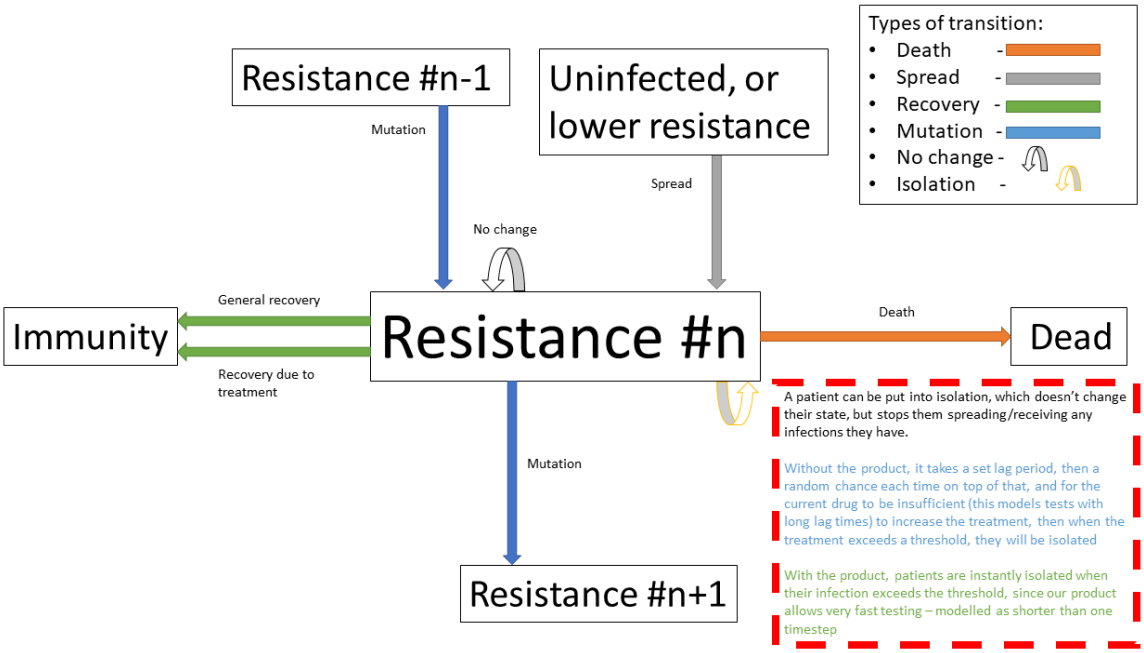
Without our product, a person is put in isolation when they exceed a threshold of **treatment**

With our product, as the pathogen can be detected, they are put into isolation when they exceed a threshold of **having the resistant strain**

```
if PRODUCT_IN_USE and decision(PROBABILIY_PRODUCT_DETECT):
    # If the product is in use, and it detects the
    # infection (which occurs a certain probability of
    # the time) immediately isolate this with the
    # resistance

```

```
    if person.infection.resistances[str(PRODUCT_DETECTION_LEVEL)]:
        person.isolate()

if int(person.treatment.drug) >= ISOLATION_THRESHOLD:
    # Isolate if in high enough treatment class (which
    # is not the same as infection class - this will
    # likely lag behind)
    person.isolate()

# Increment the number of times a person has been
# treated with the drug
person.treatment.time_treated += 1
```

Below shows the same specified diagram used above, with additional information about the isolation step to elucidate it:



5. Recovery and death

As discussed in section (1), each timestep, patients can recover (either naturally or via treatment), and patients can die.

Recovery makes the patients immune, meaning they cannot be infected again, essentially removing them from the system. Death also essentially removes patients from the system, as there cannot be any more state changes after death.

```
# Recovery generally or by treatment if currently infected
general_recovery = decision(PROBABILITY_GENERAL_RECOVERY)
treatment_recovery = (person.correct_treatment() and
                      decision(PROBABILITY_TREATMENT_RECOVERY))
if general_recovery or treatment_recovery:
    person.recover_from_infection()

# Deaths due to infection
if decision(PROBABILITY_DEATH):
    person.die()
```

The goal is to create a situation where in the limit of time, the number of uninfected and immune people is maximised, and the number of dead people is minimised.

Context

Due to the flexibility of the model, its parameters can be adjusted to simulate the spread of many real-world diseases. Adding such context to the model helps us better understand better how our product could improve the situation in such scenarios. Here we have chosen to use Neo-natal Bacterial Meningitis as an example. The disease can easily be spread within hospitals by medical staff and often has a deadly outcome [2], all of which can be simulated in the model. Furthermore, since the last line of treatment of meropenem, a carbapenem, it is relevant to the use of our product.

The parameters of the model have hence been adjusted because:

1. NBM has two lines of treatment (amoxicillin + cefotaxime/ceftriaxone, then meropenem) [3], the model has two levels of treatment and corresponding resistance levels.
2. There is a 100% mortality rate of untreated NBM [4], there is no chance of recovery if the pathogen is resistant against the current antibiotic in use.
3. There is 40% overall mortality [4] , parameters have been adjusted to end up with a 40% mortality rate

Discussion of the model

Some common questions about the model are answered below:

- Q. Is the model realistic

  A. No, very little about it is realistic. It is an abstraction of the real world which discards many unnecessary complexities, in order to simply and efficiently model how resistance spreads and is combatted. It is not viable to make a wholly realistic model, as this inevitable turns into a “hospital simulator”, and would be too complex to design, and take too long to run on current computers.

- Q. Is the model useful

  A. Yes, because it provides several helpful insights:

  - The impact our product will have on the spread of resistance just by quickly detecting who to put into isolation
  - Whether higher or lower mortality or transmissibility of a disease increase or decrease the effectiveness

- Q. What potential improvements are there

  A. It would be possible to add additional features to the model to make it more realistic, for example:

  - Spatial considerations – e.g. modelling multiple wards with movement between them
  - Asymptomatic transmission periods of infection

  however, these are beyond the scope of our project

- How does the model compare to other existent ones

- Q. Can the model be applied to current issues, i.e. the COVID pandemic

  A. Since the model is a very generic abstraction of the real world, by adjusting it’s parameters, a vast amount of different scenarios can be modelled. The key issue in adapting it to different scenarios is if they fit the inherent logic and states hard-coded into it. Since COVID is a viral infection, as opposed to a bacterial infection, antibiotics cannot be used to treat it, so the tiered system of antibiotic uses fits less cleanly to it, however, they could

instead be considered as increasingly aggressive treatment options, to which it also grows resistant. However, the logic around our product would not apply, as viral infections are not affected by carbapenem, which is the antibiotic we focus on.

References

[1] Simon, Cory M., 2020. *The SIR dynamic model of infectious disease transmission and its analogy with chemical kinetics*. Available at https://peerj.com/articles/pchem-14/ [Accessed 27 September 2021]. DOI: 10.7717/peerj-pchem.14

[2] Şah İpek, M., 2019. *Neonatal Bacterial Meningitis*. [online] IntechOpen. Available at: https://www.intechopen.com/chapters/68042. DOI: 10.5772/intechopen.87118

[3] Meningitis Research Foundation, 2017. *Management of Bacterial Meningitis in infants <3 months*. Available at: https://www.meningitis.org/getmedia/75ce0638-a815-4154-b504-b18c462320c8/Neo-Natal-Algorithm-Nov-2017 [pdf]

[4] Tesini, B., 2020. *Neonatal Bacterial Meningitis*. [online] MSD Manual Professional Edition. Available at: https://www.msdmanuals.com/en-gb/professional/pediatrics/infections-in-neonates/neonatal-bacterial-meningitis [Accessed 23 September 2021].