Summer Project

Thomas Blake supervised by Dr Aden Forrow

August 2021

1 Introduction

Improvements in single-cell RNA sequencing [4] has allowed us to use RNA abundance as an indication of cell state. By counting and classifying RNA molecules in individual cells, biologists can precisely characterise distinct cell states within any tissue using statistical methods described in Luecken and Theis 2019 [2]. These methods only give us information about a cell at a snapshot in time as we need to break apart the cell to sequence the RNA. By using the ratio of spliced and unspliced RNA we can calculate an RNA velocity [1] which can be seen as the "time-derivative of the gene expression state". RNA timestamping [3] gives us an insight into the time of creation of each RNA molecule which can be used to estimate the previous transcriptional activity. In this paper we look to create a mathematical model of RNA creation and accumulation of edits. At first we use a simplified model using the bulk sequencing data which can not give an insight into a specific cell activity but should be insightful in the modelling of editing rates. This model also does not include any degradation as we concentrate purely on normalised "editing distributions". In our more developed model we model absolute counts and stochastic cell degradation. Due to the currently limited "volume" of data on timestamping, we see if our experiment can accurately be used to simulate more complicated experiments. Finally we see if our model can be used to make a prediction of RNA velocity using the timestamps rather than using ratio of spliced to unspliced RNA.

2 Biological Background

2.1 RNA velocity

In the RNA velocity paper [1], the velocity is defined as the rate of change of spliced mRNA abundance. The model for the spliced and unspliced mRNA is given by:

$$\begin{aligned} \dot{u} &= \alpha(t) - \beta(t)u(t) \\ \dot{s} &= \beta(t)u(t) - \gamma s(t) \end{aligned} \tag{1}$$

where u and s is the expected abundance of unspliced and spliced mRNA respectively. $\alpha(t)$ is the rate of production, $\beta(t)$ is the rate of splicing and γ is the rate of degradation. We will see in the next section how this choice of equation inspires the simplified model for editing counts. They set the splicing rate $\beta(t) = 1$ and assume a constant number of unspliced molecules to simplify the model and give the single rate equation:

$$\dot{s} = u - \gamma s(t) \tag{2}$$

this therefore gives us a much simpler formulation of RNA velocity. Under the assumption that some of the single cell readings are at the steady state, we can use the phase plane to make an estimate of γ (we will use a similar method in section 4.3 to estimate degradation rate). It is assumed that the equations hold independently for all genes such that each gene has different rate constants (the same assumption is made for our models). Due to variability in cell size and "global variation of splicing efficiency and detection of unspliced molecules" the spliced and unspliced counts must be normalised separately. This is done by letting $u = U/N_u$ and $s = S/N_s$ where U and S are absolute number of unspliced and spliced counts respectively, and N_u and N_s are the total number of unspliced and spliced counts in a cell. We attempt to normalise our own definition of RNA velocity in a similar way.

2.2 Timestamping

RNA timestamping gives us an estimate an indication as to how old each RNA molecule is. In the experiments, RNA molecules are tagged with a reporter motif. A enzyme called ADAR is added to the cells leading to the gradual accumulation of A-to-I edits on the timestamps. Counting the number of these edits on the timestamps allows us to infer the age of the RNA molecule on the timescale of hours. The data used in this paper come from two distinct experiments in the timestamping paper [3].

Firstly, a group of cells were left in a medium containing deoxycycline (an RNA transcription promoter) for 1 hour. A transcription inhibiter was added, the cells left to accumulate edits for a variable amount of time before the cells were lysed and timestamps sequenced. The amount of time the cells were left was 1,2,...,12 hrs (so we had 12 different readings). Bulk sequencing was carried out on the cells (rather than single cell) to produce a normalised “editing distribution” which gives us the distribution of different edits at this time point. The “Simplified Model” is trained using this data.

The second experiment is carried out very similarly but each cell is sequenced separately using single-cell sequencing which allows us to look at the behaviour of particular cells and to calculate RNA velocity. Additionally, in the second experiment the cells were only sequenced at the 1, 2 and 4 hour time points.

3 Simplified model

First we tried a simple model for the editing rates. This model is inspired by the model of the spliced and unspliced molecules in equation (1). Suppose a_i is the number of RNA molecules with i edits and $\mathbf{k} = (k_0, k_1, \cdots, k_{40})$ be the editing rates. Then we assume a linear system of ODEs given by:

$$\begin{aligned} \dot{a}_0 &= p(t) - k_0 a_0 \\ \dot{a}_1 &= k_0 a_0 - k_1 a_1 \\ &\cdots \\ \dot{a}_{40} &= k_{39} a_{39} - k_{40} a_{40} \end{aligned} \tag{3}$$

where $p(t)$ refers to the production rate at time t. The model is “simplified” as we do not account for degradation of the RNA molecules which we know must occur. These equations are used to model the first experiment of the timestamping paper (as described in 2.2) where the data is from bulk sequencing and the observed distributions are normalised. By “normalised” we mean that each observation of edit counts is divided by the number of RNA observed so that the sum of all counts is 1. By assuming the initial production occurs at a constant rate, we can model the experiment as $p(t) = AH(1 - t)$ where H is the Heaviside function and A is the constant amplitude of production and by asserting that initially there is no RNA.

3.1 Fitting parameters

Let $a(t) = (a_0(t), a_1(t), \cdots, a_{40}(t))$ be the solution to (1) with initial condition $a(0) = \mathbf{0}$. Using the data provided we look to set up an optimisation problem to find the parameters k_i and A that fit the experimental data. As we are only provided the data for the normalised editing counts it becomes very hard to estimate the amplitude of production A. Therefore to simplify the optimisation problem we set $A = 1$ under the condition that for this model we are only concerned with the normalised distributions. We verify this simplification in the “verifying amplitude simplification” subsection.

To fit the parameters k_i we used ODE45 and used MATLAB global optimisation toolbox with the following loss function:

$$Loss(k) = \sum_{i=1}^{12} \left\| \frac{\tilde{a}}{\|\tilde{a}(i)\|_1} - D[i] \right\|_2^2 \tag{4}$$

where D[i] is the observed editing distribution at time i and $\tilde{a} = (a_1(t), a_2(t), \cdots, a_{40}(t))$. In the experiments they do not make observations of RNA molecules with no edits so we exclude a_0 from $\tilde{a}$. It is worth noting that, although we do not include the 0 edit state in our loss function we still need to solve for it to find the other editing states. Originally, we attempted to use "dsolve" which is an analytical solver to solve the set of equations (1) which gives us an analytical representation of loss with algebraic parameters k_i. The advantage of this over ODE45 is that we can run gradient descent to find the optimal parameters rather than resorting to the less efficient local solver below. However, when working with more than 7 different parameters k_i this became far too computationally expensive (in our case there were 40 parameters to track).

To set up the ODE we created a separate file `dstate.m` which took the input C which is a 40×1 array where $C = [k_0, k_1, \cdots k_{40}]$.

dstate.m

```
function dydt = dstate(t,y,C)
c = diag(C(1:39));
d=zeros(40);
d(2:40,1:39)=c;
A = d − diag(C);
B = zeros(40,1);
B(1,1)= heaviside(1−t);
dydt=A*y+B;
```

loss.m

```
function l=loss(C, BasisVectors)
[~,y] = ode45(@(t,Y) dstate(t,Y,C),[0 1 2 3 4 5 6 7 8 9 10 11 12], zeros(40,1));
% exclude the 0 edits and the data at time 0
modelledDist = y(2:13,:).';
modelledDist2 = zeros(39,12);
% We normalise the data in each timeframe
for i=1:12
   modelledDist2(:,i)=modelledDist(2:40,i)/sum(modelledDist(2:40,i));
end

l = norm(BasisVectors−modelledDist2);
```

computationalSolver.m

```
BasisVectors = table2array(readtable('BasisVectors.csv'));
BasisVectors = BasisVectors(2:40,:);

C = ones(40,1);

A = [];
b = [];
Aeq = [];
beq = [];
ub = 1000*ones(1,40);
lb = zeros(1,40);
nonlcon =[];

gs = GlobalSearch('Display','iter');
lossFunc = @(c) loss(c,BasisVectors);
problem = createOptimProblem('fmincon','x0',C,'objective',lossFunc,'lb',lb,'ub',ub);
problem.options.Display = 'final';
problem.options.MaxFunctionEvaluations = 3.000000e+04;
x = run(gs,problem);
```

The file `loss.m` implements the loss function in (2) into a MATLAB function while `computationalSolver.m` looks to find the parameters k_i which minimise the loss function. computationalSolver.m uses the GlobalSearch function from the Global Optimisation Toolbox which repeatedly runs a local solver. In this case the default "interior-point" local solver is used. The "MaxFunctionEvaluations" was set to be large so that the local solver stops when the iteration steps become small and to prevent it from stopping early. We have to work with a bounded optimisation problem as we need all our constants k_i to be larger than 0.

Using these files we found a set of parameters that fit the model well for 2 hours timepoint onwards. This did not fit the model well for the first hour as shown in Figure 1. This is because the first 12 edits all occur at a much shorter timescale. Moving forward we therefore decided to merge the first 12 edit states into one state.

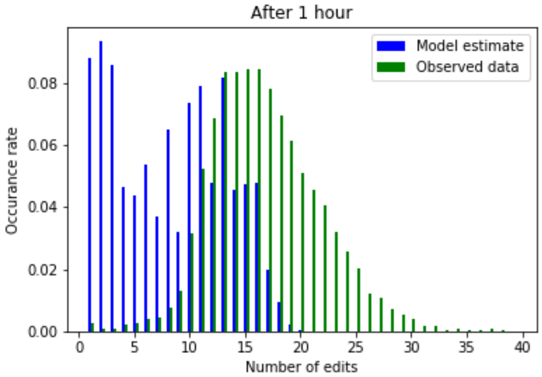


Figure 1: Before merging first 12 edit states

3.2 Verifying Amplitude simplification

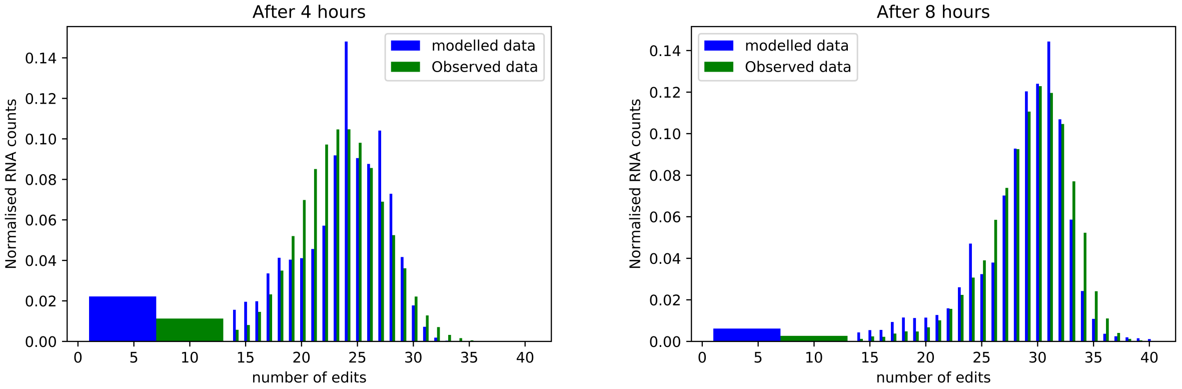
Using the parameters k_i in the previous section we ran the computational ODE solver for different values of the amplitude of production, A. The average percentage change (APC) when moving from $A = 1$ to $A = 0.1$ is 0.72% and moving from $A = 1$ to $A = 10$ the APC is 0.27%. The APC is calculated by finding the average across all edits and timepoints of the absolute difference between the modelled normalised counts with different amplitudes divided by the observed normalised count when A=1. As changing the amplitude by a factor of 10 does not seem to affect the normalised edit counts for our parameters we have justified the arbitrary choice of $A = 1$ in the previous section. This is intuitive as all RNA molecules are conserved (there is no degradation) and we are dividing by total number of RNA molecules so changing the number of RNA produced should not change the expected normalised distribution. It is important to note that the predictions of this model must always be normalised (which makes sense as we have trained using normalised data).

3.3 Testing Goodness of fit

As described in subsection 3.1 we decided to merge the first 12 edit states. Rather than use all the data to train $\mathbf{k}$, we left out hour 4 and hour 8 which we will use as our test set (see Figure 2). After training (using approach described above) to get our editing parameters, the average test error is 0.00263. This is the average squared difference between predictive and observed distribution.

We seem to have a good fit in the test data so we will proceed with this model. A lot of the inaccuracy again comes from the first 12 edits, even though the model is greatly improved by merging the first 12 edit states. This is because in the experiment RNA molecules were not counted unless they had an edit on each read. (See "Alignment and edit counting" section of timestamping paper [3]).

We then retrained using all 12 editing distributions to get a new set of parameters.



(a) test set 1

(b) test set 2

Figure 2: Test prediction vs observation

3.4 Simulations

Using the simple model above we were able to simulate the behaviour of the first experiment in the Timestamping paper [3] . We multiplied the normalised distribution by 10^5 and rounded the numbers to get an absolute count of RNA molecules as our initial absolute distribution. We assume each RNA molecule is created at a time uniformly distributed between 0 and 1 hour as the doxycycline promoter caused RNA production for the first hour. If an RNA molecule has i edits at time t, then after i.i.d random variable with $\mathrm{Exp}(k_i)$ distribution amount of time the molecule has $i+1$ edits. We drew from the exponential distribution by first drawing an i.i.d random variable X from a $U[0,1]$ distribution.

$$T = \frac{1}{k_i}\log(1/X)$$

T has a $\mathrm{Exp}(k_i)$ distribution. At each time point we then normalised the distribution so that we could compare with our observations. As shown in Figure 3, these simulations seem to match the ODE model and the observed data well. These simulations can only simulate experiments where there is constant production the first hour and then no production after (where a production inhibitor has been added).

3.5 Simulating new experiments

Using the simulation object in the Python code, we simulated two new experiments. These simulations have RNA production for the first hour but we have freedom over the initial distribution of edits for the RNA produced. Firstly, we used a Poisson distribution with mean 4 as the initial distribution. If $p(i)$ is the probability of a random RNA molecule having i edits then $p(i) = \frac{e^{-4}4^i}{i!}$. Then we sampled the simulation after 4 hours and after 8 hours.

Secondly, we used an exponential distribution but only gave edits non-zero probability if the edit count was a multiple of 3 (as shown in figure 4). We then sampled the simulation after 1 hour and 2 hours.

3.6 Reasons for error

A lot of the error in fitting the the parameters came from the fact that we do not have a reading for the 0 edits columns - it would be useful to find a way to get an experimental method to find this.

The model could also be greatly improved by only using the bases mentioned in the Supplementary Figure 3 of the Timestamping paper [3]. These bases have editing times which are closest to being exponential, so I would expect our model to fit these bases much better.

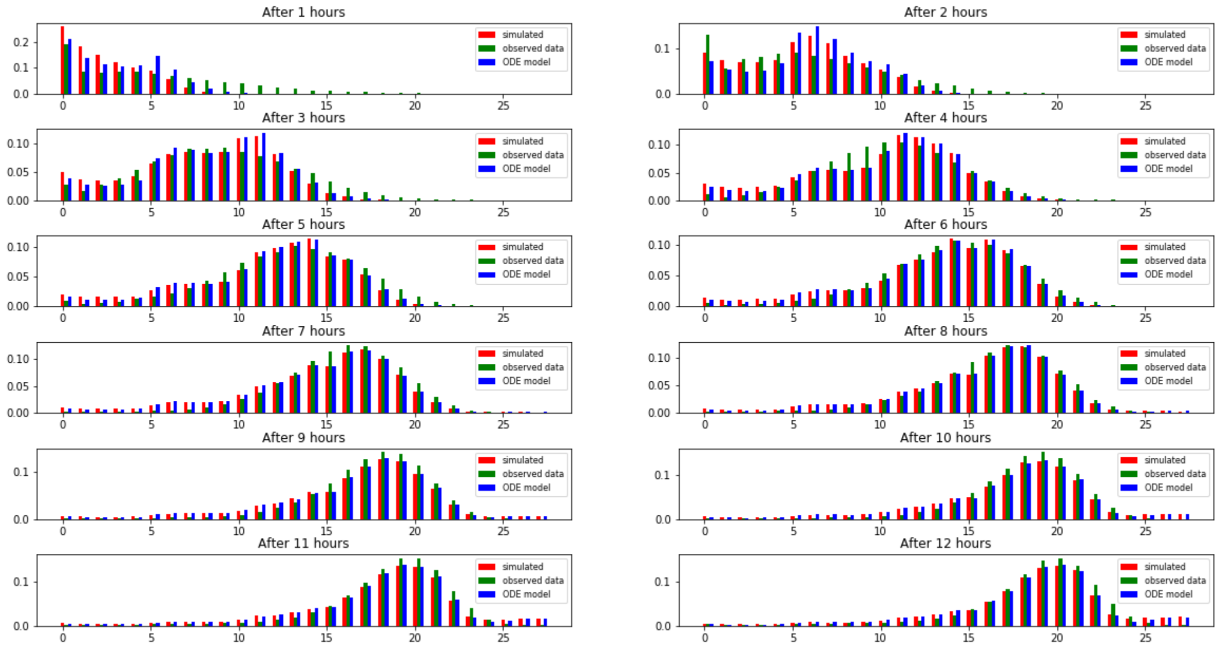
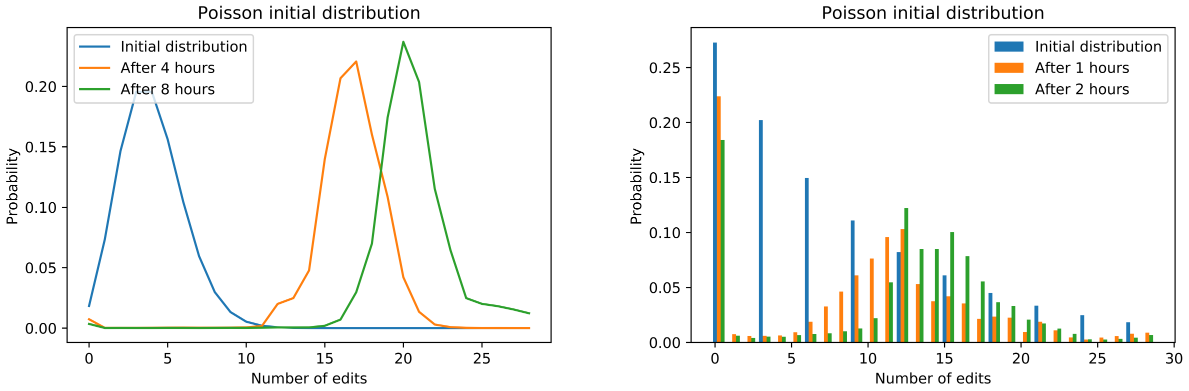


Figure 3: For ease of viewing the 0 column now shows the 1-12 edits observation, the 1 column shows the 13 edits column and so on...



(a) Experiment 1

(b) Experiment 2

Figure 4: Simulating new expermients

3.7 Model Conclusions

The model seems to be a good fit for the first experiment in the timestamping paper and allows us to make accurate simulations of experiments which are carried out. However, there a few clear issues with the model.

We have made an assumption on the production of the RNA (we assume RNA is produced at a constant rate) when we do not need to for determining k_i. Even though our assumption does seem intuitive, we could treat the first observation as our initial condition and use the fact that we know no RNA is produced after the first observation (as an inhibitor is added) to more accurately determine the constants k_i (as shown in the next model).

All the data in this experiment is from bulk sequencing data. Even though we expect edits to accumulate at the same rate for single-cell experiments, the next model is trained specifically on single-cell data. The single-cell data then allows us to make predictions on RNA velocity for each cell which is not possible for bulk sequencing data. Crucially, it also allows us to make simulations for individual cells.

Finally, the simplified model does not include a degradation rate for the RNA molecules. Naturally, the RNA molecules must degrade and the rate of degradation is something that will prove crucial to our definition of RNA velocity. Therefore it is important to find a model which includes this feature.

4 More developed model

There are two key changes in the more developed model. The first being that the a_i now represents expected absolute number of RNA molecules with i edits rather than concentrating on the relative edits. This means that we can no longer choose the rate of production as 1 arbitrarily. Secondly, we introduce the degradation of the RNA molecules at a rate c.

$$\begin{aligned}
\dot{a}_0 &= p(t) - k_0 a_0 - c a_0 \\
\dot{a}_1 &= k_0 a_0 - k_1 a_1 - c a_1 \\
&\dots \\
\dot{a}_{24} &= k_{23} a_{23} - k_{24} a_{24} - c a_{24} \\
\dot{a}_{25} &= k_{24} a_{24} - c a_{25}
\end{aligned} \tag{5}$$

The single cell experiments only concentrate on one of the reads so in these experiments we only have 25 editing bases. Note also that there is no $-k_{25}a_{25}$ term. This means that the columns in the matrix K (below) sum to 0 which makes the parameters $\mathbf{k}$ much easier to solve for (as shown in the next section).

$$K = \begin{bmatrix}
-k_0 & 0 & & \dots & \dots & 0 \\
k_0 & -k_1 & 0 & \dots & \dots & 0 \\
0 & k_1 & -k_2 & \dots & \dots & 0 \\
\vdots & \ddots & & & \dots & \\
0 & \dots & \dots & k_{23} & -k_{24} & 0 \\
0 & \dots & \dots & 0 & k_{24} & 0
\end{bmatrix} \tag{6}$$

4.1 Finding the parameters k

In this subsection we look to try and fit the parameters k_i in the model. The experiments for single-cell analysis are set up so that a promoter is added for an hour then a transcription inhibitor is added. Then we get a reading of absolute counts for single cells 1,2 and 4 hours after the promoter is added (which we will call "one hour", "two hour" and "four hour" data respectively). After the inhibitor is added we can assume there is no production ($p(t) \equiv 0$). So our problem becomes:

$$\dot{a} = Ka - ca = (K - cI)a \tag{7}$$

where I is the identity matrix. This has solution $a(t) = e^{(K-cI)t}a(0) = e^{-ct}e^{Kt}a(0)$ where $a(0)$ is the initial condition for the ODE. As shown from equation (8) and (9), the 1-norm of the solution (total number of RNA molecules for that gene) is independent of K:

$$\|a\|_1 = \|e^{-ct}e^{Kt}a(0)\|_1 = e^{-ct}\|e^{Kt}a(0)\|_1 = e^{-ct}\|a(0)\|_1 \tag{8}$$

This is a result of editing counts being conserved (we didn't have a $-k_{25}$ term in the matrix K). While this assumption might seem slightly naive as for a large production amplitude and a long timescale we will have counts accumulating in the 25th state, on the shorter timescale we are looking at this should not cause an issue.

If a vector v has only positive entries then $\|v\|_1 = \mathbf{1}^\top v$, where $\mathbf{1}$ is the vector of all ones. Assuming

$e^{Kt}a(0)$ and $a(0)$ have only positive entries (which can be argued separately), then

$$\|e^{Kt}a(0)\|_1 = \mathbf{1}^\top e^{Kt}a(0) = \mathbf{1}^\top a(0) = \|a(0)\|_1 \tag{9}$$

As $\mathbf{1}^\top K = \mathbf{0}$ (as the columns of K sum to zero), then $\mathbf{1}^\top e^{Kt} = \mathbf{1}^\top$.
Now consider a normalized version $b = a/\|a\|_1$. Then

$$b(t) = \frac{e^{-ct}e^{Kt}a(0)}{e^{-ct}\|a(0)\|_1} = e^{Kt}\frac{a_0}{\|a(0)\|_1} \text{ solves } \dot{b} = Kb$$

We will use this fact to find the parameters k that best fit our data. As the ODE above requires no production of RNA, we will treat the normalised `one hour data` as our initial condition for the ODE b(0). Using the same optimisation process as we did for the simple model, we compare the normalised `two hour data` to b(1) and `four hour data` to b(3). We now have an estimate for our parameters k_i. Ideally we would have more experimental data for the single cell parameters so that we could more accurately determine the parameters k_i.

4.2 Estimating the number of 0 edits

As in the bulk sequencing data, we do not have data for the number of RNA with 0 edits for any of our observations. As we needed to know this to carry out the optimisation in the previous section, we introduced the relative number of 0 edits as a parameter in the optimisation problem. We then had a normalised distribution including the 0 edits column. By finding the scale factor between this normalised distribution and the mean absolute counts (observed data) we can calculate the expected absolute RNA with no edits at each time point. This gives us the absolute distribution at time 0 as shown in Figure 5 (one hour after promoter was added).

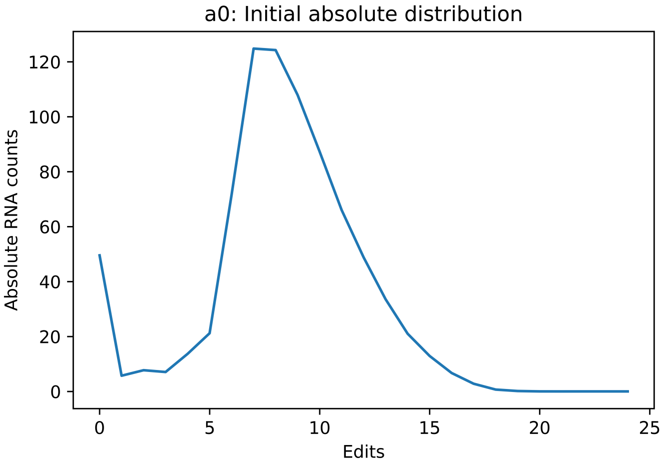


Figure 5: We have added our prediction of 0 edits to the observed data

4.3 Finding the degradation rate

In this subsection we look at how we can make an estimate for degradation rate. Initially I took a fairly naive approach in assuming that the absolute RNA counts could be used accurately to determine degradation rate c. We could then seek c to minimise the loss function in equation (10) where χ is the set of timepoints at which we make observations. So in this case $\chi = \{1, 3\}$

$$Loss(c) = \sum_{t\in\chi} \|a(t) - e^{-ct}e^{Kt}a(0)\|_2^2 \tag{10}$$

However, due to the inaccuracies of the experiment and the variability of cell size the absolute RNA count can not be used to measure the degradation rate. This can be seen in the experiment carried out. The mean number of RNA after one hour, two hours and four hours is 813.4, 888.5 and 1386.1 respectively. There is a clear increase in observed RNA count over time which is unexpected due to the degradation of RNA and the lack of production (due to the introduction of the transcription inhibitor).

I simulated a new experiment (using the simulator explained in the next section) to try and estimate c using the steady states of the ODEs. If we assume constant RNA production at rate P then we expect the steady state $\hat{a} = (\hat{a}_0, \hat{a}_1, \cdots \hat{a}_{25})$ to be given by:

$$\hat{a}_0 = \frac{P}{k_0 + c} \tag{11}$$

$$\hat{a}_1 = \frac{k_0 \hat{a}_0}{k_1 + c} \tag{12}$$

So our estimate of c can be given by $c = \frac{k_0 \hat{a}_0}{\hat{a}_1} - k_1$. As this relies on the ratio of the two steady states this method does not run into the same issues that occurred using the method above (we are relying on relative counts rather than absolute counts). We expect to reach the steady state on the timescale of hours as shown by the simulations in the next section. By carrying out reads after a long period of time, we can plot many observations on $a_0 - a_1$ axis. Then using the gradient of the straight line of best fit through the origin we can get a good estimate of $\frac{\hat{a}_0}{\hat{a}_1}$.

4.4 Running simulations

Simulations are carried out in a similar way to the simplified model with a few key differences. Firstly the input distribution is absolute number of RNA (so must be integers). Secondly, the simulator now accepts a production array. So if RNA is made at a constant rate λ for the 3rd hour, the number of RNA produced in the 3rd hour has a Poisson distribution with parameter λ and each RNA is produced at a random time uniformly distributed throughout the third hour.

To introduce degradation in our simulations we used the Gillespie Algorithm. Suppose there is an RNA with i edits. Define $\mu = k_i + c$. The time until the RNA is given another edit or degrades is:

$$T = \frac{1}{\mu} \log(1/r_1)$$

where $r_1 \sim U[0, 1]$. At this time the RNA accumulates another edit if $r_2 < k_i/\mu$, otherwise the RNA degrades, where again $r_2 \sim U[0, 1]$ independently of r_1.

I simulated the experiment that can be used to find the degradation rate. This involved starting with no initial RNA molecules, producing RNA at rate 4 for the first 4 hours before turning off the production. We can see from Figure 6 that the experiment approaches the steady state in the phase plane (red dot) before moving back to the origin after production is over. This suggests that if we sequenced the single cells after 4 hours we could accurately find the degradation rate using the method in the previous section.

4.5 Model analysis

This model is much more useful than the simplified model as it allows us to make predictions on single cell behaviour. This is useful as we can make simulations for each individual cell (see section 5.2). As we use use a different set of bases to the data from the first experiment of the Timestamping paper [3] we had to retrain to find k_i in Section 4.1. As suggested in the paper, the editing rates should be the same as we calculated for the bulk sequencing data (Section 3.1) meaning that we would only have had to calculate the degradation rate as we move to the more developed model (provided we are using the same bases).

5 RNA velocity

In the RNA velocity paper [1], they defined the RNA velocity of the gene as the rate of change of spliced RNA (as spliced RNA are used to make protein). So an analogous definition would be to define RNA velocity

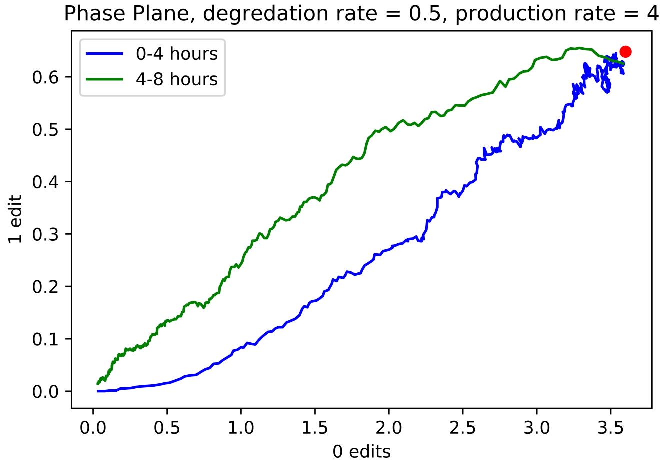
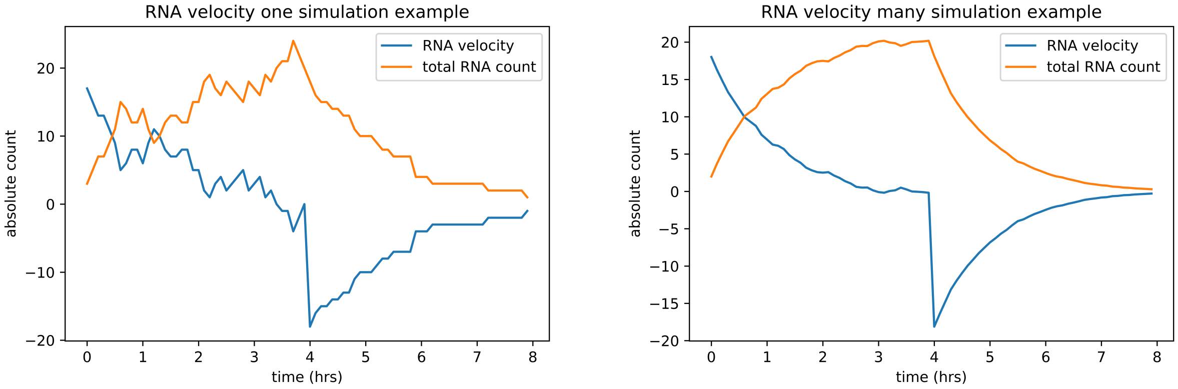


Figure 6: Simulated experiment

as the rate of change of all RNA:

$$v_1 = \sum_{i=0}^{25} \dot{a}_i = P - c \sum_{i=0}^{25} a_i \tag{13}$$

Here we assume the degradation rate c is always the same for a particular gene so that we can use the result of the previous experiment. We now carry out the same experiment (same production and initial condition) as in Figure 6 but plot the RNA velocity v_1. Initially the RNA velocity is large as RNA is being produced at a fast rate. As the system approaches a steady state, the velocity tends to 0 as expected. After the sudden drop in production, we see a sudden drop in the velocity as RNA is only being removed and not created.



(a) Single simulation

(b) Average of many simulations

Figure 7: Test observation

It seems as if v_1 gives an accurate definition of RNA velocity. However, there are numerous issues with our definition. The first being that we do not know the production rate p at this point.

5.1 Alternative definition without production

To get around the fact that we do not know p, we can instead consider the rate of change of RNA with edits greater or equal to 1:

$$v_2 = \sum_{i=1}^{25} \dot{a}_i = k_0 a_0 - c \sum_{i=1}^{25} a_i \tag{14}$$

This should gives us a "delayed" velocity as it does not account for the rate of change of the 0 edit RNA molecules. We say it is "delayed" as the rate of change of a_0 will affect the rate of change of the rest of the RNA molecules after some time as the RNA molecules get edited. v_1 is also highly sensitive on the a_0 reading meaning that it is more stochastic than v_0 . We repeated the simulation in Figure 7b and got the following result:

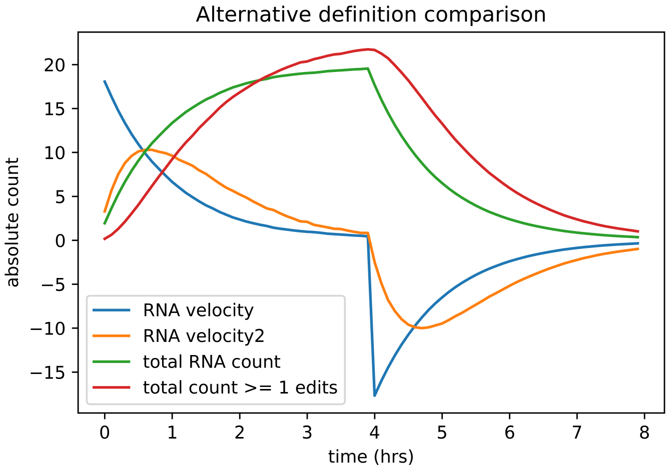


Figure 8: Comparing average of v_1 and v_2 in simple experiment

Note that the "RNA velocity2" and total count of RNA with 1 or more edits have been multiplied by 3 so that we can more easily compare them to the data in Figure 7b. v_2 seems to give a good estimation of RNA velocity in this experiment. There is the most error in the first hour as originally there is no RNA in the cell so it takes some time for our definition of velocity to calibrate.

As this new definition does not require us making an estimate of p it has an advantage over the original definition v_1. Additionally, v_1 relies us on us having an accurate reading for absolute RNA counts which can be highly variable. We can replace our observations of a_i in v_2 with normalised relative counts which will be much more reliable.

An issue with both these definitions is that currently they both require readings of a_0, an observation which is not made in current experiments. Additionally, v_0 and v_1 can not be used to compare velocities between cells. For example, a larger cell produces more RNA than a smaller cell so its velocities will have larger magnitude than the smaller cell (when RNA velocity should be independent of cell size). We attempt to combat this issue by normalising between genes in the next section.

5.2 Normalising between genes

In the RNA velocity paper they normalise their counts of spliced and unspliced RNA counts by dividing each reading by the sum of spliced or unspliced RNA across all genes.This allows us to compare RNA velocities between cells. If we suppose that a_{ij} is the number of RNA molecules with i edits relating to the jth gene, then we can redefine a relative velocity for a gene j as:

$$v_3 = \frac{k_0 a_{0j}}{\sum_{k=1}^{N} a_{0k}} - c \sum_{i=1}^{25} \frac{a_{ij}}{\sum_{k=1}^{N} a_{ik}} \tag{15}$$

where N is the number of timestamped genes. It would require timestamping multiple genes experimentally to see how useful this definition of RNA velocity could be.

We simulate an experiment to see how our new definition v_3 compares to v_1 and v_2. I simulate a single cell with 50 different genes which are timestamped. The degradation rates are randomly chosen uniformly on $[0, 2]$. The production rates are of the form $[p_{1j}, p_{2j}, p_{3j}, p_{4j}, p_{5j}, p_{6j}, p_{7j}, p_{8j}]$ for the jth gene where RNA is produced at rate p_{ij} for the ith hour. p_{ij} are i.i.d from the U[0,40] distribution. Initially we assume there is no RNA in the cell. We then chose 4 random genes and compared our different definitions of velocity. In Figure 9 we normalised each velocity by dividing by the maximum of each one so they could all be compared on the same scale.

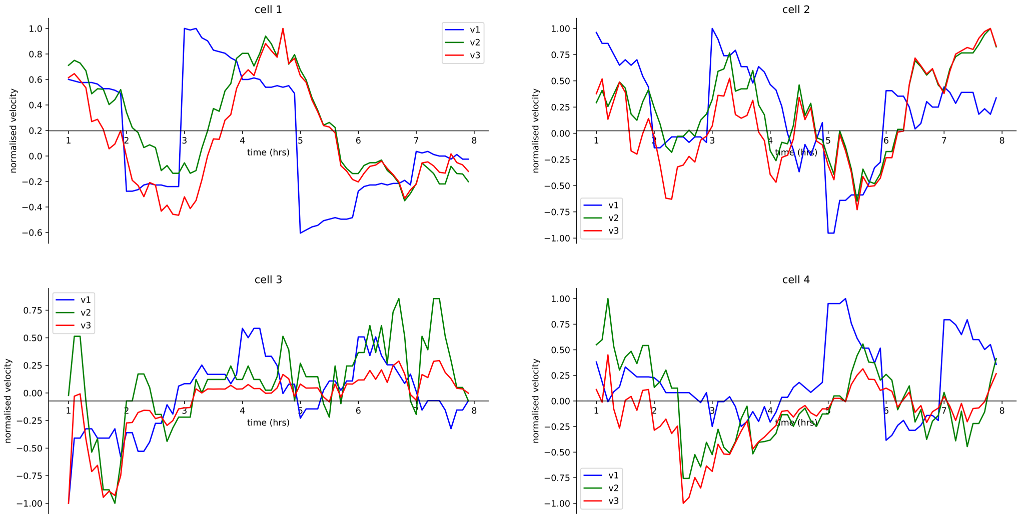


Figure 9: Comparing definitions in randomised experiment

From these four randomly chosen cells, v_3 seems to give an informative definition of RNA velocity after the first hour. A lot of the error (difference with v_1) in the first hour comes from the fact that v_2 and v_3 take a while to calibrate so we ignored the first hour in our plots. Across all genes v_1 and v_2 agree on their sign (both positive or both negative) 64% of the time while v_1 and v_3 agree on their sign 61% of the time. Finally, v_2 and v_3 agree on their sign 84% of the time.

5.3 Summary of definitions

v_1 provides the most accurate definition of RNA velocity but it relies on us knowing the production rate p. We introduced v_2 as an improvement on v_1 as it does not require us knowing p to calculate the velocity even if it is less accurate and “delayed”. The normalisation in v_3 allows us to compare RNA velocities between cells as variability in cell size means that it is difficult to compare magnitude of velocity between cells using v_1 or v_2. It is worth noting also the key assumption we have made in our simulations that all the genes have the same set of parameters k_i (which might not be true in practice).

6 Conclusion

Our simplified model gave a useful way to model the behaviour of a tissue of cells without being able to model each cell seperately. The more developed model allows us to make predictions on single cells using the simulator. We have provided a new way to define RNA velocity using the timestamps. Even though this definition uses the reading of a_0, a measurement not currently included in experimental data, I would be keen to see if there is a way to measure this experimentally or make predictions on what it might be.

Moving forward it would be interesting to see how accurately the simulator we created matches the results of real life experiments in more unusual experiments (for example having unusual production shapes). Additionally, it would be useful to have more data using on the "good" bases described in Section 3.6 which I would expect to better match the model.

Adding timestamps to multiple genes (as simulated in Section 5.2) would make our definitions of RNA velocity multidimensional and I am keen to see how our definitions of RNA velocity would compare between cells experimentally. The model would have to be adjusted greatly if there is a large difference in editing rates between different genes.

References

[1] G. La Manno, R. Soldatov, and A. et al. Zeisel. Rna velocity of single cells. *nature*, 560:494–498, 2018.

[2] Fabian J Theis Malte D Luecken. Current best practices in single-cell rna-seq analysis: tutorial. *Mol Syst Biol*, 15(e8746), 2019.

[3] S.G. Rodriques, L.M. Chen, and S. et al. Liu. Current best practices in single-cell rna-seq analysis: tutorial. *Mol Syst Biol*, 15(e8746), 2021.

[4] Linnarsson S. & Teichmann. S.a. single-cell genomics: coming of age. *Genome Biol.*, 17(97), 2016.