Evaluation of the robustness to batch effect in RNA velocity analyses

Author: Tomàs Montserrat Ayuso

Supervised by PhD Camille Stephan-Otto Attolini and the whole bioinformatics and biostatistics unit from IRB Barcelona research center.

September 30, 2022

ABSTRACT

All analyses based on NGS data suffer from batch effect when there is more than one sample involved, for various reasons, such as different technologies used or carrying out the analysis at different times. We performed a series of simulations from synthetic and real datasets to study the effect of the batch effect on the analysis and conclusions from RNA velocity analyses with scVelo. The results of these simulations reveal that when the samples are correctly integrated and scVelo uses the principal components of the corrected counts in its spliced and unspliced count imputation algorithms, the conclusions are not affected by the batch effect. When the batch effect is not confounded with any biological effect, RNA velocity analyses are suitable for integrated data, being able to take advantage of the extra power that comes of having more data. The code to reproduce the analysis is publicly available on the GitHub of the project (https://github.com/tmontserrat/rna_velocity_vs_batch_effect).

INTRODUCTION

During the last decade, many techniques have proliferated to analyse data from scRNA-seq experiments. One of the techniques to study these datasets, in particular those that come from studies of dynamic processes such as development and cell differentiation, is the so-called trajectory inference. These methods allow modelling the process by ordering the cells in a trajectory based on the similarity of the amount of RNA of each gene in each cell of the dataset[1].

Conventional methods for estimating the cellular trajectory do not allow the directionality of the process to be deduced. Without prior knowledge of biology, it is not possible to know which cell population represents a starting point and which an ending point. The RNA velocity (the temporal derivative of gene expression)[2], based on the ratio of unspliced and spliced mRNA of each gene, allows this direction to be deduced.

The premise of the RNA velocity analysis is simple. When a gene is activated, transcription begins, and mRNA is created. This newly transcribed mRNA still contains introns. It is the unspliced mRNA (or pre-mRNA). As time progresses, the cell processes this mRNA by removing the introns, obtaining the spliced mRNA. The algorithms to calculate the RNA velocity are based on the following differential equations[3]

$$\frac{du(t)}{dt} = \alpha(t) - \beta u(t),$$
$$\frac{ds(t)}{dt} = \beta u(t) - \gamma s(t),$$

Where α(t) is the rate of transcription, β the rate of splicing and γ the rate of degradation.

Thus, when a gene is in equilibrium, the production of newly spliced mRNA and its degradation must be equal. However, a newly activated gene will show a higher proportion of unspliced mRNA than it would have at equilibrium, while a repressed gene will have less.

Therefore, by estimating the parameters of these two differential equations we can establish a metric of the expected change in the expression of each gene, the RNA velocity. Positive rates indicate that this gene is activated, while negative rates indicate that it is repressed. Rates close to zero imply that this gene is in equilibrium. Combining the rates of each gene for a cell creates a vector that represents the direction and sense of transcriptional changes that cell will experience. The joint information of the vectors of all the cells of a dataset allows us to infer the directionality of the process.

In most experimental sequencing techniques, the so-called batch effect must be kept in mind. When conducting an experiment, it may be observed that some of the variation between two or more different samples is due to the time it was processed, the technology used or the batch in which it has been sequenced. This variation is not due to the biology of the sample but is a technical artifact. The effect it produces is known as the batch effect and it is advisable to eliminate or control it in order not to obtain spurious results.

There are different algorithms to correct the batch effect in scRNA-seq samples such as kMNN, fastKMNN, anchors, BBKNN, harmony, combat or limma[4]. However, these methods are designed to correct the batch effect in the count matrix or in the matrix of reduced dimensionality. It is to be expected that spliced and unspliced

matrices obtained from different samples also suffer from this effect. Development of methods to correct them has only just begun[5]. However, it has not been studied what effect the batch effect has on the estimation of RNA velocity and on the trajectories and conclusions derived from the downstream analysis.

In this work we have studied the effect of the batch effect on the analysis and conclusions of RNA velocity. We have worked under the hypothesis that the RNA velocity analysis is robust to the batch effect once the samples have been satisfactorily integrated, as long as the technical effect is not confounded with any biological effect. By eliminating the batch effect in the counting matrix, similar cells between different samples will reduce their distance in the n-dimensional space defined by gene expression. Assuming that the similar cells between the two or more samples share the same cellular dynamics, it can be thought that the estimation of the RNA velocity in these cells will be similar, especially the direction and sense of the resulting vector, being in essence a cell attribute robust to the batch effect.

In order to study the robustness to the batch effect of the RNA velocity analyses, we divided the analysis into two parts. First, we performed two simulations with Dyngen[6], with two different types of trajectories, in which we know the ground truth of the RNA velocity of each gene for each cell. From the comparison of the trajectories obtained in the unintegrated and integrated samples, we have verified the robustness of the analysis when well-estimated velocities are available. On the other hand, we also worked with a dataset of epithelial pancreatic cells from stage 15.5 of a mouse embryo, downloaded directly from the scVelo library. This dataset is annotated, and we can consider that we know the ground truth of the cell trajectories[3]. On this dataset we have performed three experiments generating a series of perturbations that want to simulate a batch effect and thus study the implications on the final conclusions of the RNA velocity analysis.

METHODS

Synthetic datasets

We generated two simulated datasets with the R package Dyngen. This simulator makes it possible to study dynamic cellular processes such as cellular differentiation. It can generate counts of spliced and unspliced mRNA from different cellular dynamics that originate different cellular trajectories. In addition, the program can create different datasets simulating two or more similar samples where the differences between them are due to the batch effect. Since scVelo does not perform well with these datasets, we have used the RNA velocity ground truth of each cell returned by the program for the analysis.

Using this software we have generated a dataset with 3 samples, whose cells follow a linear trajectory, and another with 2 samples, where the cells are subjected to a bifurcated trajectory. The generation of the batch effect has been carried out as mentioned in its article[6]. We did the pre-processing (normalization, reduction of dimensionality and visualization) with Seurat[7] with the spliced matrix as the count matrix. This pre-processing was carried out for the unintegrated samples and for the integrated samples. The integration algorithm was the same as proposed by Seurat (anchors)[7].

With these datasets, our interest has been to check if the cellular trajectories of each batch separately, delineated from the RNA velocity, were maintained once the data were integrated. The analysis of the RNA velocity has been carried out with scVelo[3] and the visual check has been carried out by projecting the velocities on the UMAP obtained in the pre-processing step.

Disturbed datasets

In order to work with a more realistic system in which we can consider we know the ground truth of cell developmental trajectories, we used the pancreatic epithelial cell dataset described above. In this dataset we know that there are four differentiated cell types (alpha, beta, delta and epsilon cells), three transient cell types (Ngn3 low EP, Ngn3 high EP and pre-endocrine cells) and a pool of cells in the cell cycle that eventually start the differentiation trajectory (ductal cells).

We divided the cells from this dataset into two groups, representing two different batches that we called batch 1 and batch 2. For batch 2 we built the *ratio* matrix, which is the proportion of unspliced and spliced for each gene. Basically, for each cell and gene, we performed the operation

$$r_{cg} = \frac{u_{cg}}{s_{cg}}$$

Where u_{cg} is the count of unspliced transcripts for gene g in cell c, s_{cg} is the count of spliced transcripts and r_{cg} is the ratio of the two.

We also generated the *zero* matrix, whose elements contain either 1 or 0 depending on whether the spliced matrix from batch 2 contains a non-zero number or a zero at that position, respectively.

Once this was done, in batch 1 we did not apply any perturbation, while in the second we added a random number (between 2 and 5, numbers arbitrarily chosen, and with as many elements as the number of cells) to each element of the spliced matrix simulating the effect that could have a batch effect. To do this, we created a vector of random numbers between 2 and 5 with as many elements as there are cells in batch 2 and added this vector to the spliced mRNA count of each gene in batch 2. To respect the original zeros of the matrix of this batch, we have multiplied each element of the perturbed matrix by the same position of the *zero* matrix.

In those cells where a gene has a zero in both arrays (spliced and unspliced) and thus *ratio* is undefined at that position, the *ratio* matrix is also assigned a zero. Then, the new unspliced matrix is generated by multiplying each element of the spliced matrix by the element in the same position of the *ratio* matrix.

$$u'_{cg} = s_{cg} * r_{cg}$$

In those positions where the spliced matrix is zero but the unspliced one is not (and therefore the *ratio* matrix is undefined) the value in the new unspliced matrix is the same as the original plus the perturbation. Finally, the values of the new unspliced matrix are rounded to the nearest integer.

We repeated this procedure three times by progressively increasing the difference between the two batches (table 1). In the first case, the two batches have approximately the same number of cells of each type. In the second, batch 2 and batch 3 do not have the same proportion of each cell type. In the third case, batch 2 has practically no pre-endocrine cells, a cell type necessary for the transcription data to be continuous throughout the dataset, thus having a large transcriptional gap between the different clusters. In this way, batch 2 only makes sense once the data is integrated.

Table 1 Cell types distribution among the different batches applied to the dataset. The ratio is given in brackets.

Cell type	**Control**	**Experiment 1**		**Experiment 2**		**Experiment 3**	
		Batch 1	**Batch 2**	**Batch 1**	**Batch 2**	**Batch 1**	**Batch 2**
Ductal	916 (0.25)	461 (0.24)	455 (0.25)	595 (0.25)	321 (0.25)	412 (0.19)	504 (0.33)
Ngn3 low EP	262 (0.07)	131 (0.07)	131 (0.07)	183 (0.07)	79 (0.06)	131 (0.06)	131 (0.09)
Ngn3 high EP	642 (0.17)	324 (0.18)	318 (0.17)	289 (0.12)	353 (0.27)	289 (0.13)	353 (0.23)
Pre-endocrine	592 (0.16)	306 (0.17)	286 (0.15)	444 (0.18)	148 (0.11)	562 (0.26)	30 (0.02)
Alpha	481 (0.13)	228 (0.12)	253 (0.14)	385 (0.16)	96 (0.07)	337 (0.16)	144 (0.09)
Beta	591 (0.16)	302 (0.16)	289 (0.16)	325 (0.14)	266 (0.20)	325 (0.15)	266 (0.17)
Delta	70 (0.02)	24 (0.01)	46 (0.02)	70 (0.03)	0 (0)	28 (0.01)	42 (0.03)
Epsilon	142 (0.04)	72 (0.04)	70 (0.04)	106 (0.04)	36 (0.03)	71 (0.03)	71 (0.05)

For both the unperturbed and the perturbed dataset, we followed a standard RNA velocity analysis pipeline. We pre-processed the samples with Seurat (normalization, integration, dimensionality reduction and visualization)

and performed the RNA velocity analysis as we did with the simulated data. In this case, however, we do not have the ground truth velocity of each gene for each cell and rely on the estimates made by the scVelo program.

The robustness of these estimates in the presence of a batch effect has been evaluated using different tools. On the one hand, through the visual analysis which has been based on the comparison of the velocity vector field displayed as streamlines. As a measure of the robustness of the analyses and subsequent conclusions from the estimated velocities, we compared the cell trajectories inferred with PAGA from the PAGA velocity-graph.

In a more analytical way, we compared the direction and sense of the vector of each cell formed by the velocities of the driver genes of each cell population between the control dataset and the problem, specifically between the shared top 100 driver genes of each sample. To do this, we first grouped the cells by cell type. We then filtered the genes to keep the 100 genes that contribute most to determining the future expression of cells of each type. Next, we only kept those driver genes present in both the control and problem datasets. In these filtered datasets, each cell is represented by a vector with as many elements as there are shared driver genes between the control and problem datasets. The value of each element is the calculated RNA velocity for that gene in that cell. To measure the similarity in direction and sense between these vectors and the control we calculated the cosine similarity between the vector of the same cell in the control dataset and in the problem according to the formula

$$s(x,y) = \frac{\sum_{i=1}^{n} x_i y_i}{\sqrt{\sum_{i=1}^{n}(x_i)^2} * \sqrt{\sum_{i=1}^{n}(y_i)^2}}$$

Where a value of -1 means that the vectors have the same direction but opposite senses, a value of 1 that they have the same direction and sense, and 0 that they are perpendicular. To obtain more compact and understandable results, we calculated the average of these values for each cell type.

We also compared the latent time values assigned to each cell between the control and problem datasets by calculating the Pearson correlation coefficient between cells in both datasets using

$$corr(x,y) = \frac{\sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_{i=1}^{n}(x_i - \bar{x})^2} * \sqrt{\sum_{i=1}^{n}(y_i - \bar{y})^2}}$$

Where a value of 1 implies a perfect positive correlation and a value of -1 a perfect negative correlation. The latent time is calculated by scVelo directly from the estimated parameters of the dynamical model and approximates the real time experienced by the cells in the process studied.

Pipelines studied

We studied two different pipelines to carry out the analyses and compared their results based on which provides more robustness against the batch effect. scVelo applies an imputation method on the spliced and unspliced counts consisting of replacing the values of the original counts with the average values (first moment) of the k-neighbours

of each cell. Neighbours are found in the space of the principal components. The two pipelines studied differ in the data used to calculate the principal components.

On the one hand, the principal components calculated from the original normalized data without integration have been used. We will refer to this pipeline as ‘pipeline 1’. On the other hand, the principal components calculated from the corrected Pearson residuals have been used, once normalization has been done using the variance stabilizing transformation[8] and integration with Seurat. We will refer to this pipeline as ‘pipeline 2’.

To compare the two methods, we repeated all the analyses described above using both. We also visualized the RNA velocity estimated in each experiment in a UMAP to study if these velocities were grouped, not only by cell type, but also by batch.

Parameters used

We performed all experiments using the combination of parameters that best captured the underlying biological signal. Basically, the parameters that have been varied from the default have been the number of genes used in the integration and the number of dimensions and neighbours used to construct the UMAP. For all experiments, the number of recursive neighbours has been kept constant to build the velocity graph.

The final parameters used in each experiment can be consulted on the GitHub of the project (https://github.com/tmontserrat/rna_velocity_vs_batch_effect).

RESULTS

Synthetic datasets

The trajectories delineated by the RNA velocity ground truth in the integrated and non-integrated datasets agree, suggesting that the trajectories are robust to integration. It maintains the directionality of the trajectory and does not distort the conclusions in this aspect (figure 1).

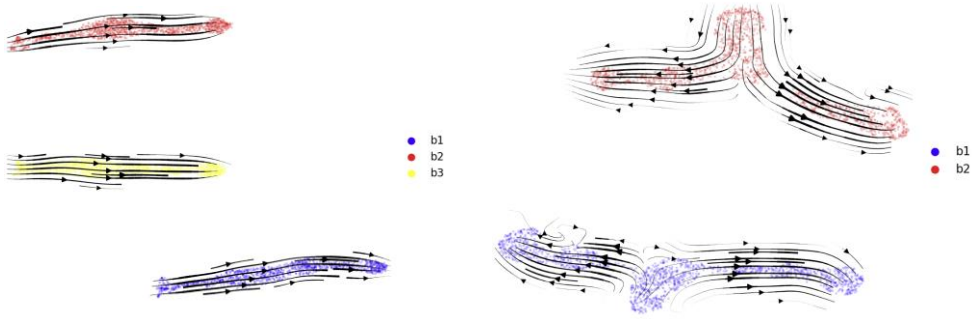
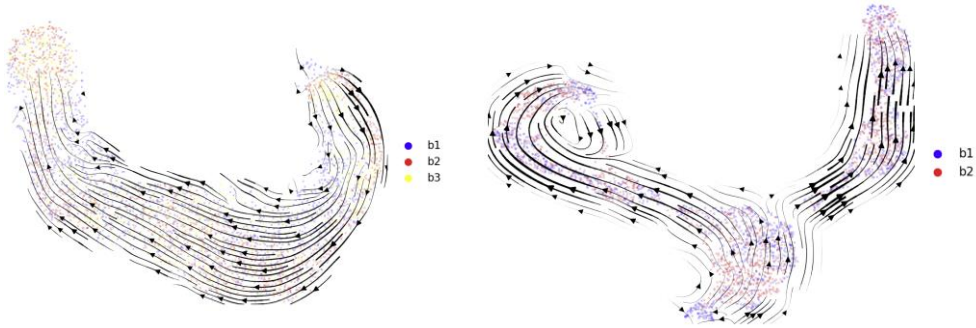
	Simulation 1	Simulation 2
Type of trajectory	Lineal	Bifurcated
Number of samples	3	2
Unintegrated data		
Integrated data		

Figure 1 Visualization of simulated datasets with both unintegrated and integrated data. The velocity vector field is displayed as streamlines to observe the directionality of the trajectories.

Disturbed dataset

In figure 2 the UMAP obtained with the original data and those obtained with the unintegrated and integrated data, coloured by the batch to which each cell belongs, can be observed. The cells are grouped by cell type and, within each cell type, a subcluster is observed for each batch. It can be visually appreciated how well the integration mixes the two samples.

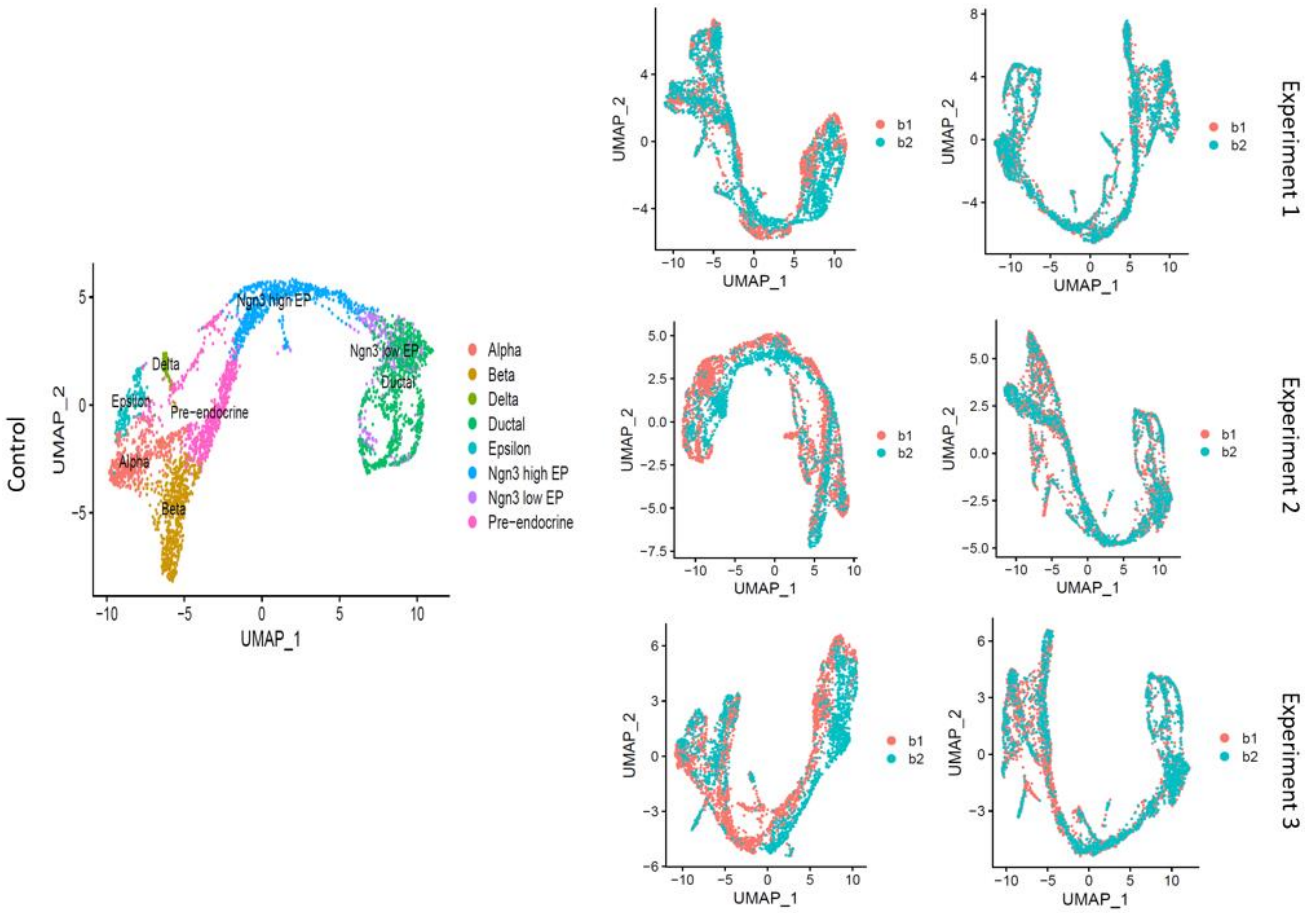


Figure 2 Visualization of the control dataset (left) and the three problem datasets with the unintegrated (center) and integrated (right) data.

As with the simulated datasets, integration does not impair the definition of cell trajectories in any of the datasets (figure 3). The cosine similarity of the cellular vectors defined by the rates of each gene are, on average, close to 1 in all cell populations, which means that the prediction of the future transcriptional state of the cells is, in overall, very similar for both the control dataset and the integrated datasets. Regarding the latent time, the average correlation between the cells in the control dataset or the disturbed ones is almost 1, indicating that this metric is highly robust to batch effect (table 2).

The definition of the trajectories and the cellular fate calculated with PAGA, we were able to reproduce the same trajectory as found in the control dataset in the three experiments carried out. The cyclic trajectory followed by the ductal cells, perhaps the most delicate, is the one that suffers the most as the datasets to be integrated are more different (that is, as the batch effect is more important).

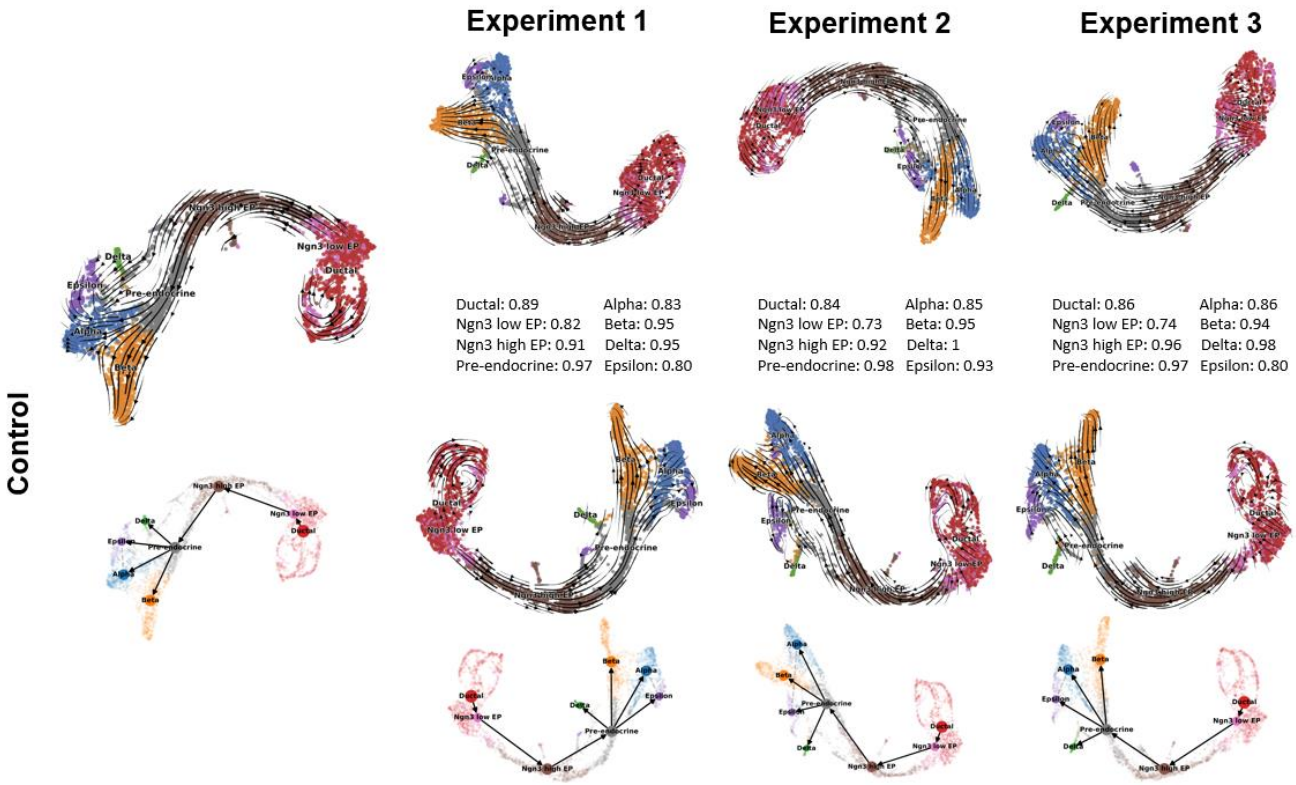


Figure 3 Summary of the results obtained for the trajectory and its directionality for the non-integrated (top) and integrated (third row) data. Also shown is the PAGA with the estimated trajectory (bottom) and the average cosine similarity of each cell type between the integrated data and the control (second row).

Pipeline comparison

For both pipelines analysed we found that using the pipeline 2 generally improves the estimation of cellular vectors as we can observe in figure 4 where the cosine similarity between the third experimental dataset and the control are almost the same or higher when using the pipeline 2.

Where we have found that the improvement is most significant is in the recovery of trajectories with PAGA. Using the pipeline 2, the predicted trajectories were the same as for the control dataset, thus obtaining the same conclusions. In contrast, using the pipeline 1, the results obtained have been unsatisfactory, the more the batches are different.

In viewing the velocity matrices using a UMAP we can see how the cells are not grouped by batch when we use the pipeline 2. In contrast, cells are grouped by batch when using the pipeline 1. Therefore, we see that the first moment calculated by scVelo allows to correct the batch effect in the velocities when the right neighbours are used.

No significant changes are observed in the plots of the velocity vector field displayed as streamlines of the integrated datasets in the UMAP between the two methods. In the same way, the correlation of latent time between

the control dataset and the experimental datasets does not show differences regarding the followed pipeline (table 2)

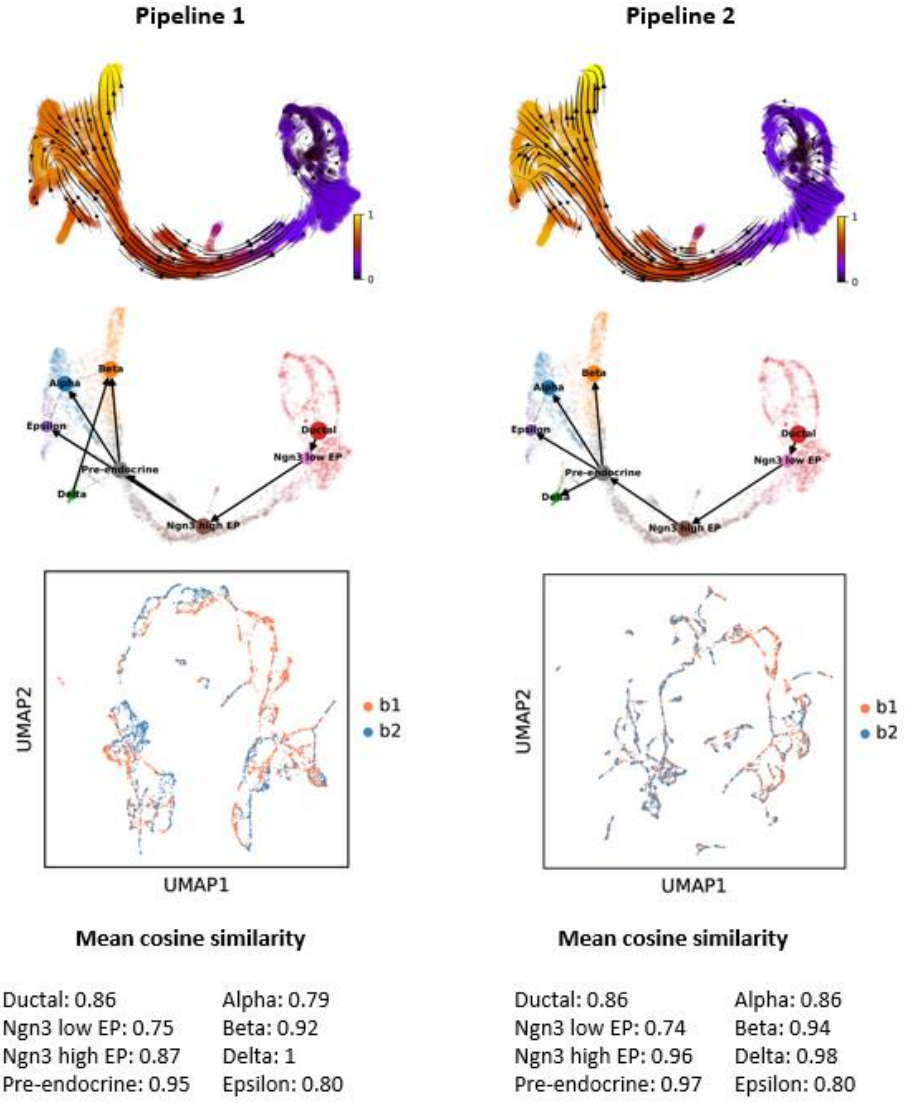


Figure 4 Comparison of the results obtained using the two different pipelines for the third experiment. Latent time and velocity embedded (above). PAGA inferred trajectories (second row). Visualization of the dataset using UMAP over the velocity matrix (third row). Difference between the two pipelines of the cosine similarity between the problem and control data (bottom). We observed similar results for the three experiments.

Table 2 Correlation of latent time between the control dataset and the three disturbed datasets.

Experiment	1	1	2	2	3	3
Pipeline	1	2	1	2	1	2
Control	0.98	0.99	0.99	0.98	0.99	0.98

DISCUSSION

The main conclusion of the study is that the RNA velocity analysis seems robust to the batch effect. The results with the simulated datasets, both with ground truth and estimated velocities, suggest that the final conclusions of the analysis are not altered by the integration. The trajectories defined by the velocities do not vary once the samples are integrated.

On the other hand, the three experiments carried out with the pancreas dataset disturbing part of the cells simulating a batch effect show that, while the biological signal is present, the integration of the different batches keeps this signal. The trajectories delineated by the RNA velocity have been recovered in all three cases, including the finer ones such as the cyclic one defined by the ductal cells (although more diffusely in the last experiment, with a stronger batch effect). The high cosine similarity between the cell's velocity vector from any of the experiments and the control reflects the observed robustness of RNA velocity analysis to batch effect.

The use of the pipeline 2 improves the capture of the biological signal of interest compared to the use of the pipeline 1. It is especially noticeable the more different are the two batches to integrate and for the reconstruction of the trajectory using PAGA.

Visualization of estimated velocities, after reducing its dimensionality, using one or another pipeline shows how these tend to cluster by batch if pipeline 1 is used. On the other hand, if the pipeline 2 is used, this separation disappears. Therefore, the batch effect affects the future cell expression vector (the vector of the velocities of each cell), but scVelo's imputation of the spliced and unspliced values from the average of the counts of k-neighbouring cells in the space of the principal components allows to correct this effect.

It should be added that the results and their interpretation depend on the quality of the integration and the resulting UMAP. In this aspect, parameters such as the number of genes to be used in Seurat's anchors algorithm, the number of principal components to be considered in the construction of the UMAP or the number of neighbours used in the generation of the velocity graph, become critical for draw accurate conclusions. In this sense, we have seen that increasing the number of recursive neighbours when building the velocity graph from 2 (default) to 4 improve the reconstruction of the trajectories, no matter which pipeline is being followed.

REFERENCES

1. Saelens, W., Cannoodt, R., Todorov, H. & Saeys, Y. A comparison of single-cell trajectory inference methods. *Nat Biotechnol* **37**, 547–554 (2019).
2. La Manno, G. *et al.* RNA velocity of single cells. *Nature* **560**, 494–498 (2018).
3. Bergen, V., Lange, M., Peidli, S., Wolf, F. A. & Theis, F. J. Generalizing RNA velocity to transient cell states through dynamical modeling. *Nat Biotechnol* **38**, 1408–1414 (2020).
4. Tran, H. T. N. *et al.* A benchmark of batch-effect correction methods for single-cell RNA sequencing data. *Genome Biol* **21**, 12 (2020).
5. Batch effects in scRNA velocity analysis. https://www.hansenlab.org/velocity_batch.
6. Cannoodt, R., Saelens, W., Deconinck, L. & Saeys, Y. Spearheading future omics analyses using dyngen, a multi-modal simulator of single cells. *Nat Commun* **12**, 3942 (2021).
7. Hao, Y. *et al.* Integrated analysis of multimodal single-cell data. *Cell* **184**, 3573-3587.e29 (2021).
8. Hafemeister, C. & Satija, R. Normalization and variance stabilization of single-cell RNA-seq data using regularized negative binomial regression. *Genome Biol* **20**, 296 (2019).