Methods

Correcting observer bias due to drop-out of reads

Let N reads from K clusters align to a reference sequence of length L. Let the proportion of reads whose 5' and 3' ends align, respectively, to coordinates a and b ($1 \leq a \leq b \leq L$) be γ_{ab} (assuming these proportions are equal for all clusters). Let the mutation rate of base j ($1 \leq j \leq L$) in cluster k ($1 \leq k \leq K$) be μ_{jk}. Let the proportion of cluster k in the ensemble be π_k.

Computing observer bias

Given the true (unobservable) proportions of alignment coordinates (γ_{ab}), mutation rates (μ_{jk}), and cluster proportions (π_k), compute what would be observed after the drop-out of reads with two mutations closer than the minimum gap (g) **?**. To express these quantities as probabilities, let C_{ab} be the event that a read aligns with 5' and 3' coordinates a and b, respectively; let S_j be the event that a read contains position j (i.e. its alignment coordinates a and b satisfy $1 \leq a \leq j \leq b \leq L$); let M_{jk} be the event that a read from cluster k has a mutation at position j; and let G_k be the event that a read from cluster k has no two mutations closer than g bases.

In terms of these events, the real mutation rates (μ_{jk}) are $P(M_{jk}|S_j)$, i.e. the probability that a read from cluster k would have a mutation at position j given that it contained position j; and the observed mutation rates (m_{jk}) are $P(M_{jk}|S_jG_k)$, i.e. the probability that a read from cluster k would have a mutation at position j given that it contained position j and had no two mutations closer than g bases. By Bayes' theorem:

$$m_{jk} = P(M_{jk}|S_jG_k) = P(M_{jk}|S_j)\frac{P(G_k|S_jM_{jk})}{P(G_k|S_j)} = \mu_{jk}\frac{P(G_k|S_jM_{jk})}{P(G_k|S_j)}$$

The term $P(G_k|S_j)$ represents the probability that a read from cluster k would have no two mutations closer than g bases given that it contained position j. It can

be computed using $P(G_k|C_{ab})$ (abbreviated d_{abk}): the probability that a read from cluster k would contain no two mutations closer than g bases given that its 5' and 3' coordinates, respectively, are a and b ($1 \leq a \leq b \leq L$). If position b were mutated (probability μ_{bk}), then the read would contain no two mutations closer than g bases if and only if none of the g bases preceding b (i.e. positions $b-g$ to $b-1$, inclusive) were mutated (probability $\prod_{j'=\max(b-g,a)}^{b-1}(1-\mu_{j'k})$, abbreviated $w_{\max(b-g,a),b-1,k}$) and two no mutations between positions a and $b-(g+1)$, inclusive, were too close (probability $d_{a,\max(b-(g+1),a),k}$)). If position b were not mutated (probability $1-\mu_{bk}$), then the read would contain no two mutations closer than g bases if and only if no mutations between positions a and $b-1$, inclusive, were too close (probability $d_{a,\max(b-1,a),k}$). These two possibilities generate a recurrence relation similar to that previously described **?**:

$$d_{abk} = \mu_{bk} w_{\max(b-g,a),b-1,k} d_{a,\max(b-(g+1),a),k} + (1-\mu_{bk}) d_{a,\max(b-1,a),k}$$

The base case is $d_{abk} = 1$ when $a = b$ because such a read would contain one position and thus be guaranteed to have no two mutations too close. Then, $P(G_k|S_j)$ is the average of d_{abk} over every read that contains position j, weighted by the proportions γ_{ab}:

$$P(G_k|S_j) = \frac{\sum_{a=1}^{j}\sum_{b=j}^{L}\gamma_{ab}d_{abk}}{\sum_{a=1}^{j}\sum_{b=j}^{L}\gamma_{ab}}$$

The term $P(G_k|S_jM_{jk})$ represents the probability that a read from cluster k would have no two mutations closer than g bases given that it contained a mutation at position j. It can be computed using $P(G_k|C_{ab}M_{jk})$ (abbreviated f_{abjk}): the probability that a read from cluster k would contain no two mutations closer than g bases given that its 5' and 3' coordinates, respectively, are a and b and that position j is mutated ($1 \leq a \leq j \leq b \leq L$). Because position j is mutated, having no two mutations closer than g bases requires that none of the g bases on both sides of position j be mutated. The probability that none of the positions $j-g$ to $j-1$ are mutated is $w_{\max(j-g,a),j-1,k}$, while that of finding no mutations among positions $j+1$ to $j+g$ is $w_{j+1,\min(j+g,b),k}$. Upstream of the g bases flanking position j (i.e. positions a

to $j-(g+1)$), the probability that no two mutations are too close is $d_{a,\max(j-(g+1),a),k}$; downstream (i.e. positions $j+(g+1)$ to b), the probability is $d_{\min(j+(g+1),b),b,k}$. Since mutations in these four sections are independent, the probability that the read contains no two mutations too close is the product:

$$f_{abjk} = d_{a,\max(j-(g+1),a),k} w_{\max(j-g,a),j-1,k} w_{j+1,\min(j+g,b),k} d_{\min(j+(g+1),b),b,k}$$

Then, $P(G_k|S_jM_{jk})$ is the average of f_{abjk} over every read that contains position j, weighted by the proportions γ_{ab}.

$$P(G_k|S_jM_{jk}) = \frac{\sum_{a=1}^{j}\sum_{b=j}^{L}\gamma_{ab}f_{abjk}}{\sum_{a=1}^{j}\sum_{b=j}^{L}\gamma_{ab}}$$

Combining the above results yields a formula for m_{jk}:

$$m_{jk} = \mu_{jk}\frac{\sum_{a=1}^{j}\sum_{b=j}^{L}\gamma_{ab}f_{abjk}}{\sum_{a=1}^{j}\sum_{b=j}^{L}\gamma_{ab}d_{abk}}$$

Screening coronavirus long-range interactions computationally

All coronaviruses with reference genomes in the NCBI Reference Sequence Database **?** were searched for using the following query:

```
refseq[filter] AND ("Alphacoronavirus"[Organism] OR
                    "Betacoronavirus"[Organism] OR
                    "Gammacoronavirus"[Organism] OR
                    "Deltacoronavirus"[Organism])
```

The complete record of every reference genome was downloaded both in FASTA format (for the reference sequence) and in Feature Table format (for feature locations). The location of the frameshift stimulating element (FSE) in each genome was estimated from the feature table, and the nearest instance of `TTTAAAC` was used as the slippery site, using a custom Python script. The 2,000 nt segment beginning 100 nt upstream of and ending 1,893 nt downstream of the slippery site was

used for predicting long-range interactions involving the FSE. Genomes with ambiguous nucleotides (e.g. `N`) in this segment were discarded. For each coronavirus genome, up to 100 secondary structure models of the 2,000 nt segment were generated using Fold version 6.3 from RNAstructure **?** with `-M 100` and otherwise default parameters. Then, for each position, the fraction of models for the coronavirus in which the base at the position paired with any other base between positions 101 (the first base of the slippery sequence) and 250 was calculated using a custom Python script. The coronaviruses were clustered by their fraction vectors using the unweighted pair group method with arithmetic mean (UPGMA) and a euclidean distance metric, implemented in Seaborn version 0.11 **?** and SciPy version 1.7 **?**. The resulting hierarchically-clustered heatmap was examined manually to select coronaviruses based on the prominence of potential long-range interactions with the FSE (relatively large fractions far from positions 101-250).