Journal of Visualized Experiments

Validating whole genome Nanopore sequencing, using Usutu virus as an example

--Manuscript Draft--

Article Type:	Invited Methods Article - JoVE Produced Video
Manuscript Number:	JoVE60906R2
Full Title:	Validating whole genome Nanopore sequencing, using Usutu virus as an example
Section/Category:	JoVE Genetics
Keywords:	Nanopore sequencing; R10 flowcell; USUV; arboviruses; whole genome sequencing
Corresponding Author:	Bas Oude Oude Munnink Erasmus MC Rotterdam, Zuid-Holland NETHERLANDS
Corresponding Author's Institution:	Erasmus MC
Corresponding Author E-Mail:	b.oudemunnink@erasmusmc.nl
Order of Authors:	Bas B. Oude Munnink
	David F. Nieuwenhuijse
	Reina S. Sikkema
	Marion Koopmans
Additional Information:	
Question	**Response**
Please indicate whether this article will be Standard Access or Open Access.	Open Access (US$4,200)
Please indicate the **city, state/province, and country** where this article will be **filmed**. Please do not use abbreviations.	Rotterdam, Zuid-Holland, The Netherlands

TITLE:

Validating Whole Genome Nanopore Sequencing, using Usutu Virus as an Example

AUTHORS AND AFFILIATIONS:

Bas B. Oude Munnink[1], David F. Nieuwenhuijse[1], Reina S. Sikkema[1] and Marion Koopmans[1]

[1]ErasmusMC, Department of Viroscience, WHO Collaborating Centre for Arbovirus and Viral Hemorrhagic Fever Reference and Research, Rotterdam, The Netherlands

Email Addresses of Co-Authors:

David F. Nieuwenhuijse	(d.nieuwenhuijsse@erasmusmc.nl)
Reina S. Sikkema	(r.sikkema@erasmusmc.nl)
Marion Koopmans	(m.koopmans@erasmusmc.nl)

Corresponding Author:

Bas B. Oude Munnink	(b.oudemunnink@erasmusmc.nl)

KEYWORDS:

Nanopore, sequencing, R10 flowcell, USUV, arboviruses, whole genome sequencing

SUMMARY:

We previously validated a protocol for amplicon-based whole genome Usutu virus (USUV) sequencing on a nanopore sequencing platform. Here, we describe the methods used in more detail and determine the error rate of the nanopore R10 flow cell.

ABSTRACT:

Whole genome sequencing can be used to characterize and to trace viral outbreaks. Nanopore-based whole genome sequencing protocols have been described for several different viruses. These approaches utilize an overlapping amplicon-based approach which can be used to target a specific virus or group of genetically related viruses. In addition to confirmation of the virus presence, sequencing can be used for genomic epidemiology studies, to track viruses and unravel origins, reservoirs and modes of transmission. For such applications, it is crucial to understand possible effects of the error rate associated with the platform used. Routine application in clinical and public health settings require that this is documented with every important change in the protocol. Previously, a protocol for whole genome Usutu virus sequencing on the nanopore sequencing platform was validated (R9.4 flowcell) by direct comparison to Illumina sequencing. Here, we describe the method used to determine the required read coverage, using the comparison between the R10 flow cell and Illumina sequencing as an example.

INTRODUCTION:

Fast developments in third generation sequence technologies allows us to move forward towards close to real-time sequencing during viral outbreaks. This timely availability of genetic information can be useful to determine the origin and evolution of viral pathogens. Gold standards in the fields of next generation sequencing however, are still the second-generation sequencers. These techniques rely on specific and time-consuming techniques like clonal amplification during an emulsion PCR or clonal bridge amplification. The third-generation sequencers are cheaper, hand-held and come with simplified library preparation methodologies. Especially the small size of the sequence device and the low purchase price makes it an interesting candidate for deployable, fieldable sequencing. This could for instance be seen during the Ebola virus outbreak in Sierra Leone and during the ongoing arbovirus outbreak investigations in Brazil[1–3]. However, the reported high error rate[4] might limit the applications for which nanopore sequencing can be used.

Nanopore sequencing is evolving quickly. New products are available in the market on a regular basis. Examples of this are for instance the 1D squared kits which enables sequencing of both strands of the DNA molecule, thereby boosting the accuracy of the called bases[5] and the development of the R10 flow cell which measures the change in current at two different instances in the pore[6]. In addition, improved bio-informatic tools like improvements in basecalling will improve the accuracy of basecalling[7]. One of the most frequently used basecallers, (e.g., Albacore), has been updated at least 12 times in a 9-month time period[5]. Recently, the manufacturer also released a novel basecaller called flip-flop, which is implemented in the default nanopore software[8]. Together, all of these improvements will lead to more accurate sequences and will decrease the error rate of the nanopore sequencer.

Usutu virus (USUV) is a mosquito-borne arbovirus of the family *Flaviviridae* and it has a positive-stranded RNA genome of around 11,000 nucleotides. USUV mainly affects great grey owls and blackbirds[9, 10], although other bird species are also susceptible to USUV infection[11]. Recently, USUV was also identified in rodents and shrews although their potential role in transmission of the virus remains unknown[12]. In humans, asymptomatic infections have been described in blood donors[13–16] while USUV infections also have been reported to be associated with encephalitis or meningo-encephalitis[17, 18]. In the Netherlands, USUV was first detected in wild birds in 2016[10] and in asymptomatic blood donors in 2018[14]. Since the initial detection of USUV, outbreaks have been reported during the subsequent years and surveillance, including whole genome sequencing, is currently ongoing to monitor the emerge and spread of an arbovirus in a previously naïve population.

Similar to what has been described for other viruses, such as Ebola virus, Zika virus and yellow fever virus[3, 19, 20], we have developed a primer set to sequence full length USUV[21]. This polymerase chain reaction (PCR)-based approach allows for the recovery of full length USUV genomes from highly host-contaminated sample types like brain samples in samples up to a Ct value of around 32. Benefits of an amplicon-based sequencing approach are a higher sensitivity compared to metagenomic sequencing and a higher specificity. Limitations of using an amplicon-based approach are that the sequences should be similar in order to design primers fitting all strains and that primers are designed on our current knowledge about the virus diversity.

Given the constant developments and improvements in third generation sequencing, there is a need to evaluate the error rate of the sequencer on a regular basis. Here, we describe a method to evaluate the performance of nanopore directly against Illumina sequencing using USUV as an example. This method is applied to sequences generated with the latest R10 flow cell and basecalling is performed with the latest version of the flip-flop basecaller.

PROTOCOL:

NOTE: List of software tools to be used: usearch v11.0.667; muscle v3.8.1551; porechop 0.2.4; cutadapt 2.5; minimap2 2.16-r922; samtools 1.9; trimmomatic 0.39; bbmap 38.33; spades v3.13.1; kma-1.2.8

1. Primer design

1.1. Start with downloading or retrieving a set of relevant reference whole genome sequences from public or private data collections. For instance, retrieve all full length USUV genomes (taxid64286) from the NCBI database[22]. USUV encodes a genome of around 11,000 nucleotides so only retrieve the sequences with a sequence length of 8,000–12,000 nucleotides. Do this using the following search entry:

- *taxid64286[Organism:noexp] AND 8000[SLEN]:12000[SLEN]*.

1.1.1. Click on **Send to** | **Complete Record** | **File**; use Format = FASTA and create the File.

1.2. To downsize the set of reference sequences, remove duplicate sequences or sequences with over 99% nucleotide identity from the dataset. Do this using the cluster fast option from usearch[23]. On the command line enter:

- *usearch -cluster_fast All_USUV.fasta -id 0.99 -centroids All_USUV_dedup.fasta*

1.3. To generate the primers, sequences need to be aligned. This is done using MUSCLE[24]. On the command line enter:

- *muscle -in All_USUV_dedup.fasta -out All_USUV_dedup_aligned.fasta -log log_muscle.txt*

NOTE: It is essential to manually inspect the alignment to check for discrepancies. These can be manually corrected if needed and the ends can be trimmed according to the length of most whole genome sequences.

1.4. Primal is used to make a draft selection of the primers which can be used for full length amplicon sequencing[19]. Upload the alignment to the primal website (http://primal.zibraproject.org/) and select the preferred amplicon length and overlap length between the different amplicons. Go to primal.zibraproject.org, fill in the **Scheme name**, upload the aligned fasta file, select the amplicon length, overlap size, and generate the scheme.

1.5. Align the complete set of available complete USUV sequences (not the downsized or deduplicated set). On the command line enter:

- *muscle -in All_USUV.fasta -out All_USUV_aligned.fasta -log log_muscle.txt*

NOTE: Map the generated primers against the complete alignment (do not use the deduplicated alignment), manually correct errors and include a maximum of 5 degenerative primer positions.

2. Multiplex PCR

2.1. Perform the multiplex PCR using the designed primers and nanopore and Illumina sequencing. The multiplex PCR for USUV was performed as previous described[19, 21].

2.2. Perform basecalling with flip-flop version 3.0.6.6+9999d81.

3. Data analysis to generate consensus sequences from nanopore data

3.1. Several samples can be multiplexed on a single nanopore sequencing run. After performing the sequence run, demultiplex the nanopore data. Use Porechop[25] for this. To prevent contamination and enhance accuracy, use the *require_two_barcodes* flag. On the command line enter:

```
-	porechop -i Run_USUV.fastq -o Run_USUV_demultiplex --require_two_barcodes
```

3.2. After demultiplexing, remove primer sequences (indicated in the file Primers_Usutu.fasta in both orientations) using cutadapt[26]. In addition, remove sequences with a length shorter than 75 nucleotides. The primers have to be removed since they can introduce artificial biases in the consensus sequence. On the command line enter:

```
-	cutadapt -b file:Primers_USUV.fasta -o BC01_trimmed.fastq BC01.fastq -m 75
```

3.3. Demultiplexed sequence reads can be mapped against a panel of distinct reference strains using minimap2[27] and a consensus sequence can be generated using samtools[28]. Follow the example below which shows the procedure of a reference-based alignment and the consensus sequence generation of one sample: BC01. On the command line enter:

```
-	minimap2 -ax map-ont Random_Refs_USUV.fasta BC01_trimmed.fastq > BC01.bam
-	samtools sort BC01.bam > BC01_sorted.bam
-	bcftools mpileup -Ou -f Random_Refs_USUV.fasta BC01_sorted.bam | bcftools call -mv -Oz -o BC01.vcf.gz
-	bcftools index BC01.vcf.gz
-	cat Random_Refs_USUV.fasta | bcftools consensus BC01.vcf.gz > BC01_consensus.fasta
```

3.4. For reference-based alignments it is essential that a closely related reference sequence is used. Therefore, perform a BlastN search with the generated consensus sequence to identify the closest reference strain. After that, repeat the reference-based alignment with the closest reference strain as reference (step 3.3 and 3.4). On the command line enter:

```
-	minimap2 -ax map-ont Ref_USUV_BC01.fasta BC01_trimmed.fastq > BC01_ref.bam
-	samtools sort BC01_ref.bam > BC01_sorted_ref.bam
-	bcftools mpileup -Ou -f Ref_USUV_BC01.fasta BC01_sorted_ref.bam | bcftools call -mv -Oz -o BC01_ref.vcf.gz
-	bcftools index BC01_ref.vcf.gz
-	cat Ref_USUV_BC01.fasta | bcftools consensus BC01_ref.vcf.gz >
```

```
BC01_ref_consensus.fasta
```

4. Analysis of the Illumina data

4.1. These sequences are automatically demultiplexed after sequencing. Reads can be quality controlled using trimmomatic[29]. For paired-end Illumina sequences, use the commonly used cut-off median PHRED score of 33 and a minimal read length of 75 to get accurate, high quality reads. On the command line enter:

```
-   trimmomatic PE -phred33 9_S9_L001_R1_001.fastq.gz 9_S9_L001_R2_001.fastq.gz 9_1P.fastq 9_1U.fastq 9_2P.fastq 9_2U.fastq LEADING:3 TRAILING:3 SLIDINGWINDOW:3:15 MINLEN:75
```

4.2. Remove primers (indicated in the file Primers_Usutu.fasta in both orientations), since they can introduce artificial biases, using cutadapt[26]. In addition, remove sequences with a length shorter than 75 nucleotides using the commands below. On the command line enter:

```
-   cutadapt -b file:Primers_USUV.fasta -o 9_1P_trimmed.fastq -p 9_2P_trimmed.fastq 9_1P.fastq 9_2P.fastq -m 75
```

4.3. Before de novo assembly, the sequence reads can be normalized for an even coverage across the genome. This is essential since de novo assemblers like SPAdes take the read coverage into account when assembling sequence reads. Normalize reads to a read coverage of 50 using BBNorm from the BBMap package[30]. On the command line enter:

```
-   bbmap/bbnorm.sh target=50 in=9_1P_trimmed.fastq in2=9_2P_trimmed.fastq out=Sample9_FW_norm.fastq out2= Sample9_RE_norm.fastq
```

4.4. The normalized reads are de novo assembled using SPAdes[31]. Default settings are used for the assembly using all different kmers (21, 33, 55, 77, 99 and 127). On the command line enter:

```
-   spades.py -k 21,33,55,77,99,127 -o Sample9 -1 Sample9.qc.f.fq -2 Sample9.qc.r.fq
```

4.5. Map the QC reads against the obtained consensus sequence using minimap2 and programs like Geneious, Bioedit or Ugene to curate the alignment. It is important to check the beginning and the end of the contig.

4.5.1. Align the QC reads against the obtained consensus sequencing using minimap2.

4.5.2. Import the alignment in Geneious/Bioedit/UGene.

4.5.3. Manually inspect, correct and curate especially the beginning and the end of the genome.

5. Determining the required read coverage to compensate for the error profile in nanopore sequencing using Illumina data as gold standard

5.1. Select sequence reads mapping to one amplicon, in this case amplicon 26. Subsequently, map the nanopore reads against this amplicon using minimap2. Use Samtools to select only the reads mapping to amplicon 26 and to convert the bam file into fastq. On the command line enter:

```
minimap2 -ax map-ont -m 150 Amplicon26.fasta BC01_trimmed.fastq > BC01.bam
samtools view -b -F 4 BC01.bam > BC01_mapped.bam
samtools bam2fq BC01_mapped.bam | seqtk seq - -> BC01_mapped.fastq
```

5.2. Randomly select subsets of for instance 200 sequence reads one thousand times. For example, changing it to 10 will result in the random selection of one thousand times a subset of 10 sequence reads. The script is provided as **Supplementary File 1**. On the command line enter:

```
python Random_selection.py
```

5.3. All randomly selected sequence reads are aligned to amplicon 26. Use KMA[32] to map the sequence reads and to immediately generate a consensus sequence. Use optimized settings for nanopore sequencing, indicated by the -bcNano flag. On the command line enter:

```
kma index -i Amplicon26.fasta
for file in random_sample*; do
	sampleID=${file%.fastq}
	kma -i ${sampleID}.fastq -o ${sampleID} -t_db Amplicon26.fasta –mem_mode -mp 5 -mrs 0.0 -bcNano
done
```

5.4. Inspect the generated consensus sequences on the command line using:

```
cat *.fsa > All_genomes.fsa
minimap2 -ax map-ont Amplicon26.fasta All_genomes.fsa > All_genomes.bam
samtools sort All_genomes.bam > All_genomes_sorted.bam
samtools stats All_genomes_sorted.bam > stats.txt
```

5.4.1. The error rate is displayed in the stats.txt under the heading **error rate #mismatches / bases mapped**. Display it on the screen with the following command:

```
grep ^SN stats.txt | cut -f 2-
```

5.4.2. The amount of indels is displayed under the heading **#Indels per cycle**. Display it on the screen with the following command:

```
grep ^IC stats.txt | cut -f 2-
```

REPRESENTATIVE RESULTS:

Recently, a new version of the flow cell version (R10) was released and offered improvements to the basecaller used to convert the electronic current signal to DNA sequences (so-called flip-flop basecaller). Therefore, we have re-sequenced USUV from brain tissue of an USUV-positive owl which was previously sequenced on a R9.4 flow cell and on an Illumina Miseq instrument[21]. Here, we described the method used to determine the required read coverage for reliable consensus calling by direct comparison to Illumina sequencing.

Using the newer flow cell in combination with the basecaller flip-flop we show that a read coverage of 40x results in identical results as compared to Illumina sequencing. A read coverage of 30x results in an error rate of 0.0002% which corresponds to one error in every 585,000 nucleotides sequenced, while a read coverage of 20x results in one error in every 63,529 nucleotides sequenced. A read coverage of 10x results in one error in every 3,312 nucleotides sequenced, meaning that over three nucleotides per full USUV genome are being called wrong. With a read coverage above 30x, no indels were observed. A read coverage of 20x resulted in the detection of one indel position while a read coverage of 10x resulted in indels in 29 positions. An overview of the error rate using different read coverage cut-offs is shown in **Table 1**.

FIGURE AND TABLE LEGENDS:

Table 1. Overview of the error rate of nanopore sequencing. Each iteration represents one thousand random samples.

DISCUSSION:

Nanopore sequencing is constantly evolving and therefore there is a need for methods to monitor the error rate. Here, we describe a workflow to monitor the error rate of the nanopore sequencer. This can be useful after the release of a new flow cell, or if new releases of the basecalling are released. However, this can also be useful for users who want to set-up and validate their own sequencing protocol.

Different software and alignment tools can yield different results[33]. In this manuscript, we aimed to use freely available software packages which are commonly used, and which have clear documentation. In some cases, preference might be given to commercial tools, which generally have a more user-friendly interfaces but have to be paid for. In the future, this method can be applied to the same sample in case big modifications in sequence technology or basecalling software are introduced Preferentially this should be done after each update of the basecaller or flowcell, however given the speed of the current developments this can be also been done only after major updates.

The reduction in the error rate in sequencing allows for a higher number of samples to be multiplexed. Thereby, nanopore sequencing is getting closer to replacing conventional real time PCRs for diagnostic assays, which is already the case for influenza virus diagnostics. In addition, the reduction of the error rate increases the usability of this technique sequencing, for instance for the determination of minor variants and for high-throughput unbiased metagenomic sequencing.

A critical step in the protocol is that close, reliable reference sequences need to be available. The primers are based on the current knowledge about virus diversity and might need to be updated every once in a while. Another critical point when setting up an amplicon-based sequencing approach is the balancing of the primer concentration to get an even balance in amplicon depth. This enables the multiplexing of more samples on a sequence run and results in a significant cost reduction.

ACKNOWLEDGMENTS:

This work has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 643476 (COMPARE).

DISCLOSURES:

The authors have nothing to disclose.

REFERENCES:

1. Faria, N. R. et al. Establishment and cryptic transmission of Zika virus in Brazil and the Americas. *Nature*. **546** (7658), 406–410 (2017).
2. Bonaldo, M. C. et al. Genome analysis of yellow fever virus of the ongoing outbreak in Brazil reveals polymorphisms. *Memórias do Instituto Oswaldo Cruz*. **112** (6), 447–451 (2017).
3. Faria, N. R. et al. Genomic and epidemiological monitoring of yellow fever virus transmission potential. *bioRxiv*. 299842 (2018).
4. Magi, A., Giusti, B., Tattini, L. Characterization of MinION nanopore data for resequencing analyses. *Briefings in Bioinformatics*. **18** (6), bbw077 (2016).
5. Rang, F. J., Kloosterman, W.P., de Ridder, J. From squiggle to basepair: computational approaches for improving nanopore sequencing read accuracy. *Genome Biology*. **19** (1), 90 (2018).
6. Nanopore Store, R10 flow cells.
7. Wick, R. R., Judd, L. M., Holt, K.E. Performance of neural network basecalling tools for Oxford Nanopore sequencing. *Genome Biology*. **20** (1), 129 (2019).
8. GitHub - nanoporetech/flappie: Flip-flop basecaller for Oxford Nanopore reads.
9. Lühken, R. et al. Distribution of Usutu Virus in Germany and Its Effect on Breeding Bird Populations. *Emerging Infectious Diseases*. **23** (12), 1994–2001 (2017).
10. Cadar, D. et al. Widespread activity of multiple lineages of Usutu virus, Western Europe, 2016. *Eurosurveillance*. **22** (4) (2017).
11. Becker, N. et al. Epizootic emergence of Usutu virus in wild and captive birds in Germany. *PLoS ONE*. **7** (2) (2012).
12. Diagne, M. et al. Usutu Virus Isolated from Rodents in Senegal. *Viruses*. **11** (2), 181 (2019).
13. Bakonyi, T. et al. Usutu virus infections among blood donors, Austria, July and August 2017 – Raising awareness for diagnostic challenges. *Eurosurveillance*. **22** (41) (2017).
14. Zaaijer, H. L., Slot, E., Molier, M., Reusken, C.B.E.M., Koppelman, M.H.G.M. Usutu virus infection in Dutch blood donors. *Transfusion*. trf.15444 (2019).
15. Cadar, D. et al. Blood donor screening for West Nile virus (WNV) revealed acute Usutu virus (USUV) infection, Germany, September 2016. *Eurosurveillance*. **22** (14), 30501 (2017).
16. Pierro, A. et al. Detection of specific antibodies against West Nile and Usutu viruses in healthy blood donors in northern Italy, 2010–2011. *Clinical Microbiology and Infection*. **19** (10), E451–E453 (2013).
17. Pecorari, M. et al. First human case of Usutu virus neuroinvasive infection, Italy, August-September 2009. *Euro surveillance : bulletin européen sur les maladies transmissibles = European Communicable Disease Bulletin*. **14** (50) (2009).
18. Simonin, Y. et al. Human Usutu Virus Infection with Atypical Neurologic Presentation, Montpellier, France, 2016. *Emerging Infectious Diseases*. **24** (5), 875–878 (2018).
19. Quick, J. et al. Multiplex PCR method for MinION and Illumina sequencing of Zika and

other virus genomes directly from clinical samples. *Nature Protocols*. **12** (6), 1261–1276 (2017).

20. Quick, J. et al. Real-time, portable genome sequencing for Ebola surveillance. *Nature*. **530** (7589), 228–232 (2016).

21. Oude Munnink, B. B. et al. Towards high quality real-time whole genome sequencing during outbreaks using Usutu virus as example. *Infection, Genetics and Evolution*. **73**, 49–54 (2019).

22. Benson, D. A., Karsch-Mizrachi, I., Lipman, D.J., Ostell, J., Sayers, E.W. GenBank. *Nucleic Acids Research*. **38** (Database issue), D46-51 (2010).

23. Edgar, R. C. Search and clustering orders of magnitude faster than BLAST. *Bioinformatics*. **26** (19), 2460–2461 (2010).

24. Edgar, R. C. MUSCLE: multiple sequence alignment with high accuracy and high throughput. *Nucleic Acids Research*. **32** (5), 1792–1797 (2004).

25. R. R. Wick GitHub - rrwick/Porechop: adapter trimmer for Oxford Nanopore reads.

26. Martin, M. Cutadapt removes adapter sequences from high-throughput sequencing reads. *EMBnet.journal*. **17** (1), 10 (2011).

27. Li, H. Minimap2: pairwise alignment for nucleotide sequences. *Bioinformatics*. **34** (18), 3094–3100 (2018).

28. Li, H. et al. The Sequence Alignment/Map format and SAMtools. *Bioinformatics*. **25** (16), 2078–2079 (2009).

29. Bolger, A. M., Lohse, M., Usadel, B. Trimmomatic: a flexible trimmer for Illumina sequence data. *Bioinformatics (Oxford, England)*. **30** (15), 2114–20 (2014).

30. BBMap download | SourceForge.net.

31. Bankevich, A. et al. SPAdes: a new genome assembly algorithm and its applications to single-cell sequencing. *J Comput Biol*. **19** (5), 455–477 (2012).

32. Clausen, P. T. L. C., Aarestrup, F.M., Lund, O. Rapid and precise alignment of raw reads against redundant databases with KMA. *BMC Bioinformatics*. **19** (1), 307 (2018).

33. Brinkmann, A. et al. Proficiency Testing of Virus Diagnostics Based on Bioinformatics Analysis of Simulated In Silico High-Throughput Sequencing Data Sets. *Journal of Clinical Microbiology*. **57** (8) (2019).

Coverage	Errors iteration 1	Error rate iteration 1	Indels:	Errors iteration 2	Error rate iteration 2	Indels:
10×	100	0.0274%	4	116	0.0297%	18
20×	4	0.0010%	0	6	0.0015%	1
30x	2	0.0005%	0	0	0.0000%	0
40x	0	0.0000%	0	0	0.0000%	0
50×	0	0.0000%	0	0	0.0000%	0

Errors iteration 3	Error rate iteration 3	Indels:
110	0.0282%	7
7	0.0018%	0
0	0.0000%	0
0	0.0000%	0
0	0.0000%	0

Coverage	Errors iteration 1	Error rate iteration 1	Indels:	Errors iteration 2	Error rate iteration 2	Indels:
10×	100	0.0274%	4	116	0.0297%	18
20×	4	0.0010%	0	6	0.0015%	1
30x	2	0.0005%	0	0	0.0000%	0
40x	0	0.0000%	0	0	0.0000%	0
50×	0	0.0000%	0	0	0.0000%	0

Errors iteration 3	Error rate iteration 3	Indels:
110	0.0282%	7
7	0.0018%	0
0	0.0000%	0
0	0.0000%	0
0	0.0000%	0

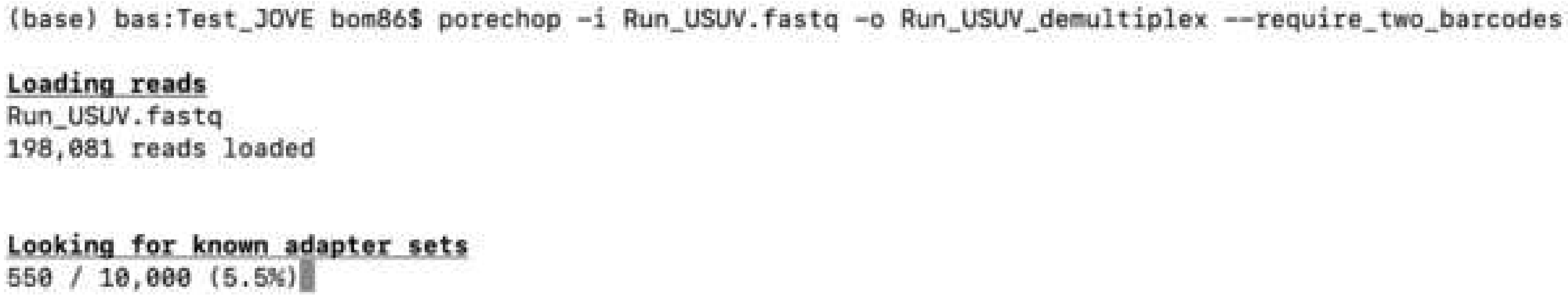
Name of Material/ Equipment	Company	Catalog Number	Comments/Description
Agencourt AMPure XP beads	Beckman Coulter	A63881	
dNTPs	Qiagen	201900	
FLO-MIN106 R10 flowcell	Nanopore	R10 flowcell	
KAPA Hyperplus libarary preparation kit	Roche	7962436001	
Library Loading Bead Kit	Nanopore	EXP-LLB001	
Ligation Sequencing Kit 1D	Nanopore	SQK-LSK109	
Native Barcoding Kit 1D 1-12	Nanopore	EXP-NBD103	
Native Barcoding Kit 1D 13-24	Nanopore	EXP-NBD104	
NEB Blunt/TA Ligase Master Mix	NEB	M0367S	
NEB Next Quick Ligation Module	NEB	E6056	
NEB Next Ultra II End Repair / dA-Tailing Module	NEB	E7546S	
Protoscript II Reverse Transcriptase	NEB	M0368X	
Q5 High-Fidelity polymerase	NEB	M0491	
Qubit dsDNA HS Assay kit	Thermo Fisher	Q32851	
Random Primers	Promega	C1181	
RNAsin Ribonuclease Inhibitor	Promega	N2111	

Dear editor,

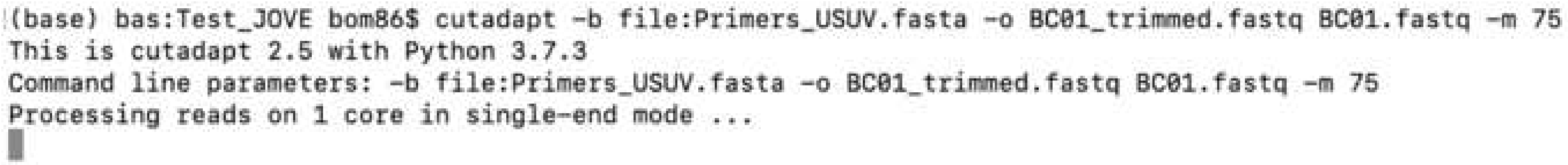
I have addressed the comments in the resubmitted file. However, the commercial language cannot completely be removed. The manuscript describes the comparison of two different sequence platform which has to be called by name in order to make it understandable for the reader.

Best regards,

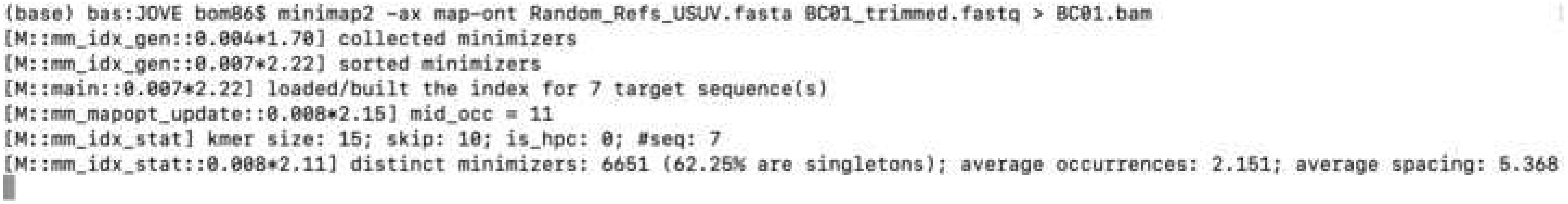
Bas Oude Munnink



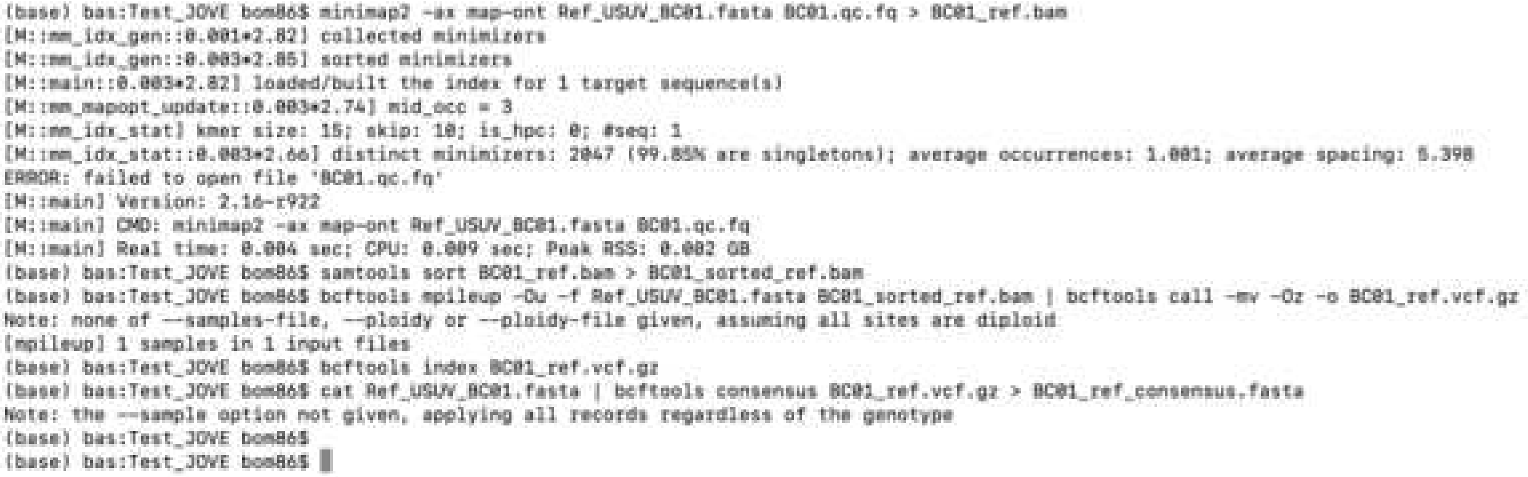
```
(base) bas:Test_JOVE bom86$ porechop -i Run_USUV.fastq -o Run_USUV_demultiplex --require_two_barcodes

Loading reads
Run_USUV.fastq
198,081 reads loaded


Looking for known adapter sets
550 / 10,000 (5.5%)
```



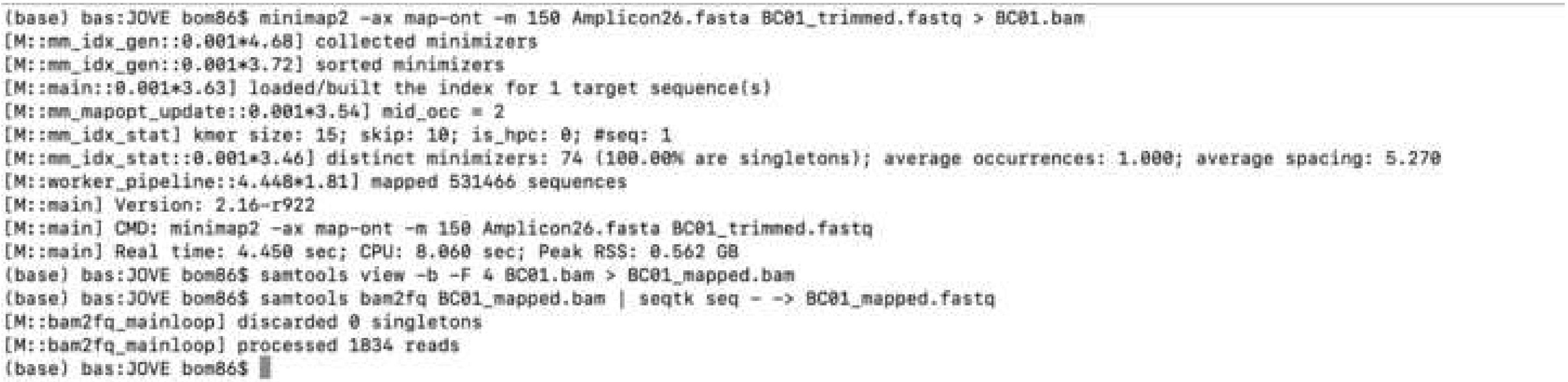
```
(base) bas:Test_JOVE bom86$ cutadapt -b file:Primers_USUV.fasta -o BC01_trimmed.fastq BC01.fastq -m 75
This is cutadapt 2.5 with Python 3.7.3
Command line parameters: -b file:Primers_USUV.fasta -o BC01_trimmed.fastq BC01.fastq -m 75
Processing reads on 1 core in single-end mode ...
█
```



```
(base) bas:JOVE bom86$ minimap2 -ax map-ont Random_Refs_USUV.fasta BC01_trimmed.fastq > BC01.bam
[M::mm_idx_gen::0.004*1.70] collected minimizers
[M::mm_idx_gen::0.007*2.22] sorted minimizers
[M::main::0.007*2.22] loaded/built the index for 7 target sequence(s)
[M::mm_mapopt_update::0.008*2.15] mid_occ = 11
[M::mm_idx_stat] kmer size: 15; skip: 10; is_hpc: 0; #seq: 7
[M::mm_idx_stat::0.008*2.11] distinct minimizers: 6651 (62.25% are singletons); average occurrences: 2.151; average spacing: 5.368
```

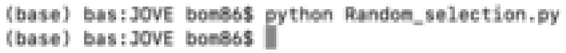


```
(base) bas:Test_JOVE bom86$ minimap2 -ax map-ont Ref_USUV_BC01.fasta BC01.qc.fq > BC01_ref.bam
[M::mm_idx_gen::0.001*2.82] collected minimizers
[M::mm_idx_gen::0.003*2.85] sorted minimizers
[M::main::0.003*2.82] loaded/built the index for 1 target sequence(s)
[M::mm_mapopt_update::0.003*2.74] mid_occ = 3
[M::mm_idx_stat] kmer size: 15; skip: 10; is_hpc: 0; #seq: 1
[M::mm_idx_stat::0.003*2.66] distinct minimizers: 2047 (99.85% are singletons); average occurrences: 1.001; average spacing: 5.398
ERROR: failed to open file 'BC01.qc.fq'
[M::main] Version: 2.16-r922
[M::main] CMD: minimap2 -ax map-ont Ref_USUV_BC01.fasta BC01.qc.fq
[M::main] Real time: 0.004 sec; CPU: 0.009 sec; Peak RSS: 0.002 GB
(base) bas:Test_JOVE bom86$ samtools sort BC01_ref.bam > BC01_sorted_ref.bam
(base) bas:Test_JOVE bom86$ bcftools mpileup -Ou -f Ref_USUV_BC01.fasta BC01_sorted_ref.bam | bcftools call -mv -Oz -o BC01_ref.vcf.gz
Note: none of --samples-file, --ploidy or --ploidy-file given, assuming all sites are diploid
[mpileup] 1 samples in 1 input files
(base) bas:Test_JOVE bom86$ bcftools index BC01_ref.vcf.gz
(base) bas:Test_JOVE bom86$ cat Ref_USUV_BC01.fasta | bcftools consensus BC01_ref.vcf.gz > BC01_ref_consensus.fasta
Note: the --sample option not given, applying all records regardless of the genotype
(base) bas:Test_JOVE bom86$
(base) bas:Test_JOVE bom86$ █
```



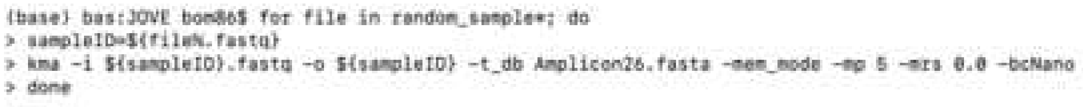
```
(base) bas:JOVE bom86$ minimap2 -ax map-ont -m 150 Amplicon26.fasta BC01_trimmed.fastq > BC01.bam
[M::mm_idx_gen::0.001*4.68] collected minimizers
[M::mm_idx_gen::0.001*3.72] sorted minimizers
[M::main::0.001*3.63] loaded/built the index for 1 target sequence(s)
[M::mm_mapopt_update::0.001*3.54] mid_occ = 2
[M::mm_idx_stat] kmer size: 15; skip: 10; is_hpc: 0; #seq: 1
[M::mm_idx_stat::0.001*3.46] distinct minimizers: 74 (100.00% are singletons); average occurrences: 1.000; average spacing: 5.270
[M::worker_pipeline::4.448*1.81] mapped 531466 sequences
[M::main] Version: 2.16-r922
[M::main] CMD: minimap2 -ax map-ont -m 150 Amplicon26.fasta BC01_trimmed.fastq
[M::main] Real time: 4.450 sec; CPU: 8.060 sec; Peak RSS: 0.562 GB
(base) bas:JOVE bom86$ samtools view -b -F 4 BC01.bam > BC01_mapped.bam
(base) bas:JOVE bom86$ samtools bam2fq BC01_mapped.bam | seqtk seq - -> BC01_mapped.fastq
[M::bam2fq_mainloop] discarded 0 singletons
[M::bam2fq_mainloop] processed 1834 reads
(base) bas:JOVE bom86$ █
```

JoVE60906_screenfile6



```
(base) bas:JOVE bom86$ python Random_selection.py
(base) bas:JOVE bom86$ 
```

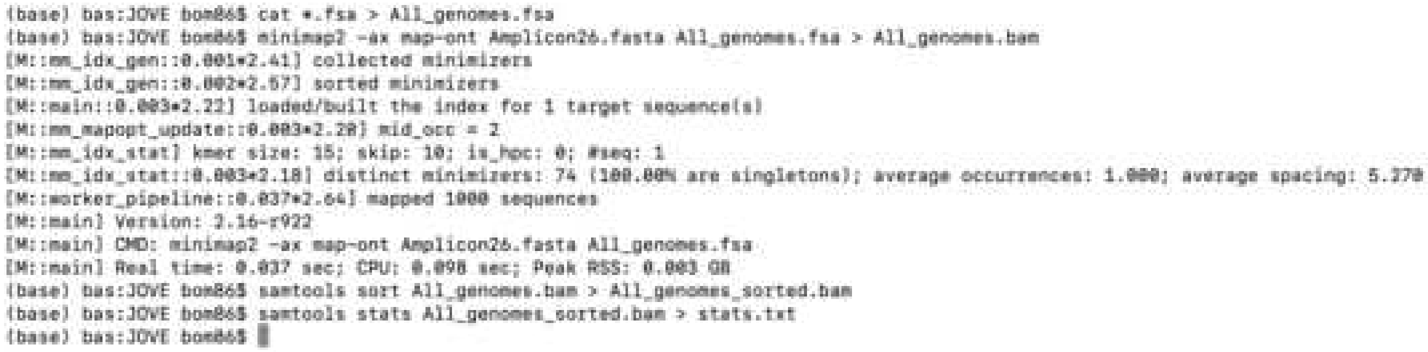
Click here to access/download;Supplemental File (Figures, Permissions, etc.);JoVE60906_screenfile7.png



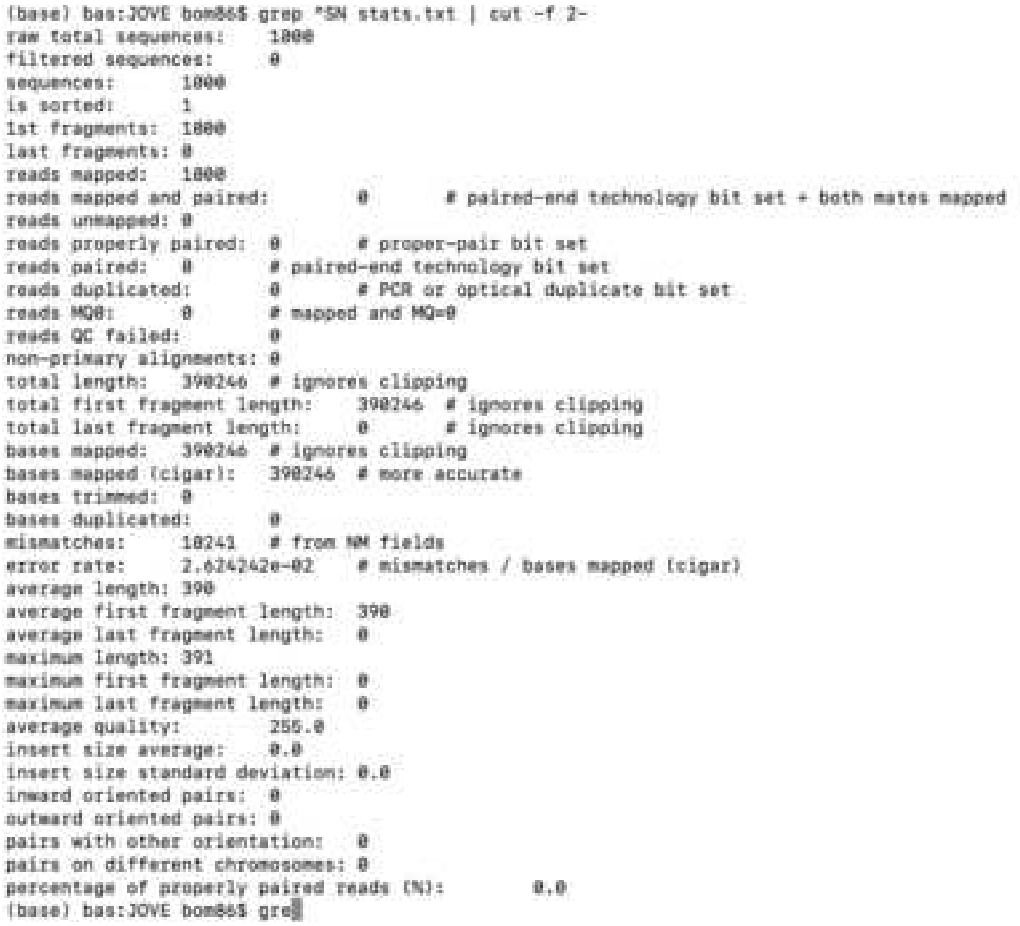
```
(base) bas:JOVE bom86$ for file in random_sample*; do
> sampleID=${file%.fastq}
> kma -i ${sampleID}.fastq -o ${sampleID} -t_db Amplicon26.fasta -mem_mode -mp 5 -mrs 0.0 -bcNano
> done
```

JoVE60906_screenfile8

Click here to access/download;Supplemental File (Figures, Permissions, etc.);JoVE60906_screenfile8.png

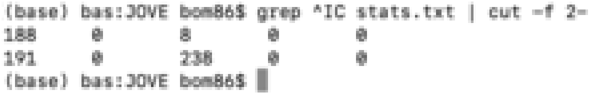


```
(base) bas:JOVE bom86$ cat *.fsa > All_genomes.fsa
(base) bas:JOVE bom86$ minimap2 -ax map-ont Amplicon26.fasta All_genomes.fsa > All_genomes.bam
[M::mm_idx_gen::0.001*2.41] collected minimizers
[M::mm_idx_gen::0.002*2.57] sorted minimizers
[M::main::0.003*2.22] loaded/built the index for 1 target sequence(s)
[M::mm_mapopt_update::0.003*2.20] mid_occ = 2
[M::mm_idx_stat] kmer size: 15; skip: 10; is_hpc: 0; #seq: 1
[M::mm_idx_stat::0.003*2.18] distinct minimizers: 74 (100.00% are singletons); average occurrences: 1.000; average spacing: 5.270
[M::worker_pipeline::0.037*2.64] mapped 1000 sequences
[M::main] Version: 2.16-r922
[M::main] CMD: minimap2 -ax map-ont Amplicon26.fasta All_genomes.fsa
[M::main] Real time: 0.037 sec; CPU: 0.098 sec; Peak RSS: 0.003 GB
(base) bas:JOVE bom86$ samtools sort All_genomes.bam > All_genomes_sorted.bam
(base) bas:JOVE bom86$ samtools stats All_genomes_sorted.bam > stats.txt
(base) bas:JOVE bom86$ ▌
```



```
(base) bas:JOVE bom86$ grep ^SN stats.txt | cut -f 2-
raw total sequences:    1000
filtered sequences:     0
sequences:      1000
is sorted:      1
1st fragments:  1000
last fragments: 0
reads mapped:   1000
reads mapped and paired:        0       # paired-end technology bit set + both mates mapped
reads unmapped: 0
reads properly paired:  0       # proper-pair bit set
reads paired:   0       # paired-end technology bit set
reads duplicated:       0       # PCR or optical duplicate bit set
reads MQ0:      0       # mapped and MQ=0
reads QC failed:        0
non-primary alignments: 0
total length:   390246  # ignores clipping
total first fragment length:    390246  # ignores clipping
total last fragment length:     0       # ignores clipping
bases mapped:   390246  # ignores clipping
bases mapped (cigar):   390246  # more accurate
bases trimmed:  0
bases duplicated:       0
mismatches:     10241   # from NM fields
error rate:     2.624242e-02    # mismatches / bases mapped (cigar)
average length: 390
average first fragment length:  390
average last fragment length:   0
maximum length: 391
maximum first fragment length:  0
maximum last fragment length:   0
average quality:        255.0
insert size average:    0.0
insert size standard deviation: 0.0
inward oriented pairs:  0
outward oriented pairs: 0
pairs with other orientation:   0
pairs on different chromosomes: 0
percentage of properly paired reads (%):        0.0
(base) bas:JOVE bom86$ gre
```

JoVE60906_screenfile10



```
(base) bas:JOVE bom86$ grep ^IC stats.txt | cut -f 2-
188     0       8       0       0
191     0       238     0       0
(base) bas:JOVE bom86$ 
```

Suppl File 1-Script for the random selection