DNA Diet Methods Pipeline Comparison Supplement

Ana Miller-ter Kuile

5/15/2020

# DNA Diet Methods Pipeline Comparison Analyses

## Motivation

This project creates a workflow for analyzing the diets of invertebrate predators using high throughput sequencing of gut contents. This method provides a promising way to get highly-resolved diet data from consumers across ecosystems.

**Bioinformatics challenge**: To extract all possible prey items from predator guts when predator and prey are taxonomoically similar, it is best practice to use a set of PCR primers that target all possible prey. However, a side effect of this is that these primers will also end up amplifying a large amount of predator DNA. As a result, these datasets are dominated by predator DNA, and so detecting relatively rare prey sequences in these datasets is key. As the molecular ecology field moves toward using amplicon sequence variants (ASVs) as biologically-real units of biodiversity in high throughput datasets, these types of datsets dominated by predator DNA are even more challenging since these ASV clustering pipelines use sequence abundance as a way to cluster sequences into similar, biologically-real groups of sequences. Therefore, any clustering pipeline used for DNA diet data dominated by the predator must

* detect prey sequences that are taxonomically similar to the DNA of predators and
* detect prey sequences that are relatively rare compared to the DNA of predators

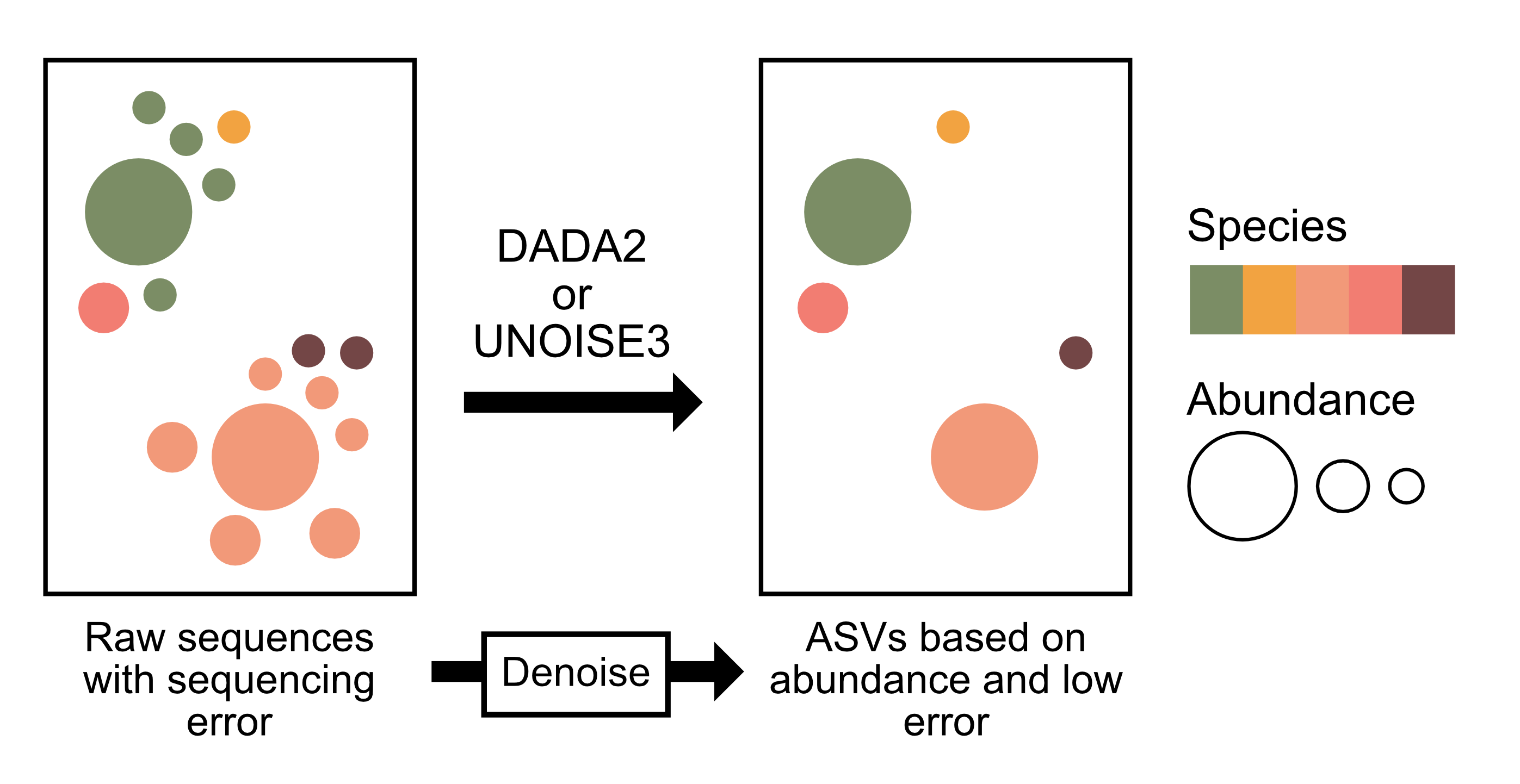


Figure 1: Denoising pipelines (including DADA2 and UNOISE3) consider sequence abundance and error rate in raw sequence datasets when removing sequences that have high error and low abundance and combining their reads with those of their most similar neighbor sequence with high abundance and low error (amplicon sequence variants or ASVs). Because our dataset has high abundance predator DNA with lower abundance prey DNA, any denoising pipeline we use must accurately be able to distinguish low abundance prey ASVs from high abundance predator ASVs.

This supplement assesses how different denoising/clustering pipelines perform with these types of datasets and provide a template for other studies interested in interactions between invertebrate predators and prey, but with the idea that the same sort of process could work well for other study systems (i.e. vertebrates that eat vertebrates) as well.

## The data

The data for this project consists of 56 individuals of the spider species *Heteropoda venatoria* from Palmyra Atoll National Wildlife Refuge. We collected 37 of these individuals in the field in 2015 and immediately froze them in a -80C freezer until processing in 2019. The remaining 19 individuals we collected in 2017, starved in mesocosms in the lab for 24 hours, and then fed a known diet item (the grasshopper *Oxya japonica*). After spiders had been in mesocosms with prey items and eaten them (12-24 hours), we froze them at -20C. Prior to final preservation in 80% EtOH, we surface sterilized 8 of these 19 individuals by submerging them in a 10% bleach to DI water solution for 2 minutes and then washing them in DI water for 2 minutes. We performed the same sterilization protocol on 18 of the 37 field-collected individuals after we had brought them to the lab from the field (in 2019). (see Methods document for full extraction, PCR, and sequencing protocols).

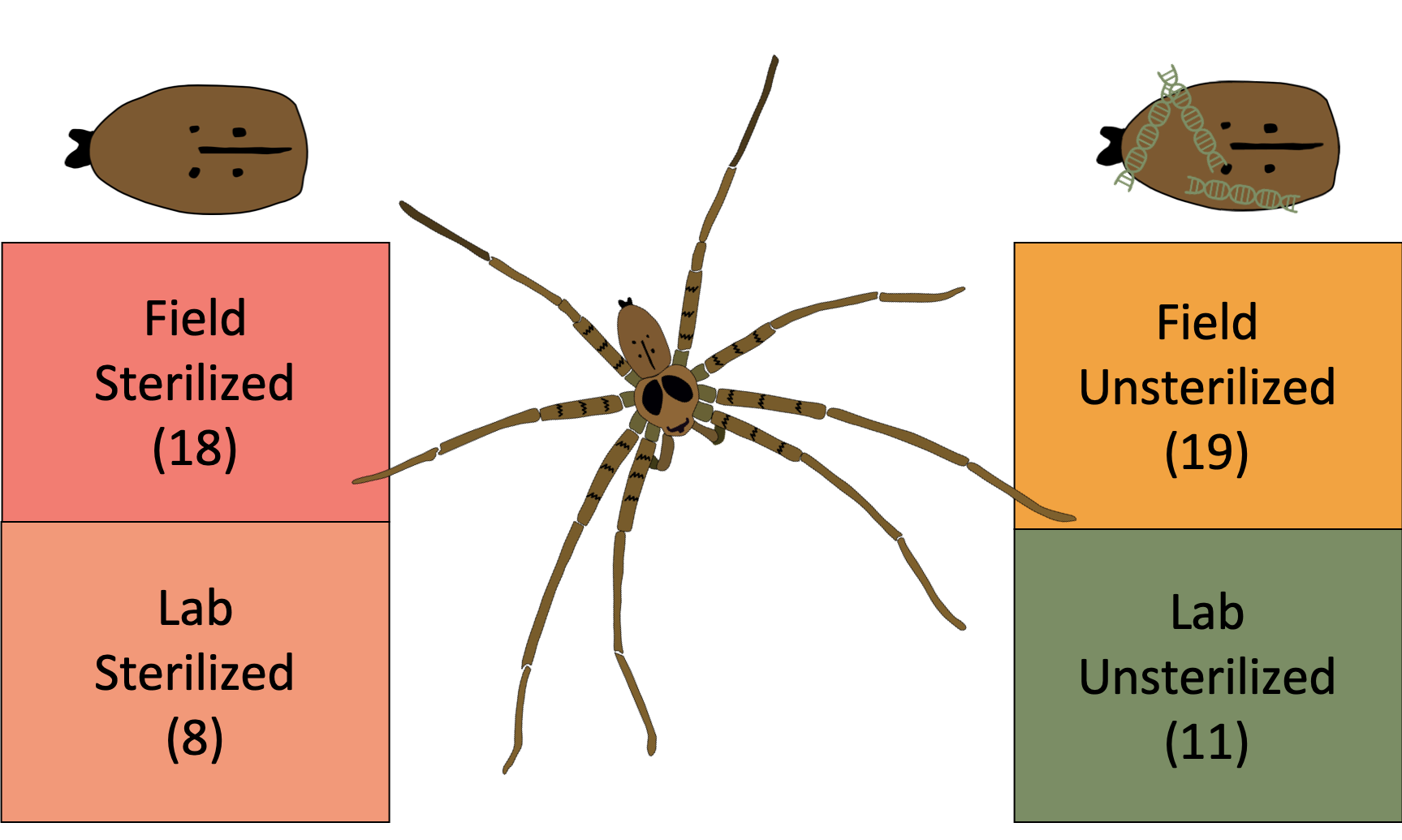


Figure 3: Our surface-contamination experiment included four groups of varying sample sizes: both field-collected and lab-fed sterilized individuals and unsterilized individuals.

We ran all samples on an Illumina MiSeq platform at the UCSB Genetics Core. We cleaned low-quality sequences from this set of ~15 million sequences returned from the genetics core, and then used two clustering pipelines (DADA2 and UNOISE3) that use sequence abundance and quality to combine similar sequences (which are often different because of errors in amplification or sequencing and actually represent the same sequence) into clusters representing the most common real sequence. We performed each of these clustering pipelines twice: once on the complete list of sequences, and once on a set of sequences that had been “cleaned” of predator and non-diet sequences using the program BBSplit. After the denoising/clustering process, we are left with a matrix very similar to any community matrix, with columns corresponding to each sample (spider individual) and rows corresponding to ASVs (sequences) in that individual. In this way, each spider becomes functionally a “community” of species which can be treated very similarly to any other matrix of community data (i.e. in vegan with multidimensional analyses, etc).

## X HEV.100\_S22 HEV.101\_S23 HEV.102\_S1 HEV.103\_S2 HEV.104\_S3 HEV.105\_S4  
## 1 ASV\_1 177549 219335 186660 199570 134902 145046  
## 2 ASV\_2 0 0 0 0 0 0  
## 3 ASV\_3 0 0 0 0 0 0  
## 4 ASV\_4 521 0 0 0 15393 0  
## 5 ASV\_5 5 0 0 0 0 0  
## 6 ASV\_6 0 0 0 0 0 0  
## 7 ASV\_7 0 0 32 0 0 0  
## 8 ASV\_8 0 0 0 80 0 8  
## 9 ASV\_9 0 0 0 0 16 0  
## 10 ASV\_10 0 0 0 0 0 0

## Outline of pipeline performance

* *Clustering specificity*: Positive and negative controls map to fewer ASVs
* *Total ASVs (richness)*: A greater number of ASVs means a higher likelihood the full diversity of the sample has been detected
* *Number of prey ASVs (diet richness)*: A higher diet richness means the pipeline better distinguishes prey ASVs from predator
* *Phylogenetic diversity of prey ASVs*: A better pipeline will detect a broader phylogenetic diversity of prey
* *Prey read abundance*: More prey reads means a higher likelihood of picking up rarer prey items
* *Percent of ASVs that are prey*: This corrects for variable sequencing depth and a higher percentage means a higher likelihood that all prey are detected
* *Percent of reads that are prey*: Again, correcting for sequencing depth, a higher percentage means a higher likelihood that all prey are detected
* *Amount of known diet reads*: A better pipeline will be better at detecting a diet item we KNOW to be present in all samples

## Want just the highlights? Jump to the [summary](#pipelines)

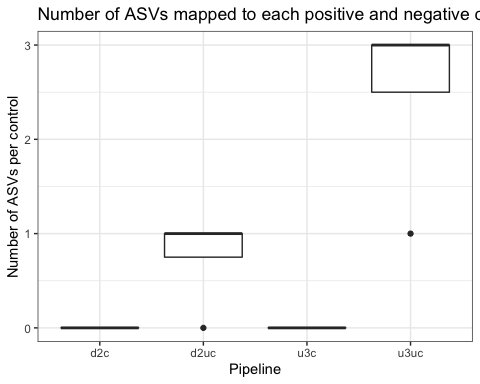
## Pipeline performance

#non-zero reads should only be in the positive control samples:  
u3\_pos\_ct <- u3\_uc %>%  
 filter(ASV %in% c("Zotu4", "Zotu2", "Zotu3")) %>%  
 gather(sample, reads, CL1:QC1) %>%  
 filter(reads > 0) %>%  
 mutate(type = "u3")  
  
############################  
#Justify UNOISE vs. DADA2####  
############################  
#also, in justifying UNOISE, want to check that the positives got more reads  
#NOTE: PUT THIS IN THE PIPELINE COMPARISON SCRIPT!!!!####  
d2 <- read.csv(here("data", "denoised\_data", "ASV\_tables",   
 "dada2\_uc\_asv\_tab.tsv"), sep = "\t")   
  
colnames(d2) <- sapply(str\_split(colnames(d2), "\_"), function(x){return(x[[1]])})  
d2 <- rename(d2, "ASV" = "X")  
  
d2\_pos\_cont <- d2\_uc %>%  
 filter(ASV %in% c("ASV\_10", "ASV\_2", "ASV\_16")) %>%  
 gather(sample, reads, CL1:QC1) %>%  
 filter(reads > 0) %>%  
 mutate(type = "d2")  
  
positive <- u3\_pos\_ct %>%  
 bind\_rows(d2\_pos\_cont) %>%  
 group\_by(sample, type) %>%  
 summarise(reads = sum(reads))  
  
pos\_reads <- positive %>%  
 group\_by(type) %>%  
 summarise(mean = mean(reads), sd =sd(reads), total = n(), se = sd/sqrt(total)) %>%  
 ggplot(aes(x = type, y = mean)) +  
 geom\_bar(stat = "identity", position = "dodge") +  
 theme\_bw() +  
 labs(x = "Pipeline", y ="Average reads per positive control") +  
 geom\_errorbar(aes(ymin = mean -se, ymax = mean+se))

We looked at a bunch of different ways the pipelines performed both bioinformatically and ecologically. If you don’t want to spend time on each one, you can jump to the pipeline performance [summary table](#pipelines), which provides the scores of each pipeline in each measure of performance.

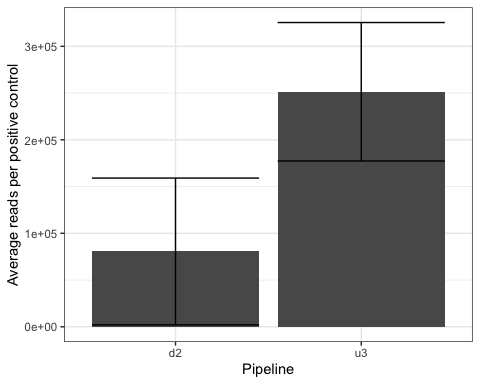
## a. Clustering specificity (positive and negative control ASVs)

Different clustering methods may have more or less specificity in their clustering. We checked for this by running several positive control samples and one negative control sample and looking at how different clustering pipelines assigned ASVs to these controls. The positive controls are cloned fungal species which Austen uses in his work. Because they are clones, they should map to one or very few ASVs. The negative controls were samples we ran through PCRs with no DNA in them and quantified to zero before submitting. Therefore, these should map to zero ASVs. **Takeaway**: smaller is better for both positive and negative control ASV counts.



It seems that, first of all, cleaning removes all ASVs for the controls, which we would expect, since we filtered these based on known prey and predator ASVs. Furthermore, it seems that DADA2 does a better job of assigning fewer ASVs to positive controls than UNOISE3. Furthermore, the negative control had zero reads assigned to any reads in DADA2, while UNOISE3 assigned one ASV a value of 1 read for the negative control.

On the other hand, when we look at how many sequence reads each control received (indicative of the pipeline to match DNA to denoised clusters), it looks like UNOISE3 does a much better job on average of mapping more sequence reads to positive controls:

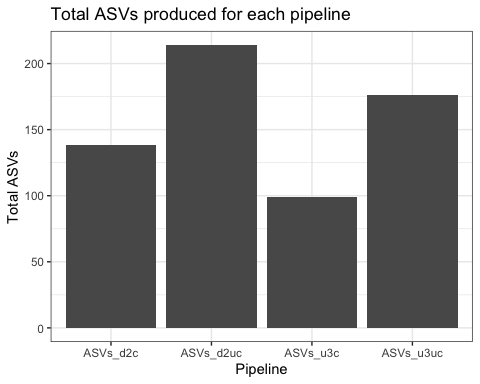


**Winner: UNOISE uncleaned**

## b. Total ASVs (richness)

A good clustering pipeline will be able to pick up all the diversity in a dataset. In this case, a higher number of total ASVs (roughly, species richness) means better clustering method, since we want to capture the greatest diversity possible. However, we will go into some more specific measures of richness (prey richness) next, since just the total number of ASVs may not actually be prey richness, but may instead be the pipeline clustering more predator or non-prey ASVs.

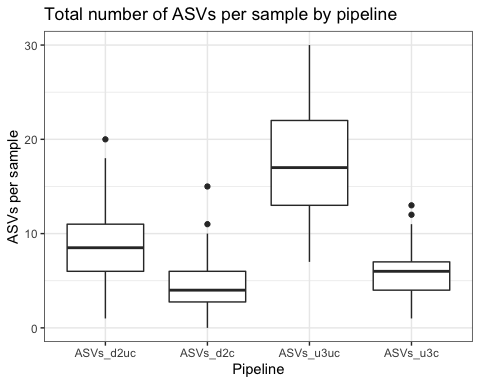
We looked at both total ASVs produced by each pipeline as well as the number of ASVs each clustering pipeline assigned to each sample.



The uncleaned datasets produce more ASVs, and DADA2 produces more total ASVs than UNOISE3.

**Winner: DADA2**

When we look at the total number of ASVs produced per sample, we see a different pattern, however:

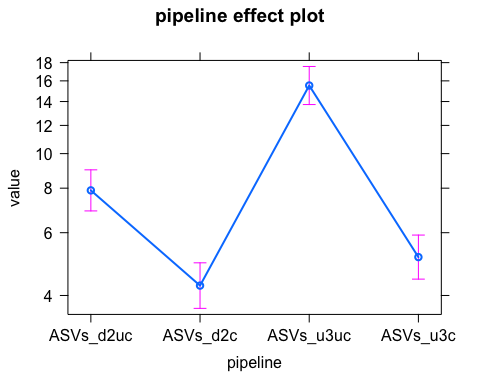


Indeed, these differences among pipelines are statistically significant based on a random effects model with a full model structure of (comparing it to a null without pipeline, and then estimating marginal means between each pair):

tot\_ASV\_mod <- glmmTMB(value ~ pipeline + (1|sample),  
 data = by\_sample\_long,  
 family = "genpois")

With significant pairwise differences between all pipelines:

## contrast estimate SE df t.ratio p.value  
## ASVs\_d2uc - ASVs\_d2c 0.615 0.0593 218 10.367 <.0001   
## ASVs\_d2uc - ASVs\_u3uc -0.677 0.0434 218 -15.595 <.0001   
## ASVs\_d2uc - ASVs\_u3c 0.431 0.0564 218 7.644 <.0001   
## ASVs\_d2c - ASVs\_u3uc -1.292 0.0540 218 -23.921 <.0001   
## ASVs\_d2c - ASVs\_u3c -0.184 0.0648 218 -2.839 0.0254   
## ASVs\_u3uc - ASVs\_u3c 1.108 0.0507 218 21.881 <.0001   
##   
## Results are given on the log (not the response) scale.   
## P value adjustment: tukey method for comparing a family of 4 estimates



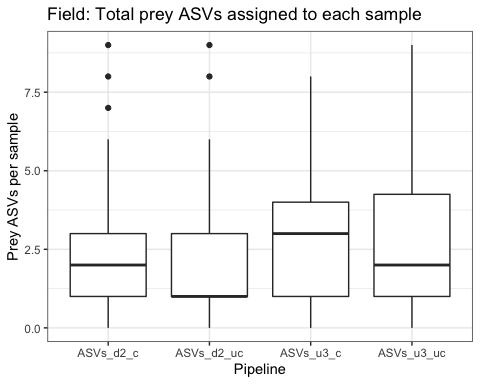
The takeaway here is that both UNOISE3 unclean and clean produce more ASVs than their DADA2 counterparts.

**Winner: UNOISE3 uncleaned and cleaned**

## c. Number of prey ASVs (diet richness)

A higher total number of ASVs does not necessarily mean that a clustering pipeline is the best option, since these newly discovered ASVs may not actually be ASVs of interest, but rather ASVs of predators or other non-prey items (contamination, for example). Therefore, a better measure of whether a pipeline works best for the purposes of this study is to look at how well it clusters prey ASVs. We did this by combining taxonomic data to our ASV list, and assigning broad taxonomic categories to these ASVs. We combined taxonomic data from both GenBank (using BLAST and MEGAN) and from the BOLD IDEngine database. We combined taxonomic assignments from both sources and removed any assignment which did not match between these two databases. Anything that mapped to the predator Family, Genus, or Species was given a taxonomic category of “predator”, anything that could possibly be prey (including arthropods and vertebrates in this dataset) were given a taxonomic category of “prey”. Anything that was not given a specific taxonomic assignment but which could have been prey or predator was given the category of “unknown” (i.e. a BLAST taxonomic assignment of “Arthropoda”). There were other items in the dataset that were definitely not predators or prey (mostly fungi) and these were given a taxonomic assignment of “non-prey”. Our final category included any ASV that was clustered but which was not assigned a taxonomy; this category was “no hit”.

We then subset just the prey ASVs from this dataset and compared the per sample number of prey ASVs assigned by each pipeline, which is visualized here:

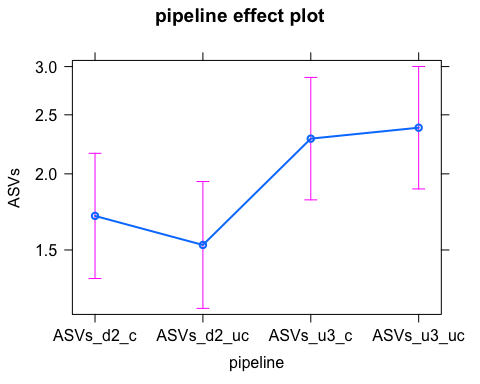


We again looked for significant across group differences in prey ASVs using a mixed model (comparing it to a null without pipeline, and then estimating marginal means between each pair):

lme\_prey\_ASVs <- glmmTMB(ASVs ~ pipeline + (1|sample),  
 data = taxa\_ASVs\_prey,  
 family = "genpois")

And looked at pairwise differences between these groups:

## contrast estimate SE df t.ratio p.value  
## ASVs\_d2\_c - ASVs\_d2\_uc 0.1098 0.0707 218 1.552 0.4084   
## ASVs\_d2\_c - ASVs\_u3\_c -0.2920 0.0677 218 -4.312 0.0001   
## ASVs\_d2\_c - ASVs\_u3\_uc -0.3334 0.0675 218 -4.936 <.0001   
## ASVs\_d2\_uc - ASVs\_u3\_c -0.4018 0.0699 218 -5.753 <.0001   
## ASVs\_d2\_uc - ASVs\_u3\_uc -0.4432 0.0693 218 -6.399 <.0001   
## ASVs\_u3\_c - ASVs\_u3\_uc -0.0414 0.0615 218 -0.672 0.9074   
##   
## Results are given on the log (not the response) scale.   
## P value adjustment: tukey method for comparing a family of 4 estimates



What we see is that cleaning does not significantly increase the number of ASVs assigned to prey items for either DADA2 or UNOISE3. However, what we see is that UNOISE3 assigns more prey ASVs to each sample, suggesting that UNOISE3 is better at detecting a greater prey richness than DADA2

**Winner: UNOISE unclean**

## d. Phylogenetic diversity of prey ASVs

Not only might one pipeline be better at detecting total prey richness, but might also be able to cluster a broader phylogenetic diversity of prey items. This might be especially important in systems where predators are generalists and feed on organisms across broad taxonomic groups. Because the number of ASVs assigned to prey per sample is fairly low (less than 10 per sample) and because we are interested in just detection here and not necessarily abundance variations among samples, we will just qualitatively assess Faith’s PD for each pipeline, summing the prey ASVs for each pipeline. Furthermore, we will combine phylogenetic analyses at the Order and Family level, since we have good taxonomic representation at these levels (15 orders, 19 families)

UNOISE3 unclean picks up the greatest phylogenetic diversity. It is important to note that this is in spite of the fact that order richness is greater in the DADA2 dataset. UNOISE3 unclean has the lowest phylogenetic diversity, which is probably because we trained the cleaning program with DADA2 and some of the phylogenetic diversity of the UNOISE3 unclean dataset was removed. Based on the phylogenetic tree, DADA2 picked up Lepidoptera (moths and butterflies) and Sarcoptiformes (mites), while UNOISE3 picked up Squamata (geckos).

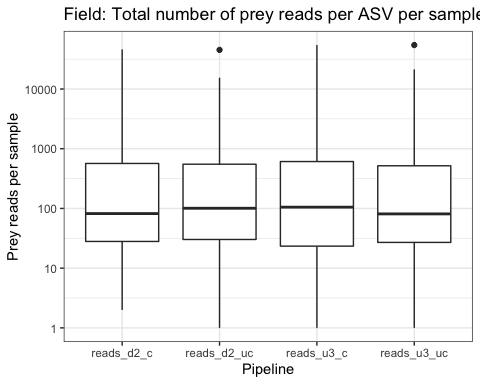
Again, UNOISE3 unclean picks up the greatest phylogenetic diversity. It is important to note that this is in spite of the fact that, again, family richness is greater in the DADA2 dataset. UNOISE3 unclean has the lowest phylogenetic diversity, which is probably because we trained the cleaning program with DADA2 and some of the phylogenetic diversity of the UNOISE3 unclean dataset was removed. DADA2 picks up Formicidae (ants) and Suidasiidae (mites), while UNOISE3 picks up Xiphydridae (wasps) and Gekkoniidae (geckos).

**Winner: UNOISE unclean**

## e. Prey read abundance

Similar to the number of ASVs assigned to prey, the number of sequence reads assigned to each of these ASVs is important because rarer prey items are more likely to be detected the greater abundance each prey ASV is in each sample.

We can visualize the number of reads of each ASV in each sample:



And again, we ran a mixed model to determine whether there are significant among pipeline detections in prey reads (comparing it to a null without pipeline, and then estimating marginal means between each pair):

read\_mod <- glmmTMB(reads ~ pipeline + (1|sample),  
 data = taxa\_reads\_prey,  
 family = "genpois")

pairs(model.emm\_preyreads)

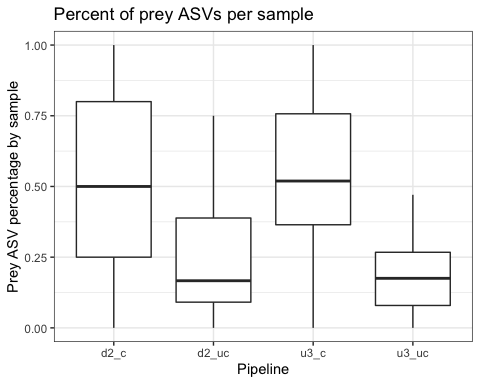
## contrast estimate SE df t.ratio p.value  
## reads\_d2\_c - reads\_d2\_uc 0.03215 0.0355 218 0.905 0.8021   
## reads\_d2\_c - reads\_u3\_c -0.20844 0.0335 218 -6.221 <.0001   
## reads\_d2\_c - reads\_u3\_uc -0.20328 0.0336 218 -6.052 <.0001   
## reads\_d2\_uc - reads\_u3\_c -0.24060 0.0339 218 -7.089 <.0001   
## reads\_d2\_uc - reads\_u3\_uc -0.23543 0.0339 218 -6.937 <.0001   
## reads\_u3\_c - reads\_u3\_uc 0.00517 0.0317 218 0.163 0.9985   
##   
## Results are given on the log (not the response) scale.   
## P value adjustment: tukey method for comparing a family of 4 estimates

Cleaning did not significantly increase prey read abundance. UNOISE3 produced more prey reads per ASV per sample than DADA2

**Winner: UNOISE3 unclean**

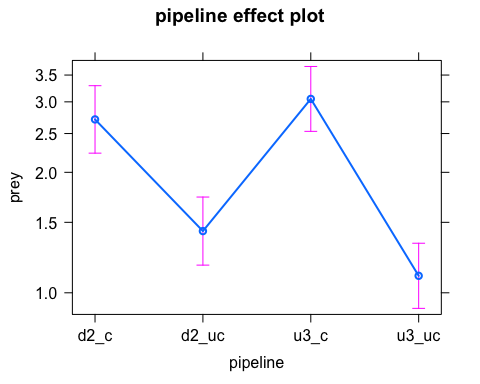
## f. Percent of ASVs that are prey

So far, we have been assuming that the total number of ASVs is independent of sequencing depth, which we know is not the case. Therefore, we ran the above ASV model again correcting for sequencing depth (using an offset of the total number of ASVs). These data can then be interpreted as the percent of all ASVs assigned to prey.



prey\_prop\_mod <- glmmTMB(prey ~ pipeline + (1|sample),  
 data = ASV\_totals\_samples,  
 family = "genpois",  
 offset = log(tot))

## contrast estimate SE df t.ratio p.value  
## d2\_c - d2\_uc 0.642 0.0661 215 9.719 <.0001   
## d2\_c - u3\_c -0.118 0.0638 215 -1.851 0.2525   
## d2\_c - u3\_uc 0.900 0.0631 215 14.249 <.0001   
## d2\_uc - u3\_c -0.760 0.0638 215 -11.921 <.0001   
## d2\_uc - u3\_uc 0.257 0.0629 215 4.090 0.0004   
## u3\_c - u3\_uc 1.018 0.0562 215 18.106 <.0001   
##   
## Results are given on the log (not the response) scale.   
## P value adjustment: tukey method for comparing a family of 4 estimates

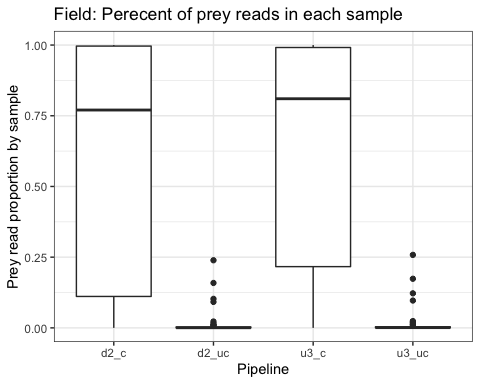


Unsurprisingly, cleaning increases the total proportion of prey ASVs in each sample. DADA2 has a higher proportion of prey ASVs than UNOISE3 for both clean and uncleaned datasets.

**Winner: DADA2 clean**

## g. Percent of reads that are prey

Again, we have been assuming that the total number of reads is independent of sequencing depth, which we know is not the case. Therefore, we ran the above reads model again correcting for sequencing depth (using an offset). These data can then be interpreted as the percent of all reads assigned to prey.



prey\_rprop\_mod <- glmmTMB(prey ~ pipeline + (1|sample),  
 data = read\_totals\_samples,  
 family = "genpois",  
 offset = log(tot))

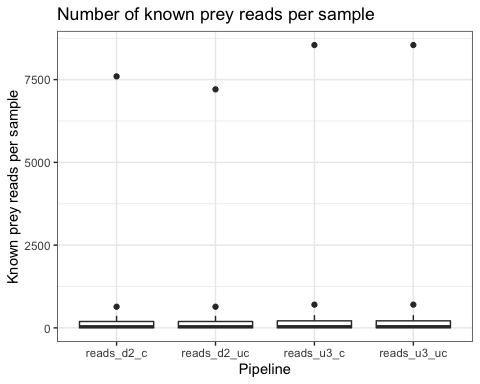
Cleaning increased the total proportion of reads that are assigned to prey ASVs. There is no difference in the proportion of prey reads in samples for either DADA2 or UNOISE3.

**Winner: cleaned datasets**

## h. Amount of known diet reads

We know we fed lab organisms a specific diet item (*Oxya japonica*), and so we should trust the pipeline that detects this prey item better.

Let’s look at detection across samples of this prey item (we subset our dataset to just be those that we fed the item to):

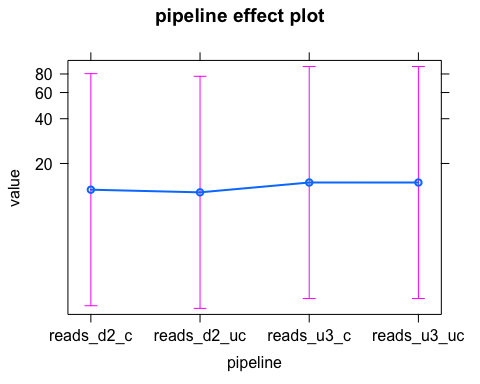


Again, we built a mixed model that we compared to a null model:

known\_reads\_mod <- glmmTMB(value ~ pipeline + (1|sample),  
 data = known\_reads\_long,  
 family = "genpois")

And looked at pairwise differences:

## contrast estimate SE df t.ratio p.value  
## reads\_d2\_c - reads\_d2\_uc 0.040970 0.00610 70 6.712 <.0001   
## reads\_d2\_c - reads\_u3\_c -0.111342 0.00588 70 -18.935 <.0001   
## reads\_d2\_c - reads\_u3\_uc -0.111167 0.00588 70 -18.904 <.0001   
## reads\_d2\_uc - reads\_u3\_c -0.152312 0.00595 70 -25.616 <.0001   
## reads\_d2\_uc - reads\_u3\_uc -0.152137 0.00595 70 -25.586 <.0001   
## reads\_u3\_c - reads\_u3\_uc 0.000175 0.00571 70 0.031 1.0000   
##   
## Results are given on the log (not the response) scale.   
## P value adjustment: tukey method for comparing a family of 4 estimates



Cleaned datasets have significantly more reads of the known prey items per sample for DADA2, however cleaning did not increase detection of known prey for UNOISE3. UNOISE3 has better detection of known prey items.

## Pipeline Performance Summary

Compiling the results from this, we get a summary of all the measures of pipeline performance. In the case of a non-significant difference between pipelines, the pipeline with the most parsimonious protocol (i.e. no cleaning) will win. In the case that the parsimony is the same between pipelines, none is considered to be outperforming. (in the following table, an “X” designates the best performing pipeline, either by significant difference between groups or because it is the most parsimonious of a tie, “NS” signifies that these pipelines performed equally well to the winning, parsimonious pipeline, and “TIE” signifies that these pipelines performed equally well and have the same parsimony.)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Measure/Pipeline | DADA2 UC | UNOISE3 UC | DADA2 C | UNOISE3 C |
| Positive control ASVs |  | X |  |  |
| Negative conotrol ASVs | X |  | NS | NS |
| Total ASVs | X |  |  |  |
| ASVs per sample |  | X |  | NS |
| Prey ASVs |  | X |  |  |
| Prey reads |  | X |  |  |
| Prey ASV % |  |  | X |  |
| Prey read % |  |  | TIE | TIE |
| Known Diet |  | X |  | NS |
| NA | NA | NA | NA | NA |
| NA | NA | NA | NA | NA |

From this, we see that in most cases, the uncleaned pipelines perform equally well or better than the cleaned pipelines. So cleaning does not add much to our bioinformatic performance nor our ecological inference from this dataset. As a result, for our analyses of sterilized vs. unsterilized individuals (both field and lab), the best options in terms of time are both unclean DADA2 and UNOISE3. The “winner” between these two pipelines is unclear - UNOISE3 outperformed ecologically (providing more diet data to work with); conversely, DADA2 outperformed bioinformatically (controls mapped more accurately).

#### Takeaway: either UNOISE3 or DADA2 will give you good diet data with good specificity. The choice, then, may be dependent on access rather than performance. While DADA2 is open access and user-friendly in R, it takes longer to run than UNOISE3; conversely, while UNOISE3 runs much more quickly, it is only free in a 32-bit version that does not run on some new operating systems (e.g. 64-bit Mac Catalina OS). These and other costs-benefits are reviewed more thoroughly in Nearing et al. (2018)