**Prerequisite environment & files:**

**Environment:**

1. Blastx: <https://blast.ncbi.nlm.nih.gov/Blast.cgi?CMD=Web&PAGE_TYPE=BlastDocs&DOC_TYPE=Download>
2. Augustus: <http://bioinf.uni-greifswald.de/augustus/binaries/>
3. Edirect: [https://dataguide.nlm.nih.gov/edirect/install.html](%20https:/dataguide.nlm.nih.gov/edirect/install.html)
4. BioPerl: <http://bioperl.org/INSTALL.html>

**Files:**

1. **probe.fas**: the file contains all kinds of opsin protein sequences, including 5 LW, 7 B, 7 UV, 6 Rh7, 6 Copsin, 6 GO/RGR opsins. Used for blastp against Proteome or tblastn against Genome.
2. **protein\_profile.prf1**: the file contains protein profile for opsin. Generated from alignment of probe.fas, using ‘msa2prfl.pl probe.aln > fam.prfl’. Used for gene predicting when running Augustus.
3. **fasta-splitter.pl**: split large fasta file to multiple small fasta files for online blast.

**Workflow:**

**Proteome:**

1. Download Proteome data from “NCBI Assembly” or “Ensemble” (using Edirect).
2. Make protein blast library.
3. Blastp probe.fas against the library (evalue=1e-20)
4. Extract seqid of potential opsin sequences from blastp output files.
5. Blast potential opsin sequences against online database, exclude incorrect sequences or pseudogenes.
6. Align new-found opsin sequences together with opsin sequences in probe.fas, construct a tree.
7. Determine number and type of new-found opsin genes.

**Genome:**

1. Download Genome data from “NCBI Assembly” or “Ensemble” (using Edirect).
2. Make nucleotide blast library.
3. tblastn probe.fas against the library.
4. Extract information from tblastn output files, including seqid, start and end position of matching part of subject sequence (usually the subject sequences are assembled scaffolds or contigs, which is very long)
5. Efetch the partial sequences base on the information (expanding the original matching range to 10000 bp upstream and 10000 bp downstream in order to include the complete gene) from NCBI
6. Run Augustus with input of protein profile (protein\_profile.prf1) against the fetched partial genomic file to predict proteins.
7. Extract predicted protein sequences form Augustus output file.
8. Make protein blast library using predicted protein sequences.
9. Blastp probe.fas against the library (evalue=1e-20)
10. Extract seqid of potential opsin sequences from blastp output files.
11. Blast potential opsin sequences against online database, exclude incorrect sequences or pseudogenes.
12. Align new-found opsin sequences together with opsin sequences in probe.fas, construct a tree.
13. Determine number and type of new-found opsin genes.

**Problems still need to address:**

1. Hard to tell whether the new-found opsin sequences are actual real genes or pseudogenes.
2. Might get partial opsin sequences, 3 possibilities: 1) Augustus failed to predict upstream or downstream exon. 2) The sequencing quality is not so good that a gap appears at the range of the opsin gene. 3) That is debris of opsin that had lost the function early in evolution. How to tell?