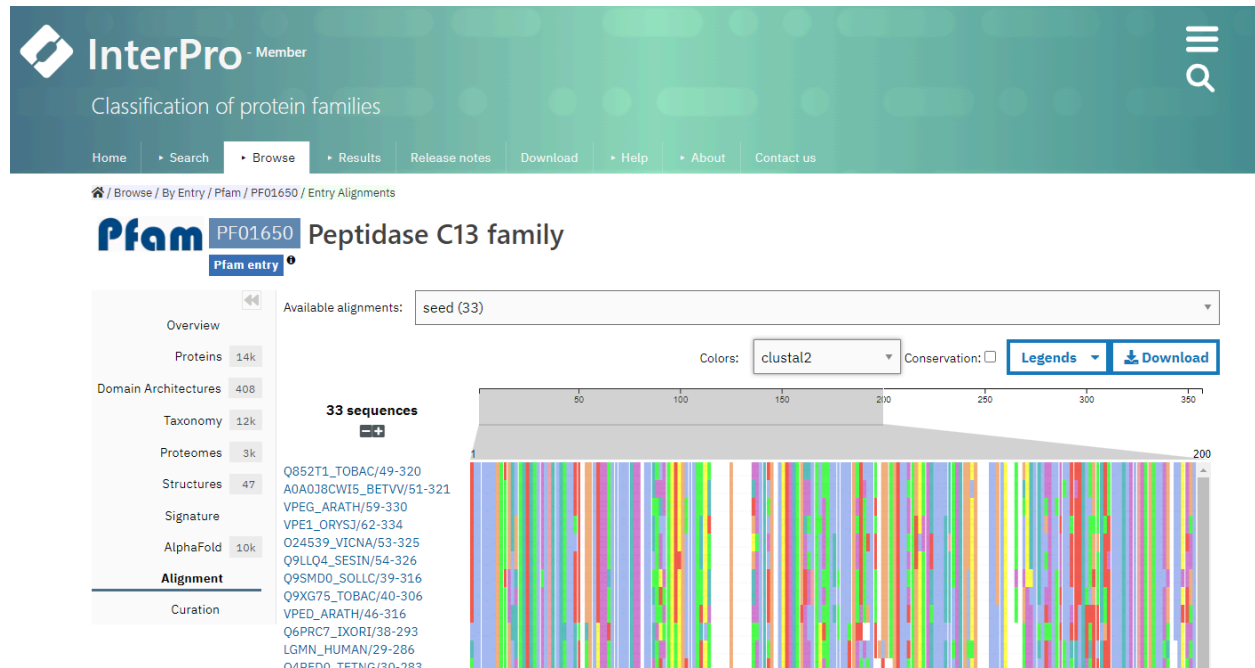


## 1. Collect Pfam “seed” entry:

Select the Pfam entry of your protein of interest. Go to the “Alignment” tab, and from “Available alignments” select the “seed” alignment. Download the seed alignment in clustal2 format. For this tutorial, Peptidase C13 was used as an example protein.



The screenshot shows the Pfam website interface for the Peptidase C13 family (PF01650). The page includes a navigation bar with links like Home, Search, Browse, Results, Release notes, Download, Help, About, and Contact us. The main content area displays the Pfam logo, the family name 'Peptidase C13 family', and the entry ID 'PF01650'. A sidebar on the left lists various categories: Overview, Proteins (14k), Domain Architectures (408), Taxonomy (12k), Proteomes (3k), Structures (47), Signature, AlphaFold (10k), Alignment, and Curation. The 'Alignment' tab is selected, showing 'Available alignments: seed (33)'. A 'Colors: clustal2' dropdown and a 'Conservation: [ ]' checkbox are visible. A 'Download' button is present. The main area displays '33 sequences' and a corresponding sequence alignment visualization. The alignment is shown in a color-coded format, with a scale bar indicating positions from 50 to 350. The sequences are listed on the left, and the alignment is shown on the right.

## 2. Use the “Visual” sequence specific selection pairwise sequencing code:

To find the pairwise sequence alignment between 2 homologous sequences of the neurotransmitter membrane channel, a stockholm file from Pfam was input into a biopython code using the pairwise2 package. This step was repeated with 2 homologous sequences of the protein of interest: Peptidase C13. This was also done using a stockholm file of the protein of interest found on Pfam. This was done using the code below:

```
1 from Bio import AlignIO, pairwise2
2
3 sto_file = r"C:\Users\Tania\Downloads\PF01650.alignment.seed\PF01650.alignment.seed" #Insert your .seed stockholm/clustal2 filepath here
4
5 alignment = AlignIO.read(r"C:\Users\Tania\Downloads\PF01650.alignment.seed\PF01650.alignment.seed", "stockholm") #Insert your .seed stockholm/clustal2 filepath again
6
7 seq1 = alignment[0].seq
8 seq2 = alignment[1].seq
9
10 alignments = pairwise2.align.globalxx(seq1, seq2)
11
12 for alignment in alignments:
13     print(pairwise2.format_alignment(*alignment))
```

You should receive an output like this:

```
KNAVLVAGSR-GYMYRHRQDVCHAYQ-LKK-GGLKDNITVPMYDD-IAHIF--ENRPRQVI-INSPI-----G----D-DVYK-G-VPKDYTGHH-VTA-INPL-AV-ILQNK-ALS---GSGSKVE-SGPNHIFIFYSOHGGP-G--VLGMP--S--G-PVLYADD-LIDV--LKR-KHASGTYSKLVF-YIEACESGSIFEG-L-
LPEGLNIYATTASNA-EES-IGTKCPGQ-----YPG-P--PPQYQ-TCLGDLYA-VSM-EDSEK--HILRR---ET-LGM-Q-YELV-KRRT--ANS--F--P-YAS-SHMYGDLX-LVD--DP-LS-L-Y-----MET
|||||
RWAVLTAGS-SGYMYRHRQDVCHAYQVL-KK-GGLKDNITVPMYDD-IA-Y-DEENPRPGV-LTNSP-Y-----G-----HDVY-AG-VPKDYTG--EDVT-VNMF-FA-AILQNK-DA--IT-GSGSKV-NSGPNHIFIFYSOHGG-AG--VLGMP--T--YPVLYA-DELT--ETLK-EKHASGTYSKLV-VYIEACESGSIFEG--
ILPEGLNIYATTASNA-VE-SSMGTCYCPG-Q-----DPMVPEY-DTCLGDLY-SVSM-IEDSE-R-HNL--HT--E-SL--KQYQ-VVK--TKTA-E--KPFY--GSHVQVGD-KEL--TQD-NL-YL-Y-----MGT
Score=300
```

## Complete pairwise sequencing code:

**WARNING, PLEASE READ:** You can use the code below to conduct a sequence alignment between each sequence in the SEED file; however, this will take up a lot of memory, a lot of computer processing power, and it may even crash your computer. Running the code on VS studio may give you a bunch of random numbers in the terminal. You can simply exit VS studio, and you should still have some, if not all pairwise sequences, analyzed on the output .txt file.

```
1  from Bio import AlignIO, pairwise2
2
3  sto_file = r"C:\Users\Tania\Downloads\PF01650.alignment.seed\PF01650.alignment.seed" #Insert your .seed filepath here
4  output_file = r"C:\Users\Tania\Documents\Bio notepad reads\pepc13_read.txt" #insert your .txt filepath here
5
6  alignment = AlignIO.read(sto_file, "stockholm")
7
8  with open(output_file, "w") as outfile:
9      for i in range(len(alignment)):
10         for j in range(i + 1, len(alignment)):
11             seq1 = str(alignment[i].seq)
12             seq2 = str(alignment[j].seq)
13
14             alignments = pairwise2.align.globalxx(seq1, seq2)
15
16             for aligned_pair in alignments:
17                 outfile.write(f"Pairwise alignment between sequences {i+1} and {j+1}:\n")
18                 outfile.write(pairwise2.format_alignment(*aligned_pair))
19                 outfile.write("\n")
20
```