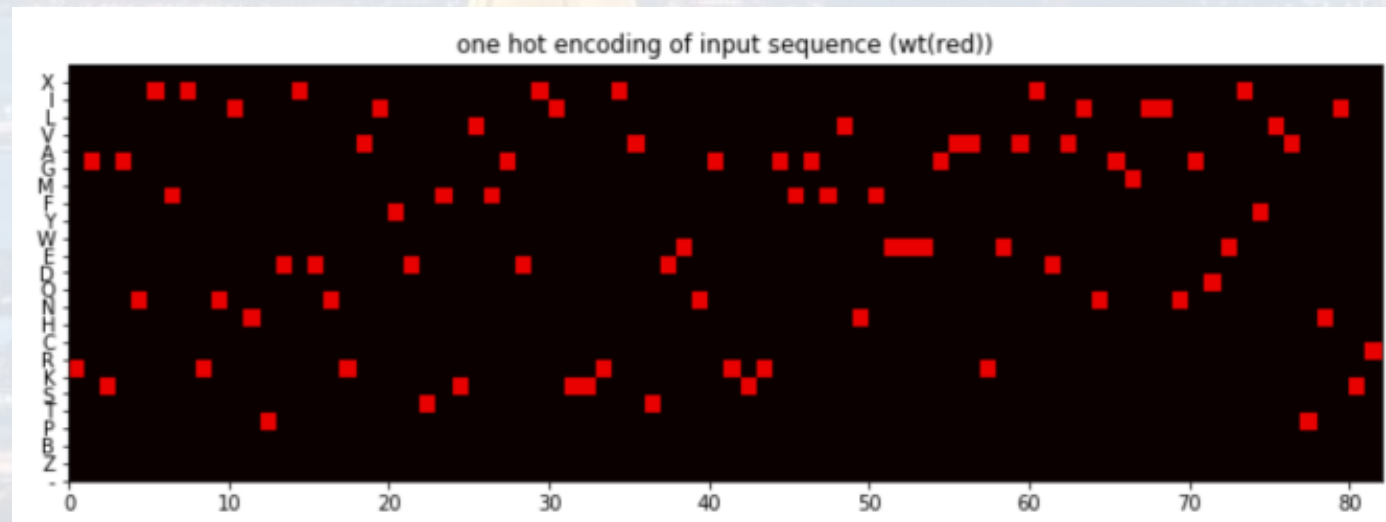


motif finding / sequence analysis

	C	A	G	T	C	T	A
4	1	0	0	0	1	0	0
	0	0	1	0	0	0	0
	0	0	0	1	0	1	0
	0	1	0	0	0	0	1
	Sequence Length						

one – hot encoded NT or AA sequences
can be interpreted as b/w images!



- barcodes are short DNA sequence for identifying species
- we want to see, if we can use a CNN for classification
- loading a so called *fasta* file

```
>BEISA025-19|Culex|COI-5P
AACATTATATTTTCATTTTTTGGTGCTTGAGCAGGAATAATTGGAACCTTCTTTAAGTCTTCTTATTCG
AGCTGAATTAAGTCAACCAGGAGTTTTTATTGGGAATGATCAAATTTATAATGTAATTGTTACAGC
TCATGCTTTTATTATAATTTTTTTTTTATAGTAATACCTATTATAATTGGAGGATTTGGAAATTGATT
AGTTCCTTTAATACTAGGAGCTCCTGATATAGCTTTTCCTCGAATAAATAATATAAGATTTTGAAT
ACTTCCCCCCTCATTAACACTTCTACTTTCTAGTAGTATAGTAGAAAATGGAGCTGGTACAGGTTG
AACAGTATATCCTCCTCTTTCTTCTGGAACAGCTCATGCTGGAGCTTCTGTTGATTTAGCTATTTT
TTCTTTACATTTAGCCGGAATTTCTTCAATTTTAGGAGCTGTAAATTTTATTACTACTGTAATTAA
TATGCGATCTTCTGGTATTACCCTTGATCGAATACCTTTATTTGTTTGATCAGTTGTAATTACTGC
TATTCTTTTATTATTATCTCTTCCTGTTTTAGCTGGAGCTATTACTATATTATTAACAGATCGTAA
TTTAAATACTTCTTTTTTTCGATCCTATTGGAGGAGGAGATCCTATTTTATATCAACATTTATTT
>BEISA121-19|Anopheles|COI-5P
AACATTATATTTTATTTTCGGTGCTTGAGCAGGAATAGTAGGAACCTTCTTTAAGTATTCTTATTCG
```

true label

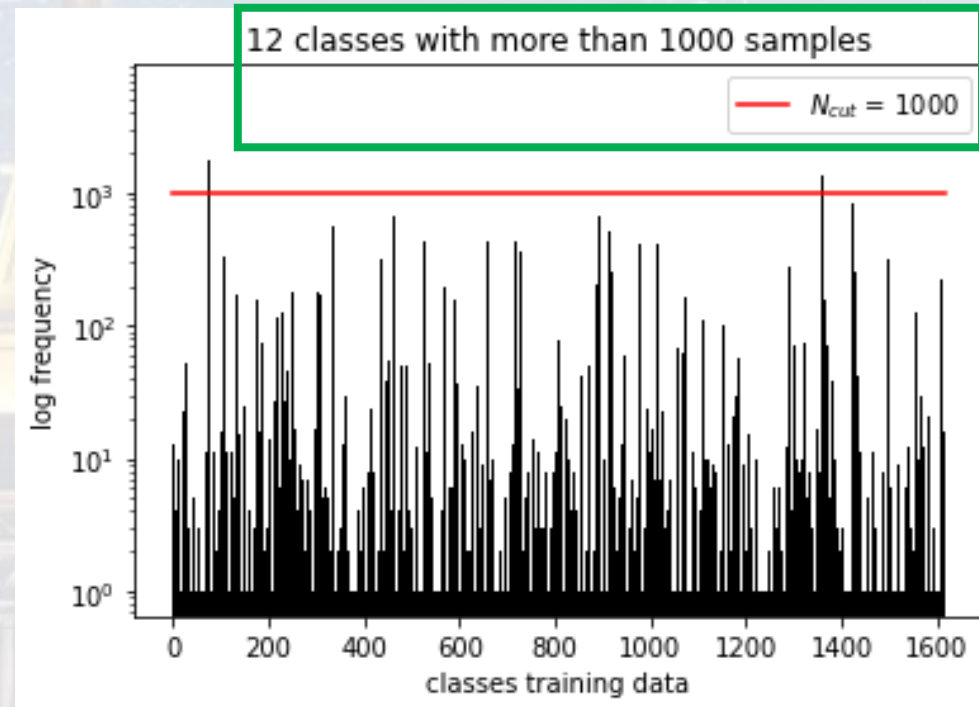
sequence
as image

the data set:

- 86k samples
 - 1.6k classes
 - classes are not evenly distributed
- picking only those with >1k samples (12 classes)

run the package `AnalyzBarcode2.py`

```
A = Analyzer()
```



`__init__`
reads the data,
one-hot encodes
the nucleotides
and passes only
those samples
where `Nsample`
> `Ncut`

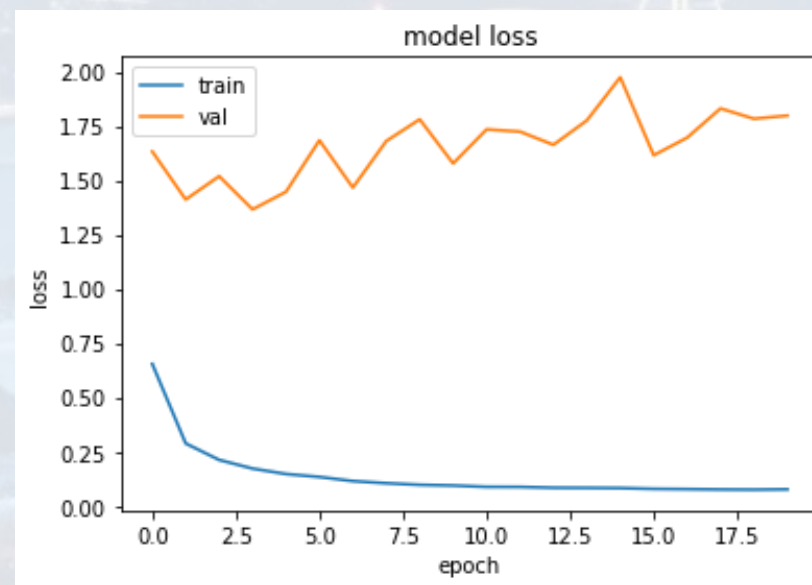
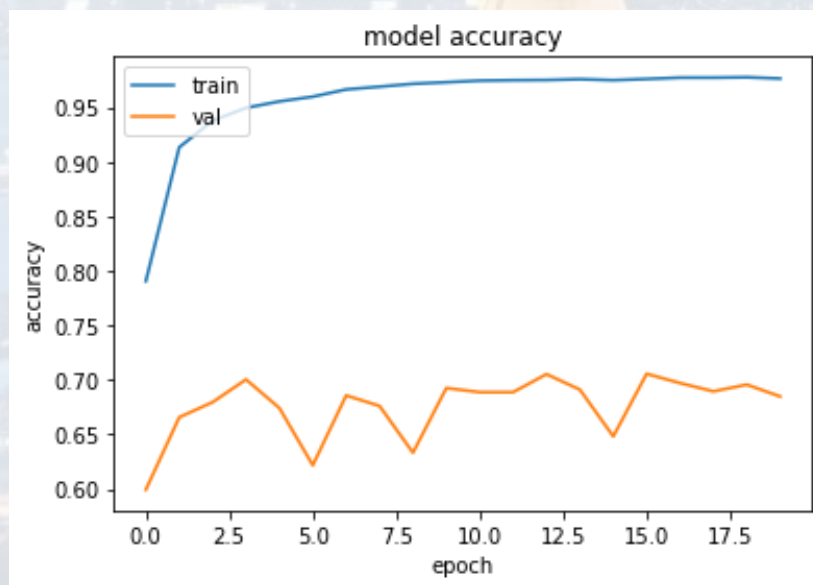
A.RunCNN()

Layer (type)	Output Shape	Param #
conv2d_6 (Conv2D)	(None, 1256, 1, 24)	408
flatten_6 (Flatten)	(None, 30144)	0
dense_12 (Dense)	(None, 84)	2532180
dense_13 (Dense)	(None, 12)	1020

=====

Total params: 2533608 (9.66 MB)
Trainable params: 2533608 (9.66 MB)
Non-trainable params: 0 (0.00 Byte)

runs very simple CNN

results are not great,
but it shows the principle

A.EvalModel()

evaluation (note: here with training data)

