The SVM (*Support Vector Machine*) three-step classifier was used in the classification process. In the first step, a decision is made whether the EEG belongs to grade 4 or to the remaining groups (grades 1,2, 3). In the second step, other signals that are not classified as grade 4 (inactive) are checked. A distinction is made between group 1 and the remaining groups (ie grades 2 and 3). In the last step, a binary decision is made between the remaining signals if they belong to grade 2 or 3. Seven features were extracted by analyzing both the time and frequency domains of the signal. EEG time series are preprocessed by a bandpass filter. EEG is divided on non-overlapped windows duration 30s for the first four features from list. EEG is divided on non-overlapped windows duration 30s for the first four features from list. For the rest of the features in list we used overlapped epochs of duration 8s.

**Feature List:**

1. Sum of copula parameters for each window of two EEG channels.  
2. Standard deviation of copula parameters for each window of two EEG channels.  
3. Maximum value of singular value of 9 EEG channels for each window.  
4. Maximum value of the estimated distribution density function of one EEG channel.  
5. Maximum value of spectral power-overlapped epochs (used code \*)  
6. Minimum value of spectral entropy overlapped epochs (used code \*)  
7. Minimum value of upper margin overlapped epochs (used code \*)

The main function is *main*.m

*feature\_extraction*-extract seven features from list above

*class**\_4\_vs\_1\_2\_3* - use svm for selection EEG that belongs to grade 4.

*class\_1\_vs\_2\_3* – use svm for selection EEG that belongs to grade 1 of remaining time series.

*class 2\_vs\_3 -* use svm for selection EEG that belongs to grade 2 of grade 3 of remaining time series.

\*Use code <https://github.com/otoolej/qEEG_feature_set>

Please, download the code NEURAL: a neonatal EEG feature set in Matlab from [https://github.com/otoolej/qEEG\_feature\_set](https://github.com/otoolej/qEEG_feature_setand%20function%20do_bandpass_filtering.m) and *do\_bandpass\_filtering.m* from <https://github.com/otoolej/nonlinear-energy-operators> to execute programm.