**Cleaning the data steps**

**Spatial filtering (Each elecdtrode only does filter to a reference electrode):**

* Input 1: The data in the electrode which we want to get
* Input 2: The data in the neighbors of input 1, input 2 will be the mean of the neighbors (i.e. input 2 = ) or Laplacian
* Channel “X” = input 1 – input 2, in this way we are filtering all the noise from the neighbors from input 1 electrode

Spatial filtering transform a multi-channel signal X(n) such that each Y(n) depends only on X(n). Most spatial filters are linear: Y(n) = MX(n)

Linear spatial filters can be used for remap channel signals to source signals.

**Temporal filters:**

* Moving Average (MA):

Acts like a smoothing (low-pass) operator, where m is the number of past samples that we take

**Spectral filtering:**

* FIR filters (e.g., “low-pass” at 40 [Hz], low-pass, band-pass, notch)

Where is a kernel which generate the frequency, we want to filter

* The band we want to work on is at approximately 0.5-40 [Hz]

**Independent Component Analysis (ICA):**

* We use ICA for artifacts in the data such as eye or muscle movement or pulse

**Segmentation**

In this part we want to segment the data to trials per category

* We want to create struct (class) which contains type, latency, duration, start trial
  + We want to get the starting place of each trial in different array, for example, trial No.3 starts at index No. 8. In this way we can count the number of trials and know the place of each trial and the starting point of each one (from the latency)
  + Type represents the name of each trigger (start of trial, name of class, etc.) for example ‘1111’ is the start of each trial, the name of each class for example ‘1’ is left ‘2’ is right ‘3’ is idle
  + Latency is the time/sample the trigger appears
* We want to take some buffer from the beginning and the end of the trial