Integrated Lecture and Tutorials
"Brain-Computer Interfacing"

Wintersemester 2014/15

Übungsleiter: Matthias Schultze-Kraft
schultze-kraft@tu-berlin.de

Fachgebiet Neurotechnologie
Institut für Softwaretechnik und
theoretische Informatik
Fakultät IV, Technische Universität Berlin
Prof. Dr. Benjamin Blankertz
Email: benjamin.blankertz@tu-berlin.de

Exercise Sheet 2

Materials

- `oddballVPei.mat`: EEG data
- `scalpmap.m`: Matlab function that plots a scalp topography
- `roc_auc.m`: Incomplete Matlab function for computing ROC and AUC
- `signed_r_squared.m`: Incomplete Matlab function for computing signed r^2 values

Recap: Measures for Class Discriminability

In the last two lectures you have seen two different approaches for assessing the discriminability of dichotomous data. The Receiver Operator Characteristic (ROC) curve – and its corresponding area under the curve (AUC) – comes from signal detection theory and shows the performance of a binary classifier by plotting the true positive against the false positive rate at varying thresholds. An alternative is the signed and squared version of the point-biserial correlation coefficient (`sgn` r^2) which quantifies how much of the variance in the distribution of all samples is explained by the class membership.

Exercise 1 (4 points)

Implement the function `roc_auc` such that the missing structure fields `.x` and `.roc` are computed. Field `.roc` is a `T x C x E x 2` matrix containing the ROC values for each time point T_i and channel C_i, such that

```
plot(squeeze(auc.roc(ti,ci,:,1)),squeeze(auc.roc(ti,ci,:,2)))
```

plots the ROC curve for time index `ti` and channel index `ci`. Use the function `cat` to concatenate matrices along a specific dimension. Field `.x` is a `T x C` matrix containing the corresponding AUC values of each ROC curve. The AUC of one ROC curve can be computed using the trapezoidal rule

$$\int_a^b f(x)dx \approx (b-a)\left[\frac{f(a)+f(b)}{2}\right].$$

Exercise 2 (3 points)

Implement the function `signed_r_squared.m` such that the missing field `.x` is computed, which is a `T x C` matrix containing the signed r^2 values for each time point and channel.

Bonus Exercise (1.5* bonus points)

- Implement the function `roc_auc` using only one for-loop for computing the ROC curves.
- Implement the computation of the AUC with the trapezoidal rule without using a for-loop.
- Implement the function `signed_r_squared` without using a for-loop.

Make appropriate use of the functions `reshape` and `repmat`.

Exercise 3 (3 points)

Now let's see how both class discriminative approaches apply to the dataset of an oddball paradigm. In a script `investigate_discriminability.m` do the following:

- Load the dataset `oddballVPei.mat`. Segment and baseline correct the data using `ival = [-200 1000]` and `base_ival = [-200 -100]` and functions `make_epochs` and `baseline_epochs` from the last exercise sheet.
- Obtain ROC curves and AUC and signed r^2 values for all time points and channels with
  `auc = roc_auc(epo)`
  `rsq = signed_r_square(epo)`
- Using different colors, show for channels `Cz`, `Pz` and `P6` in one plot the AUC values, in the other the signed r^2 values. Use `legend` to indicate the color code of the curves. Plot with a horizontal line the corresponding zero discriminability value.
- For t_1 `= 250 ms` in one plot and t_2 `= 340 ms` in another, plot the ROC curves of channel `Cz`. Add to both plots with a dotted line the diagonal. Show the corresponding AUC value in the title of the plots.
- In a `2 x 2` figure, plot for t_1 `= 250 ms` and t_2 `= 340 ms` (figure columns) the AUC and signed r^2 values (figure rows) as scalp topographies using the function `scalpmap`. Adjust the color limits of the scalp maps appropriately.
- Shortly comment what you observe in all generated plots.

Note: Please write your name(s) on all submitted files.