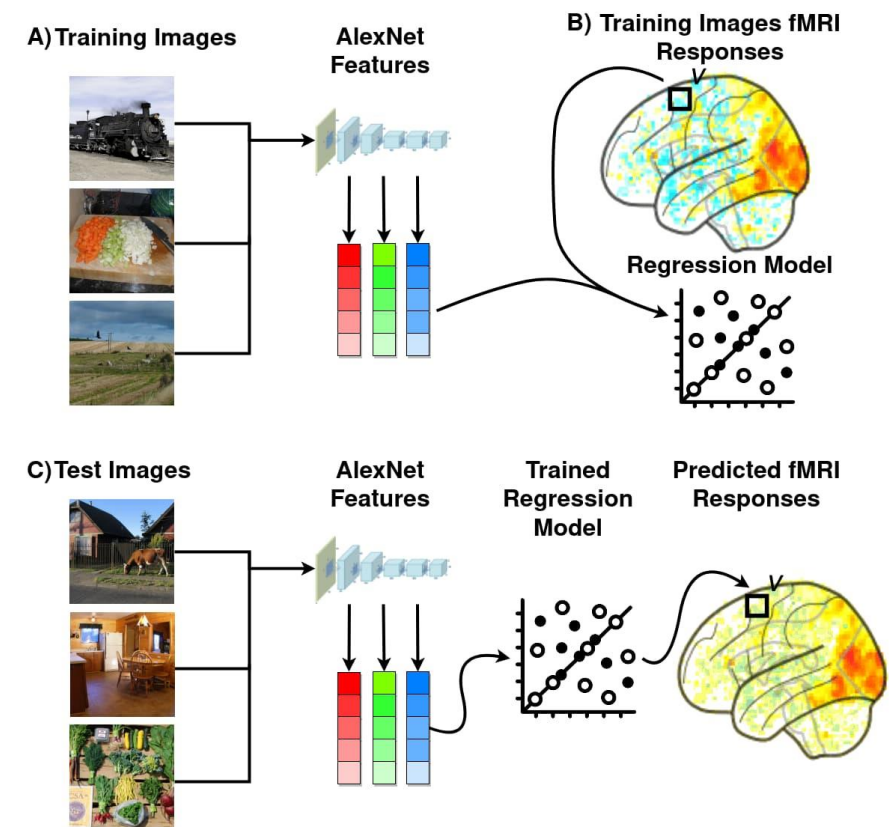
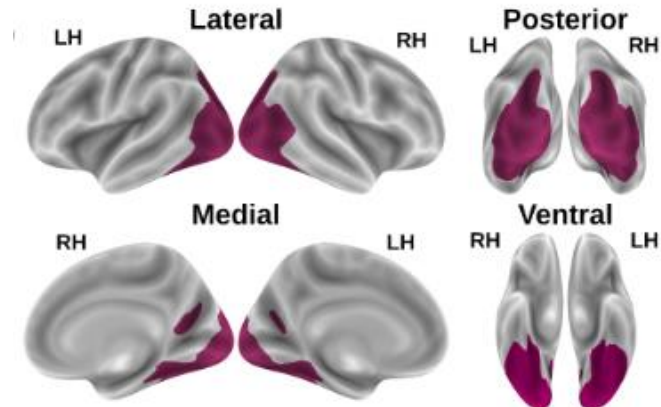


Obiettivo Challenge: **Algonauts 2023**

Obiettivo Challenge

1. Sviluppo di un **encoding model** che preveda la risposta neurale umana (Bold Response Amplitudes) a stimoli visivi che consistono in scene naturali complesse
 - ❑ I dati - Algonauts Challenge 2023 Dataset: provengono dal Natural Scene Dataset (NSD): ma i neuroni presi in considerazione sono quelli appartenenti alla sola **visual cortex** che si mostrassero particolarmente responsivi agli stimoli visivi



Algonauts Challenge

2023: Dataset

Dataset: Riassunto

- ❑ Versione pre-processata di una parte del Natural Scene Dataset (NSD) -> tenendo solo vertici appartenenti alla corteccia visiva, che abbiamo massima responsività
- ❑ 8 soggetti, 2 emisferi per soggetto (LH e RH)
- ❑ A ogni soggetto sono mostrate ~8000-10000 immagini
- ❑ Per ogni emisfero (2, RH e LF) di ogni soggetto (8 totali), per ogni immagine mostrata, è disponibile un vettore di attivazioni (**Bold Response Amplitudes**) per ~20000 vertici (unità tridimensionale di tessuto cerebrale) della corteccia visiva
 - ❑ Per ogni emisfero si ha una matrice 2D (per un totale di $8 \times 2 = 16$ matrici)
 - ❑ Righe -> Immagine Mostrata (~9000 righe)
 - ❑ Colonne -> Vertice della corteccia visiva (~20000 vertici per emisfero)
 - ❑ Valori -> **BOLD Response Amplitudes** o Betas (floats distribuiti tra -2 e +2, indicano l'attivazione/disattivazione del vertice, in risposta allo stimolo) -> single o multi-trial (mediati)
- ❑ Dati aggiuntivi, per ogni vertice:
 - ❑ **ROI Indices**: indica a quale **region-of-interest** appartiene, ognuna associata a determinate proprietà funzionali
 - ❑ **ROI Classes**: ogni ROI appartiene a una certa classe (body-selective regions, face selective regions, ...)
 - ❑ **fsaverage space ROI indices**: indice che consente di mappare ogni vertice in una **common cortical surface**, ovvero una superficie corticale ottenuta come "media" di tutti i soggetti -> e.g. usata per plottare le attivazioni graficamente come nell'esempio



Stimuli

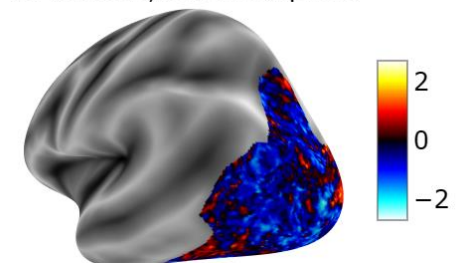


LH, Subj1

All vertices, left hemisphere

	v1	v2	v3	v4
img1	-1.002848	0.694632	-0.243869	-0.324497
img2	-0.147516	-0.596537	-0.675691	0.810392
img3	-0.135000	0.912252	0.602520	0.305391
img4	0.582822	0.212471	0.058079	-0.126217
img5	-0.204780	0.227659	-0.544403	-0.292768
...

All vertices, left hemisphere



BOLD Response Amplitudes (single o multi-2/3-trial)

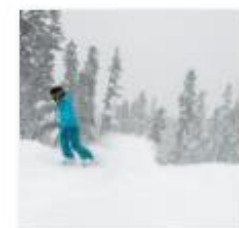
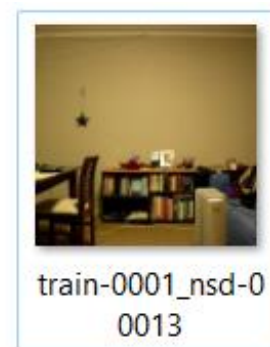
16 matrici di questo tipo disponibili:
8 soggetti x 2 emisferi

Algonauts Challenge

Dataset: Dettaglio

Immagini

- ❑ 8 soggetti, rispettivamente associati a:
 - ❑ 9841, 9841, 9082, 8779, 9841, 9082, 9841, 8779 Immagini di training (di cui sono **disponibili le risposte**)
 - ❑ 159, 159, 293, 395, 159, 293, 159, 395 -> Test Images (risposte non disponibili)
 - ❑ ~9000 Uniche per soggetto, ~1000 condivise
 - ❑ Ogni immagine è mostrata 1, 2 o 3 volte al soggetto, le risposte sono state mediate quando presenti più trials
 - ❑ .png - 425x425px
- ❑ **Fonte: COCO Image Dataset**
 - ❑ Cropped (rispetto alle immagini originali)
- ❑ **Nomenclatura:** es. train-0001_nsd-00013
 - ❑ **Primo indice:** train-0001 -> ordina le immagini per matchare la rispettiva riga nella matrice delle attivazioni
 - ❑ **Secondo indice:** nsd-000013 -> ID dell'immagine nelle 73000 immagini di NSD
 - ❑ Consente poi di risalire all'ID dell'immagine in **COCO**, quindi ai relativi **metadati**



Risposte: BOLD response amplitudes

- ❑ Disponibili solo per i dati di training
- ❑ Per ogni emisfero (LH, RH) di ogni soggetto (1-8) è disponibile una **matrice 2D** dove:
 - ❑ **Righe** -> **Indice Immagine** Mostrata (~9000 righe)
 - ❑ **Colonne** -> **Vertice** della corteccia visiva (~20000 vertici per emisfero)
 - ❑ **Valori** -> **BOLD Response Amplitudes**
- ❑ **Bold Response Amplitudes** (valore risposta per ogni vertice) o **Betas**
 - ❑ Valori singoli per vertice **ottenuti** a partire da serie temporali/segnali (per vertice): fMRI blood-oxygen-level-dependent (BOLD) signal
 - ❑ È il segnale in output per il **BOLD Imaging**: *misura indiretta dell'attività celebrale di un vertice* (unità tridimensionale del tessuto celebrale) della corteccia in risposta a uno stimolo, *rilevata mediante fMRI*.
 - ❑ Il segnale aumenta con la diminuzione di concentrazione di emoglobina deossigenata (dotata di proprietà magnetiche) nel flusso sanguigno locale. (effetto BOLD: Quando una determinata regione del cervello è attiva, richiede maggiori quantità di ossigeno e di glucosio per produrre energia.)
 - ❑ Da ogni segnale è ottenuto un **Single Trial Beta**: *stima dell'ampiezza della risposta fMRI per ogni vertice e trial condotto, mediante GLM analysis*
 - ❑ Rispetto ai dati di NSD (single trial betas), le attività di **Algonauts Data** sono: 1) z-scored rispetto alla sessione di scan 2) **multi trials betas**: mediate sui trials (ripetizioni) delle immagini
 - ❑ **Betas sono floats distribuiti tra -2 e +2**
 - ❑ **Positive BOLD Responses**: aumento del flusso sanguigno rispetto al valore base, aumento attività celebrale
 - ❑ **Negative BOLD Responses**: diminuzione flusso, diminuzione attività celebrale
 - ❑ Prossimi a 0: flusso simile al livello base, non presente attivazione/disattivazione
- ❑ I vertici resi disponibili per la challenge sono un subset di quelli di NSD:
 - ❑ **fsaverage space** -> insieme dei vertici della corteccia di cui sono registrate le risposte (~360k per emisfero) in NSD (intera corteccia e non solo quella visiva)
 - ❑ **challenge space** -> insieme dei vertici della corteccia VISIVA di cui si hanno le risposte nel dataset della challenge (~20k per emisfero)

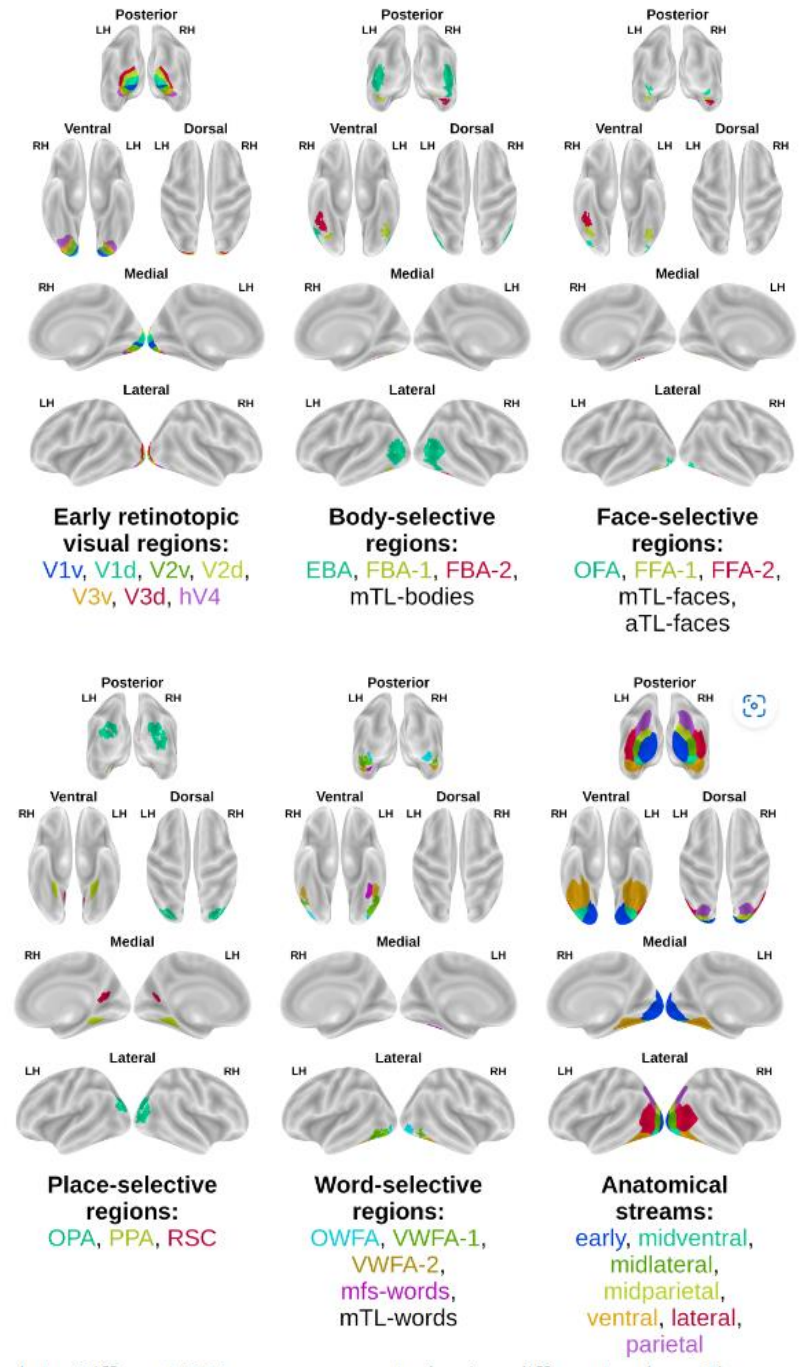
LH, Subj 1

	v1	v2	v3	v4
img1	-1.002848	0.694632	-0.243869	-0.324497
img2	-0.147516	-0.596537	-0.675691	0.810392
img3	-0.135000	0.912252	0.602520	0.305391
img4	0.582822	0.212471	0.058079	-0.126217
img5	-0.204780	0.227659	-0.544403	-0.292768
...

Soggetto 1, lh, shape: (9841, 19004)
Soggetto 1, rh, shape: (9841, 20544)
Soggetto 2, lh, shape: (9841, 19004)
Soggetto 2, rh, shape: (9841, 20544)
Soggetto 3, lh, shape: (9841, 19004)
Soggetto 3, rh, shape: (9841, 20544)
Soggetto 4, lh, shape: (9841, 19004)
Soggetto 4, rh, shape: (9841, 20544)
Soggetto 5, lh, shape: (9841, 19004)
Soggetto 5, rh, shape: (9841, 20544)
Soggetto 6, lh, shape: (9841, 19004)
Soggetto 6, rh, shape: (9841, 20544)
Soggetto 7, lh, shape: (9841, 19004)
Soggetto 7, rh, shape: (9841, 20544)
Soggetto 8, lh, shape: (9841, 19004)
Soggetto 8, rh, shape: (9841, 20544)

Dati Aggiuntivi: ROI (Region-Of-Interest)

- ❑ **Regions-of-interest:** regioni della corteccia visiva con differenti proprietà funzionali
- ❑ Le ROI sono suddivisibili in **classi di funzione:**
 - ❑ **Early retinotopic visual regions/area (prf-visualrois):**
 - ❑ *Primi siti corticali per l'elaborazione sensoriale bottom-up delle informazioni visive (V1 direttamente legato all retina)*
 - ❑ *compute lower-order features such as edges or contours.*
 - ❑ ROIs: V1v, V1d, V2v, V2d, V3v, V3d, hV4.
 - ❑ **Body-selective regions (floc-bodies):**
 - ❑ Parti della extrastriale visual cortex impegnate nel processing dei corpi umani in relazione alle parti del corpo (EBA) e in contrasto con essi (FBA)
 - ❑ EBA, FBA-1, FBA-2, mTL-bodies.
 - ❑ **Face-selective regions (floc-faces):**
 - ❑ Region specialised for face perception (OFA) and recognition (FFA).
 - ❑ OFA, FFA-1, FFA-2, mTL-faces, aTL-faces.
 - ❑ **Place-selective regions (floc-places):**
 - ❑ Scene perception relies on a set of cortical regions (PPA, RSC, OPA) exhibiting dissociable functional selectivity to various scene properties.
 - ❑ OPA, PPA, RSC.
 - ❑ **Word-selective regions (floc-words):**
 - ❑ OWFA, VWFA-1, VWFA-2, mfs-words, mTL-words.
 - ❑ **Anatomical streams (streams):**
 - ❑ reflecting large-scale divisions of visual cortex
 - ❑ early, midventral, midlateral, midparietal, ventral, lateral, parietal.



Dati Aggiuntivi: ROI (Region-Of-Interest)

- ❑ Per ogni **emisfero** di ogni **soggetto**, si hanno i seguenti dati aggiuntivi:
 - ❑ **Per Class – Vertex / ROI Index Maps:** liste che, per ogni classe, per ogni vertice del challenge space (~360k vertici) o del fsaverage space (~20k vertici), associano un indice ROI:
 - ❑ 0 indica che il vertice non appartiene a alcun ROI noto della classe specifica
 - ❑ Mi permette di selezionare i vertici in funzione dell'appartenenza a una determinata ROI / Classe di ROI
 - ❑ **Per Class – ROI Indices / ROI Name Maps:** dizionari che per ogni classe ROI, associano un ID intero al nome di ogni ROI che vi appartiene
 - ❑ **fsaveragespace to challengespace maps:** vettori binari, che per ognuno dei ~163k vettori della fsaverage space, assegna
 - ❑ 0 se appartiene alla challenge space, 1 se non appartiene
 - ❑ Permette di
 - ❑ Mappare i vertici della challenge space nel fsaverage space (e.g. per visualizzare i vertici in 3D)
 - ❑ Individuare le corrispondenze tra i dati nel challenge dataset e nel NSD dataset, quindi poter estrarre da esso dati aggiuntivi
- ❑ **Le ROI maps** sono disponibili sia per i vertici del fsaverage space (tutti i vertici di NSD) che per il challenge space (dati della challenge, ~19k vertici per emisfero/soggetto)

Subj1 - LH

```
ROI Class 0 : [0 0 0 ... 0 0 0] shape: (19004,)
ROI Class 1 : [0 0 0 ... 0 0 0] shape: (19004,)
ROI Class 2 : [0 0 0 ... 0 0 0] shape: (19004,)
ROI Class 3 : [0 3 1 ... 0 0 0] shape: (19004,)
ROI Class 4 : [0 0 0 ... 0 0 0] shape: (19004,)
ROI Class 5 : [3 0 7 ... 5 5 5] shape: (19004,)
```

```
ROI Class 0 {0: 'Unknown', 1: 'V1v', 2: 'V1d', 3: 'V2v', 4: 'V2d', 5: 'V3v', 6: 'V3d', 7: 'hV4'}
ROI Class 1 {0: 'Unknown', 1: 'EBA', 2: 'FBA-1', 3: 'FBA-2', 4: 'mTL-bodies'}
ROI Class 2 {0: 'Unknown', 1: 'OFA', 2: 'FFA-1', 3: 'FFA-2', 4: 'mTL-faces', 5: 'aTL-faces'}
ROI Class 3 {0: 'Unknown', 1: 'OPA', 2: 'PPA', 3: 'RSC'}
ROI Class 4 {0: 'Unknown', 1: 'OWFA', 2: 'VWFA-1', 3: 'VWFA-2', 4: 'mfs-words', 5: 'mTL-words'}
ROI Class 5 {0: 'Unknown', 1: 'early', 2: 'midventral', 3: 'midlateral', 4: 'midparietal', 5: 'v'}
```

Subj1 - LH

```
[0 0 1 0 1 0 0 0 0 0] Shape (163842,)
```

Dati Aggiuntivi: NCSNR e Images Trials

- ❑ Per ogni **emisfero** di ogni **soggetto**, ho ottenuto dai creatori della challenge i seguenti dati aggiuntivi:

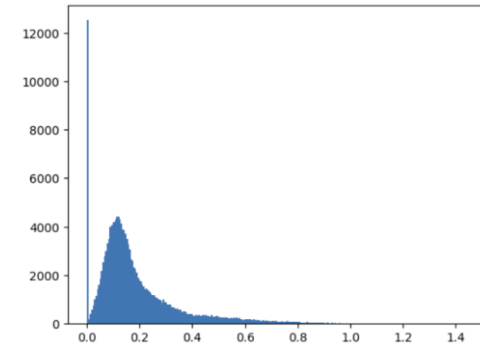
- ❑ **signal-to-noise ratio (ncsnr) per vertex** `[0.11920288 0.31043354 0.11843289 0.16619019]` `shape (163842,)`

$$\text{ncsnr} = \frac{\hat{\sigma}_{\text{signal}}}{\hat{\sigma}_{\text{noise}}}$$

- ❑ Vettori monodimensionali
- ❑ Per ogni vertice del fsaverage space indicano il signal-to-noise ratio dello stesso (un valore unico indipendente dalle immagini)
- ❑ Tra 0 e 0.6 (ma possono anche essere maggiori)
- ❑ Utili per calcolare il **Noise Ceiling (NC)** per ogni vertice

- ❑ **Number of trials per image:** `[3 3 3 ... 3 3 3]` `shape (9841,)`

- ❑ Per ogni immagine indica il numero di trials, quindi il numero di volte che è stata mostrata al soggetto (1, 2 o 3 volte)
- ❑ Indica anche quindi il **numero di BOLD responses mediati per ottenere il dato disponibile**
- ❑ Permette anche esso di calcolare il Noise Ceiling (NC)



Possibili dati aggiuntivi

- ❑ Dal Natural Scene Dataset è possibile ottenere (dati per tutto il fsaverage space):
 - ❑ **Raw Data** (~946 GB) -> Tutti i dati acquisiti -> structural, functional e diffusion MRI scans + dati aggiuntivi
 - ❑ **Time series** leggermente pre-processate (~3.4 TB)
 - ❑ pre-processed (upsampling spazio-temporale) **BOLD-fMRI time-series** data for each voxel
 - ❑ Da essi sono ottenuti i valori BOLD Response Amplitude per ciascun trial (esistono diverse tecniche per ottenerli)
 - ❑ Associati alle time-series si hanno **dati di natura psicologica** sul soggetto durante l'esperimento
 - ❑ physiological data (pulse and respiratory)
 - ❑ experimental design information (i.e. which images were shown when)
 - ❑ motion parameter estimates from the pre-processing of the fMRI data, and eyetracking data
 - ❑ **Single-trial BOLD responses** (simili ai dati della challenge)
 - ❑ **BOLD responses** per i singoli trial di ogni immagine (quindi non mediati come nel challenge dataset)
 - ❑ 3 versioni dei BOLD responses ottenuti dalle serie temporali con 3 diversi metodi
 - ❑ Sono simili ai dati della challenge, differiscono da essi per 1) sono relativi a tutta la corteccia 2) non sono mediati sui trials 3) non sono z-scored
 - ❑ **Misure Ausiliarie**
 - ❑ Anatomical neuroimaging measures (T1, T2, diffusion, venogram, angiogram)
 - ❑ Functional neuroimaging measures (resting-state data)
 - ❑ Misure comportamentali
 - ❑ Dati fisiologici e **eye-tracking** (solo per alcune sessioni)
- ❑ **Metadati** da COCO dataset

Baseline Encoding **Model**

Baseline Encoding Model: Pipeline

- ❑ Approccio: **Linearizing Encoding Model** -> response of each vertex is predicted independently (linear regression) using the multiple features provided by a computational model
- ❑ Per ogni EMISFERO di ogni SOGGETTO si esegue separatamente:
 1. **Split randomico** training data nelle partizioni di training (90%) e validation
 2. Training Partition: **Feature Extraction** delle immagini, **in batch**, usando **AlexNet** (pre-trained su ImageNet) e **FIT di una Incremental PCA** per la dimensionality reduction delle features (progressivamente sui batch)
 - Le feature estratte sono interpretate come le feature ipotetiche che una certa area del cervello genera per rappresentare le immagini stimolo
 1. Loading del batch di immagini e **trasformazione immagini**
 - Resize 224x224, Normalizzazione [0,1.0], normalizzazione rispetto a ImageNet mean e std
 2. Estrazione feature da AlexNet e flattening per ogni immagine
 - Layer: (2): MaxPool2d(kernel_size=3, stride=2, padding=0, dilation=1, ceil_mode=False)
 - Features Matrix Shape: **n_batch x n_features** (300, 46656)
 3. **Partial fit PCA** to feature batch
 - n components: **100**
 - L'incremental PCA viene utilizzata per limitazioni di RAM, applicarla alle feature dell'intero dataset non è possibile (io ho 32 gb e non sono riuscito)

```
AlexNet(  
  (features): Sequential(  
    (0): Conv2d(3, 64, kernel_size=(11, 11), stride=(4, 4), padding=(2, 2))  
    (1): ReLU(inplace=True)  
    (2): MaxPool2d(kernel_size=3, stride=2, padding=0, dilation=1, ceil_mode=False)  
    (3): Conv2d(64, 128, kernel_size=(5, 5), stride=(1, 1), padding=(2, 2))  
    (4): ReLU(inplace=True)  
    (5): MaxPool2d(kernel_size=3, stride=2, padding=0, dilation=1, ceil_mode=False)  
    (6): Conv2d(128, 256, kernel_size=(3, 3), stride=(1, 1), padding=(1, 1))  
    (7): ReLU(inplace=True)  
    (8): Conv2d(256, 512, kernel_size=(3, 3), stride=(1, 1), padding=(1, 1))  
    (9): ReLU(inplace=True)  
    (10): Conv2d(512, 512, kernel_size=(3, 3), stride=(1, 1), padding=(1, 1))  
    (11): ReLU(inplace=True)  
    (12): MaxPool2d(kernel_size=3, stride=2, padding=0, dilation=1, ceil_mode=False)  
  )  
  (avgpool): AdaptiveAvgPool2d(output_size=(6, 6))  
  (classifier): Sequential(  
    (0): Dropout(p=0.5, inplace=False)  
    (1): Linear(in_features=9216, out_features=4096, bias=True)  
    (2): ReLU(inplace=True)  
    (3): Dropout(p=0.5, inplace=False)  
    (4): Linear(in_features=4096, out_features=4096, bias=True)  
    (5): ReLU(inplace=True)  
    (6): Linear(in_features=4096, out_features=1000, bias=True)  
  )  
)
```

Baseline Encoding Model: Pipeline

3. Feature Extraction e Downsampling immagini usando la PCA addestrata (ripetuto su training, validation e test set)
 1. Caricamento batch di immagini e trasformazione (come prima)
 2. Estrazione feature da AlexNet e flattening per ogni immagine
 - ottenendo matrice $n_batch \times n_features \rightarrow (300, 46656)$
 3. Downsampling delle feature usando la PCA fittata precedentemente
 - ottenendo matrice $n_batch \times n_pca_components \rightarrow (300, 100)$
 4. Ripetizione per tutti i **batch** e **hstack** delle feature ottenute
 - ottenendo una matrice $\rightarrow n_images_per_partition \times n_pca_components \rightarrow e.g (8000, 100)$
4. Train del **linear encoding model** che mappi AlexNet image features nelle fMRI responses (**training set**)
 - Model: **Ordinary Least Squares Regression**
 - **Objective Functions**: minimizzazione **residual sum of squares** tra attivazioni reali e previste
 - **X = training images feature matrix (extracted e downsampled)** $\rightarrow n_images_per_partition \times n_pca_components \rightarrow e.g (8000, 100)$
 - **Y = images/responses matrix** $\rightarrow n_images_per_partition \times n_vertices \rightarrow e.g (8000, 19000)$
 - Si ottiene **un modello per ogni vertice**
 - Che mappi un vettore delle feature estratte da una immagine nell'attivazione BOLD del singolo vertice per quell-immagine
5. Applicazione dei modelli di regressione allenati per prevedere le attivazioni dei vertici reattive alle immagini di **validation e test set**
 - Si ottengono dei vettori delle predizioni delle BOLD responses per val e test set
 - **Matrici 2D** $\rightarrow n_images_val/test_partition \times n_pca_component e.g (8000, 19000)$
6. Ripetizione passaggi 1-5 per gli emisferi RH/LH di tutti i (8) soggetti

Challenge Evaluation Metric:
Median Noise-Normalized Squared
Correlation Percentage Score

Median Noise-Normalized Squared Correlation

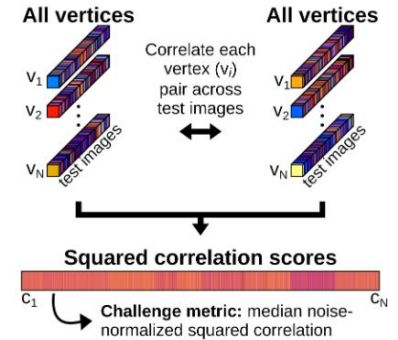
Median Noise-Normalized Squared Correlation

- Indica la prediction accuracy mediana, normalizzata sul rumore, tra tutti i vertici, di tutti gli emisferi, di tutti i soggetti
- Indica quanta della predictable variance è prevista dal modello
- Baseline Model Score** (Pretrained AlexNet): 40.48% of the total predictable variance

$$\text{metric} = \text{Median} \left\{ \frac{R_1^2}{NC_1}, \dots, \frac{R_v^2}{NC_v} \right\} \times 100,$$

$$R_v = \text{corr}(G_v, P_v) = \frac{\sum_t (G_{v,t} - \bar{G}_v)(P_{v,t} - \bar{P}_v)}{\sqrt{\sum_t (G_{v,t} - \bar{G}_v)^2 \sum_t (P_{v,t} - \bar{P}_v)^2}},$$

where v is the index of vertices (over all subject and hemispheres), t is the index of the test stimuli images, G and P correspond to, respectively, the ground truth and predicted fMRI test data, \bar{G} and \bar{P} are the ground truth and predicted fMRI test data averaged across test stimuli images, R is the Pearson correlation coefficient between G and P and NC is the noise ceiling.



Calcolo (si ripete per i vertici di ogni emisfero, di ogni soggetto)

$$\text{ncsnr} = \frac{\hat{\sigma}_{\text{signal}}}{\hat{\sigma}_{\text{noise}}}$$

- Si ottengono i vettori 1D **noise ceiling signal-to-noise ratio (ncsnr)** \longrightarrow `[0.11920288 0.31043354 0.11843289 0.16619019]` shape (163842,) Shape: 1 x n_fsaverage_vertices
 - 1D, Indicano il SNR di ogni vettore della corteccia dello specifico emisfero del fsaverage space (~163k)
- Selezione **noise ceiling signal-to-noise ratio (ncsnr)** per i vertici del **challenge space** \longrightarrow `[0.45753199 0.30325422 0.69066209 1.016029 0.27059516]` shape (19004,) Shape: 1 x n_challenge_vertices
- Importazione dei vettori dei **Number of trials / image** per ogni soggetto \longrightarrow `[3 3 3 ... 3 3 3]` shape (9841,) Shape: 1 x n_images
 - Per ogni immagine indica il numero trial, quindi il numero di volte che è stata mostrata al soggetto (1, 2 o 3 volte) / il numero di BOLD responses mediati per ottenere il dato disponibile per l'immagine
 - Shape: 1 x n_images
- ncsnr to Noise Ceiling (NC)** per vertice: \longrightarrow `[0.35793416 0.19672465 0.55953275 0.73327037 0.16317555]` shape (19004,)

- Noise Ceiling (NC):** total predictable variance given the level of noise in the data, maximum percentage of variance in the voxel's responses that can, in theory, be explained, given the presence of measurement noise. Quantifying the percentage of variance in a voxel's responses that can be attributed to the stimulus and not to measurement noise

$$NC = 100 \times \frac{\text{ncsnr}^2}{\text{ncsnr}^2 + \frac{1}{n}}$$

$$NC = 100 \times \frac{\text{ncsnr}^2}{\text{ncsnr}^2 + \frac{\frac{A}{3} + \frac{B}{2} + \frac{C}{1}}{A + B + C}}$$

- Il SNR di ogni vertice viene normalizzato rispetto al numero di trial delle immagini di test/val set (A: numero di immagini con 3 trial, B: 2, C:1), quindi al numero di BOLD responses mediate per ogni immagine $\rightarrow + \text{trials} = \rightarrow \text{SNR}$

$$NC = 100 \times \frac{\sigma_{\text{signal}}^2}{\sigma_{\text{signal}}^2 + \sigma_{\text{noise}}^2}$$

- Calcola, per ogni vertice, il **coefficiente di correlazione di pearson** tra le risposte reali (ground truth) e quelle previste, e

- ❑ Imposta correlazioni negative a 0
- ❑ Ottieni le correlazioni al quadrato

LH, Subj 1, **Ground Truth**
n_images x n_vertices

	v1	v2	v3
img1	-1.362684	0.105403	-0.634885
img2	-0.755504	-0.152065	-0.998677
img3	0.864619	0.136462	-0.621923
img4	1.017041	-1.972292	0.777953
img5	0.468452	0.146705	-0.446328

LH, Subj 1, **Predictions**
n_images x n_vertices

	v1	v2	v3
img1	-0.290258	-0.260766	0.398556
img2	-0.231810	-0.086052	-0.523469
img3	0.167813	0.094185	0.182432
img4	0.179203	0.060119	-0.010540
img5	0.304406	0.443552	0.005249

LH, Subj 1,
Pearson Correlations
1 x n_vertices

[0.30821148 0.36844289 0.58986917 0.55375606 0.25530387] Shape: (19004,)

Squared Correlations

[0.09499432 0.13575016 0.34794564 0.30664577 0.06518006] shape (19004,)

Noise Ceiling (NC)

[0.35793416 0.19672465 0.55953275 0.73327037 0.16317555] shape (19004,)

- Calcola le **noised normalized squared correlations** per i singoli vertici dividendo per i Noise Ceiling (NC)

$$\frac{R_1^2}{NC_1}, \dots, \frac{R_v^2}{NC_v}$$

[0.26539606 0.6900516 0.62185036 0.41818923 0.3994475] shape (19004,)

- Lo score indica how much of the predictable variance has been accounted for by the models for the single vertex
- Ripeti il processo per tutti gli emisferi di tutti i soggetti
- Concatenazione dei vettori ottenuti
- Ottenimento **median** noise normalized squared correlation score su tutti i vertici





















$$\text{metric} = \text{Median} \left\{ \frac{R_1^2}{NC_1}, \dots, \frac{R_v^2}{NC_v} \right\} \times 100,$$

Modifiche effettuate rispetto al **baseline model**

Modifiche effettuate rispetto al baseline model

1. Da Linear Regression OLS a Ridge Regression + Ottimizzazione parametro alpha (GridSearch)
 - ❑ Preferibile nel caso di multicollinearità tra le variabili (come per le feature estratte dalle CNN), riduce overfitting
 - ❑ Infatti, anche se viene effettuata PCA, le componenti ottenute rimangono correlate tra loro
 - ❑ Si ottiene tipicamente un **miglioramento** dello score di **2-3 punti percentuali**
2. Separazione feature extraction da fit della Incremental PCA
 - ❑ Nel baseline model, la dimensione del batch di immagini passate al modello per la feature extraction coincide con quella del batch di feature su cui viene applicato il **partial fit** della **incremental PCA**
 - ❑ Questo causava problemi, un batch troppo grande era insostenibile in termini di VRAM in fase di feature extraction, un batch troppo piccolo non permetteva di fittare una PCA con un numero sufficiente di componenti
 - ❑ Nella mia implementazione quindi vengono passati mini-batch di immagini al feature extractor (AlexNet), con una dimensione consona alla VRAM disponibile, concatenate poi verticalmente fino a ottenere un `pca_batch` coerente con il numero di componenti da ottenere con la PCA
3. Testati diversi layer di output di AlexNet, **raggiungendo il baseline** score di **40%**, con:
 - ❑ Layer di Output: «(12):MaxPool2d» (MaxPool Layer del 4° e ultimo blocco di convoluzione di AlexNet, 16384 features)
 - ❑ PCA a 150 componenti
 - ❑ Ridge Regression ($\alpha = 2e4$)
4. Testati altri **modelli** per il **feature extraction**: ResNet-50, EfficientNet-B2, VGG-16, VGG-19

Risultati

Aa Name	# Test Sco...	# Val Score	# S1L	# S1R	Feature ...	layer	# n Featur...	weights	downsample	Encoding Model
 vgg19_features.43-pca_2048-ridge-alpha_20000.0		49.84	50.56	49.02	VGG-19_BN	avgpool	25088	pre-trained ImageNet-1K	partial-pca 2048	RidgeR GridSearch
 vgg16_features.43-pca_2048-ridge-alpha_20000.0	43.18	50.34	50.32	48.39	VGG-16_BN	avgpool	25088	pre-trained ImageNet-1K	partial-pca 2048	RidgeR GridSearch
 vgg16-flatten-pca512-RidgeR			49.86	48.08	VGG-16_BN	Flatten		pre-trained ImageNet-1K	partial-pca 512	RidgeR GridSearch
 vgg16-flatten-pca300-linearR	41.049	47.97	48.03	46.2	VGG-16_BN	Flatten		pre-trained ImageNet-1K	partial-pca 300	LinearR OLS
 vgg16-features43-pca100-linearR			46.69	44.97	VGG-16_BN	Features.43		pre-trained ImageNet-1K	partial-pca 100	LinearR OLS
 efficientnetb2_features.8-pca_2048-ridge-alpha_10000.0			46.48	44.97	EfficientNet	Features.8	114048	pre-trained ImageNet-1K	partial-pca 2048	RidgeR GridSearch
 alexnet-pca-RidgeR-1.3		46.95	46.203	45.257	AlexNet	features.12 - Ma	16384	pre-trained ImageNet-1K	partial-pca 300	RidgeR GridSearch
 alexnet-pca-RidgeR-1.3		46.7645	45.536	44.589	AlexNet	features.12 - Ma	16384	pre-trained ImageNet-1K	partial-pca 200	RidgeR GridSearch
 alexnet-pca-RidgeR-1.2	40.702	46.527	45.009	44.065	AlexNet	features.12 - Ma	16384	pre-trained ImageNet-1K	partial-pca 150	RidgeR GridSearch
 vgg16-features43-pca100-RidgeR-a1e5			44.66	43.04	VGG-16_BN	Features.43		pre-trained ImageNet-1K	partial-pca 100	RidgeR GridSearch
 efficientnetb2-flatten-pca512-RidgeR			44.4	43.28	EfficientNet	Features.8	68992	pre-trained ImageNet-1K	partial-pca 512	RidgeR GridSearch
 alexnet-pca-RidgeR-1.1		45.745	43.8	43.05	AlexNet	features.12 - Ma	16384	pre-trained ImageNet-1K	partial-pca 100	RidgeR GridSearch
 alexnet-pca-linearR-1.2	39.56	45.25	43.49	42.67	AlexNet	features.12 - Ma	16384	pre-trained ImageNet-1K	partial-pca 100	LinearR OLS
 alexnet-pca-RidgeR-1.2		45.22	43.15	42.24	AlexNet	features.12 - Ma	16384	pre-trained ImageNet-1K	partial-pca 100	RidgeR GridSearch
 efficientnetb2-flatten-pca512-RidgeR 			 34.53	34.25	EfficientNet	Flatten		pre-trained ImageNet-1K	partial-pca 512	RidgeR GridSearch
 alexnet-pca-RidgeR-1.0		15.987			AlexNet	2 - MaxPool2D	46656	not-trained	partial-pca 100	RidgeR GridSearch
 alexnet-pca-linearR-1.1	14.338	15.796			AlexNet	features.12 - Ma	16384	not-trained	partial-pca 100	LinearR OLS
 alexnet-pca-linearR-1.0	10.279	11.49			AlexNet	2 - MaxPool2D	46656	not-trained	partial-pca 100	LinearR OLS

↳ New

Test Score: score su test set (ottenuto mediante submission)

Val Score: score su validation set (calcolato)

S1L e S1R: Val score solo su emisferi del primo soggetto (calcolato)

Possibili Sviluppi

Possibili sviluppi

1. Feature Extraction

- A. Scelto il modello migliore tra quelli testati, provare a concatenare orizzontalmente vettori di features da più layer
- B. Merging Layers:
 - Estrazione di feature da **layer diversi** in funzione del ROI del vertice da prevedere
 - Fissato il modello, trovare i layer che performano meglio per un certo ROI, in modo da usarne le feature estratte per predire i vertici associati.
 - Idea: gli early layers dei modelli, tipicamente, sono migliori nel prevedere gli early visual layer della mente (come ROI V1), discorso analogo vale per i late layers
- C. Merging Networks
 - Estrazione di feature da **modelli diversi** in funzione del ROI del vertice da prevedere
 - DNNs trained for low-level visual tasks predict activity in early brain regions accurately, while DNNs trained for 3-dimensional perception tasks and semantic tasks perform well at predicting dorsal and ventral stream activity
- D. Passare da task oriented/optimized CNN/net a brain-oriented net o brain-optimized models: networks closely representing available anatomical and neurophysiological information about brain areas.
 - e.g. GNET:
- E. Finetune delle reti su COCO
- F. Utilizzo di una **rete per il captioning**
- G. Utilizzo dei metadati ricavabili da COCO Database

2. Downsampling

- A. Passare da **Incremental PCA** a **PCA** (potrebbe non essere sufficientemente accurata?)
- B. Altri algoritmi di downsampling, che tengano conto della multicollinearità tra le feature estratte

Possibili sviluppi

3. Regressione

A. Regressione Multivariata (?)

4. Utilizzo di dati aggiuntivi

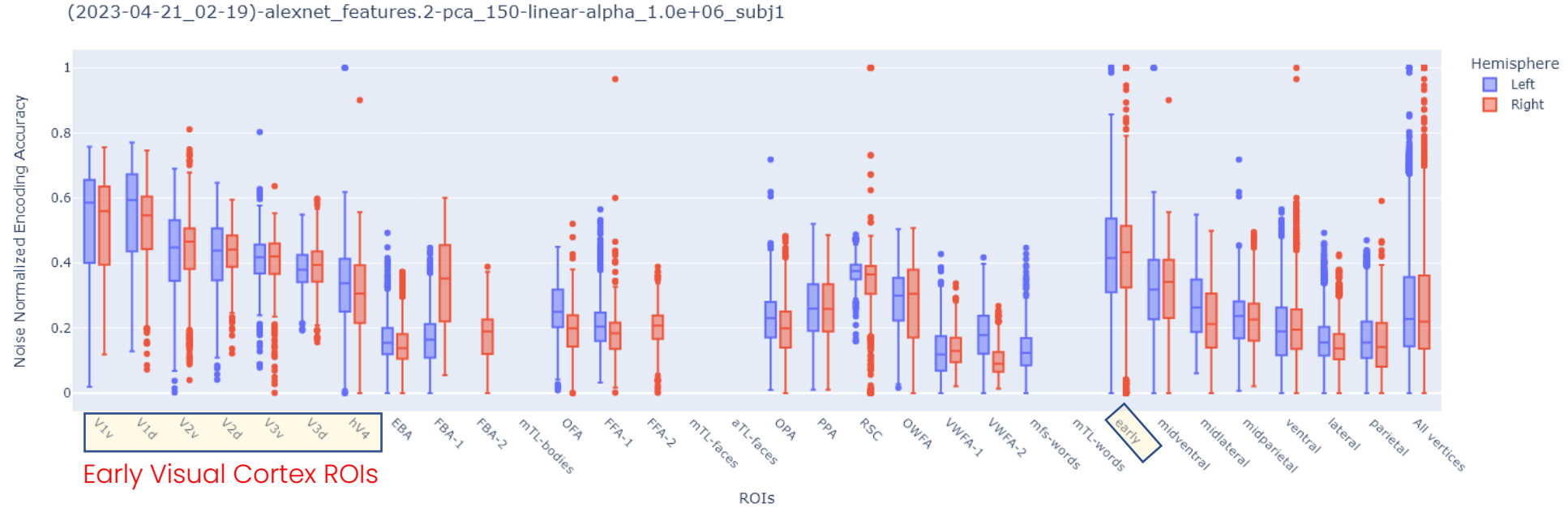
C. Metadati di COCO

D. Dati da Natural Scene Dataset (NSD)

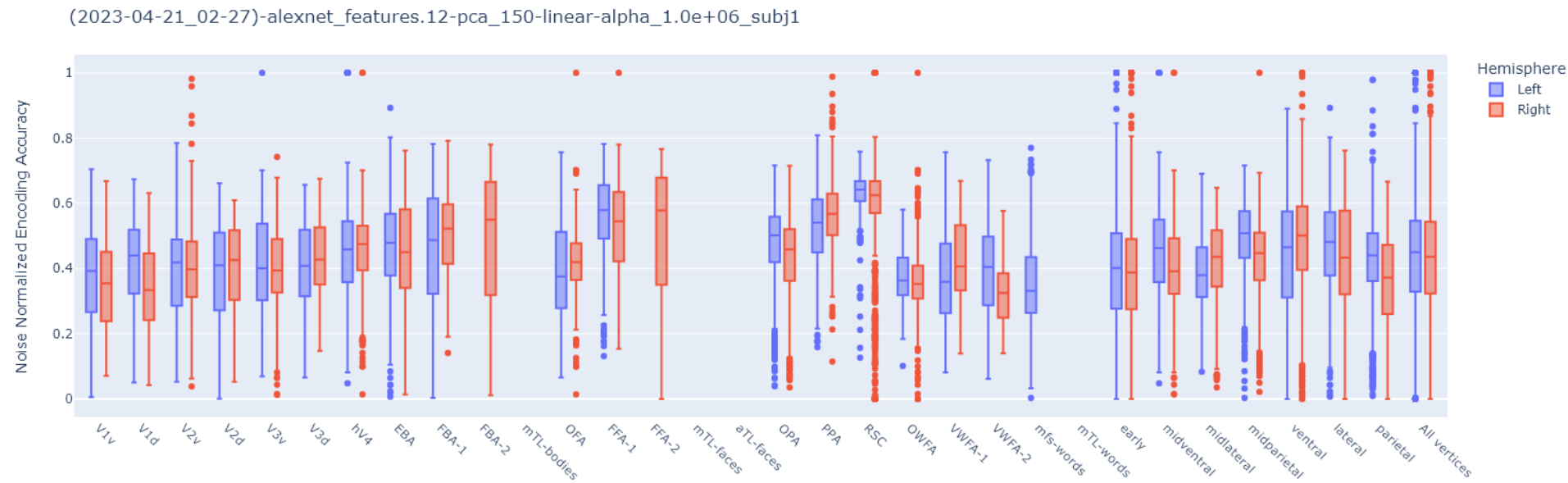
- Raw Data
- Time-series
- Single Trial Bold non mediati e z-scored
- Misure Ausiliarie
 - Anatomical neuroimaging measures (T1, T2, diffusion, venogram, angiogram)
 - Functional neuroimaging measures (resting-state data)
 - Misure comportamentali
 - Dati fisiologici e eye-tracking (solo per alcune sessioni)

Merging Layers: Esempio Alexnet early layer vs late layer

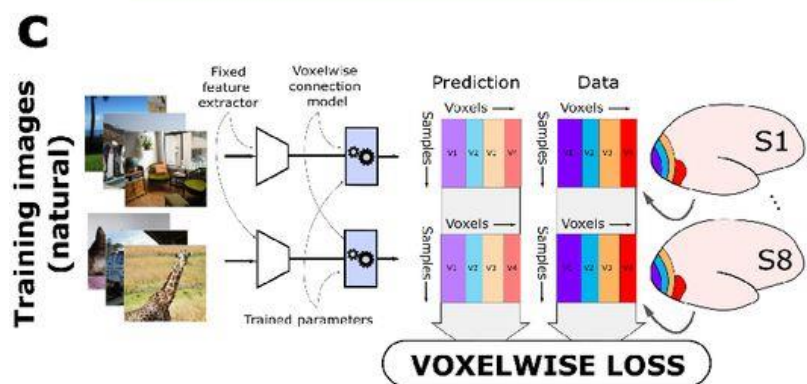
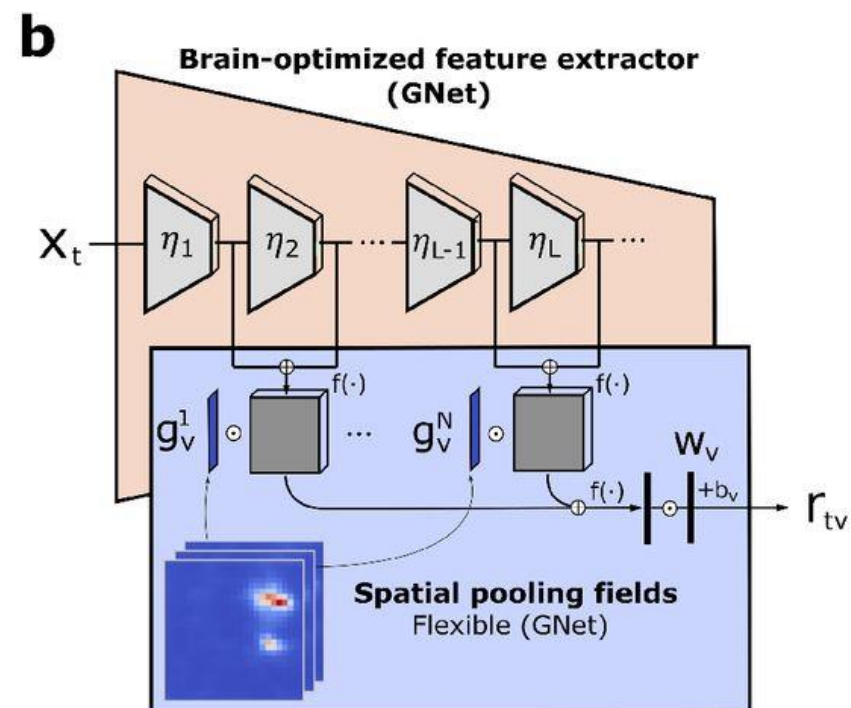
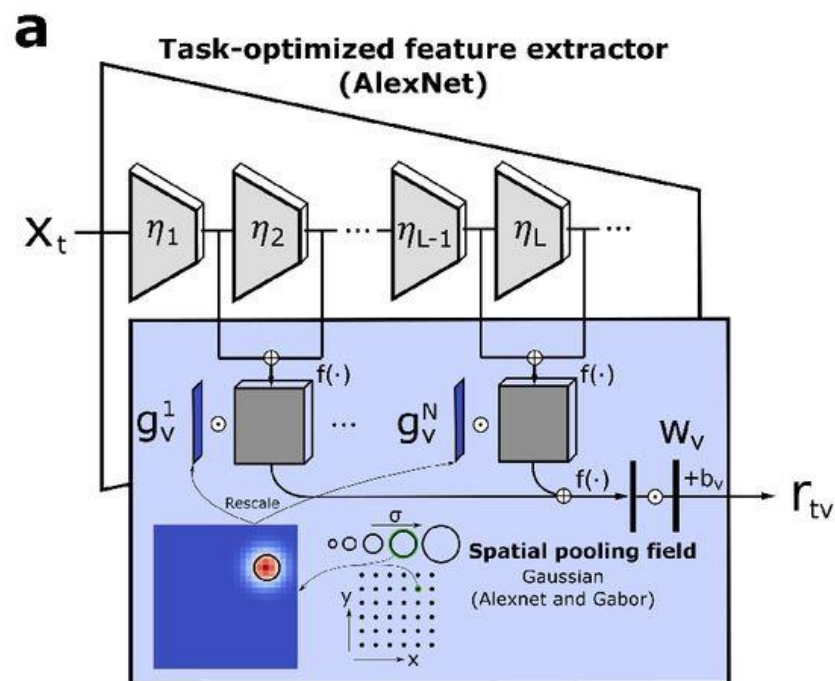
Alexnet **Early Layer**: Layer
MaxPool2d of the first
convolutional block
(features.2)



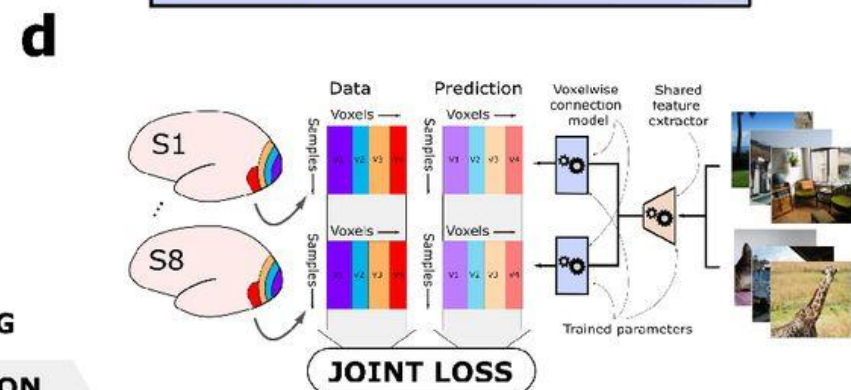
Alexnet **Late Layer**: Layer
MaxPool2d Layer of the last
convolutional block (features.2)



Brain Optimized vs Task Optimized CNN



TRAINING
VALIDATION



Bibliografia

❑ Challenge

- ❑ Gifford, A. T., Lahner, B., Saba-Sadiya, S., Vilas, M. G., Lascelles, A., Oliva, A., Kay, K., Roig, G., & Cichy, R. M. (2023). *The Algonauts Project 2023 Challenge: How the Human Brain Makes Sense of Natural Scenes* (arXiv:2301.03198). arXiv. <https://doi.org/10.48550/arXiv.2301.03198>
- ❑ Allen, E. J., St-Yves, G., Wu, Y., Breedlove, J. L., Prince, J. S., Dowdle, L. T., Nau, M., Caron, B., Pestilli, F., Charest, I., Hutchinson, J. B., Naselaris, T., & Kay, K. (2022). A massive 7T fMRI dataset to bridge cognitive neuroscience and artificial intelligence. *Nature Neuroscience*, 25(1), Article 1. <https://doi.org/10.1038/s41593-021-00962-x>

❑ Encoding Models

- ❑ van Gerven, M. A. J. (2017). A primer on encoding models in sensory neuroscience. *Journal of Mathematical Psychology*, 76, 172–183. <https://doi.org/10.1016/j.jmp.2016.06.009>
- ❑ St-Yves, G., Allen, E. J., Wu, Y., Kay, K., & Naselaris, T. (2022). *Brain-optimized neural networks learn non-hierarchical models of representation in human visual cortex* (p. 2022.01.21.477293). bioRxiv. <https://doi.org/10.1101/2022.01.21.477293>
- ❑ Wu, M. C.-K., David, S. V., & Gallant, J. L. (2006). Complete functional characterization of sensory neurons by system identification. *Annual Review of Neuroscience*, 29, 477–505. <https://doi.org/10.1146/annurev.neuro.29.051605.113024>
- ❑ Naselaris, T., Kay, K. N., Nishimoto, S., & Gallant, J. L. (2011). Encoding and decoding in fMRI. *NeuroImage*, 56(2), 400–410. <https://doi.org/10.1016/j.neuroimage.2010.07.073>
- ❑ Qiao, K., Zhang, C., Chen, J., Wang, L., Tong, L., & Yan, B. (n.d.). *Neural encoding and interpretation for high-level visual cortices based on fMRI using image caption features*.
- ❑ Willeke, K. F., Fahey, P. G., Bashiri, M., Pede, L., Burg, M. F., Blessing, C., Cadena, S. A., Ding, Z., Lurz, K.-K., Ponder, K., Muhammad, T., Patel, S. S., Ecker, A. S., Tolias, A. S., & Sinz, F. H. (2022). *The Sensorium competition on predicting large-scale mouse primary visual cortex activity* (arXiv:2206.08666). arXiv. <https://doi.org/10.48550/arXiv.2206.08666>
- ❑ Yamins, D. L. K., & DiCarlo, J. J. (2016). Using goal-driven deep learning models to understand sensory cortex. *Nature Neuroscience*, 19(3), Article 3. <https://doi.org/10.1038/nn.4244>