Final Project - fMRI Data
Stat 215A, Fall 2014

Fanny Perraudeau

This project can be reproduced from the github repository
`https://github.com/fperraudeau/final_project_git`.
Make sure the read the README file before to start the reproduction of the project.

Abstract

In this project, I explored fMRI data provided by the Gallant Lab at UC Berkeley that measure the brain's response to visual images. The study was focused on the response of twenty voxels located in the visual cortex responsible for visual functions. A subject was shown 1,750 pictures while his brain activity was registered. Each image could be reduced to a vector of dimension 10,921. For this project, I focused on two main goals. The first was to predict the voxels response to new images by fitting GLM with ridge, elastic net (with elastic net parameters α of 0.25, 0.5 and 0.75) and LASSO regularizations. I compared these five models using the model selection criteria: AICc, BIC, CV and ESCV. The performance of my models was measured using the correlation between the fitted values and observed values on a data set untouched when the models were fitted. The two models with the highest correlation and the smallest number of features selected were LASSO and elastic net with $\alpha = 0.5$. Looking at the residuals and stability of my best two models, I had good confidence that the choice of these two models was appropriate. The second goal of this project was to interpret the prediction models and understand how voxels respond to images. To select the most important features, the two models were fitted accross 1,000 different bootstrap samples. Features with non-null coefficients for most of the bootstrap samples were selected for two voxels. The real part of these Gabor wavelets was plotted according to the pixel location on the image. For the two voxels chosen, voxels seemed to respond only to a particular area of the image. Finally, voxels were clustered according to their parameters of interest β. With five clusters, some voxels seemed to cluster according to their location in the brain. It means that some voxels with similar responses to images could be located close to each other.

1 Introduction

Functional Magnetic Resonance Imaging (fMRI) is one of the tools used in neuroscience to decode brain activity. For this project, we looked at fMRI data provided by the Gallant Lab at UC Berkeley measuring activity in visual cortex. In an experiment, a subject is shown a series of randomly selected natural images and the fMRI response from his primary visual cortex is recorded. The fMRI response is recorded at the voxel level, where each voxel corresponds to a tiny volume of the visual cortex. For this project, we focused on two main goals. First, we predicted the voxel response to new images. Secondly, we interpreted the prediction models to understand how voxels respond to images.

fMRI data were recorded from 20 voxels while a subject viewed 1,750 natural images. Each image is a

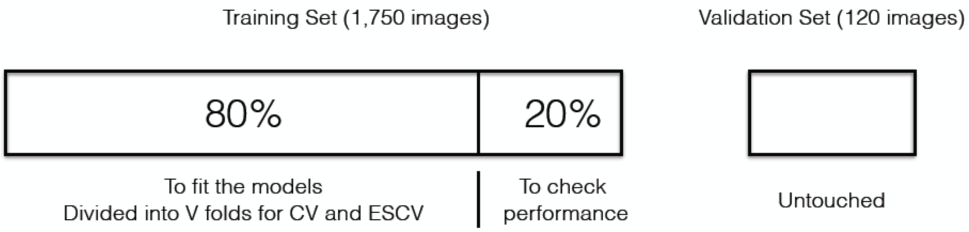
Training Set (1,750 images)
Validation Set (120 images)
80%
20%
To fit the models
Divided into V folds for CV and ESCV
To check
performance
Untouched

Figure 2.1: Partition of the data

128 by 128 pixel gray scale image, which can be represented by a vector of length $128^2 = 16,384$. Through a Gabor wavelet transformation each image was reduced to a vector of 10,921 coefficients. The prediction performance of my best model will be evaluated by looking at correlation scores against a validation set of 120 images.

2 Predict the voxel response

By modeling each of the 20 voxels response to 1,750 images of the training set, I built models that can predict the response to new images. The strength of these models is that the 120 images in the validation set were not part of the training set. To evaluate the performance of my best model I will predict voxels response to totally new images.

One of the simplest prediction method would be to use ordinary least squares estimates. As our data set had high dimensions, least squares would predict the responses with low bias but high variance making this method a bad prediction method. Moreover, it would be difficult to interpret the results of a least squares method because the number of features would not be reduced. In order to see the big picture and reduce prediction error, we wanted to reduce the number of predictors and choose the features that have the strongest influence on the response. Reducing the number of features is likely to increase the bias but at the same time will probably reduce the variance. To reduce the number of features, I fitted Generalized Linear Models (GLM) with different regularizations.

2.1 Partition of the data

To predict the voxel response to new images, I used the training set provided for the project (with 1,750 images). The validation set with the 120 new images was not used for this part. As suggested in the tasks, the size of the data set is large so it is difficult to perform cross-validation on the entire training set. Thus, inspired by section 7.2 of Hastie's book[**2**], I divided the training set into two parts. 80% of the training set was used to train the models while the remaining 20% was used to check the performance of the models. See Figure2.1.

2.2 GLM with Ridge, Elastic Net and Lasso regularization

The training set was fitted to GLM with ridge, elastic net and Least Absolute Shrinkage and Selection Operator (LASSO) regularizations using Gaussian family models. Thus, for a response with only one voxel the objective function to minimize over β was

$$\frac{1}{2N}\sum_{i=1}^{N}(y_i-\beta_0-x_i^T\beta)^2 \;+\; \lambda\;[\;\frac{(1-\alpha)}{2}\|\beta\|_2^2+\alpha\|\beta\|_1\;]$$

where $x_i \in \mathbb{R}^p$ with p = 10,921 the observations, $y_i \in \mathbb{R}$ the response, N the size of the training set (80% of the 1,750 images), $\beta \in \mathbb{R}^p$ *and* $\beta_0 \; \in \mathbb{R}$. The elastic-net penalty is controlled by α, and bridges the gap between lasso ($\alpha = 1$) and ridge ($\alpha = 0$). The tuning parameter λ controls the overall strength of the penalty.

As our response was not a vector but a 1,750 by 20 matrix, I fitted a multi-response linear regression with the objective function

$$\frac{1}{2N}\sum_{i=1}^{N}\|(y_i-\beta_0-\beta^T x_i)\|_F^2 \;+\; \lambda\;[\;\frac{(1-\alpha)}{2}\|\beta\|_F^2+\alpha\sum_{j=1}^{p}\|\beta_j\|_2\;]$$

where β_j is the jth row of the p by K matrix β. I replaced the absolute penalty on each single coefficient by a group lasso penalty on each K-vector β_j for a single predictor x_j. It is known that the ridge penalty shrinks the coefficients of correlated predictors towards each other while the LASSO tends to pick one of them and discard the others. The elastic-net penalty mixes these two. An $\alpha = 0.5$ tends to select the groups in or out together. The small elastic net parameters perform much like the ridge regularization whereas big elastic net parameters perform more like a LASSO regularization.

The R function glmnet from the package glmnet[**3**] was used to fit the linear models with ridge ($\alpha = 0$), elastic net with $\alpha = (0.25,\ 0.5,\ 0.75)$ and LASSO ($\alpha = 1$) regularizations. The option family = "mgaussian" was used to fit multi- response linear regression for our 20 voxels at a time.

2.3 Model selection criteria

To select the smoothing parameter in the five proposed models, four different selection criteria were used

1. Akaike Information Criterion corrected (AICc)
2. Bayesian Information Criterion (BIC)
3. Cross-Validation (CV)
4. Estimation Stability with Cross-Validation (ESCV)

Once the smoothing parameter was chosen for each criterion using the training set (80% of the data set with 1,750 images), the model performance was checked on the validation set (the remaining 20% of the data set). The model with the best performance on the validation set was considered the best of the five models.

2.3.1 AICc

Akaike Information Criterion (AIC) is

$$AIC = 2k - 2\ln(L)$$

where k is the number of parameters in the model, and L is the maximized value of the likelihood function. AIC not only rewards goodness of fit, but also includes a penalty that is an increasing function of the number of estimated parameters. The penalty discourages overfitting that is very likely with our 10,921 initial parameters. It can tell nothing about the quality of the model in an absolute sense (if all the candidate models fit poorly, AIC will not give any warning of that), but can be used to compare different models.

When the number of parameters if large compared to the number of samples, it is recommended to use the Akaike Information Criterion with a correction (AICc) instead of the AIC. AICc is

$$AICc = AIC + \frac{2k(k+1)}{n-k-1}$$

where k is the number of parameters and n is the sample size. Thus, AICc is AIC with a greater penalty for extra parameters because our data set has $n << k$. The best model chosen according to the AICc was the one with the minimum AICc value.

2.3.2 BIC

To penalize even more strongly the number of parameters, I also used the Bayesian Information Criterion (BIC)

$$BIC = k[\ln(n) - ln(2\pi)] - 2\ln(L)$$

where k is the number of parameters in the model, n is the sample size and L is the maximized value of the likelihood function. The idea behind BIC is the same as the one behind AICc but with a penalty term larger in BIC than in AICc. See part 2.4.2 for a more detailed comparison between AICc and BIC.

2.3.3 Cross-Validation

To choose the smoothing parameter using Cross-validation (CV), I divided my training set (80% of the data set with 1,750 images) into 10 folds. At each step, the models were fitted on nine folds with a set of 100 different values for the smoothing parameter λ. The range of the λ tested was from 0 (unregularized solution) to λ_{max} (the smallest λ with all coefficients equal to zero). We will see in the section about ESCV why I chose to index the set of the smoothing parameters. The remaining fold not used to fit the models was used to check the performance of the models. As the performance indicator used by the Gallant lab is the correlation between the fitted values and the response values, I chose to use the correlation to measure the performance of my models on the validation sets. Once the ten steps were completed (in parallel using the R function foreach), the validation correlations were averaged over the steps and the voxels. For each of my five models, the smoothing parameters giving the highest correlation between the fitted and the response values were chosen to be the best parameters for the models. Thus, for each of the five models, the best smoothing parameter was chosen. The steps of my algorithm are described below.

Finally, the performance of the models was evaluated on the untouched validation set (20% of the training set) using the correlation between the fitted and response values. Using CV as a model selection criteria, the best model was the one with the highest correlation.

The strength of CV is to be able to find good solutions with less coefficients than AICc or BIC. However, I realized that performing several times my CV algorithm, I got different solutions each time. It was confirmed by the paper of Bin Yu and Chinghway Lim on ESCV[**1**], CV can lead to models that are unstable in high-dimensions, and consequently not suited for reliable interpretation.

Summary of my algorithm for CV

```
for $\alpha$ = 0 (ridge), 0.25, 0.5, 0.75, 1 (lasso) do
    for $k$ = 1, 2, ..., $V$ = 10 do
        for $\lambda = \lambda_{max}$, ..., 0 do
            1) Train Model on 9 folds using R function glmnet
            2) Predict the response for the remaining fold
            3) Compute the correlation between the fitted and response values on this remaining fold
        end
    end
    4) Average the correlation over the k's and the 20 voxels
    5) Choose the $\lambda$ with the highest correlation
    6) Compute the correlation for the untouched validation set with this $\lambda$
end
7) Choose the model with the highest correlation
```

2.3.4 Estimation Stability with Cross-Validation

The idea of the model selection criteria ESCV[**1**] was to be able to find stable solution for the smoothing parameter in high dimension. To do so the algorithm was nearly the same as the CV algorithm. The training set was divided into V folds. The models were fitted to V - 1 folds of the training set and the performance was checked on the remaining fold. The difference with the CV algorithm was that instead of checking the performance of the fitted model using correlation, the performance was evaluated using the ES metric defined in Bin Yu's paper[**1**]

$$ES[\lambda] = \frac{\widehat{Var}(\widehat{Y}[\lambda])}{\|\overline{\widehat{Y}}[\lambda]\|_2^2}$$

with

$$\widehat{Var}(\widehat{Y}[\lambda]) = \frac{1}{V}\sum_{k=1}^{V}\|\widehat{Y}[k;\lambda] - \overline{\widehat{Y}}[\lambda]\|_2^2 \qquad \textit{and} \qquad \overline{\widehat{Y}}[\lambda] = \frac{1}{V}\sum_{i=1}^{V}\widehat{Y}[i;\lambda]$$

At each step, the same set of indexed lambda as for the CV was tested from λ_{max} to $\lambda = 0$. It is important to keep the same indexation of λ at each step of the ESCV algorithm because the L1-norm of the unregularized solution corresponding to the saturated fit can vary a lot depending on the folds selected. Because it was easiest to implement with the R glmnet package, I chose to represent the regularization parameter with λ whereas the regularization parameter chosen in Bin Yu's paper[**1**] was $\tau = \|\widehat{\beta}(\lambda)\|_1$. I also implemented the ESCV algorithm for ridge and elastic net regularizations, not implemented in Bin Yu's paper. They seemed to perform as well as the LASSO regularization.

Table 1: Correlation on the validation set and number of non null coefficients

Alpha	**Correlation**				**Non null coefficient**			
	AICc	BIC	CV	ESCV	AICc	BIC	CV	ESCV
0 (Ridge)	0.41	0.41	0.54	0.40	7.6	7.3	10	9
0.25	0.42	0.42	0.52	0.42	9.0	8.8	34	31
0.5	0.42	0.42	0.50	0.45	15.2	14.3	69	57
0.75	0.41	0.41	0.51	0.43	27.7	19.3	166	148
1 (LASSO)	0.39	0.40	0.52	0.43	56.8	20.9	296	276

The main strength of the ESCV criterion is its stability. I run several times my ESCV algorithm and unlike the CV algorithm I found about the same smoothing parameter each time. The other strength of ESCV is that it can be computed at the same time as the CV algorithm because it uses the same folds and steps as the CV algorithm. Thus, it gave us two criteria for the computation cost of one.

Summary of my algorithm for ESCV

```
for $\alpha$ = 0 (ridge), 0.25, 0.5, 0.75, 1 (lasso) do
    for $k$ = 1, 2, ..., $V$ = 10 do
        for $\lambda = \lambda_{max}$, ..., 0 do
            1) Train Model on 9 folds using R function glmnet
            2) Predict the response for the remaining fold
            3) Compute the ES metric on this remaining fold
        end
    end
    4) Average the ES metric over the k's and the 20 voxels
    5) Compute the $\lambda$ with the smallest ES metric
    5 bis) Choose the highest $\lambda$ between $\lambda$ from CV and $\lambda$ from ESCV
    6) Compute the correlation for the untouched validation set with this $\lambda$
end
7) Choose the model with the highest correlation
```

2.4 Results

2.4.1 Comparison between the regularization methods

All the models with regularization parameters try to minimize a penalized sum of residuals, but used different penalization. As the sample size n of our data set is small compared to the number of features, LASSO should not perform well. Indeed, LASSO selects at most n variables before it saturates. Thus, if in our data set there were more than 1,750 important features, then LASSO would not be appropriate. Looking at the results of GLM with ridge and elastic net regularizations, it seemed than the number of important features is not higher than 1,750. So this restriction of the LASSO is not a real problem here. However, there is another reason why the LASSO should perform poorly. The Gabor wavelets of the observation matrix were correlated, even highly correlated for some of the features. One of the properties of LASSO is that from a group of correlated features it tends to select one from the group and does not care which one is selected. We would expect that our model would automatically select groups of correlated features and give similar coefficients to similar features. With the LASSO, it is not the case. On the contrary, elastic net regularization tends to take into account the grouping of correlated features and selects the groups in and out of the model.

With my computation, the best model seemed to be the GLM with elastic net parameter $\alpha = 0.5$. It is not surprising as it is a compromise between the ridge and LASSO regularizations. To choose the best second model, it was less clear as a good model is a model with a good balance between high correlation and small number of features. I chose the LASSO because the correlation was high for this model. As we started with 10,921 features, even if the LASSO is not the best model to reduce the number of parameters, the numbers of features selected by the LASSO was still only one fifth of the initial number of features.

2.4.2 Comparison between the model selection criteria

Both AICc and BIC are methods to penalize the number of features. When the number of samples n is large, AICc and BIC produce generally different results because BIC applies a much larger penalty for complex models and hence leads to simpler models than AICc. Thus, for n large BIC usually underfits whereas AICc presents more the danger of overfitting[**4**]. It can be seen looking at how AICc and BIC penalize free parameters (2k for AICc and kln(n) for BIC). However, here the number of images n = 1,750 is small compared to the number of features n = 10,921 and there is no big different between the results from AICc and BIC. See results table. For such dimensions (with p » n), it was not surprising to find similar results for the two methods.

Stone[**5**] showed that AIC (or BIC) and cross-validation are asymptotically equivalent. Here, the number of samples is not large enough to consider that we have an approximation of asymptotical results. For my computations, cross-validation got slightly better results in term of prediction with slightly higher correlations for CV than for AICc and BIC. However, the number of features selected by CV was higher than with AICc and BIC. I did not really understood why.

According to Bin Yu's paper[**1**], ESCV should have about the same prediction power as CV and should select simpler and more reliable models. However, with my implementation of the ESCV algorithm, I got slightly better prediction results with the CV than with the ESCV method. It is likely to be due to my implementation of the ES metric which might be slightly different from the one in Bin Yu's paper. The number of selected features was slightly smaller for the ESCV method than for CV confirming the results in Bin Yu's paper.

2.4.3 Model selected

To choose the best two models, I looked at the different model selection criteria and took into account both the prediction power (ie. the higher correlation between the fitted and response values) and the number of features selected. Thus, for further investigation, I chose the GLM with LASSO and elastic net parameter $\alpha = 0.5$ regularizations.

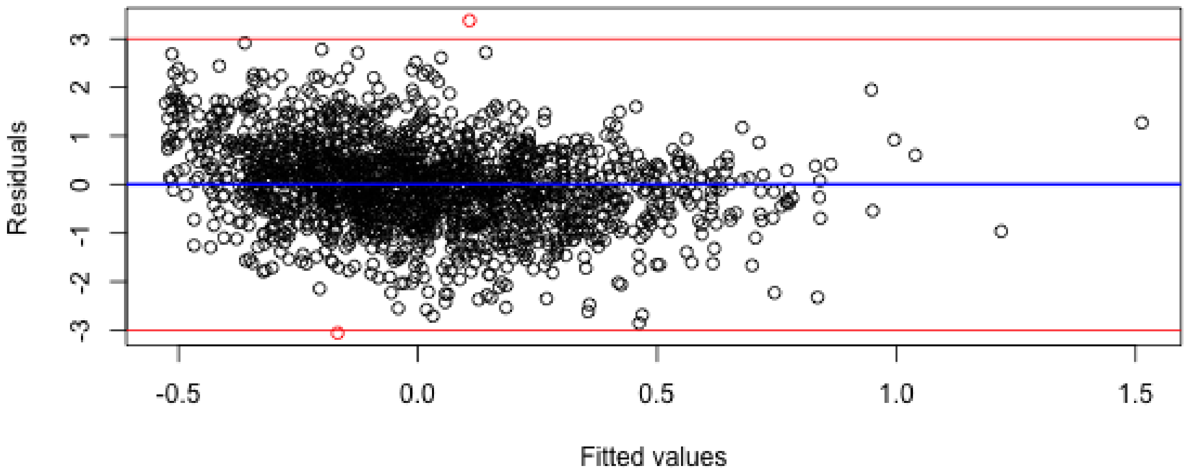

Figure 3.1: Residuals against the fitted values for GLM with elastic net regularization (elastic net parameter is 0.5). The red points are the fitted values with absolute value of the residual up to 3.

3 How do voxels respond to images ?

For this part of the project, I chose to investigate on my best two models, GLM with LASSO regularization and GLM with elastic net parameter equals to 0.5. I also restricted my analysis to two voxels.

3.1 The choice of the models was appropriate

3.1.1 Fit

First, I looked at the fit of my two models plotting the residuals against the fitted values for each of the two voxels. See Figure3.1 the residuals for the GLM with elastic net regularization and voxel 1. The plots for voxel 2 and GLM with LASSO regularization for voxels 1 and 2 were similar. One can see that the residuals were (approximately) independently distributed with a mean of zero. The histograms of the residuals for the two models showed the residuals were approximately normally distributed (data not shown). The residuals versus fitted values plots did not reveal outlier and showed that the choice of the two models was appropriate. Now, let's look at the stability of the models to confirm that.

3.1.2 Stability

To evaluate the stability of the models, correlation between the fitted values and observed values was calculated accross 1,000 bootstrap samples. Observations and responses were sampled with replacement and the coefficients from my two models were used to calculate the fitted values. See Figure3.2 the boxplots of correlation calculated for 1,000 bootstrap samples. The mean of correlation for the two models was about 0.5 with a variance of about 0.01. The stability of the predicted results and of the models seemed to be good.

The analysis of the residuals and stability of the models showed that the choice of these two models was appropriate.

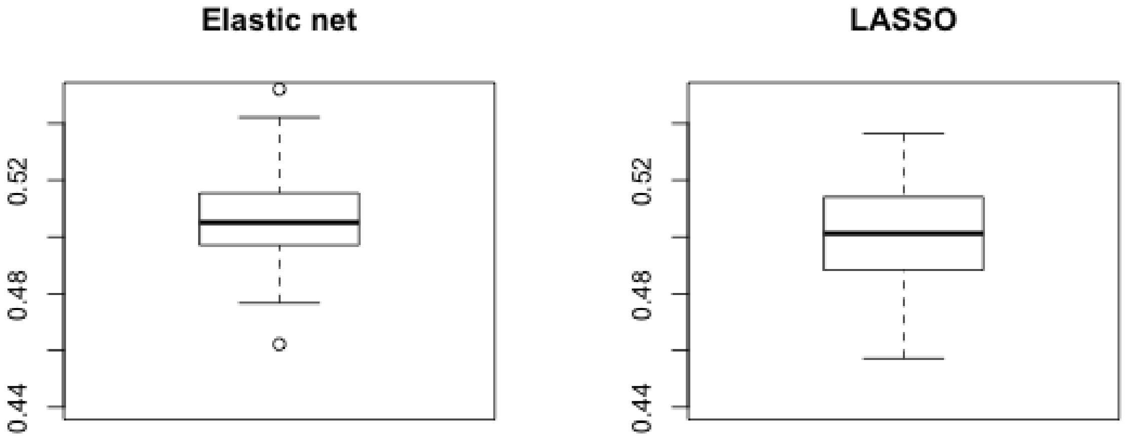

Figure 3.2: Boxplot of the correlation between the fitted and observed values accross 1000 bootstrap samples. On the left, boxplot for GLM with elastic net regularization. On the right, boxplot with GLM with LASSO regularization.

3.2 Important features

Now that we are confident in our two models, we would like to know which features were important for prediction.

I looked at the β coefficients of the models. 107 non-null coefficients were overlapping between the models with elastic net and LASSO regularization. The fact that the features were present in both models could make us think that these features were important to predict the response to images.

To be more confident in the non-null coefficients of the two models, I fitted GLM with elastic net and LASSO regularizations accross 1,000 bootstrap samples of the observations and responses for the first two voxels. Over the 1,000 bootstraps, I counted the number of times a coefficient was non-null. Then, I extracted a list of the ten most frequently non-null coefficients in both of the models for voxels 1 and 2 separately. All the features of these two lists were also present in the overlapping list of the coefficients for my two models (see paragraph above) making me more confident in my results. For voxels 1 and 2, the coefficients were non-null for respectively more than 75% and 65% of the bootstrap samples. Three of the top non-null coefficients overlapped between voxels 1 and 2 (features 1145, 6076 and 10586). Performing this bootstrapping, we could do hypothesis testing on the estimated coefficients because we could compute the sample mean and sample standard error of the coefficients.

Now that we have a criteria to select important features, let's look at the voxels response to images with only the most important features for both of my models. For voxels 1 and 2, I selected the first nine features with the most frequently non-null coefficients in the 1000 bootstrap samples and plotted the real part of the selected Gabor wavelets. See Figure3.3 and Figure3.4 where the axes represent the pixel location of the images for respectively voxel 1 and voxel 2. One can see that most of the wavelets were located in about the same part of the image. Biologically, we expect each voxel to respond only to a particular area of the visual receptive field[**1**]. Thus, my models seemed to be appropriate from a biological point of view. More time would be needed to look at the other eighteen voxels and understand why voxels focus their response

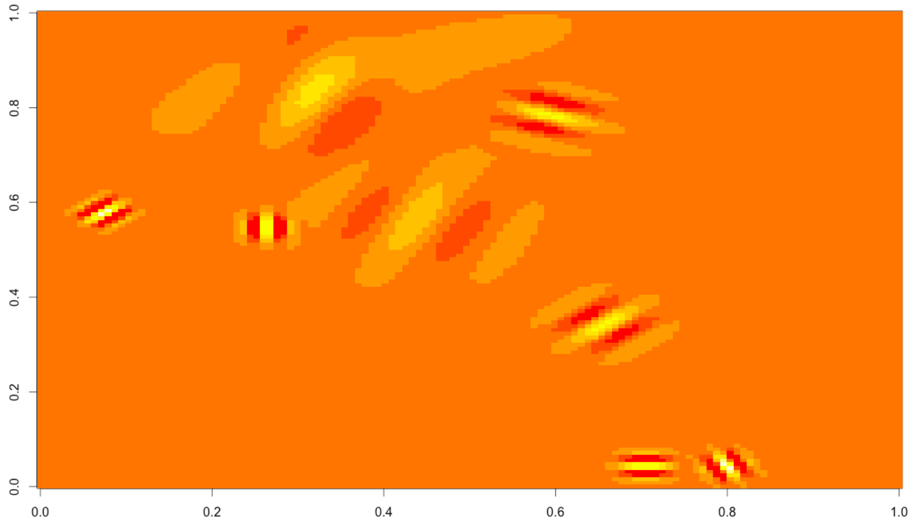

Figure 3.3: Real part of the nine most important wavelets in the response to images for Voxel 1. The most important wavelets chosen were the wavelets with the most frequently non-null coefficients after 1000 bootstrap samples. The axes represent the pixel location of the images.

on a part of the images.

3.3 Clustering the voxels

Another approach to interpret the response of voxels to images was to study the relation between the responses of the twenty voxels and the location of these voxels in the brain. Here I clustered voxels that seem to have similar responses to images. Fitting my models to the responses of the twenty voxels, each voxel got a different parameter of interest β with a different regularization parameter λ. I used the parameters of interest of the twenty voxels for the clustering. The idea was that if voxels have similar parameters of interest, they might have similar responses to images. Then, I wondered if voxels having similar responses were located close to each other in the brain. We could think that similar functions of the visual cortex are executed by voxels that are close to each other. For example, we could think that voxels on the left of the brain are responsible for the analysis of the left part of images and the right part of the brain take care of the right part of images.

To cluster the twenty voxels, I used the R function kmeans with different numbers of center from two to nine. When the number of clusters was three, all the voxels clustered in the same cluster except for voxels four and nine clustering separately. Thus, one can think that these two voxels have the most different responses compared to the other voxels. It could be interesting to study these two voxels with more care. However, this clustering with three clusters did not seem to be the most interesting clustering because there was no relation between the clusters and the location of the voxels.

To infer the best number of clusters, I used subject matter selection looking at the location of the voxels in the visual cortex. I colored the voxels according to the cluster they belonged to. I found that five clusters was the best number of clusters if we are interested in the location of the voxels in the brain. See Figure3.5. With five clusters, five voxels close to each other in the visual cortex clustered together. Three other voxels clustered separately and were also close to the five voxels clustering together. The voxels we studied in this project were arranged in kind of a circle shape (see Figure3.5). These eight voxels that seemed interesting (the five voxels in the same cluster plus the three others clustering separately) were located in the middle and on one side of the twenty voxels we studied in this project. This could suggest that these voxels have different functions than the voxels in the periphery. Further investigation would be needed to

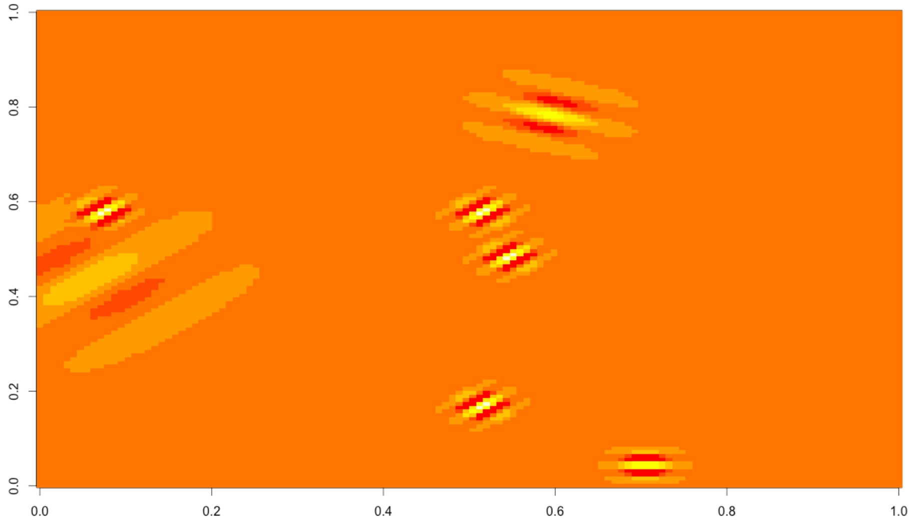

Figure 3.4: Real part of the nine most important wavelets in the response to images for Voxel 2. The most important wavelets chosen were the wavelets with the most frequently non-null coefficients after 1000 bootstrap samples. The axes represent the pixel location of the images.

better understand the responses of voxels to images.

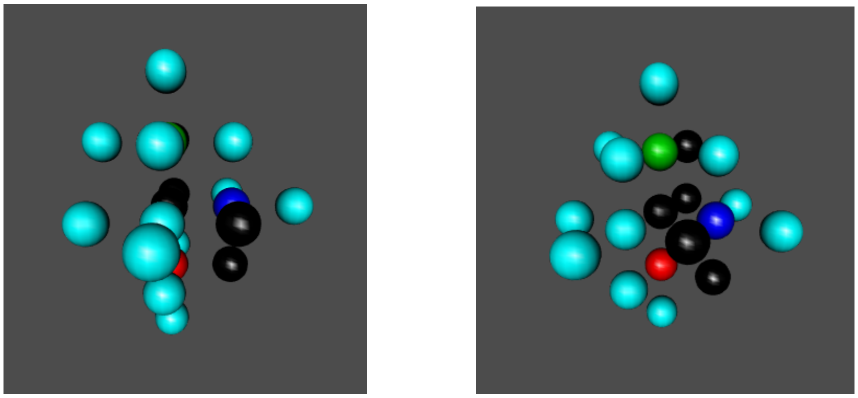

Figure 3.5: Location of the twenty voxels in the visual cortex. The five different colors correspond to the clusters the voxels belong. The two images correspond to the same figure but with different views.

Conclusion

My best models were the GLM with elastic net (elastic net parameter $\alpha = 0.5$) and LASSO regularizations. Looking at the residuals and stability of these two models, I had a good confidence that the choice of my models was appropriate. Selecting the most important features using 1,000 bootstrap samples, it seemed that the selected features focused on a particular area of the images. This result matches with what is biologically expected. Finally, I clustered the voxels according to their response to images. With five clusters, I found that some voxels with similar response to images were located close to each other. It could mean that voxels with similar functions are located in the same area of the visual cortex. Further investigation would be needed to better understand how voxels respond to images.

Prediction for the validation set

To predict the response of the first voxel on the 120 images of the validation set with 120 images provided for the project, I used the model Generalized Linear Model with elastic net regularization with the elastic net parameter $\alpha = 0.5$. The regularization parameter was $\lambda = 0.25$ and the number of non-null coefficients was 67.

References

[1] Lim Chinghway, Yu Bin, *Estimation Stability with Cross Validation*, 2013

[2] Trevor Hastie, Robert Tibshirani, and Jerome Friedman, *The Elements of Statistical Learning*, Springer Series in Statistics Springer New York Inc., 2001

[3] Jerome Friedman, Trevor Hastie, Robert Tibshirani, *Regularization Paths for Generalized Linear Models via Coordinate Descent*, Journal of Statistical Software 2010

[4] Burnham, K. P.; Anderson, D. R., *Model Selection and Multimodel Inference: A Practical Information-Theoretic Approach*, Springer-Verlag, 2002

[5] M. Stone, *An Asymptotic Equivalence of Choice of Model by Cross-Validation and Akaike's Criterion*, Journal of the Royal Statistical Society, 1997