Temporal Feature Selection for fMRI Analysis

Mark Palatucci
School of Computer Science
Carnegie Mellon University
Pittsburgh, Pennsylvania
markmp@cmu.edu

ABSTRACT

Recent work in neuroimaging has shown that it is possible to classify cognitive states from functional magnetic resonance images (fMRI). Machine learning classifiers such as Gaussian Naive Bayes, Support Vector Machines, and Nearest Neighbors have all been applied successfully to this domain. Although it is a natural question to ask which classifiers work best, research has shown that the accuracy of a classifier is intimately tied to the underlying feature selection (or generation) method.

Most of these feature selection methods search spatially for voxels that discriminate classes well. An empirical analysis shows, however, that voxels that discriminate well at a given time point may not discriminate well at another time point. Thus without considering this temporal component we risk passing more noise to the classifier than necessary. Choosing features temporally, focusing on *regions of time* when voxels discriminate well, can reduce this noise. We present empirical results that show that this method yields highly accurate classifiers with far fewer features than methods that only consider spacial information.

Categories and Subject Descriptors

I.5.2 [**Pattern Recognition**]: Design Methodology—*classifier design and evaluation, feature evaluation and selection*

Keywords

functional magnetic resonance imaging, neuroimaging, cognitive state classification

1. INTRODUCTION

1.1 Cognitive State Classification

Functional magnetic resonance images (fMRI) have proven useful for studying the behavior of human brains. These images measure the neural activity [1] of the brain at different locations and can be used to decipher how the brain responds to various stimuli. For example, researchers may want to understand how the brain responds to human faces. By placing a human subject into a fMRI scanner and exposing him to images of faces, they may discover that a particular region has activity highly correlated with the exposure of faces. This may lead to the conclusion that this region of the brain is specialized to the function of face recognition. In this fashion, a great literature of research has been generated, much of it dedicated to discovering the functional purpose of the various regions of the brain.

Beyond discovering the functional purpose of different brain regions, recent research has shown that it is possible to classify cognitive states from these neural images. One study [8] has shown that it is possible to determine when a subject is either reading a sentence or viewing a picture. In a similar study, the group was able to determine when a subject was reading words within different semantic categories (e.g. fruits, buildings, tools, etc.). Another study has shown it is possible to classify between drug addicted persons and non-drug-using controls [13]. Classification methods have also been used successfully for the purpose of lie detection [2].

1.2 fMRI Datasets

Despite recent progress, training effective classifiers is still a challenging problem. This difficulty is mainly caused by the high dimensional, sparse, and noisy nature of the data. The shear volume of data produced also poses problems. A typical experiment takes a 3D image of the brain every second. Each image is composed of roughly 10,000 voxels [2], each of which measures the neural activity at a particular location within the brain. Figure 1 shows a 2D example of this neural activity.

A typical experiment may be divided into trials, with each lasting approximately 60 seconds. A trial is often repeated several times within an experiment. Figure 2 shows the time series data produced for a single voxel. An experiment with V voxels, T images per trial, and N trials would have $V * T * N$ total data points. A typical experiment may have $V = 10,000$, $T = 60$, and $N = 30$, yielding 18 million data points.

[1]Technically, fMRI measures the *hemodynamic response* which is a measure of the blood oxygenation level. Current theory suggests that neural activity increases the amount of oxygen brought into the active area.

[2]The actual number of voxels depends on the resolution of the scanner as well as the size of the subject's brain

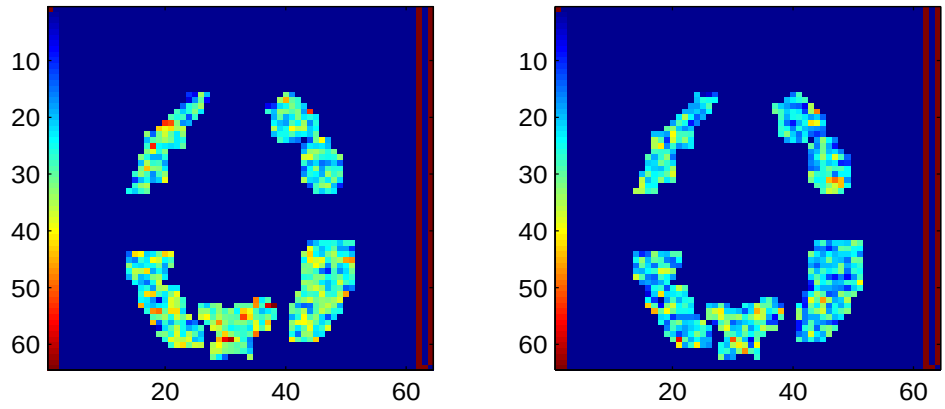

Figure 1: A 2D snapshot of neural activity in response to different stimuli

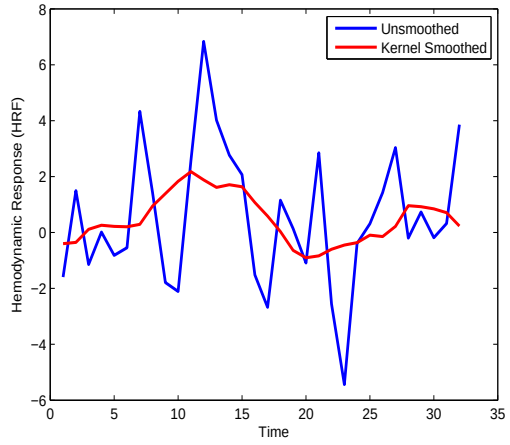

Figure 2: The time series for a single voxel in a fMRI experiment

1.3 Feature Selection

Dealing with this much data is a challenge. Since reducing the dimensionality of the data is critical to making good predictors, numerous methods have been developed that attempt to address this problem.

Classical fMRI analysis has taken a univariate approach [5]. Statistical tests are performed on voxels individually to select which voxels have the highest activation level. The selected voxels are then used in a general linear model to make predictions.

More recently, researchers have tried multivariate techniques such as independent components analysis (ICA) [7], principal component analysis (PCA) [10], and manifold learning techniques [11].

Another popular approach is the *searchlight* method [6]. Voxels with high activation (signal) are often correlated with their neighboring voxels. The searchlight is a simple scheme that passes a 3D window through the brain looking for collective regions of high activation. The window size is usually fixed.

The data becomes more manageable when its dimensionality is reduced. Of course, any time we remove data we risk loosing valuable information that could help our prediction task. To our knowledge, there does not appear to be any gold standard for handling this problem.

1.3.1 Are voxels features?

In many of the feature selection methods described above there appears to be a synonymy between voxels and features. The thinking goes: in order to make a good predictor, we just need to find the underlying voxels that discriminate the classes well. This seems completely natural and appropriate since often the overall goal of fMRI analysis is to find the regions of the brain that are responsible for a particular task.

There is a subtlety, however, that is often ignored. fMRI data are a time series, and voxels that discriminate well at a given time point may not discriminate well at another time point. This is due to the nature of the experiments: typically a stimulus is presented and only after several seconds does the hemodynamic response fully develop (Figure 3). Thus including regions of time when the hemodynamic response is not fully developed can hurt classification, since these regions have lower signal-to-noise ratio.

1.4 Contribution

The main contribution of this work is to address the importance of time when performing feature selection in fMRI analysis. We show a simple method where features are *no longer voxels, but rather voxel-timepoints*. We demonstrate that choosing highly discriminating voxel-timepoints improves classification performance and significantly reduces the number of features necessary for classification. We also show the importance of temporally smoothing data, and show its impact on classification.

Table 1: Percent Accuracy of Temporally Smoothed vs. Unsmoothed Data

	No Smoothing	Smoothed
Subject 1	82.5	92.5
Subject 2	50.0	67.5
Subject 3	80.0	82.5
Subject 4	67.5	90.0
Subject 5	62.5	82.5
Subject 6	25.0	47.5
Subject 7	32.5	60.0
Subject 8	55.0	75.0
Subject 9	75.0	77.5
Subject 10	92.5	92.5
Subject 11	65.0	77.5
Subject 12	10.0	37.5
Subject 13	82.5	95.0

2. TEMPORAL FEATURES

2.1 Kernel Smoothing

fMRI data are inherently noisy. Since the hemodynamic response is assumed continuous, it seems appropriate to consider temporally smoothed data. This technique appears under various names in different communities but is usually called low-pass filtering or kernel smoothing (e.g. Gaussian smoothing). (See Figure 2).

The basic idea is to remove noise while preserving the underlying signal. The difficulty is choosing the amount of smoothing. Too much smoothing destroys signal while too little preserves noise. Choosing the appropriate smoothing is a problem of statistical decision theory and the bias/variance decomposition. Smoothing removes variance and introduces bias. Much work has been done trying to balance bias and variance and to estimate the optimal smoothing factor. Some work [3] also suggests that for problems of classification (rather than signal reconstruction) introducing additional bias (more smoothing) may be beneficial.

We implemented Gaussian kernel smoothing and chose a smoothing factor as described in [12]. Although smoothing is often noted as a footnote in fMRI studies, we found it *can dramatically increase classification accuracy*. Thus we feel it important to explicitly note its use and we present the technical implementation in Appendix A.2.

2.1.1 Classification accuracy of smoothed data

We present the results of a simple fMRI classification task to demonstrate the importance of temporal smoothing. In this experiment, a subject was presented a stimulus for eight seconds. There were two stimuli, pictures and words and each was presented twenty times (twenty word trials, twenty picture trials). The experimental data produced for each subject are a matrix $Voxels * Timepoints * Trials$.

The classification task is to learn the mapping between the trial and the stimulus. If a trial has V voxels over T timepoints, then the classification function f is:

$$f : V \times T \rightarrow \{\text{Picture, Word}\}$$

We used a Gaussian Naive Bayes classifier and treated each voxel-timepoint as a feature in the classifier. We estimated

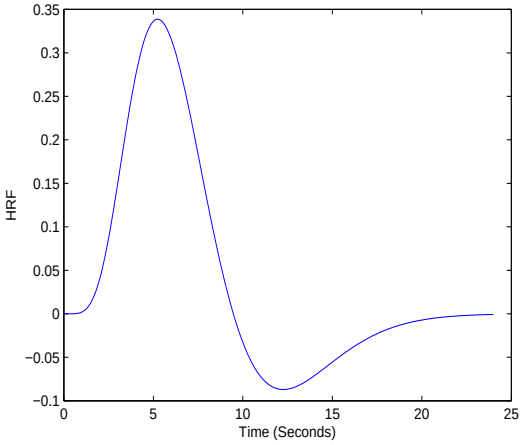

Figure 3: Typical Hemodynamic Response Function

parameters for each feature by calculating the class conditional mean and variance for each voxel-timepoint over the relevant trials. We detail the implementation in Appendix A.1.

We operated directly on all the voxels ($\approx$ 5,000 per subject) and did not perform any feature selection. We used kernel smoothing to generate a new feature set of exactly the same dimension. We performed a leave-one-out-cross-validation (LOOCV) to compare the effects of smoothing.

Table 1 shows the accuracy of classification for smoothed vs. unsmoothed data. For twelve of thirteen subjects in this study, smoothed data produced a more accurate classifier. On the remaining subject (which was the most accurate subject in the unsmoothed experiment), smoothing had no effect.

Smoothing did not hurt performance in any subject and we see large performance gains for the poorer performing subjects. In some subjects, unsmoothed data had near random accuracy. Without smoothing, a researcher might look at these subjects and conclude that classification is not possible for the given task. Smoothing could lead the researcher to an entirely different conclusion.

As a result, we feel that temporal smoothing is an important preprocessing step that should be used for fMRI classification studies.

2.2 Voxel Discrimination Over Time

We now turn to the task of choosing subsets of features. The hope is that if we can remove noisy, non-discriminating features, we can improve our classification performance. As mentioned in the introduction there are many ways to do this feature selection step, most of which focus on spacial selection of voxels.

A spacial classification scheme might consider voxels, rather than voxel-timepoints as features. In this scenario the time series for each voxel may be replaced by a simple statistic such as the mean. Each feature would then become the

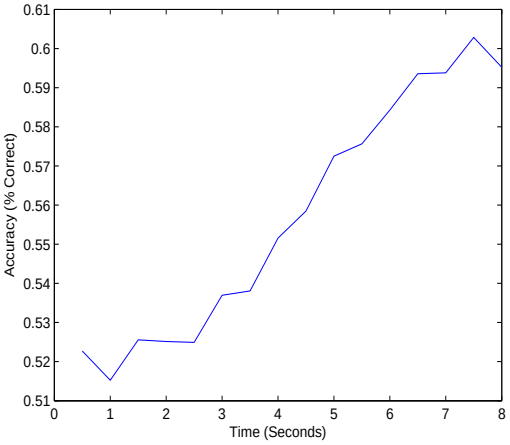

Figure 4: Average accuracy of all voxels at each timepoint (Subject 1)

mean activity of a particular voxel, and then voxels could be ranked by how well their mean activity discriminates a training set. Since we reduce the entire temporal dimension to a simple statistic, we may destroy valuable information.

Instead, we could treat each feature as a voxel-timepoint, yet still rank by voxel. Essentially, each voxel would represent a subset of features (i.e. all the timepoints for that voxel). Ranking voxels would basically mean ranking the subsets of features against each other. The danger here is that some voxels may discriminate well at a given time point but poorly at other timepoints. Ranking by voxel may add individual voxel-timepoints that discriminate poorly, although the voxel's aggregate performance for the entire time series may be reasonable.

Figure 4 demonstrates the discriminating power over the time series (i.e. we compute the average accuracy of all voxels for each timepoint). We see that the discriminating ability increases several seconds after the onset of the stimulus. In hindsight, this effect is not surprising. When we look at a typical hemodynamic response curve in Figure 3 we see peak activation several seconds after the onset of the stimulus.

At the peak of the hemodynamic response, we have the strongest signal-to-noise ratio. We would expect classification accuracy to improve near this peak.

This leads to the simple intuition that we should not rank features entirely by voxel (spatially). Rather, we should take advantage of this difference in temporal discrimination to further reduce noise.

Therefore, we should treat each feature as a voxel-timepoint, and we should rank each voxel-timepoint individually when selecting features.

2.2.1 Classification accuracy of temporally selected features

We present results of two feature selection methods in Table 2. In each case, features are considered voxel-timepoints. The first method, *Discriminate-By-Voxel*, chooses the voxels (all the timepoints for that voxel) that best discriminate the training set. In this experiment, choosing a single voxel resulted in sixteen voxel-timepoints being selected. We tested groups of 50 voxels at a time, up to 4,000 voxels (i.e. 50, 100, 150, ..., 4,000). Thus we tested classifiers with 800, 1600, ..., 60,000 features.

The second method, *Discriminate-By-Voxel-Timepoint* , chooses the voxel-timepoints that best discriminate the training set. We tested groups of 10 voxel-timepoints at a time, up to 2,500 voxel-timepoints (i.e. 10, 20, 30, ... 2,500).

Lessons Learned
There are some interesting insights that can be gained from these results. First, both methods improved performance on all subjects with the largest gains coming from the noisiest subjects. Without feature selection, several of these subjects performed with near random accuracy. Feature selection shows us that there is discriminating signal even within these noisy subjects. This is an important point since often the goal of fMRI research is to determine whether two cognitive states can be discriminated at all (e.g. can fMRI be used to detect lies from truth?). *Feature selection might be the key between two different conclusions.*

Another insight gained is that it is possible to classify with high accuracy with only a small number of features. In both methods, the highest accuracy appeared close to the smallest number of features considered. The original data contained ≈80,000 features per example. With the *Discriminate-By-Voxel-Timepoint* method, we see that a classifier with ≈100 features, leads to a good predictor for all subjects. This leads us to a heuristic that could be useful for this domain:

Feature Selection Heuristic: Treat each voxel-timepoint as a feature. Rank all features by how well they individually classify a training set. Choose the top 100 features for the final classifier

Although there seems to be a minor improvement with the *Discriminate-By-Voxel-Timepoint* method over the *Discriminate-By-Voxel* method, more tests would be required to confirm if this is statistically significant. There is a large computational benefit, however, for the *Discriminate-By-Voxel-Timepoint* method. Since we're only evaluating a very small number of features for each test example, testing time can be greatly reduced.

Future Experiments
We think the Gaussian Naive Bayes is a great classifier for this domain because it appears to be very robust to noise. Adding a few hundred noisy features only seems to have a minor degradation on performance. We suspect other classification methods would be more susceptible to spurious features and thus show a larger discrepancy in performance for the two methods described above. Since feature selection methods are intimately tied to the underlying classifier, it would be interesting to know the results of these two methods with other classification techniques.

Table 2: Accuracy of Feature Selection Methods for 13 Subjects

	1	2	3	4	5	6	7	8	9	10	11	12	13
No feature selection	92.5	67.5	82.5	90.0	82.5	47.5	60.0	75.0	77.5	92.5	77.5	37.5	95.0
Discrim-by-voxel (DV)	95.0	90.0	95.0	97.5	95.0	92.5	92.5	90.0	85.0	97.5	82.5	80.0	100
Discrim-by-voxel-timepoint (DVT)	97.5	95.0	90.0	97.5	97.5	95.0	95.0	92.5	85.0	97.5	90.0	77.5	100
NumFeatures used (DV)	800	800	1600	800	1600	1600	800	800	1600	800	1600	800	1600
NumFeatures used (DVT)	20	10	80	120	60	180	220	10	70	10	50	130	110

Also, it would be useful to run these experiments with a smaller granularity and also to consider a smaller number of voxels for the *Discriminate-By-Voxel* method. With such a small number of training examples, we run the risk that certain features may discriminate well by random chance. As a result, we were surprised that both methods appeared to have highest accuracy with only a small number of features. It would be interesting to know how few voxels or voxel-timepoints we could use before classification performance degrades and to see how this amount changes as a function of the number of training examples.

3. CONCLUSION

There are many ways to create features from fMRI time series data. Most traditional methods have focused on spatially selecting the most discriminating voxels while ignoring the temporal dimension of the data. We discovered empirically, however, that voxels that discriminate well at a given timepoint may not discriminate well at another timepoint. This seems obvious (only in hindsight) given the changing signal strength of the hemodynamic response. By explicitly treating features as voxel-timepoints, we can improve performance by removing regions of time that discriminate poorly.

Using this idea, we demonstrated a simple heuristic that reduces an example of 80,000 dimensions to one of 100 useful features. We found that it yields excellent performance and computational speed. Further, we showed that temporal smoothing further increased classification performance, especially for noisy subjects.

Dealing with noisy subjects is an important problem for fMRI analysis. Often the goal of fMRI research is to determine whether two cognitive states can be discriminated at all. Currently, experiments are run using many subjects in order to minimize the effect of a single noisy subject on the experimental conclusion. In the experiment we considered, several subjects produced very noisy data and had near random classification accuracy. Through kernel smoothing and feature selection, we were able to make dramatic improvements in classification accuracy for these subjects. As we become more confident in our ability to handle noisy subjects, we can run fMRI experiments and test cognitive hypotheses with fewer human subjects. This would lead to a large savings in both cost and time.

We believe time is an essential element for selecting features in fMRI analysis. We hope this work demonstrates temporal feature engineering is useful and stimulates additional research into temporal methods.

4. ACKNOWLEDGMENTS

We would like to thank Tom Mitchell and Francisco Pereira for useful discussions. We would also like to thank Francisco again for his quick implementation of kernel smoothing code.

Mark Palatucci is supported by a NSF Graduate Research Fellowship.

5. REFERENCES

[1] D. D. Cox and R. L. Savoy. Functional magnetic resonance imaging (fmri) brain reading: detecting and classifying distributed patterns of fmri activity in human visual cortex. *NeuroImage*, 19(2):261–270, June 2003.

[2] C. Davatzikos and et al. Classifying spatial patterns of brain activity with machine learning methods: application to lie detection. *Neuroimage*, 28(1):663–668, 2005.

[3] J. H. Friedman. On bias, variance, 0/1loss, and the curse-of-dimensionality. *Data Mining and Knowledge Discovery*, 1(1):55–77, 1997.

[4] I. Guyon and A. Elisseeff. An introduction to variable and feature selection. *Journal of Machine Learning Research*, 3:1157–1182, March 2003.

[5] J.-D. Haynes and G. Rees. Decoding mental states from brain activity in humans. *Natural Reviews: Neuroscience*, 7(1):523–534, July 2006.

[6] N. Kriegeskorte, R. Goebel, and P. Bandettini. Information-based functional brain mapping. *PNAS*, 103(10):3863–3868, March 2006.

[7] M. McKeown, L. Hansen, and T. Sejnowski. Independent component analysis of functional mri: what is signal and what is noise? *Current Opinion in Neurobiology*, 13(1):620–629, 2003.

[8] T. M. Mitchell, R. Hutchinson, R. S. Niculescu, F. Pereira, X. Wang, M. Just, and S. Newman. Learning to decode cognitive states from brain images. *Machine Learning*, 57(1-2):145–175, 2004.

[9] K. Norman, S. Polyn, G. Detre, and J. Haxby. Beyond mind-reading: multi-voxel pattern analysis of fmri data. *Trends in Cognitive Sciences*, 10(3), 2006.

[10] K. Petersson and et al. Statistical limitations in functional neuroimaging. non-inferential methods and statistical models. *Philosophical Transactions of the Royal Society B:*, 354:1239–1260, 1999.

[11] X. Shen and F. G. Meyer. Nonlinear dimension reduction and activation detection for fmri dataset. In *CVPR Workshop*, 2006.

[12] L. Wasserman. *All of Statistics*. Springer, New York, NY, 2005.

[13] L. Zhang, D. Samaras, D. Tomasi, N. Alia-Klein, L. C. A. Leskovjan, N. Volkow, and R. Goldstein. Exploiting temporal information in functional

magnetic resonance imaging brain data. In *MICCAI Conference Proceedings*, pages 679–687, 2005.

APPENDIX

A. TECHNICAL DETAILS

A.1 Gaussian Naive Bayes Classifier

The Gaussian Naive Bayes classifier is based on Bayes rule:

$$P(Y|X) = \frac{P(X|Y)P(Y)}{P(X)}$$

where Y is our class label $Y \in \{0, 1\}$ and X is a real valued number $X \in \Re$. We model the likelihood $P(X|Y)$ using a Gaussian. Further, if X is a vector, we make the naive assumption that all the X_i are conditionally independent given the class label. Specifically,

$$P(X_1, X_2, \ldots, X_n|Y) = \prod_{i=1}^{n} P(X_i|Y)$$

Our classification rule is to find the class label k that maximizes $P(Y = k|X)$. Since the denominator is the same regardless of k, we can choose the k that maximizes just the numerator. Often the numerator is called the *score function* and is denoted by $\mathcal{S}$. We find k to maximize the score:

$$\begin{aligned} \text{Predicted Class} &= \text{argmax}_k \mathcal{S}(k) \\ &= \text{argmax}_k P(Y = k) \prod_{i=1}^{n} P(X_i|Y = k) \end{aligned}$$

Now for our fMRI classification task we treat each *voxel-timepoint* as a feature. Let V be the number of voxels, T be the number of timepoints, and N be the number of trials. We index a voxel-timepoint for a particular trial as X_{ivt} where $1 \le v \le V, 1 \le t \le T, 1 \le i \le N$. We assume that each feature is distributed normally across trials in the experiment given the class:

$$X_{itv}|Y = k \sim N(\mu_{tv}^{(k)}, \sigma_{tv}^{2(k)})$$

If we let $\widehat{\pi}_k$ and $\widehat{f}(x|Y_i = k)$ denote estimates for the the prior $P(Y = k)$ and likelihood $P(X|Y)$ then our score function then becomes:

$$\begin{aligned} \mathcal{S} &= \widehat{\pi}_k \widehat{f}(x|Y_i = k) \\ &= \widehat{\pi}_k \prod_{T,V} \frac{1}{\sqrt{2\pi}\widehat{\sigma}_{tv}^{(k)}} \exp\Big\{ \frac{-(X_{tv} - \widehat{\mu}_{tv}^{(k)})^2}{2\widehat{\sigma}_{tv}^{2(k)}} \Big\} \end{aligned}$$

It is important to note that we cannot find this score directly. The problem is that we have a very large number of features $(T * V)$. The likelihood of a single voxel-timepoint will be a very small number and multiplying thousands of small numbers together will quickly break the numerical precision of any computer. To correct this problem we can work in *log-space*. Since *log* is a monotonic function, the max of the log will also be the max of the original function. Therefore we compute the log-likelihood instead:

$$\mathcal{S} = log(\widehat{\pi}_k) + \sum_{T,V} -log(\sqrt{2\pi}\sigma_{tv}^{(k)}) - \frac{(X_{tv} - \widehat{\mu}_{tv}^{(k)})^2}{2\widehat{\sigma}_{tv}^{2(k)}}$$

We estimate the class specific mean $\mu^{(k)}$ and variance $\sigma^{2(k)}$ for each voxel-timepoint. (Note: we use $\delta(\cdot)$ as the indicator function. It is 1 when the predicate is true and 0 otherwise)

$$\begin{aligned} \hat{\mu}_{tv}^{(k)} &= \frac{1}{\sum_{i=1}^{N} \delta(Y_i = k)} \sum_{i=1}^{N} \delta(Y_i = k) X_{itv} \\ \hat{\sigma}_{tv}^{2(k)} &= \frac{1}{\sum_{i=1}^{N} \delta(Y_i = k)} \sum_{i=1}^{N} \delta(Y_i = k)(X_{itv} - \hat{\mu}_{tv}^{(k)})^2 \end{aligned}$$

We can compute estimates for the priors by just taking the average as usual:

$$\hat{\pi}_k = \frac{1}{N} \sum_{i=1}^{N} \delta(Y_i = k)$$

A.2 Kernel Smoother

We implemented kernel regression from Wasserman [12] as our smoother. To keep notation consistent with the book, we let our output variable Y be a real number $Y \in \Re$. The basic idea is that each data point is replaced by a weighted combination of the nearby data points. This is sometimes referred to as the Nadaraya-Watson kernel estimator:

$$\widehat{r}(x) = \sum_{i=1}^{n} w_i(x) Y_i$$

The weights $w_i(x)$ are computed using a kernel function K:

$$w_i(x) = \frac{K(\frac{x - x_i}{h})}{\sum_{j=1}^{n} K(\frac{x - x_j}{h})}$$

There are many different kernel functions that can be substituted for K. In the above equation, the variable h is the *bandwidth* or scale of the kernel. Wider bandwidths increase the weight on points farther away. For smoothing tasks, the Gaussian kernel is often used:

$$K(x) = \frac{1}{\sqrt{2\pi}} e^{-x^2/2}$$

The choice of bandwidth is the tricky part. If we choose a bandwidth that is too large we risk over smoothing the data and destroying the underlying signal. If we choose too little smoothing we leave high frequency noise in our data. There has been much work trying to find estimators for the optimal bandwidth. One way to choose the bandwidth is to minimize the cross validation risk:

$$\widehat{J}(h) = \sum_{i=1}^{n} (Y_i - \widehat{r}_{-i}(x_i))^2$$

In this equation $r_{-i}(x)$ is the estimate of r(x) with bandwidth h and the ith data point left out. Like many cross validation estimators, there is a shortcut formula that avoids explicitly computing each r_{-i}.

$$\widehat{J}(h) = \sum_{i=1}^{n} (Y_i - \widehat{r}(x_i))^2 \frac{1}{\Big(1 - \frac{K(0)}{\sum_{j=1}^{n} K\left(\frac{x_i - x_j}{h}\right)}\Big)^2}$$

So we just need to test different values of the bandwidth to find the one that minimizes $\widehat{J}(h)$.