Chapter 10: Generalized Additive Models[a]

In this chapter we consider data points $\{(y_i^0, x_i^0)\}_{i=1}^n$ in $\mathbb{S}\times\mathbb{R}^p$, with $\mathbb{S}\subseteq\mathbb{R}$, and assume the following regression model

$$Y_i^0 \sim f(y;\mu_i,\phi)\mathrm{d}y, \quad g(\mu_i)=\alpha+\sum_{j=1}^p f_j(x_{ij}^0), \quad i=1,\dots n \tag{10.1}$$

where $\alpha\in\mathbb{R}$ and $\phi\in(0,\infty)$ are two parameters, $g\in\mathcal{C}^2(\mathbb{R})$ is an invertible link function and where $f(y;\mu_i,\phi)\mathrm{d}y$ belongs to an exponential family of distributions, that is, for all $\mu\in\mathbb{R}$ and $\phi\in\mathbb{R}$ we have

$$f(y;\mu,\phi)=\exp\Big(\frac{y\,\theta-b(\theta)}{a(\phi)}+c(y,\phi)\Big), \quad \forall y\in\mathbb{S} \tag{10.2}$$

for some invertible function $b\in\mathcal{C}^3(\mathbb{R})$ and function $c:\mathbb{R}\times\mathbb{R}\to\mathbb{R}$, and where $\theta=(b')^{-1}(\mu)$.

Let $\theta_i=(b')^{-1}(\mu_i)$ for all i. Then, it can be shown that, under the model (10.1) and assuming that the x_i^0's are fixed,

$$\mathbb{E}[Y_i^0]=b'(\theta_i)=\mu_i, \quad \mathrm{Var}(Y_i^0)=\phi\, b''(\theta_i), \quad \forall i\in\{1,\dots,n\}.$$

Remark: The additive model (Chapter 9) corresponds to the case where, in (10.1), $f(y;\mu_i,\phi)\mathrm{d}y$ is a Gaussian distribution with mean μ_i and variance equal to ϕ (in which case, in (10.2), b, a and c are defined by $b(x)=x^2/2$, $a(x)=x$ and $c(y,\phi)=y^2/(2\phi)$).

In a first step we focus on the estimation of α and $\{f_j\}_{j=1}^p$ for a given value of the dispersion parameter ϕ.

[a] The main reference for this chapter is [14, Chapter 6]

Estimation of α and $\{f_j\}_{j=1}^p$ in the model (10.1)

Let $l^*(\alpha, \{f_j\}_{j=1}^p) = -\sum_{i=1}^n \log f(y_i^0; \mu_i, \phi)$ be minus the log-likelihood function of the model and, for all j, let $m'_j \in \mathbb{N}$, $\{\tilde{b}_{(j),k}\}_{k=1}^{m'_j}$ be as defined in Chapter 9, and let $\boldsymbol{S}_\lambda$ be the corresponding penalty matrix. In addition, let $m = \sum_{j=1}^p m'_j$ and

$$\tilde{C}_j^2(\mathbb{R}) = \mathrm{span}\big(\tilde{b}_{(j),1}, \dots, \tilde{b}_{(j),m'_j}\big), \quad j = 1, \dots, p.$$

Then, generalizing the approach introduced in Chapter 9 for additive models, our goal is to estimate $(\alpha, \{f_j\}_{j=1}^p)$ using

$$(\hat{\alpha}_\lambda, \{\hat{f}_{\lambda,j}\}_{j=1}^p) \in \underset{\alpha\in\mathbb{R}, f_j\in\tilde{C}_j^2(\mathbb{R}), \forall j}{\mathrm{argmin}} \; l^*\big(\alpha, \{f_j\}_{j=1}^p\big) + \frac{1}{2a(\phi)} \sum_{j=1}^p \lambda_j \int_{\mathbb{R}} (f''_j(x))^2 \mathrm{d}x$$

or, equivalently, to compute (with an obvious definition of $l(\alpha, \beta)$)

$$(\hat{\alpha}_\lambda, \hat{\beta}_\lambda) \in \underset{\alpha\in\mathbb{R},\, \beta\in\mathbb{R}^m}{\mathrm{argmin}} \; l\big(\alpha, \beta\big) + \frac{1}{2a(\phi)} \beta^\top \boldsymbol{S}_\lambda \beta. \tag{10.3}$$

Reminder: The basis functions $\{\tilde{b}_{(j),k}\}_{k=1}^{m_j}$ are such that, for all j, we have $\sum_{i=1}^n \hat{f}_{\lambda,j}(x_{ij}^0) = 0$ and thus α identifiable.

Remark: Using $\lambda_j/(2a(\phi))$ instead of λ_j as penalty terms (a) simplify the notation in what follows and (b) makes $(\hat{\alpha}_\lambda, \hat{\beta}_\lambda)$ independent of ϕ

Remark: It may be useful to remove one basis function $\tilde{b}_{(j),k}$ for all j to make the Gram matrix invertible (see Chapter 9).

In general, we cannot explicitly solve the optimization problem (10.3) and below we show how the Fisher scoring algorithm can be used to numerically optimize the function

$$(\alpha, \beta) \mapsto F_\lambda(\alpha, \beta), \quad F_\lambda(\alpha, \beta) = l\big(\alpha, \beta\big) + \frac{1}{2a(\phi)} \beta^\top \boldsymbol{S}_\lambda \beta. \tag{10.4}$$

Preliminaries: Newton's algorithm

Newton's algorithm is used to minimize a smooth convex objective functions $F : \mathbb{R}^q \to \mathbb{R}$, that is, to compute $z^* := \operatorname{argmin}_{z\in\mathbb{R}^q} F(z)$.

Let $\nabla F(z)$ denote the gradient of F evaluated at the point $z \in \mathbb{R}^q$ and $\boldsymbol{H}(z)$ denote the Hessian matrix of F evaluated at the point $z \in \mathbb{R}^q$.

Then, Newton's algorithm for computing z^* is as follows.

Newton's algorithm for minimizing a convex function

Input: Starting value $z_0 \in \mathbb{R}^q$

for $k \geq 1$ **do**

(i) Let $\tilde{F}_{k-1} : \mathbb{R}^q$ be defined by

$$\begin{aligned}\tilde{F}_{k-1}(z) &= F(z_{k-1}) + (z - z_{k-1})^\top \nabla F(z_{k-1}) \\ &\quad + \frac{1}{2}(z - z_{k-1})^\top \boldsymbol{H}(z_{k-1})(z - z_{k-1})\end{aligned}$$

(ii) Let $z_k = \operatorname{argmin}_{z\in\mathbb{R}^m} \tilde{F}_{k-1}(z) = z_{k-1} - \boldsymbol{H}(z_k)^{-1}\nabla F(z_{k-1})$

if Convergence=TRUE **then**

return z_k and **break**

end if

end for

In practice it is of course unnecessary to perform Step (i) but this steps makes clear how Newton's algorithm works: At iteration k, instead of minimizing the objective function F Newton's algorithm minimizes $\tilde{F}_{k-1}$ which, by Taylor's theorem, is a quadratic approximation of F around z_{k-1}.

Fisher scoring for solving (10.4).

Let $\gamma = (\alpha, \beta)$ and let $\nabla F_\lambda(\gamma)$ and $\boldsymbol{H}_\lambda(\gamma)$ denote the gradient and the Hessian matrix of the function $F_\lambda(\gamma)$ we want to minimize (defined in (10.4)), evaluated at the point γ.

Remark that $\boldsymbol{H}_\lambda(\gamma)$ depends on $\{y_i^0\}_{i=1}^n$ and let $\bar{\boldsymbol{H}}_\lambda(\gamma) = \mathbb{E}[\boldsymbol{H}_\lambda(\gamma)]$ where the expectation is taken w.r.t. to the distribution of $\{Y_i^0\}_{i=1}^n$ induced by the model (10.1) when the parameter value is γ.

Then, the Fisher scoring algorithm for computing $\hat{\gamma} = (\hat{\alpha}_\lambda, \hat{\beta}_\lambda)$ is obtained by replacing $\boldsymbol{H}_\lambda(\gamma)$ in Newton's algorithm by $\bar{\boldsymbol{H}}_\lambda(\gamma)$.

Fisher's scoring algorithm for computing $\hat{\gamma} = (\hat{\alpha}_\lambda, \hat{\beta}_\lambda)$

Input: Staring value $\gamma_0 \in \mathbb{R}^{1+m}$
  for $k \geq 1$ **do**
    (i) Let $\gamma_k = \gamma_{k-1} - \bar{\boldsymbol{H}}_\lambda(\gamma_k)^{-1} \nabla F_\lambda(\gamma_{k-1})$
    if Convergence=TRUE **then**
      return z_k and **break**
    end if
  end for

A justification for Fisher's scoring algorithm is given in the following proposition.

Proposition 10.1 *Let $F : \mathbb{R}^q \to \mathbb{R}$ and $\boldsymbol{B} \in \mathbb{R}^{q \times q}$ be a positive-definite matrix. Then, $\Delta_z := -\mathbf{B}\nabla F(z)$ is a descent direction, in the sense that there exists a $\delta_{z'} > 0$ such that $F(z' + \delta_{z'}\Delta_{z'}) < F(z')$.*

Proof: By Taylor's theorem, as $\delta_z \to 0$ we have

$$F(z + \delta_z \boldsymbol{\Delta}_z) - F(z) = \delta_z \nabla F(z)^\top \boldsymbol{\Delta}_z + o(\delta_z^2) = -\delta_z \nabla F(z)^\top \mathbf{B} \nabla F(z) + o(\delta_z^2).$$

Since $\mathbf{B}$ is positive definite we have $\nabla F(z)^\top \mathbf{B} \nabla F(z) > 0$ and since the remainder term converges to zero faster than δ_z the result is proven.

Fisher's scoring as a penalized iterative weighted least squares (IWLS) algorithm

The following proposition shows that each update of (α, β) in the above Fisher's scoring algorithm amounts to solve a penalized weighted least squares problem.

Proposition 10.2 *Let $\gamma = (\alpha, \beta) \in \mathbb{R}^{m+1}$ and let*

$$(\alpha', \beta') = \gamma - \bar{\boldsymbol{H}}_\lambda(\gamma)^{-1}\nabla F_\lambda(\gamma).$$

For all $i \in \{1, \ldots, n\}$ let

$$\eta_i = \alpha + \beta^\top \tilde{z}_i, \quad \mu_i = g^{-1}(\eta_i), \quad \hat{y}_i = \eta_i + (y_i^0 - \mu_i)g'(\mu_i)$$

and let

$$\boldsymbol{W} = \mathrm{diag}\Big(\frac{1}{b''(\mu_1)g'(\mu_1)^2}, \ldots, \frac{1}{b''(\mu_n)g'(\mu_n)^2}\Big).$$

Then, $\alpha' = \frac{1}{n}\sum_{i=1}^n \hat{y}_i$ and

$$\beta' = \operatorname*{argmin}_{b \in \mathbb{R}^m} \|\hat{y} - \tilde{\boldsymbol{Z}}b\|_{\boldsymbol{W}} + b^\top \boldsymbol{S}_\lambda b$$

where $\|u\|_{\boldsymbol{W}} = u^\top \boldsymbol{W} u$ for all $u \in \mathbb{R}^m$ and where $\tilde{\boldsymbol{Z}}$ is as defined in Chapter 9.

Proof: We first compute the matrix $\bar{\boldsymbol{H}}_\lambda(\gamma)$. To this aim recall that $\theta_i = (b')^{-1}(\mu_i)$, where $\mu_i = g^{-1}(\alpha + \beta^\top x_i)$, and that, for an invertible function $h \in \mathcal{C}(\mathbb{R})$, we have

$$\frac{h^{-1}(u)}{\mathrm{d}u}(u) = \frac{1}{h' \circ h^{-1}(u)}$$

Let $\tilde{\boldsymbol{S}}_\lambda \in \mathbb{R}^{(m+1)\times(m+1)}$ be such that $(\alpha, \beta)^\top \tilde{\boldsymbol{S}}_\lambda(\alpha, \beta) = \beta^\top \boldsymbol{S}_\lambda \beta$ for all $(\alpha, \beta) \in \mathbb{R}^{m+1}$, and for all i let $q_i = (1, x_i^0)$ and

$$l_i(\gamma) := -\frac{y_i^0\theta_i - b(\theta_i)}{a(\phi)} - c(y_i^0, \phi) + \frac{1}{2na(\phi)}\gamma^\top \tilde{\boldsymbol{S}}_\lambda \gamma.$$

Proof of Proposition 10.2 (end)

Let $i \in \{1, \dots, n\}$, $\boldsymbol{Q} = [q_{il}]$ and note that

$$\frac{\partial l_i(\gamma)}{\partial \gamma_j} = -\frac{q_i(y_i^0 - \mu_i)}{a(\phi)b''(\mu_i)g'(\mu_i)} + \frac{1}{na(\phi)}\tilde{\boldsymbol{S}}_\lambda \gamma \tag{10.5}$$

and, thus, for all $l \in \{1, \dots, m+1\}$,

$$\begin{aligned} h_{i,jl} := \mathbb{E}_{Y_i^0 \sim f(y;\mu_i,\phi)\mathrm{d}y}\left[\frac{\partial^2 l_i(\gamma)}{\partial \gamma_j \partial \gamma_l}\right] &= -\frac{\partial \mu_i}{\partial \beta_l}\frac{q_{ij}a(\phi)b''(\mu_i)g'(\mu_i)}{(a(\phi)b''(\mu_i)g'(\mu_i))^2} + \frac{1}{na(\phi)}(\tilde{\boldsymbol{S}}_\lambda)_{kl} \\ &= \frac{q_{il}}{g'(\mu_i)}\frac{q_{ij}b''(\mu_i)g'(\mu_i)}{a(\phi)(b''(\mu_i)g'(\mu_i))^2} + \frac{1}{na(\phi)}(\tilde{\boldsymbol{S}}_\lambda)_{kl} \\ &= \frac{q_{il}q_{ij}}{a(\phi)b''(\mu_i)g'(\mu_i)^2} + \frac{1}{na(\phi)}(\tilde{\boldsymbol{S}}_\lambda)_{kl}. \end{aligned}$$

Therefore, $\bar{\boldsymbol{H}}_\lambda(\gamma) = [\sum_{i=1}^n h_{i,lj}]_{l,j=1}^{m+1}$ and thus, letting

$$\boldsymbol{W}_\gamma = \mathrm{diag}\Big(b''(\mu_1)g'(\mu_1)^2, \dots, b''(\mu_1)g'(\mu_1)^2\Big)^{-1} + \tilde{\boldsymbol{S}}_\lambda,$$

it is easily checked that $\bar{\boldsymbol{H}}(\gamma) = \frac{1}{a(\phi)}\big(\boldsymbol{Q}^\top \boldsymbol{W}_\gamma \boldsymbol{Q} + \tilde{\boldsymbol{S}}_\lambda\big)$.

Using (10.5) we have

$$\nabla F_\lambda(\gamma) - \frac{1}{a(\phi)}\tilde{\boldsymbol{S}}_\lambda \gamma = -\sum_{i=1}^n \frac{q_i(y_i^0 - \mu_i)}{a(\phi)b''(\mu_i)g'(\mu_i)} = -\sum_{i=1}^n \frac{q_i(y_i^0 - \mu_i)g'(\mu_i)}{a(\phi)b''(\mu_i)g'(\mu_i)^2} = -\frac{\boldsymbol{Q}^\top}{a(\phi)}\boldsymbol{W}_\gamma \tilde{y}_\gamma$$

where the vector $\tilde{y}_\gamma$ has $(y_i^0 - \mu_i)g'(\mu_i)$ as ith element.

Therefore, letting $\eta_\gamma = \boldsymbol{Q}\gamma$, we have

$$\begin{aligned} \gamma' &= \gamma - \bar{\boldsymbol{H}}_\lambda(\gamma)^{-1}\nabla F_\lambda(\gamma) \\ &= \gamma - \big(\boldsymbol{Q}^\top \boldsymbol{W}_\gamma \boldsymbol{Q} + \tilde{\boldsymbol{S}}_\lambda\big)^{-1}\big(-\boldsymbol{Q}^\top \boldsymbol{W}_\gamma \tilde{y}_\gamma + \tilde{\boldsymbol{S}}_\lambda\big) \\ &= \gamma + \big(\boldsymbol{Q}^\top \boldsymbol{W}_\gamma \boldsymbol{Q} + \tilde{\boldsymbol{S}}_\lambda\big)^{-1}\big(\boldsymbol{Q}^\top \boldsymbol{W}_\gamma \tilde{y}_\gamma - \tilde{\boldsymbol{S}}_\lambda \gamma\big) \\ &= \big(\boldsymbol{Q}^\top \boldsymbol{W}_\gamma \boldsymbol{Q} + \tilde{\boldsymbol{S}}_\lambda\big)^{-1}\Big((\boldsymbol{Q}^\top \boldsymbol{W}_\gamma \boldsymbol{Q} + \tilde{\boldsymbol{S}}_\lambda)\gamma + \big(\boldsymbol{Q}^\top \boldsymbol{W}_\gamma \tilde{y}_\gamma - \tilde{\boldsymbol{S}}_\lambda \gamma\big)\Big) \\ &= \Big(\boldsymbol{Q}^\top \boldsymbol{W}_\gamma \boldsymbol{Q} + \tilde{\boldsymbol{S}}_\lambda\Big)^{-1}\Big(\boldsymbol{Q}^\top \boldsymbol{W}_\gamma(\eta_\gamma + \tilde{y}_\gamma)\Big) \\ &= \underset{\gamma \in \mathbb{R}^{m+1}}{\mathrm{argmin}} \|\boldsymbol{W}_\gamma^{1/2}(\eta_\gamma + \tilde{y}_\gamma) - \boldsymbol{Q}\gamma\| + \gamma^\top \tilde{\boldsymbol{S}}_\lambda \gamma \end{aligned}$$

and the result follows. □

The penalized IWLS algorithm for solving (10.3)

Using Proposition 10.2 we obtain the following convenient representation of the above Fisher's scoring algorithm for solving (10.3).

Fisher's scoring algorithm for computing $(\hat{\alpha}_\lambda, \hat{\beta}_\lambda)$

Input: Staring value $(\alpha_0, \beta_0) \in \mathbb{R}^{1+m}$

for $k \geq 1$ **do**

(i) For all i let $\eta_{ki} = \alpha_{k-1} + \beta_{k-1}^\top \tilde{z}_i$, $\mu_{k,i} = g^{-1}(\eta_{k,i})$ and

$$\hat{y}_{k,i} = \eta_{k,i} + (y_i^0 - \mu_{k,i})g'(\mu_{k,i})$$

(ii) Let

$$\boldsymbol{W}_k = \mathrm{diag}\Big(\frac{1}{a(\phi)b''(\mu_{k,1})g'(\mu_{k,1})^2)}, \dots, \frac{1}{a(\phi)b''(\mu_{k,n})g'(\mu_{k,n})^2)}\Big).$$

(iii) Let $\alpha_k = \frac{1}{n}\sum_{i=1}^n \hat{y}_{k,i}$ and

$$\beta_k = \operatorname*{argmin}_{b \in \mathbb{R}^m} \|\hat{y}_k - \tilde{\boldsymbol{Z}} b\|_{\boldsymbol{W}_k} + b^\top \boldsymbol{S}_\lambda b$$

with $\tilde{\boldsymbol{Z}}$ is as defined in Chapter 9

if Convergence=TRUE **then**

return (α_k, β_k) and **break**

end if

end for

Remark: The maximum likelihood estimator in generalized linear models is usually estimated using the above algorithm (with $\boldsymbol{S}_\lambda = \boldsymbol{O}$).

Remark: For computational reasons it may be useful in Step (iii) to compute β_k using a weighted version of the Backfitting algorithm introduced in Chapter 9.

Choice of λ

For generalized additive models the cross-validation criteria are based on the deviance, defined by $D(\alpha,\beta)=\sum_{i=1}^{n} D_i(\alpha,\beta)$ with

$$D_i(\alpha,\beta)=2\big(\log f(y_i^0,\mu_i^{(s)},\phi)-\log f(y_i^0,g^{-1}(\alpha+\beta^\top\tilde{z}_i),\phi)\big),\quad i=1,\dots,n$$

and where $\mu_i^{(s)}$ denotes the fitted value of μ_i for the saturated model, that is, for the model that contains as many parameters as observations.

Then, on can choose $\lambda=\hat{\lambda}$ where

$$\hat{\lambda}\in\underset{\lambda\in[0,\infty)^p}{\operatorname{argmin}} D_{\mathrm{cv}}(\lambda),\quad D_{\mathrm{cv}}(\lambda)=\sum_{i=1}^{n} D_i\big(\alpha_\lambda^{(-i)},\beta_\lambda^{(-i)}\big)$$

where $(\alpha_\lambda^{(-i)},\beta_\lambda^{(-i)})$ is the estimate of (α,β) obtained after having removed the ith observation from the sample.

Remark: When $f(y;\mu_i,\phi)\mathrm{d}y$ is a Gaussian distribution with mean μ_i and variance equal to ϕ (additive model) this procedure for choosing λ reduces to minimizing the OCV criterion.

Computing $D_{\mathrm{cv}}(\lambda)$ requires to compute n estimates of (α,β) but, using a simple Taylor expansion, we can obtain an approximation $\tilde{D}_{\mathrm{cv}}(\lambda)$ of $D_{\mathrm{cv}}(\lambda)$ which only requires to estimate the model once, that is, to compute $(\hat{\alpha}_\lambda,\hat{\beta}_\lambda)$ (see [14, Section 6.2.5]). The value $\hat{\lambda}$ is then approximated by minimizing $\tilde{D}_{\mathrm{cv}}(\lambda)$ numerically, e.g., using Newton's algorithm.

Remark: The deviance can also be used to generalize the GCV criterion (see [14, Section 6.2.5]).

Estimation of ϕ

Letting $\mu_{\lambda,i} = g^{-1}(\hat{\alpha}_\lambda + \hat{\beta}_\lambda \tilde{z}_i)$ for all i, we estimate ϕ using

$$\hat{\phi}_\lambda \in \operatorname*{argmax}_{\phi \in (0,\infty)} \sum_{i=1}^{n} \log f(y_i^0, \mu_{\lambda,i}, \phi).$$

It is trivial to see that $\hat{\alpha}_\lambda$ and $\hat{\beta}_\lambda$ does not depend on ϕ, and therefore $(\hat{\alpha}_\lambda, \hat{\beta}_\lambda, \hat{\phi}_\lambda)$ is such that

$$(\hat{\alpha}_\lambda, \hat{\beta}_\lambda, \hat{\phi}_\lambda) \in \operatorname*{argmax}_{\alpha \in \mathbb{R}, \beta \in \mathbb{R}^m, \phi \in (0,\infty)} \sum_{i=1}^{n} \log f(y_i^0, \mu_{\lambda,i}, \phi) - \frac{1}{2a(\phi)} \beta^\top \boldsymbol{S}_\lambda \beta.$$

An illustrative example: The ozone dataset

We consider again the ozone dataset that we used in Chapter 9. We use a logistic generalized additive model (GAM) to predict the probability that the atmospheric ozone concentration it at least equal to 10 as a function $p = 5$ meteorological variables. For this example we let $m'_j = 10$ for all p and use generalized cross-validation to choose $\{\lambda_j\}_{j=1}^p$.

The resulting estimates of the functions $\{f_j\}_{j=1}^p$ are shown in Figure 10.1. We observe that in the fitted model all the function but the one for the variable humidity are non-linear.

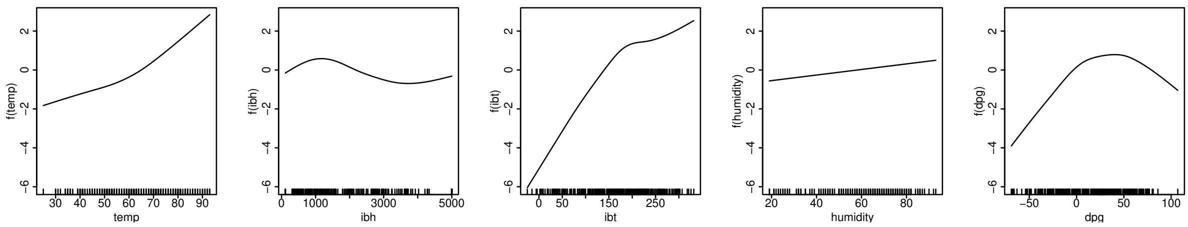

Figure 10.1: Estimation of $\{f_j\}_{j=1}^p$ for the ozone dataset (logistic GAM).

References

[1] Bishop, C. M. (2006). Pattern recognition. *Machine learning*, 128(9).

[2] Comon, P. (1994). Independent component analysis, a new concept? *Signal processing*, 36(3):287–314.

[3] Friedman, J., Hastie, T., Tibshirani, R., et al. (2001). *The elements of statistical learning*, volume 1. Springer series in statistics New York.

[4] Hastie, T., Tibshirani, R., and Wainwright, M. (2019). *Statistical learning with sparsity: the lasso and generalizations*. Chapman and Hall/CRC.

[5] Hyvärinen, A. and Oja, E. (2000). Independent component analysis: algorithms and applications. *Neural networks*, 13(4-5):411–430.

[6] Inaba, M., Katoh, N., and Imai, H. (1994). Applications of weighted voronoi diagrams and randomization to variance-based k-clustering. In *Proceedings of the tenth annual symposium on Computational geometry*, pages 332–339.

[7] Mairal, J. and Yu, B. (2012). Complexity analysis of the lasso regularization path. *arXiv preprint arXiv:1205.0079*.

[8] Mardia, K., Kent, J., and Bibby, J. (1979). *Multivariate analysis*. Probability and mathematical statistics. Academic Press Inc.

[9] Ng, A., Jordan, M., and Weiss, Y. (2001). On spectral clustering: Analysis and an algorithm. *Advances in neural information processing systems*, 14.

[10] Paulsen, V. I. and Raghupathi, M. (2016). *An Introduction to the Theory of Reproducing Kernel Hilbert Spaces*. Cambridge Studies in Advanced Mathematics. Cambridge University Press.

[11] Sande, E., Manni, C., and Speleers, H. (2020). Explicit error estimates for spline approximation of arbitrary smoothness in isogeometric analysis. *Numerische Mathematik*, 144(4):889–929.

[12] van Wieringen, W. N. (2015). Lecture notes on ridge regression. *arXiv preprint arXiv:1509.09169*.

[13] Von Luxburg, U. (2007). A tutorial on spectral clustering. *Statistics and computing*, 17(4):395–416.

[14] Wood, S. N. (2017). *Generalized additive models: an introduction with R*. CRC press.