Learning About Learning

George Zyskind Memorial Lecture
Iowa State University

Ronald Christensen
Department of Math and Statistics
University of New Mexico

May 7, 2018
My connections:

- Dan Nettleton and David Harville
- Frank Martin
- George Casella and Alistair Scott
Oscar Kempthorne (*D&A-E* in my top 10 books)

George Zyskind

Frank Martin

Charles Henderson

Shayle Searle, David Harville

Experimental Randomization: Who Needs It?  
(TAS, 1975)
George Zyskind, Strathcona Academy High School student who placed sixth in the Provincial high school leaving examinations, is one of a group of New Canadians brought to Canada almost two years ago by the Canadian Jewish Congress. In Montreal, he came under the BQMrrMon [sic] of the Jewish Child Welfare Bureau and the Jewish Vocational Service, both agencies of the Federation of Jewish Philanthropies and supported by the Combined Jewish Appeal. Born in Poland nineteen years ago, he had only intermittent schooling in Poland and France, and understood English with difficulty.

1975 Obituary in TAS by Oscar Kempthorne.
Introduction

- I’m just an old linear models guy trying to understand something about Statistical Learning.
- David Blackwell (paraphrase): Not interested in doing research but in understanding things.
- Working on new editions of PA and ALM. Share some of the more interesting things I have come up with.
The Trumpian Assumption

I hope I’m not just telling you things that everybody already knows.

“Nobody knew that health care could be so complicated.”
Plan

- Discuss the role of reproducing kernels in Statistical Learning. (It’s all about the penalty function.)
- Motivate that via Support Vector Machines for binary data. (Different strokes for different folks.)
- If time permits: Comments on bagging and boosting will be included. (Bagging gives results MORE like least squares?)
SVM: My first experience

Discriminate between these groups:

Figure: Doughnut data.
LDA cannot separate these. SVMs can.

Speaker did not add, “If you use the kernel trick on SVM but don’t on LDA.”
SVM: My first experience

Figure: Doughnut data in polar coordinates.
Penalized Estimation

Predictive estimators $\tilde{f}$ are often chosen to achieve

$$\inf_{f \in \mathcal{F}} \left\{ \sum_{h=1}^{n} \mathcal{L}[y_h, f(x_h)] + k \mathcal{P}(f) \right\},$$
Penalized Estimation

Using a linear structure $X\beta$? Pick $\beta$ by minimizing

$$\sum_{i=1}^{n} L(y_i, x'_i\beta) + kP(\beta),$$
Binary Outcomes

Support vector machines pick $\beta = (\beta_0, \beta^*)'$ by minimizing

$$\sum_{i=1}^{n} L_S(y_i, x_i' \beta) + k \beta^* \beta^*.$$ 

Standard ridge regression penalty.

The loss function is

$$L_S(y, u) = \begin{cases} (1 - u)_+ & \text{if } y = 1 \\ (1 + u)_+ & \text{if } y = 0. \end{cases}$$

Traditional Logistic Regression: $k=0$, different loss.
Loss Functions

Figure: Logistic regression and SVM loss functions.
SVM

- Quadratic minimization problem.
- Minimizing the quadratic penalty function subject to linear inequality constraints determined by the loss function.
- FYI: LASSO minimizes quadratic loss function subject to linear inequality constraints determined by the penalty.
- Quite creative how SVMs turn piecewise linear loss functions into inequality constraints.
- Lots of technical considerations.
- Necessary KKT (Karush, Kuhn, and Tucker) conditions.
Separating Hyper-Hogwash

Figure: Logistic Regression, R default SVM, LR augmented data ridge.
SVM: 3 Issues

- I thought it’s dependence only on inner products was a unique advantage.
- I thought it’s dependence only on a small number of support vectors was a computational advantage.
- I don’t see why the KKT conditions do not uniquely determine $\beta_0$ from $\beta_*$
Nonparametric Regression

\[ y_i = f(x_i) + \varepsilon_i, \quad E(\varepsilon_i) = 0, \quad i = 1, \ldots, n. \]

- \( f \) continuous on a compact set, say \([0,1]\).
- \( \varepsilon_i \)'s independent; \( \text{Var}(\varepsilon_i) = \sigma^2 \).

Matrix form,

\[ Y = F(X) + e, \quad E(e) = 0, \quad \text{Cov}(e) = \sigma^2 I, \]

where \( X' \equiv (x_1, \ldots, x_n) \) and \( F(X) \equiv [f(x_1), \ldots, f(x_n)]' \)
Spanning (basis) Functions

\[ f(x) = \sum_{j=0}^{\infty} \beta_j \phi_j(x), \]

Approximate linear model

\[ y_i = \sum_{j=0}^{p-1} \beta_j \phi_j(x_i) + \varepsilon_i. \quad (1) \]

Define \( \Phi_j \equiv [\phi_j(x_1), \ldots, \phi_j(x_n)]' \) and \( \Phi \equiv [\Phi_0, \Phi_1, \ldots, \Phi_{p-1}] \), so

\[ Y = \Phi \beta + e. \]

Pick \( p \). Estimate \( \beta \).
Simple Nonparametric Regression

Example

- Battery voltage drops.
- One predictor: time
Polynomial: 4th Degree
6 Cosines
Cubic Splines: 4 Knots
Haar Wavelets
30 Cosines
Cubic Splines: 30 Knots
Cosines: 6, 10, 14, 30
Haar wavelets: $p = 8, 16$
Simple NPR

One predictor $x$, the spanning set of functions matters.
Multiple NPR

Nonparametric multiple regression:

\[ x_i = (x_{i1}, x_{i2}, \ldots, x_{is})', \]

\( \phi_j(x) \) reindexed and redefined.

Typical example, \( s = 2 \) variables \( x_1 \) and \( x_2 \),

\[ \phi_{jk}(x_1, x_2) \equiv \phi_j(x_1)\phi_k(x_2), \]

and

\[ f(x_1, x_2) = \sum_{j=0}^{p_1} \sum_{k=0}^{p_2} \beta_{jk} \phi_{jk}(x_1, x_2). \]
Curse of Dimensionality

\[ f(x) = \sum_{k_1=0}^{p_1} \cdots \sum_{k_s=0}^{p_s} \beta_{k_1\ldots k_s} \phi_{k_1}(x_1) \cdots \phi_{k_s}(x_s). \]  

Suppose \( s = 5 \). If we need \( p = 8 \) for each dimension, fitting \( 8^5 = 32,768 \) parameters.
Reproducing Kernels

An alternative to specifying spanning functions, use a reproducing kernel, say,

\[ R(u, v). \]
Fundamental Theorem (for Statistics) of RKHSs

I am hoping for,

but not expecting,

laughter.
Fundamental Theorem (for Statistics) of RKHSs

\[ C(X) = C(XX'). \]
Kernel Trick

Any problem that involves

\[ X_{n \times p} \beta \]

replace it with

\[ \tilde{R}_{n \times n} \gamma \]

where

\[ \tilde{R} \equiv [R(x_i, x_j)]. \]
Kernel Trick 2: More Restrictive

Any problem with a solution that depends only on

\[ XX' \]

replace it with

\[ \tilde{R}. \]

This is the usual application to SVMs.

(Based on the "Fundamental Theorem"

I conjecture both tricks give the same SVMs.)
Singular Value Decomposition of Function

\[ R(u, v) = \sum_{k=0}^{p-1} \eta_k \phi_k(u) \phi_k(v). \]

\( p \) is finite or infinite \hspace{1cm} \eta_k > 0

\[ \tilde{R} = [R(x_i, x_j)] \quad R(x_i, x_j) = \sum_{k=0}^{p-1} \eta_k \phi_k(x_i) \phi_k(x_j). \]
Popular SVM Kernels

- linear [finite]
- polynomial [finite]
- radial basis (Gaussian) [infinite]
- sigmoid (hyperbolic tangent) [infinite]
Singular Value Decomposition of Function: $p < \infty$

\[ \tilde{R} = \Phi \text{Diag}(\eta_j) \Phi'. \]

Not the matrix SVD, just a diagonalization, but still

\[ C(\tilde{R}) = C(\Phi). \]

so who cares whether you use

\[ \Phi \beta \quad \text{or} \quad \tilde{R} \gamma? \]

- The permissable vectors are equivalent.
- $\tilde{R}$ may be easier to program.
- (It’s all about the penalty function.)
Kernel Trick: $p = \infty$

- How does this help with the curse of dimensionality?
- $\Phi$ may be $n \times 32,768$.
- If the $x_i$s are distinct, expect $\tilde{R}$ nonsingular!

$$C(\tilde{R}) = C(I_n) = \mathbb{R}^n.$$ 

- Traditional estimation gives “perfect” fitted values!
- No possibility for estimating error!
- It’s all about the penalty function.
Kernel Trick: $p = \infty$

Does the choice of kernel matter?

$$R_1(u, v) \neq R_2(u, v)$$

$$\tilde{R}_1 \neq \tilde{R}_2$$

but

$$C(\tilde{R}_1) = C(\tilde{R}_2) = C(I_n) = \mathbb{R}^n.$$ 

We aren’t modeling differently.
Kernel Trick: $p = \infty$

- Picking $R$ amounts to picking $\Phi$.
- We saw that the choice of $\Phi$ matters when $C(\Phi) \neq \mathbb{R}^n$.
- Even when $C(\tilde{R}) = C(\Phi) = \mathbb{R}^n$,
  with an “off the shelf” penalty function,
  the choice of $R$ matters a lot!

But you can find penalty functions that give identical results.
Reparameterization of $X_1\beta_1$

For $A$ nonsingular

$$X_2 = X_1A,$$

so

$$X_2\beta_2 = X_1A\beta_2 = X_1\beta_1 \quad \text{and} \quad \beta_1 = A\beta_2.$$

Thus

$$\mathcal{P}(\beta_1) = \mathcal{P}(A\beta_2).$$

For example:

$$\| Y - X_1\beta_1 \|^2 + k\| \beta_1 \|^2 = \| Y - X_2\beta_2 \|^2 + k\| A\beta_2 \|^2.$$

Minimizing $\| Y - X_2\beta_2 \|^2 + k\| \beta_2 \|^2$ typically gives different answers.
Reparameterization: $\Phi \beta$ versus $\tilde{R} \gamma$

General reparameterizations with just $C(X_1) = C(X_2)$ are more complicated but for

ridge regression

any tuning parameter $k_\Phi$, there exists $k_{\tilde{R}}$

with fitted values $\Phi \hat{\beta}_R = \tilde{R} \hat{\gamma}_R$

and same predictions.

(later slides)
\(x_i\)'s NOT distinct

**Fisher's Lack-of-Fit Test** (OLS fitting)

- Row structures of \(X\), \(\Phi\), and \(\tilde{R}\) (no longer nonsingular) are the same.
- \(X\beta\), \(\Phi\beta\), and \(\tilde{R}\gamma\) would give exactly the same \textit{MSPE}. (Ridge also.)
- \(\Phi\beta\) and \(\tilde{R}\gamma\) would give exactly the same \textit{MSLF}.
- Good chance that \(\Phi\beta\) and \(\tilde{R}\gamma\) would give \(\text{SSLF} = 0\) on 0 \textit{df}. 
PCR and Ridge Regression

\[ Y = X\beta + e, \quad \mathbb{E}(e) = 0. \]

Eigenvalues and vectors:

\[ X'XV = VL^2, \quad L \equiv D (\lambda_i). \]

\[ XX'U = UL^2, \quad U \equiv XVL^{-1}. \]

Eigenvectors are not unique. We need \( U \) and \( V \) to correspond.
Singular value decompositions

Using orthonormal eigenvectors:

\[ X'X = VL^2V', \quad L \equiv D(\lambda_i). \]

\[ XX' = UL^2U', \quad U \equiv XVL^{-1}. \]

\[ X = ULV'. \]
Principal Component Regression

\[ Y = \tilde{X}\gamma + e, \quad \tilde{X} \equiv UL \]

(Ignoring complications due to centering and scaling.)

\[
E(Y) = \tilde{X}\gamma = UL\gamma = ULV'V\gamma
\]

\[ = XV\gamma = X\beta \]

To transform any estimate of \( \gamma \) into an estimate of \( \beta \):

\[ \beta = V\gamma. \]
PCR Estimates

Least squares:

\[ \hat{\gamma} \equiv (\tilde{X}'\tilde{X})^{-1}\tilde{X}'Y = L^{-1}U'Y. \]

Ridge:

\[ \hat{\gamma}_R \equiv (\tilde{X}'\tilde{X} + kI)^{-1}\tilde{X}'Y = D \left( \frac{\lambda_i^2}{\lambda_i^2 + k} \right) \hat{\gamma}. \]

Reproducing kernel ridge:

\[ \hat{\gamma}_{RR} \equiv (\tilde{X}\tilde{X}'\tilde{X}\tilde{X}')^{-1}\tilde{X}\tilde{X}'Y = D \left( \frac{\lambda_i^4}{\lambda_i^4 + k} \right) \hat{\gamma}. \]

Kernel trick is just like ridge except nonlinear transformation of tuning parameter.
Original Model Estimates

Least squares:
\[ \hat{\beta} = V \hat{\gamma} = (X'X)^{-1}X'Y. \]

Ridge:
\[ \hat{\beta}_R = V \hat{\gamma}_R \equiv (X'X + kI)^{-1}X'Y. \]

Reproducing kernel ridge:
\[ \hat{\beta}_{RR} = V \hat{\gamma}_{RR} = (XX'XX' + kI)^{-1}XX'Y. \]

RR is nasty algebra.
Boosting, Bagging, and Random Forests

- **Boosting:**
  - Method of biased estimation (Lousy)
  - **BUT** what I think is a bug, HTF suggest may be a feature.
- Random Forests (Improved Bagging)
- Bagging
  - Overfit the model: important stuff always shows up, unreal stuff averages out.
Simplest Bagging Example

Predicting $y$. BP is $E(y) \equiv \mu$. Estimate BP

<table>
<thead>
<tr>
<th>Symmetry</th>
<th>Tails</th>
<th>Median</th>
<th>Midrange</th>
<th>Mean</th>
</tr>
</thead>
<tbody>
<tr>
<td>Yes</td>
<td>Thick</td>
<td>good</td>
<td>poor</td>
<td>moderate</td>
</tr>
<tr>
<td>Yes</td>
<td>Thin</td>
<td>poor</td>
<td>good</td>
<td>moderate</td>
</tr>
<tr>
<td>Yes</td>
<td>Mid</td>
<td>poor</td>
<td>poor</td>
<td>good</td>
</tr>
<tr>
<td>Nonparametric</td>
<td>poor</td>
<td>poor</td>
<td>good</td>
<td></td>
</tr>
</tbody>
</table>

Bagging moves everything towards the sample mean.

<table>
<thead>
<tr>
<th>Symmetry</th>
<th>Tails</th>
<th>Bag Median</th>
<th>Bag Midrange</th>
<th>Bag Mean</th>
</tr>
</thead>
<tbody>
<tr>
<td>Yes</td>
<td>Thick</td>
<td>worse</td>
<td>better</td>
<td>same</td>
</tr>
<tr>
<td>Yes</td>
<td>Thin</td>
<td>better</td>
<td>worse</td>
<td>same</td>
</tr>
<tr>
<td>Yes</td>
<td>Mid</td>
<td>better</td>
<td>better</td>
<td>same</td>
</tr>
<tr>
<td>Nonparametric</td>
<td>better</td>
<td>better</td>
<td>same</td>
<td></td>
</tr>
</tbody>
</table>
References


References


