Machine Learning for Operations Research

Gah-Yi Ban

INFORMS 2017

Outline

- What is Machine Learning?
- What can Machine Learning do for Operations Research?
- Model Assessment, Regularization and Portfolio Optimization
  - In-sample vs. out-of-sample error
  - Regularization
  - Cross-validation
- Statistical Learning Theory & Newsvendor Problem
  - Generalization error
  - Vapnik-Chervonenkis Theory
  - Stability Theory
  - Application: Newsvendor Problem
- Take Aways/Future Directions

What is Machine Learning?

- The study and construction of algorithms that can learn from and take actions on data
- Supervised (input and output) vs. Unsupervised (input only)
  - e.g. defining customer segmentation groups is unsupervised learning, whereas predicting total spending given customer characteristics is supervised learning
  - Supervised learning is more relevant for OR, since this involves a performance metric (e.g. MSE, total revenue, etc)
- Traditionally focused on prediction problems, but same principles apply to data-driven prescription (decision) problems

What can Machine Learning do for OR?

Research opportunities:

- Modeling: new models that integrate data and decision models
- Computational: efficient algorithms to solve data-driven problems
- Analytical: statistical analysis (asymptotic convergence, finite-sample error bounds, analysis of learning rates)

Impact on Practice:

- Data-driven models are by design implementable and automatable
- We're already seeing data-driven OR models at the heart of many organizations

PART I

Model Assessment, Selection

and

Portfolio Optimization

Part I References

- Friedman, Jerome, Trevor Hastie, and Robert Tibshirani. The elements of statistical learning. Vol. 2. Springer, Berlin: Springer series in statistics, 2013. (available free online) **[FHT13]**
- Ban, Gah-Yi, Noureddine El Karoui, and Andrew EB Lim. "Machine Learning and Portfolio Optimization." Management Science, Articles in Advance, 21 Nov 2016. Available on `gahyiban.com` **[BEKL16]**

Portfolio Optimization

Consider the portfolio optimization problem (Markowitz, 1952):

$$\begin{array}{lrcl} \min\limits_{\mathbf{w}\in\mathbb{R}^p} & \mathbf{w}^\top\Sigma\mathbf{w} & & \\ s.t. & \mathbf{w}^\top\boldsymbol{\mu} & = & R \\ & \mathbf{w}^\top\mathbf{1} & = & 1 \end{array} \qquad \text{(MV)}$$

where

- $\mathbf{X}$: $p \times 1$ random vector of relative returns
- $\boldsymbol{\mu} = E(\mathbf{X})$: mean returns
- $\Sigma = Cov(\mathbf{X})$: $p \times p$ covariance matrix for the relative returns
- Solution: $\mathbf{w}_0(R)$
- Same if return constraint is relaxed to $\mathbf{w}^\top\boldsymbol{\mu} \geq R$

Sample Average Approximation

- In practice, we don't know the distribution P of $\mathbf{X}$ but have data
- Suppose we have n iid observations of asset returns from P: $\mathcal{X}_n = [\mathbf{x}_1, \ldots, \mathbf{x}_n]$.
- Then solve

$$\begin{aligned} \min_{\mathbf{w}\in\mathbb{R}^p} \quad & \mathbf{w}^\top \hat{\Sigma}_{1:n} \mathbf{w} \\ s.t. \quad & \mathbf{w}^\top \hat{\boldsymbol{\mu}}_n = R \\ & \mathbf{w}^\top \mathbf{1} = 1 \end{aligned} \tag{SAA}$$

where

- $\hat{\Sigma}_{1:n}$ is the sample covariance matrix of $[\mathbf{x}_1, \ldots, \mathbf{x}_n]$.
- $\hat{\boldsymbol{\mu}}_n$ is the sample average of the returns
- Solution: $\hat{\mathbf{w}}_{SAA}(R)$

In-sample vs. Out-of-sample performance

Three types of performance measures:

- In-sample performance: the performance of the learned action in the (training) sample, i.e. the data you used to learn
- Out-of-sample, or test, or generalization performance: the average performance of the learned action over all possible new observations
- Expected test, or true performance: the average performance of the learned action over all possible training sets and over all possible new observations

Note 1: for typical ML prediction problems, think error not performance. E.g. in-sample error, out-of-sample error, prediction error

Note 2: Training performance always overestimates (w.p. 1) both the out-of-sample and expected performances (why?)

In-sample vs. Out-of-sample return

In-sample (aka “training”) return:

$$\hat{\mathbf{w}}_{SAA}^{\top} \hat{\boldsymbol{\mu}}_n$$

Out-of-sample (aka “test” or “generalization”) return:

$$\mathbb{E}_{\mathbf{X}_{n+1}}[\hat{\mathbf{w}}_{SAA}^{\top} \mathbf{X}_{n+1} | \mathcal{X}_n] = \hat{\mathbf{w}}_{SAA}^{\top} \boldsymbol{\mu}$$

Expected test (aka “true”) return:

$$\mathbb{E}_{\mathcal{X}_n}[\mathbb{E}_{\mathbf{X}_{n+1}}[\hat{\mathbf{w}}_{SAA}^{\top} \mathbf{X}_{n+1} | \mathcal{X}_n]]$$

In-sample vs. Out-of-sample risk

In-sample risk:

$$\hat{\mathbf{w}}_{SAA}^{\top}\hat{\Sigma}_{1:n}\hat{\mathbf{w}}_{SAA}$$

Out-of-sample risk:

$$Var_{\mathbf{X}_{n+1}}[\hat{\mathbf{w}}_{SAA}^{\top}\mathbf{X}_{n+1}|\mathcal{X}_n] = \hat{\mathbf{w}}_{SAA}^{\top}\Sigma\hat{\mathbf{w}}_{SAA}$$

Expected test risk:

$$\mathbb{E}_{\mathcal{X}_n}[Var_{\mathbf{X}_{n+1}}[\hat{\mathbf{w}}_{SAA}^{\top}\mathbf{X}_{n+1}|\mathcal{X}_n]]$$

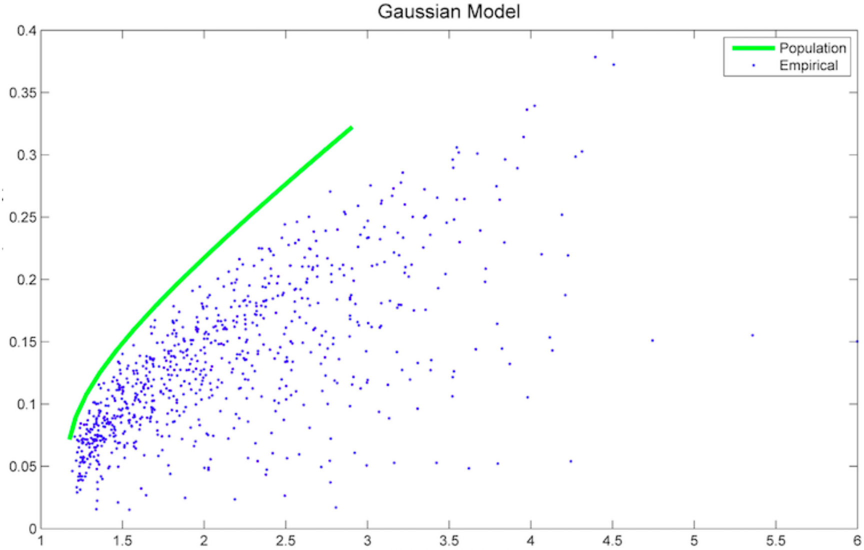
Performance of SAA: Simulated Data

Fix $(\boldsymbol{\nu}, Q)$ and target return level R. Then for $b = 1, \ldots, B$,

- Generate $\mathcal{X}_{b,n} = [\mathbf{x}_{b,1}, \ldots, \mathbf{x}_{b,n}]$, where $\mathbf{X}_{b,i} \overset{iid}{\sim} \mathcal{N}(\boldsymbol{\nu}, Q)$ for all $i = 1, \ldots, n$
- Solve the SAA problem for $\hat{\mathbf{w}}_{b,SAA}$
- Compute its out-of-sample return and risk: $\hat{\mathbf{w}}_{b,SAA}^{\top} \boldsymbol{\nu}$ and $\hat{\mathbf{w}}_{b,SAA}^{\top} Q \, \hat{\mathbf{w}}_{b,SAA}$

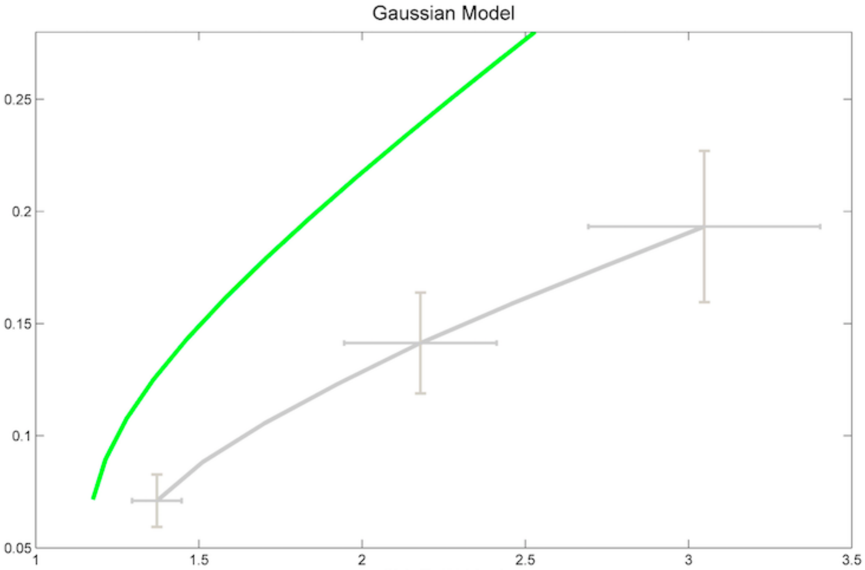
Performance of SAA

Return vs. Risk



Performance of SAA

Return vs. Risk



SAA is an error-maximizing algorithm

- Although SAA makes intuitive sense, it is highly unreliable for portfolio optimization with real stock return data
- This is well-documented across finance, statistics and OR:
  - Markowitz: Frankfurter et al. (1971), Frost & Savarino (1986, 1988b), Michaud (1989), Best & Grauer (1991), Chopra & Ziemba (1993), Broadie (1993), Lim et al. (2011)
- Michaud (1989): The (in-sample) portfolio optimization solution is an “error-maximizing” solution

Regularization

- Regularization: perturbing a linear operator problem for improved stability of solution [Ivanov (1962), Phillips (1962), Tikhonov (1963)]
- E.g. Least-squares regression with regularization:

$$\min_{\beta \in \mathbb{R}^p} ||\mathbf{y} - \mathbf{X}\beta||_2 + \lambda_n ||\beta||_k,$$

  where λ_n is the degree of regularization, and $k = 1$ (LASSO), $k = 2$ (ridge regression) yield popular penalty functions.
- Intuition: perturbing the in-sample problem reduces over-fitting; it adds bias but can improve the variance, which is good for generalization
- In general, $L - 1$ norm penalty yields sparse (many elements are exactly zero) solution vector and $L - 2$ norm penalty yields dense (many small, but non-zero elements) solution vector
- While these have justifications in regression problems, it's not clear why one would want sparse or dense portfolio solutions

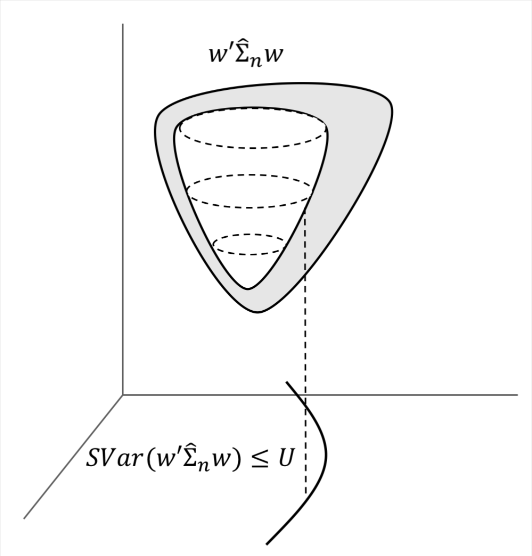
Performance-based regularization (PBR)

- Performance-based regularization: perturb portfolio problem for improved performance of the solution

$$
\begin{array}{lrl}
\min_{\mathbf{w}} & \mathbf{w}^\top \hat{\Sigma}_n \mathbf{w} & \\
s.t. & \mathbf{w}^\top \hat{\mu}_n = & R \\
 & \mathbf{w}^\top \mathbf{1} = & 1 \\
 & SVar(\mathbf{w}^\top \hat{\Sigma}_n \mathbf{w}) \leq & U
\end{array}
$$

- Intuition: penalize solutions **w** associated with greater estimation errors of objective

Schematic for PBR



PBR for Mean-Variance problem

The sample variance of the sample variance of the portfolio, $SVar(w'\hat{\Sigma}_n w)$ is given by:

$$SVar(w'\hat{\Sigma}_n w) = \Sigma_{i=1}^{p} \Sigma_{j=1}^{p} \Sigma_{k=1}^{p} \Sigma_{l=1}^{p} w_i w_j w_k w_l \hat{Q}_{ijkl},$$

where

- $\hat{Q}_{ijkl} = \frac{1}{n}(\hat{\mu}_{4,ijkl} - \hat{\sigma}_{ij}^2 \hat{\sigma}_{kl}^2) + \frac{1}{n(n-1)}(\hat{\sigma}_{ik}^2 \hat{\sigma}_{jl}^2 + \hat{\sigma}_{il}^2 \hat{\sigma}_{jk}^2),$
- $\hat{\mu}_{4,ijkl}$ is the sample average estimator for $\mu_{4,ijkl}$, the fourth central moment of the elements of $\boldsymbol{X}$
- $\hat{\sigma}_{ij}^2$ is the sample average estimator for σ_{ij}^2, the covariance of the elements of $\boldsymbol{X}$.

PBR constraint for Markowitz is thus a quartic polynomial. However, determining whether a quartic function is convex or not is an NP-hard problem [Ahmadi et al. (2013)]

PBR for Mean-Variance problem

Convex approximation I

- Rank-1 approximation:

$$(\mathbf{w}^\top \hat{\alpha})^4 \approx \Sigma_{i=1}^{p} \Sigma_{j=1}^{p} \Sigma_{k=1}^{p} \Sigma_{l=1}^{p} w_i w_j w_k w_l \hat{Q}_{ijkl},$$

  where $\hat{\alpha}_i = \sqrt[4]{\hat{Q}_{iiii}}$.
- Approximate PBR constraint: $\mathbf{w}^\top \hat{\alpha} \leq \sqrt[4]{U}$

PBR for Mean-Variance problem

Convex approximation II

- Best convex quadratic approximation:

$$(\mathbf{w}^\top A\mathbf{w})^2 \approx \Sigma_{i=1}^p \Sigma_{j=1}^p \Sigma_{k=1}^p \Sigma_{l=1}^p w_i w_j w_k w_l \hat{Q}_{ijkl},$$

  such that the elements of A are as close as possible to the pair-wise terms of Q, i.e. $A_{ij}^2 \approx \hat{Q}_{ijij}$
- Solve semidefinite program: $A^* = \underset{A \succeq 0}{\operatorname{argmin}} ||A - Q_2||_F$, where Q_2 is a matrix with ij-th element equalling $\hat{Q}_{ijij}$ and $||\cdot||_F$ denotes the Frobenius norm:

$$||A||_F = \sqrt{\sum_{i=1}^{m} \sum_{j=1}^{n} |a_{ij}|^2}$$

- Approximate PBR constraint: $\mathbf{w}^\top A^* \mathbf{w} \leq \sqrt{U}$

Cross-Validation (CV)

Cross-Validation: if there's enough data, put aside some for tuning free parameters (the "validation data set"). E.g. 50% for training, 25% for validation and 25% for testing

Figure Source: FHT13

k-fold Cross Validation: divide into $2 \leq k \leq n$ sub-training sets to maximize use of scarce training data.

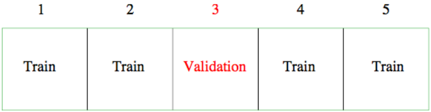


Figure Source: FHT13

Larger then k, the better the estimation of expected test error, but greater the computational burden and variance. $k = 5, 10$ are known to balance the trade-offs well. $k = n$ is leave-one-out CV

Performance-based CV

- CV: common technique in machine learning to tune free parameters
- k-fold CV: split training data into k equally-sized bins, train statistical model on every possible combination of $k-1$ bins, then tune parameter on the remaining bin.
- Performance-based k-fold CV: (1) search boundary for U_1 needs to be set carefully in order to avoid infeasibility and having no effect; (2) tune parameters by the Sharpe ratio, not by the mean squared error

Performance-based Cross-Validation

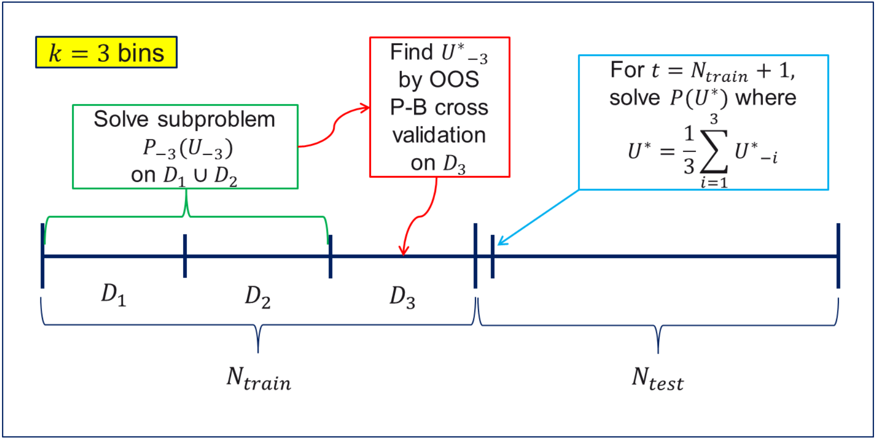


Figure: A schematic explaining the out-of-sample performance-based k-cross validation (OOS-PBCV) algorithm used to calibrate the constraint rhs, U, for the case $k = 3$. The training data set is split into k bins, and the optimal U for the entire training data set is found by averaging the best U found for each subset of the training data.

Empirical Results: Fama-French data sets

OOS Average Sharpe Ratio (Return/Std)

	FF 5 Industry p=5		FF 10 Industry p=10	
Mean-Variance R=0.04				
SAA	1.1459		1.1332	
	2 bins	3 bins	2 bins	3 bins
PBR (rank-1)	1.2603	1.3254	1.1868	1.2098
	(0.0411)	(0.0286)	(0.0643)	(0.0509)
PBR (PSD)	1.1836	1.1831	1.1543	1.1678
	(0.0743)	(0.071)	(0.0891)	(0.0816)
NS	1.0023		0.9968	
	(0.1404)		(0.1437)	
L1	1.0136	1.0386	1.1185	1.1175
	(0.1568)	(0.1396)	(0.1008)	(0.1017)
L2	0.9711	1.0268	1.0579	1.0699
	(0.1781)	(0.1452)	(0.1482)	(0.1280)

Parentheses: *p*-values of tests of differences from the SAA method.

Empirical Results: Fama-French data sets

OOS Average Sharpe Ratio (Return/Std)

	FF 5 Industry p=5		FF 10 Industry p=10	
Markowitz R=0.08				
SAA	1.1573		1.1225	
	2 bins	3 bins	2 bins	3 bins
PBR (rank-1)	1.3286	1.3551	1.1743	1.2018
	(0.0223)	(0.0208)	(0.0668)	(0.0510)
PBR (PSD)	1.1813	1.1952	1.1467	1.1575
	(0.0648)	(0.0614)	(0.0893)	(0.0844)
NS	0.9664		0.9405	
	(0.1514)		(0.1577)	
L1	0.9225	0.9965	1.0318	1.0779
	(0.1857)	(0.1403)	(0.1332)	(0.1181)
L2	0.9703	1.0284	1.0671	1.0776
	(0.1649)	(0.1398)	(0.1398)	(0.1209)

Parentheses: *p*-values of tests of differences from the SAA method.

Part I Summary

Model Assessment & Selection

- In general, in-sample optimal actions (predictions/decisions) do not generalize well out-of-sample. For the portfolio selection problem, solutions overweigh idiosyncratic observations in the training data.
- Regularization: L_1, L_2 norm penalties are standard, we explored more complex ones (PBR) in Ban et al. (2016) to focus on the performance of a decision, rather than the prediction error.
- Cross-Validation: data-driven methods to tune regularization parameters
- Can expect better out-of-sample performance with optimal amount of regularization that balances bias and variance.
- PBR solutions are better than SAA and L_1, L_2 regularized solutions on two well-known, publicly available data sets.

PART II

Statistical Learning Theory

and

Newsvendor Problem

Part II References

Vapnik-Chervonenkis Theory:

- Vapnik, Vladimir N. The nature of statistical learning theory. Springer Science Business Media, 2013. **[VNV13]**
- V.N. Vapnik. Estimation of Dependences Based on Empirical Data. Springer-Verlag, New York, 1982. **[VNV82]**

Stability Theory:

- Bousquet, Olivier, and André Elisseeff. “Stability and generalization.” Journal of Machine Learning Research 2. Mar (2002): 499-526. **[BE02]**

Newsvendor application:

- Ban, Gah-Yi and Rudin, Cynthia (2014). “The Big Data Newsvendor: Practical Insights from Machine Learning”. Minor revision in *Operations Research*. Available on `gahyiban.com` **[BR14]**

What is Statistical Learning Theory?

Consider:

- Input $\mathbf{X} \in \mathcal{X}$ and output response $Y \in \mathcal{Y}$
- Data: $D_n = [(\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_n, y_n)]$, $(x_i, y_i) \overset{iid}{\sim} P$, where P is an unknown distribution
- Learning algorithm A is a function that maps D_n to a function $f : \mathcal{X} \mapsto \mathcal{Y}$.
- $f \in \mathcal{F}$ is also known the hypothesis, and $\mathcal{F}$ the hypothesis class
- Loss function: $\ell : \mathcal{F} \times \mathcal{Y} \mapsto \mathbb{R}$

What is Statistical Learning Theory?

- Empirical (in-sample) risk:

$$R_{emp}(f, D_n) = \frac{1}{n}\sum_{i=1}^{n} \ell(f(\mathbf{x}_i), y_i)$$

- Out-of-sample (or test) risk/Generalization error:

$$R(f, D_n) = \mathbb{E}_{y_{n+1}}[\ell(f(\mathbf{x}_{n+1}), y_{n+1}))|D_n]$$

- Statistical learning theory is a theoretical framework for understanding the performance of a learning algorithm
- In particular, the literature has focused on understanding how well an algorithm generalizes out-of-sample

Vapnik-Chervonenkis (VC) Theory

- Theorem [VNV82, binary classification]: Let $\mathcal{F}$ be a hypothesis class with a VC dimension $d < n$. Then for every $n > 4$,

$$\sup_{f \in \mathcal{F}} |R_{emp}(f, D_n) - R(f, D_n)| < 2\sqrt{\frac{d(\log(2n/d) + 1) + \log(9/\delta)}{n}},$$

  with probability at least $1 - \delta$, for all $0 < \delta < 1$.
- This implies

$$R(f, D_n) \leq R_{emp}(f, D_n) + 2\sqrt{\frac{d(\log(2n/d) + 1) + \log(9/\delta)}{n}}$$

  for all $f \in \mathcal{F}$ with probability at least $1 - \delta$, for all $0 < \delta < 1$.
- In other words, we have an upper bound on the generalization error in terms of quantities we can compute.

Vapnik-Chervonenkis (VC) Theory

- A set of points is said to be shattered by a class of functions if, no matter how we assign a binary label to each point, a member of the class can perfectly separate them.
- The VC dimension of the class $\mathcal{F}$ is defined to be the largest number of points (in some configuration) that can be shattered by members of $\mathcal{F}$.

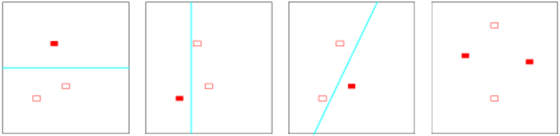


FIGURE 7.6. *The first three panels show that the class of lines in the plane can shatter three points. The last panel shows that this class cannot shatter four points, as no line will put the hollow points on one side and the solid points on the other. Hence the VC dimension of the class of straight lines in the plane is three. Note that a class of nonlinear curves could shatter four points, and hence has VC dimension greater than three.*

Source: FHT13

Vapnik-Chervonenkis (VC) Theory

- Proof of Theorem [VNV82]: derive a uniform bound on the error:

$$\mathbb{P}_{D_n}\left(\sup_{f\in\mathcal{F}}|R_{emp}(f,D_n)-R(f,D_n)|\geq\varepsilon\right)$$

  then invert the statement
- See [VNV82, VNV13] for full details
- Related to uniform convergence of empirical processes [see, e.g. Pollard (1984) or Van der Vaart, Aad W. Asymptotic statistics (2000).]

Vapnik-Chervonenkis (VC) Theory

- Vapnik's Empirical risk minimization principle (equiv. to SAA): find the hypothesis that minimize the empirical risk function

$$\hat{f} = \operatorname*{argmin}_{f \in \mathcal{F}} R_{emp}(f, D_n)$$

- Then from the VC theory, we have with probability at least $1 - \delta$,

$$R(\hat{f}, D_n) \leq R_{emp}(\hat{f}, D_n) + 2\sqrt{\frac{d(\log(2n/d) + 1) + \log(9/\delta)}{n}}.$$

- The first term, $R_{emp}(\hat{f}, D_n)$ is smaller the larger the class $\mathcal{F}$ of hypothesis considered. The second term however grows in $d < n$, the VC dimension of $\mathcal{F}$.
- Thus, to keep the generalization error small, one should consider $\mathcal{F}$ with just the right amount of complexity to minimize the upper bound. This is the idea behind Vapnik's Structural Risk Minimization, where one finds the ERM hypothesis over an increasingly complex class of functions, as measured by its VC dimension: $\mathcal{F}_0 \subset \mathcal{F}_1 \subset \ldots$

VC Theory vs. Stability Theory

Limitations of VC Theory:

- VC dimensions of classes of functions are very hard to compute! Conversely, defining classes of functions with given VC dimension is difficult as well.
- Worst-case bound: applies to all hypothesis in $\mathcal{F}$.

Stability Theory is a more recently developed framework for learning theory that addresses the shortcomings of VC Theory.

- Key: derive bounds that are algorithm-specific, rather than over the whole hypothesis class (i.e. customized not worst-case bound)
- As such, measuring the complexity of the hypothesis class is not needed.

VC Theory vs. Stability Theory

Instead of a uniform bound

$$\mathbb{P}_{D_n}\left(\sup_{f\in\mathcal{F}} |R_{emp}(f, D_n) - R(f, D_n)| \geq \varepsilon\right),$$

Stability Theory derives an algorithm-specific bound

$$\mathbb{P}_{D_n}\left(|R_{emp}(A, D_n) - R(A, D_n)| \geq \varepsilon\right),$$

Stability Theory

- Training set: $D_n = \{z_1 = (\mathbf{x}_1, y_1), \dots, z_n = (\mathbf{x}_n, y_n)\}$, $z \in \mathcal{Z}$,
- Modified training set I:

$$D_n^{\setminus i} := \{z_1, \dots, z_{i-1}, z_{i+1}, \dots, z_n\},$$

  which leaves i-th observation out.
- Modified training set II:

$$D_n^{i} := \{z_1, \dots, z_{i-1}, z_i', z_{i+1}, \dots, z_n\},$$

  where z_i' is drawn independently from $\mathcal{X} \times \mathcal{Y}$
- Learning algorithm A is symmetric with respect to D_n if for all permutations $\pi : D_n \to D_n$ of the set D_n,

$$A_{D_n} = A_{\pi(D_n)} = A_{\{\pi(z_1), \dots, \pi(z_n)\}}.$$

Stability Theory

Definition (Uniform stability)

A symmetric algorithm A has uniform stability β with respect to a loss function ℓ if for all $i \in \{1, \ldots, n\}$,

$$\sup_{D_n \in \mathcal{Z}^n} \sup_{Y \in \mathcal{Y}} \|\ell(A_{D_n}, Y) - \ell(A_{D_n^{\setminus i}}, Y)\| \leq \beta. \tag{1}$$

Theorem (BE02)

Let A be an algorithm with uniform stability β wrt a loss function ℓ where $0 \leq \ell(A_{D_n}, z) \leq M$ for all $z \in \mathcal{Z}$ and all D_n. Then for any $n \geq 1$ and any $\delta \in (0, 1)$,

$$R(A, D_n) \leq R_{emp}(A, D_n) + 2\beta + (4n\beta + M)\sqrt{\frac{\log 1/\delta}{2n}}$$

with probability at least $1 - \delta$.

Note: the results are tight when $\beta = O(1/n)$. Call an algorithm uniformly stable if this is the case.

Stability Theory: Proof

Theorem (McDiarmid, 1989)

For any measurable function $F : \mathcal{Z}^n \mapsto \mathbb{R}$, if there exists c_i, $i = 1, \ldots, n$ such that

$$\sup_{D_n \in \mathcal{Z}^n} \sup_{z_i' \in \mathcal{Z}} |F(D_n) - F(D_n^i)| \leq c_i,$$

then

$$\mathbb{P}_{D_n}\left(|F(D_n) - \mathbb{E}_{D_n}[F(D_n)]| \geq \varepsilon\right) \leq \exp\left\{\frac{-2\varepsilon^2}{\sum_{i=1}^n c_i^2}\right\}$$

Note: the results are tight when $\beta = O(1/n)$. Call an algorithm uniformly stable if this is the case.

Strategy: Let $F := R - R_{emp}$ and show this satisfies the conditions for McDiarmid with $c_i = 4\beta + \dfrac{M}{n}$.

Stability Theory: Proof

Strategy: Let $F := R - R_{emp}$ and find bounding constant c_i's:

$$\sup_{D_n \in \mathcal{Z}^n, z_i' \in \mathcal{Z}} |F(D_n) - F(D_n^i)| \leq c_i,$$

Jensen's ineq: $|R - R^{\setminus i}| \leq \mathbb{E}_z[|\ell(A_{D_n}, z) - \ell(A, z)|] \leq \beta$

+ Triangle ineq: $|R - R^i| \leq |R - R^{\setminus i}| + |R^{\setminus i} - R^i| \leq 2\beta$

Stability Theory: Proof

Invoking the Triangle inequality twice:

$$
\begin{aligned}
&|R_{emp} - R^i_{emp}| \\
&\quad \leq \frac{1}{n}\sum_{j\neq i} |\ell(A_{D_n}, z_j) - \ell(A_{D^i_n}, z_j)| + \frac{1}{n}|\ell(A_{D_n}, z_i) - \ell(A_{D^i_n}, z'_i)| \\
&\quad \leq \frac{1}{n}\sum_{j\neq i} |\ell(A_{D_n}, z_j) - \ell(A_{D_n^{\setminus i}}, z_j)| + \frac{1}{n}\sum_{j\neq i} |\ell(A_{D_n^{\setminus i}}, z_j) - \ell(A_{D^i_n}, z_j)| \\
&\quad + \frac{1}{n}|\ell(A_{D_n}, z_i) - \ell(A^i_{D_n}, z'_i)| \\
&\quad \leq 2\beta + \frac{M}{n}
\end{aligned}
$$

Finally, by β-stability, can show

$$\mathbb{E}_{D_n}[R - R_{emp}] \leq 2\beta.$$

Stability Theory: Proof

Putting everything together, we get, for $F := R - R_{emp}$,

$$\sup_{D_n \in \mathcal{Z}^n} \sup_{z_i' \in \mathcal{Z}} |F(D_n) - F(D_n^i)| \leq 4\beta + \frac{M}{n} \; \forall \; i = 1, \ldots, n.$$

Thus, by McDiarmid's inequality, we have

$$\mathbb{P}_{D_n} \left(R - R_{emp} \geq \varepsilon + 2\beta \right) \leq \exp \left\{ \frac{-2n\varepsilon^2}{(4n\beta + M)^2} \right\}.$$

Setting the RHS to δ and inverting we arrive at the statement of the theorem.

Examples of Uniformly Stable Algorithms

- Soft margin SVM classification, where $f(x) = w^\top x - b$ for some $w \in \mathbb{R}^p$ and $b \in \mathbb{R}$, for $\mathcal{Y} = \{-1, 1\}$ with loss $\ell(f, z) = (1 - yf(x))^+$
- L_2 regularized Least-Squares regression, for bounded $\mathcal{Y}$ with loss $\ell(f, z) = (f(x) - y)^2$
- Some algorithms for the Newsvendor loss function [BR14]

The Newsvendor Problem

- $D \sim \mu$ is the random future demand,
- q is the order quantity
- Order according to:s

$$q^* \in \operatorname*{argmin}_{q \geq 0} \; \mathbb{E}_{D \sim \mu}[b(D-q)^+ + h(q-D)^+],$$

where

- b is the underage cost
- h is the overage cost

The Data-Driven Newsvendor

- Assume you have past demand data $d_1, \ldots, d_n$
- Then order according to:

$$\hat{q}_n \in \underset{q \geq 0}{\operatorname{argmin}} \ \frac{1}{n} \sum_{i=1}^{n} [b(d_i - q)^+ + h(q - d_i)^+],$$

- Stochastic Programming: Sample Average Approximation (SAA)
- Can show: $\hat{q}_n \xrightarrow{P} q^*$ exponentially fast as $n \to \infty$
- Levi, Roundy & Shmoys (2007)

The "Big" Data Newsvendor

- Past data are now $(\mathbf{x}_1, d_1), \ldots, (\mathbf{x}_n, d_n)$, where $\mathbf{x}_i \in \mathcal{X} \subset \mathbb{R}^p$
- Problem is now finding the optimal function $q : \mathcal{X} \to \mathbb{R}$:

$$\min_{q \in \mathcal{Q} = \{q : \mathcal{X} \to \mathbb{R}\}} \frac{1}{n} \sum_{i=1}^{n} [b(d_i - q(\mathbf{x}_i))^+ + h(q(\mathbf{x}_i) - d_i)^+]$$

- How should we choose $\mathcal{Q}$?

The "Big" Data Newsvendor

- Consider linear decisions:

$$\mathcal{Q} = \left\{ q : \mathcal{X} \to \mathbb{R} : q(\mathbf{x}) = \mathbf{q}^\top \mathbf{x} = \sum_{j=1}^{p} q^j x^j \right\},$$

  where $x^1 = 1$, to allow for a feature-independent term
- **Not very restrictive:** nonlinear transformation of the basic features can capture nonlinear dependencies

The “Big” Data Newsvendor

Newsvendor with Features is thus:

$$
\begin{aligned}
&\min_{\mathbf{q}=[q_1,\ldots,q_p]} \quad \frac{1}{n}\sum_{i=1}^{n}(bu_i + ho_i)\\
&s.t.\ \forall\ i = 1,\ldots,n:\\
&\qquad\qquad u_i \geq d_i - \mathbf{q}^\top \mathbf{x}_i\\
&\qquad\qquad o_i \geq \mathbf{q}^\top \mathbf{x}_i - d_i\\
&\qquad\qquad u_i, o_i \geq 0
\end{aligned}
\tag{NV-ML}
$$

- Equivalent to quantile regression

Big Data Newsvendor with Regularization

$$\begin{aligned}
&\min_{\mathbf{q}=[q_1,\dots,q_p]} \quad \frac{1}{n}\sum_{i=1}^{n}(bu_i + ho_i) + \lambda||\mathbf{q}||_k \\
&s.t.\ \forall\, i = 1,\dots,n: \\
&\qquad u_i \geq d_i - \mathbf{q}^\top \mathbf{x}_i \\
&\qquad o_i \geq \mathbf{q}^\top \mathbf{x}_i - d_i \\
&\qquad u_i, o_i \geq 0, \qquad \text{(NV-ML-Reg)}
\end{aligned}$$

where

- λ is the regularization parameter
- $k = 0, 1, 2$: MIP, LP, SOCP

The Big Data Newsvendor: Kernel-weights Optimization

- SAA is only one way to approximate expected value
- Nadaraya (1964) and Watson (1964): given $(\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_n, y_n)$, estimate $\mathbb{E}[Y|\mathbf{x}_{n+1}]$, by locally weighted average

$$\mathbb{E}[Y|\mathbf{x}_{n+1}] = \frac{\sum_{i=1}^{n} K_w(\mathbf{x}_{n+1} - \mathbf{x}_i) y_i}{\sum_{i=1}^{n} K_w(\mathbf{x}_{n+1} - \mathbf{x}_i)},$$

  where $K_w(\cdot)$ is a kernel function with bandwidth w
- Common kernel functions:
  - Uniform kernel

$$K_w(\mathbf{u}) = \frac{1}{2w}\mathbb{I}(\|\mathbf{u}\|_2 \le w)$$

  - Gaussian kernel

$$K_w(\mathbf{u}) = \frac{1}{\sqrt{2\pi}w}\exp^{-\|\mathbf{u}\|_2^2/2w^2}$$

The Big Data Newsvendor: Kernel-weights Optimization

- For an order quantity q, the BDNV expected cost after observing $\mathbf{x}_{n+1}$ is:

$$\mathbb{E}[C(q; D)|\mathbf{x}_{n+1}]$$

where $C(q; D) = b(D - q)^+ + h(q - D)^+$

- This motivates a new approach to solving BDNV:

$$\min_{q \geq 0} \frac{\sum_{i=1}^{n} K_w(\mathbf{x}_{n+1} - \mathbf{x}_i) C(q, d_i)}{\sum_{i=1}^{n} K_w(\mathbf{x}_{n+1} - \mathbf{x}_i)} \qquad \text{(NV-KO)}$$

The Big Data Newsvendor: Kernel-weights Optimization

Proposition The optimal feature-based newsvendor decision $\hat{q}_n^\kappa$ obtained by solving (NV-KO) is given by

$$\hat{q}_n^\kappa = \hat{q}_n^\kappa(\mathbf{x}_{n+1}) = \inf\left\{ q : \frac{\sum_{i=1}^n \kappa_i \mathbb{I}(q \le d_i)}{\sum_{i=1}^n \kappa_i} \ge \frac{b}{b+h} \right\},$$

where $\kappa_i = K_w(\mathbf{x}_{n+1} - \mathbf{x}_i)$.

- i.e. we can find $\hat{q}_n^\kappa$ by plugging-in the past demand in increasing order, and choosing the smallest value at which the inequality above is satisfied.

Stability bounds for Big Data NV

Proposition (Uniform stability of (NV-ML))

The learning algorithm (NV-ML) with iid data is symmetric and uniformly stable with respect to the newsvendor cost function $C(\cdot,\cdot)$ with stability parameter

$$\beta = \frac{\bar{D}(b \vee h)^2}{(b \wedge h)} \frac{p}{n}.$$

Theorem (Bound on the Gen. error of (NV-ML))

Let $\hat{q}$ be the solution to (NV-ML). Then with probability at least $1-\delta$,

$$|R(\hat{q}; S_n) - \hat{R}_{in}(\hat{q}; S_n)| \leq (b \vee h)\bar{D}\left[\frac{2(b \vee h)}{b \wedge h}\frac{p}{n} + \left(\frac{4(b \vee h)}{b \wedge h}p + 1\right)\sqrt{\frac{\ln(2/\delta)}{2n}}\right]$$

Stability bounds for Big Data NV

Proposition (Uniform stability of (NV-ML-Reg))

The learning algorithm (NV-ML-Reg) with $k = 2$ is symmetric, and is uniformly stable with respect to the newsvendor cost function C with stability parameter

$$\beta = \frac{(b \vee h)^2 X_{\max}^2 p}{2n\lambda},$$

where the feature vector $\mathbf{X}$ is normalized ($\mathbf{X}_1 = 1$ almost surely, $\mathbf{X}_{[2:p]}$ has mean zero and standard deviation one) and that it lives in a closed unit ball: $||\mathbf{X}||_2 \leq X_{\max}\sqrt{p}$.

Theorem (Bound on the Gen. error of (NV-ML-Reg))

Denote the solution to (NV-ML-Reg) by $\hat{q}_\lambda = \hat{q}_\lambda(\mathbf{x}_{n+1})$. Then with probability at least $1 - \delta$,

$$\begin{aligned} &|R(\hat{q}_\lambda; S_n) - \hat{R}_{in}(\hat{q}_\lambda; S_n)| \\ &\leq (b \vee h)\bar{D}\left[\frac{(b \vee h)X_{\max}^2 p}{n\lambda\bar{D}} + \left(\frac{2(b \vee h)X_{\max}^2 p}{\lambda\bar{D}} + 1\right)\sqrt{\frac{\log(2/\delta)}{2n}}\right] \end{aligned}$$

Stability bounds for Big Data NV

Proposition (Uniform stability of (NV-KO))

The algorithm (NV-KO) with iid data and the Gaussian kernel is symmetric with respect to the newsvendor cost function $C(\cdot,\cdot)$ with uniform stability parameter

$$\beta = \frac{\bar{D}(b \vee h)^2}{(b \wedge h)} \frac{1}{1+(n-1)r_w},$$

where $r_w = \exp(-2X_{\max}^2 p/w^2)$.

Theorem (Bound on the Gen. error of (NV-KO))

Denote the solution to (NV-KO) with the Gaussian kernel by $\hat{q}^\kappa = \hat{q}^\kappa(\mathbf{x}_{n+1})$. Then with probability at least $1-\delta$,

$$|R(\hat{q}^\kappa; S_n) - \hat{R}_{in}(\hat{q}^\kappa; S_n)| \leq$$

$$(b \vee h)\bar{D}\left[\frac{2(b \vee h)}{b \wedge h}\frac{1}{1+(n-1)r_w(p)} + \left(\frac{4(b \vee h)}{1/n+(1-1/n)r_w(p)} + 1\right)\sqrt{\frac{\log(2/\delta)}{2n}}\right]$$

where $r_w(p) = \exp(-2X_{\max}^2 p/w^2)$, w the kernel bandwidth.

Stability bounds for Big Data NV

- (NV-ML): bound scales as $O(p/\sqrt{n})$
- (NV-ML-Reg): bound scales as $O(p/\sqrt{n}\lambda)$; want λ large for better generalization
- (NV-KO): bound scales as $O(1/\sqrt{n}r_w(p))$, so can be controlled by increasing $r_w(p)$ by increasing the kernel bandwidth w. This makes sense: larger w corresponds to smoother comparisons of past features to the one in $n+1$.
- Of course, generalization error isn't everything- improved generalization error comes at the cost of increased finite-sample bias. See [BR14] for details.

Nurse Staffing in a Hospital Emergency Room

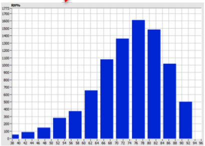
- Mandatory/recommended nurse-to-patient ratio
- Underage: must call expensive agency nurses; Overage: idle regular nurses
- Data: emergency room of a large UK hospital from July 2008-Feb 2009, recorded every two hours
- Features: day of the week, time of the day, 2 weeks of past demand (171 features)

Nurse Staffing in a Hospital Emergency Room

Training: 4 months	Validation: 2 months	Testing: 2 months

Jul ‘08

Feb ‘09



Methods considered

Abbreviation	Description	Reg.?	Free parameter
1. SAA-day	SAA by day of the week	None	None
2. Ker-0	(NV-KO) with Gaussian kernel	None	bandwidth
3. Ker-OS	"	None	"
4. NV-0	solve (NV-ML)	None	no. of days of past demand
5. NV-OS	"	None	"
6. NVreg1	solve (NV-ML-Reg)	Yes, ℓ_1	regularization parameter
7. NVreg1-OS	"	Yes, ℓ_1	"
8. NVreg2	"	Yes, ℓ_2	"
9. NVreg2-OS	"	Yes, ℓ_2	"
10. SEO-0	OLS regression + NV opt.	None	no. of days of past demand
11. SEO-OS	"	None	"
12. SEOreg1	Lasso regression + NV opt.	Yes, ℓ_1	regularization parameter
13. SEOreg1-OS	"	Yes, ℓ_1	"
14. SEOreg2	Ridge regression + NV opt.	Yes, ℓ_2	"
15. SEOreg2-OS	"	Yes, ℓ_2	"
16. Scarf	Minimax optimization	None	no. of days of past demand

Out-of-Sample Results

Method	Calibrated parameter	Mean (95 % CI)	Annual cost savings rel. to SAA-day
1. SAA-day	—	1.523 ($\pm$ 0.109)	—
3. Ker-OS	$h = 1.62$	1.156 ($\pm$ 0.140)	£46,555 ($ 74,488)
4. NV-0	12 days	1.326 ($\pm$ 0.100)	£24,909 ($ 39,854)
7. NVreg1-OS	1×10^{-7}	1.174 ($\pm$ 0.113)	£44,219 ($ 70,750)
9. NVreg2-OS	1×10^{-7}	1.215 ($\pm$ 0.111)	£39,065 ($ 62,503)
10. SEO-0	1 day	1.279 ($\pm$ 0.099)	£30,952 ($ 49,523)
16. Scarf	12 days	1.593 ($\pm$ 0.114)	—

Table:

- Assuming hourly wage of an agency nurse is 2.5 times that of a regular nurse.
- Assuming a regular nurse salary of £25,000 (which is the Band 4 nurse salary for the National Health Service in the United Kingdom in 2014) and standard working hours. Cost savings in USD assumes an exchange rate of £1: USD 1.6.

Out-of-Sample Results

Method	Calibrated parameter	Mean (95 % CI)	Annual cost savings rel. to SAA-day
1. SAA-day	—	1.523 ($\pm$ 0.109)	—
3. Ker-OS	$h = 1.62$	1.156 ($\pm$ 0.140)	£46,555 ($ 74,488)
4. NV-0	12 days	1.326 ($\pm$ 0.100)	£24,909 ($ 39,854)
7. NVreg1-OS	1×10^{-7}	1.174 ($\pm$ 0.113)	£44,219 ($ 70,750)
9. NVreg2-OS	1×10^{-7}	1.215 ($\pm$ 0.111)	£39,065 ($ 62,503)
10. SEO-0	1 day	1.279 ($\pm$ 0.099)	£30,952 ($ 49,523)
16. Scarf	12 days	1.593 ($\pm$ 0.114)	—

Table:

- Assuming hourly wage of an agency nurse is 2.5 times that of a regular nurse.
- Assuming a regular nurse salary of £25,000 (which is the Band 4 nurse salary for the National Health Service in the United Kingdom in 2014) and standard working hours. Cost savings in USD assumes an exchange rate of £1: USD 1.6.

Out-of-Sample Results

Method	Calibrated parameter	Avg. Time (per iteration)	Annual cost savings rel. to SAA-day
1. SAA-day	—	14.0 s	—
3. Ker-OS	$h = 1.62$	0.0494 s	£46,555 ($ 74,488)
4. NV-0	12 days	325 s	£24,909 ($ 39,854)
7. NVreg1-OS	1×10^{-7}	114 s	£44,219 ($ 70,750)
9. NVreg2-OS	1×10^{-7}	107 s	£39,065 ($ 62,503)
10. SEO-0	1 day	10.8 s	£30,952 ($ 49,523)
16. Scarf	12 days	20.8 s	—

Table:

- Assuming hourly wage of an agency nurse is 2.5 times that of a regular nurse.
- Assuming a regular nurse salary of £25,000 (which is the Band 4 nurse salary for the National Health Service in the United Kingdom in 2014) and standard working hours. Cost savings in USD assumes an exchange rate of £1: USD 1.6.

Part II Summary

Statistical Learning Theory:

- Framework for analyzing in-sample and out-of-sample errors
- Origins: VC Theory (uniform convergence of empirical processes), more recently Stability Theory (McDiarmid's inequality)
- Value: can theoretically compare different learning algorithms, in particular how they scale with problem parameters and data size
- Complements other dimensions of algorithmic performance: e.g. computational efficiency, performance on real data sets, interpretability

Take Aways & Future Directions

- “Traditional” stochastic modeling won’t become obsolete — if anything, more important/appreciated by the outer world as more and more OR results become implementable and generate economic value
- However, starting from real data or evidence-based assumptions about the data-generating process will become increasingly important
- New research opportunities (beyond glorified case-studies): building innovative data-integrated decision models, designing efficient algorithms, performance analysis (asymptotics, finite-sample bounds, learning rates)
- Will upload slides on gahyiban.com soon