Introduction to Machine Learning and Cross-Validation

Jonathan Hersh

February 27, 2019
Plan

1. Introduction
2. Preliminary Terminology
3. Bias-Variance Trade-off
4. Cross-Validation
5. Conclusion
Machine learning versus econometrics

### Machine Learning
- Developed to solve problems in computer science

### Econometrics
- Developed to solve problems in economics

Machine Learning
- Prediction/classification
- Want: goodness of fit
- Huge datasets (many terabytes), large # variables (1000s)
- Whatever works

Econometrics
- Explicitly testing a theory
- "Statistical significance" more important than model fit
- Small datasets, few variables
- "It works in practice, but what about in theory?"
- Can we utilize some of this machinery to solve problems in development economics?
## Machine learning versus econometrics

### Machine Learning
- Developed to solve problems in computer science
- Prediction/classification ✓

### Econometrics
- Developed to solve problems in economics
- Explicitly testing a theory
Machine learning versus econometrics

Machine Learning
- Developed to solve problems in computer science
- Prediction/classification ✓
- Want: goodness of fit

Econometrics
- Developed to solve problems in economics
- Explicitly testing a theory
- “Statistical significance” more important than model fit
Machine learning versus econometrics

**Machine Learning**
- Developed to solve problems in computer science
- Prediction/classification ✓
- Want: goodness of fit
- Huge datasets (many terabytes), large # variables (1000s)

**Econometrics**
- Developed to solve problems in economics
- Explicitly testing a theory
- “Statistical significance” more important than model fit
- Small datasets, few variables
Machine learning versus econometrics

Machine Learning
- Developed to solve problems in computer science
- Prediction/classification ✓
- Want: goodness of fit
- Huge datasets (many terabytes), large # variables (1000s)
- Whatever works

Econometrics
- Developed to solve problems in economics
- Explicitly testing a theory
- “Statistical significance” more important than model fit
- Small datasets, few variables
- “It works in practice, but what about in theory?”
Introduction

Preliminary Terminology

Bias-Variance Trade-off

Cross-Validation

Conclusion

Machine learning versus econometrics

Machine Learning

- Developed to solve problems in computer science
- Prediction/classification ✓
- Want: goodness of fit
- Huge datasets (many terabytes), large # variables (1000s)
- Whatever works

Econometrics

- Developed to solve problems in economics
- Explicitly testing a theory
- “Statistical significance” more important than model fit
- Small datasets, few variables
- “It works in practice, but what about in theory?”

Can we utilize some of this machinery to solve problems in development economics?
Applications of Machine Learning in economics (due to Sendhil Mullainathan)

1. New data
2. Predictions for policy
3. Better econometrics

- See Machine learning: an applied econometric approach, JEP
This course

- **Primer in statistical learning theory, which grew out of statistics**
- How does this differ from ML? Machine learning places more emphasis on large scale applications and prediction accuracy. Statistical learning covers
This course

- **Primer in statistical learning theory, which grew out of statistics**
- How does this differ from ML? Machine learning places more emphasis on large scale applications and prediction accuracy. Statistical learning covers
- There is much overlap and cross-fertilization
This course

- **Primer in statistical learning theory, which grew out of statistics**

- How does this differ from ML? Machine learning places more emphasis on large scale applications and prediction accuracy. Statistical learning covers

- There is much overlap and cross-fertilization

- Very little coding, but example code provided: jonathan-hersh.com/machinelearningdev
Topics covered

1. **Cross-validation** [Chapter 2]
Topics covered

1. **Cross-validation** [Chapter 2]
2. **Shrinkage methods** (Ridge and LASSO) [Chapter 6]
Topics covered

1. **Cross-validation** [Chapter 2]
2. **Shrinkage methods** (Ridge and LASSO) [Chapter 6]
3. **Classification** [Chapter 4, APM Chapter 11-12]
Topics covered

1. **Cross-validation** [Chapter 2]
2. **Shrinkage methods** (Ridge and LASSO) [Chapter 6]
3. **Classification** [Chapter 4, APM Chapter 11-12]
4. **Tree-based methods** (Decision trees, bagging, random forest boosting) [Chapter 8]
Topics covered

1. **Cross-validation** [Chapter 2]
2. **Shrinkage methods** (Ridge and LASSO) [Chapter 6]
3. **Classification** [Chapter 4, APM Chapter 11-12]
4. **Tree-based methods** (Decision trees, bagging, random forest boosting) [Chapter 8]
5. **Unsupervised learning** (PCA, k-means clustering, hierarchical clustering) [Chapter 10]
Topics covered

1. **Cross-validation** [Chapter 2]
2. **Shrinkage methods** (Ridge and LASSO) [Chapter 6]
3. **Classification** [Chapter 4, APM Chapter 11-12]
4. **Tree-based methods** (Decision trees, bagging, random forest boosting) [Chapter 8]
5. **Unsupervised learning** (PCA, k-means clustering, hierarchical clustering) [Chapter 10]
6. **Caret** Automated Machine Learning [APM]
Plan

1. Introduction
2. Preliminary Terminology
3. Bias-Variance Trade-off
4. Cross-Validation
5. Conclusion
Supervised vs. unsupervised learning

Def: Supervised learning
for every $x_i$ we also observe a response $y_i$
Supervised vs. unsupervised learning

Def: Supervised learning
for every $x_i$ we also observe a response $y_i$

Ex: Estimating housing values by OLS or random forest;
Supervised vs. unsupervised learning

**Def: Supervised learning**
for every $x_i$ we also observe a response $y_i$

*Ex: Estimating housing values by OLS or random forest;*

**Def: Unsupervised learning**
for each observation we **only** observe $x_i$, **but do not observe** $y_i$
Supervised vs. unsupervised learning

Def: Supervised learning
for every $x_i$ we also observe a response $y_i$

Ex: Estimating housing values by OLS or random forest;

Def: Unsupervised learning
for each observation we only observe $x_i$, but do not observe $y_i$

Ex: Clustering customers into segments; using principle component analysis for dimension reduction
Test, Training, and Validation Set

- **Training set**: (observation-wise) subset of data used to develop models
Test, Training, and Validation Set

- **Training set**: (observation-wise) subset of data used to develop models

- **Validation set**: subset of data used during intermediate stages to “tune” model parameters

- **Test set**: subset of data used sparingly to approximate out of sample fit
Test, Training, and Validation Set

- **Training set**: (observation-wise) subset of data used to develop models

- **Validation set**: subset of data used during intermediate stages to “tune” model parameters

- **Test set**: subset of data (used sparingly) to approximate out of sample fit
Assessing model accuracy

Mean squared error
measures how well model predictions match observed data

\[
MSE\left(\hat{f}(x)\right) = \frac{1}{N} \sum_{i=1}^{N} \left( y_i^{\text{data}} - \hat{f}(x_i)^{\text{model}} \right)^2
\]
Assessing model accuracy

Mean squared error
measures how well model predictions match observed data

\[
MSE(\hat{f}(x)) = \frac{1}{N} \sum_{i=1}^{N} \left( y_i - \hat{f}(x_i) \right)^2
\]

Training MSE vs Test MSE: good in-sample fit (low training MSE)
can often obscure poor out of sample fit (high test MSE)
Plan

1. Introduction
2. Preliminary Terminology
3. Bias-Variance Trade-off
4. Cross-Validation
5. Conclusion
Quick example on out-of-sample fit

- Let’s compare three estimators to see how estimator complexity affects out of sample fit

1. \( f_1 = \) linear regression (in orange)
2. \( f_2 = \) third order polynomial (in black)
3. \( f_3 = \) very flexible smoothing spline (in green)
Case 1: True $f(x)$ slightly complicated

**FIGURE 2.9.** Left: Data simulated from $f$, shown in black. Three estimates of $f$ are shown: the linear regression line (orange curve), and two smoothing spline fits (blue and green curves). Right: Training MSE (grey curve), test MSE (red curve), and minimum possible test MSE over all methods (dashed line). Squares represent the training and test MSEs for the three fits shown in the left-hand panel.
Case 2: True $f(x)$ not complicated at all

FIGURE 2.10. Details are as in Figure 2.9, using a different true $f$ that is much closer to linear. In this setting, linear regression provides a very good fit to the data.
Case 2: True $f(x)$ not complicated at all

**FIGURE 2.10.** Details are as in Figure 2.9, using a different true $f$ that is much closer to linear. In this setting, linear regression provides a very good fit to the data.
Case 3: True $f(x)$ very complicated

**FIGURE 2.11.** Details are as in Figure 2.9, using a different $f$ that is far from linear. In this setting, linear regression provides a very poor fit to the data.
How to select right model complexity?

- High bias (underfit)
- "Just right"
- High variance (overfit)
Generalizing this problem: Bias-Variance tradeoff

![Bias-Variance tradeoff diagram]

**FIGURE 2.11.** Test and training error as a function of model complexity.
Bias Variance Tradeoff in Math

Prediction Error: \[ E \left[ \left( \frac{y_i}{\text{data}} - \hat{f}(x_i) \right) \right]^2 \]

\[ \equiv \text{Var}(y) + \text{Var}(\hat{f}) + \text{bias}^2 - \text{var}(\hat{f}) \]

\[ \equiv \text{Var}(y) + \text{Var}(\hat{f}) + \text{bias}^2 - \text{var}(\hat{f}) \]

\[ \hat{y} = f(x) + \epsilon, \quad E[\epsilon] = 0 \]

\[ \text{def'ns var} \]
Bias Variance Tradeoff in Math

Prediction Error: \[ \mathbb{E} \left[ \left( y_i - \hat{f}(x_i) \right)^2 \right] \]

\[ \mathbb{E} \left[ (y - \hat{f})^2 \right] = \mathbb{E} \left[ y^2 + \hat{f}^2 - 2y\hat{f} \right] \quad \text{(expanding terms)} \]
Bias Variance Tradeoff in Math

**Prediction Error:**

\[ \mathbb{E} \left[ \left( y_i - \hat{f}(x_i) \right)^2 \right] \]

\[ \mathbb{E} \left[ (y - \hat{f})^2 \right] = \mathbb{E} \left[ y^2 + \hat{f}^2 - 2y\hat{f} \right] \quad \text{(expanding terms)} \]

\[ = \mathbb{E} \left[ y^2 \right] + \mathbb{E} \left[ \hat{f}^2 \right] - \mathbb{E} \left[ 2y\hat{f} \right] \]

\[ \equiv \text{Var}(y) + \mathbb{E}[y]^2 \quad \equiv \text{Var}(\hat{f}) + \mathbb{E}[\hat{f}]^2 \]
Bias Variance Tradeoff in Math

Prediction Error: \[ E \left[ \left( \underbrace{y_i}_{\text{data}} - \hat{f} (x_i) \right)^2 \right] \]

\[ E \left[ (y - \hat{f})^2 \right] = E \left[ y^2 + \hat{f}^2 - 2y\hat{f} \right] \quad \text{(expanding terms)} \]

\[ = E \left[ y^2 \right] + E \left[ \hat{f}^2 \right] - E \left[ 2y\hat{f} \right] \]

\[ \equiv \text{Var}(y) + \text{E}[y]^2 \] \quad \text{\( \equiv \text{Var}(\hat{f}) + \text{E}[\hat{f}]^2 \)}

\[ = \text{Var}(y) + \text{E}[y]^2 + \text{Var}(\hat{f}) + \text{E}[\hat{f}]^2 - \text{E}[2y\hat{f}] \quad \text{(def’n var)} \]

\[ y \equiv f(x) + \varepsilon, \text{E}[\varepsilon] = 0 \]
Bias Variance Tradeoff in Math

**Prediction Error:**

\[
\mathbb{E} \left[ \left( \frac{y_i}{\text{data}} - \hat{f}(x_i) \right)^2 \right]
\]

\[
\mathbb{E} \left[ (y - \hat{f})^2 \right] = \mathbb{E} \left[ y^2 + \hat{f}^2 - 2y\hat{f} \right] \quad \text{(expanding terms)}
\]

\[
= \mathbb{E} \left[ y^2 \right] + \mathbb{E} \left[ \hat{f}^2 \right] - \mathbb{E} \left[ 2y\hat{f} \right]
\]

\[
= \text{Var}(y) + \mathbb{E}[y]^2 + \text{Var}(\hat{f}) + \mathbb{E}[\hat{f}]^2 - \mathbb{E} \left[ 2y\hat{f} \right] \quad \text{(def'\ n var)}
\]

\[
= \text{Var}(y) + \text{Var}(\hat{f}) + \left( \mathbb{E} [\hat{f}]^2 - \mathbb{E} \left[ 2y\hat{f} \right] + f^2 \right) \quad \text{(def'\ n y)}
\]
Bias Variance Tradeoff in Math

\[ \text{Var} (y) = \mathbb{E}[(y - \mathbb{E}[y])^2] = \mathbb{E}[(y - f)^2] = \mathbb{E}[^2] = \text{Var}(\varepsilon) = \sigma^2 \]

\[ = \text{Var} (\hat{f}) + (f - \mathbb{E} [\hat{f}])^2 \]

Bias is minimized when

\[ f = \mathbb{E} [\hat{f}] \]

But total error (variance + bias) may be minimized by some

\[ \hat{f}(x) \]

with smaller variance

⇒ fewer variables, smaller magnitude coefficients
Bias Variance Tradeoff in Math

\[ \text{Bias} \quad \text{Variance} \quad \text{Irreducible Error} \]

Prediction Error

\[ \mathbb{E} \left[ (y_i - \hat{f}) \right] = \underbrace{\sigma_{\varepsilon}^2}_{\text{irreducible error}} + \underbrace{\text{Var} (\hat{f})}_{\text{variance}} + \underbrace{\left( f - \mathbb{E} [\hat{f}] \right)^2}_{\text{bias}} \]
Bias Variance Tradeoff in Math

\[ \begin{align*}
\text{Prediction Error} & \quad \mathbb{E} \left[ (y_i - \hat{f})^2 \right] \\
& = \underbrace{\sigma^2_{\varepsilon}}_{\text{irreducible error}} + \underbrace{\text{Var} \left( \hat{f} \right)}_{\text{variance}} + \underbrace{(f - \mathbb{E} \left[ \hat{f} \right])^2}_{\text{bias}}
\end{align*} \]

- Bias is minimized when \( f = \mathbb{E} \left[ \hat{f} \right] \)
- But total error (variance + bias) may be minimized by some other \( \hat{f} \)
Bias Variance Tradeoff in Math

\[ = \underbrace{\text{Var } (y)} + \text{Var } (\hat{f}) + (f - \mathbb{E}[\hat{f}])^2 \]

\[ = \mathbb{E}[(y - \mathbb{E}[y])^2] = \mathbb{E}[(y - f)^2] = \mathbb{E}[\epsilon^2] = \text{Var } (\epsilon) = \sigma_\epsilon^2 \]

Prediction Error

\[ \mathbb{E} \left[ (y_i - \hat{f}) \right] = \underbrace{\sigma_\epsilon^2}_{\text{irreducible error}} + \underbrace{\text{Var } (\hat{f}) + (f - \mathbb{E}[\hat{f}])^2}_{\text{variance + bias}} \]

- Bias is minimized when \( f = \mathbb{E}[\hat{f}] \)
- But total error (variance + bias) may be minimized by some other \( \hat{f} \)
- \( \hat{f}(x) \) with smaller variance \( \Rightarrow \) fewer variables, smaller magnitude coefficients
Plan

1. Introduction
2. Preliminary Terminology
3. Bias-Variance Trade-off
4. Cross-Validation
5. Conclusion
Cross-Validation

- Cross-validation is a tool to approximate out of sample fit
Cross-Validation

- Cross-validation is a tool to approximate out of sample fit

- In machine learning, many models have parameters that must be “tuned”

- We adjust these parameters using using cross-validation
Cross-Validation

- Cross-validation is a tool to approximate out of sample fit

- In machine learning, many models have parameters that must be “tuned”

- We adjust these parameters using cross-validation

- Also useful to select between large classes of models
  - e.g. random forest vs lasso
### K-fold Cross-Validation (CV)

#### K-fold CV Algorithm

1. Randomly divide the data into $K$ equal sized parts or "folds".

<table>
<thead>
<tr>
<th></th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Train</td>
<td>Train</td>
<td>Validation</td>
<td>Train</td>
<td>Train</td>
</tr>
</tbody>
</table>

---

J.Hersh (Chapman)  
Intro & CV  
February 27, 2019  
23 / 29
K-fold Cross-Validation (CV)

1. Randomly divide the data into $K$ equal sized parts or “folds”.
2. Leave out part $k$, fit the model to the other $K - 1$ parts.
K-fold Cross-Validation (CV)

K-fold CV Algorithm

1. Randomly divide the data into $K$ equal sized parts or “folds”.
2. Leave out part $k$, fit the model to the other $K - 1$ parts.
3. Use fitted model to obtain predictions for left-out $k$-th part
K-fold Cross-Validation (CV)

K-fold CV Algorithm

1. Randomly divide the data into $K$ equal sized parts or "folds".
2. Leave out part $k$, fit the model to the other $K-1$ parts.
3. Use fitted model to obtain predictions for left-out $k$-th part
4. Repeat until $k = 1, \ldots, K$ and combine results
Fig. 4.6: A schematic of threefold cross-validation. Twelve training set samples are represented as symbols and are allocated to three groups. These groups are left out in turn as models are fit. Performance estimates, such as the error rate or $R^2$ are calculated from each set of held-out samples. The average of the three performance estimates would be the cross-validation estimate of model performance. In practice, the number of samples in the held-out subsets can vary but are roughly equal size.
Details of K-Fold CV

- Let $n_k$ be the number of test observations in fold $k$, where $n_k = N/K$
- Cross-Validation Error for fold $k$:

$$CV\{k\} = \sum_{k=1}^{K} \frac{n_k}{N} MSE_k$$

where $MSE_k = \sum_{i \in C_k} (y_i - \hat{y})^2 / n_k$ is the mean squared error of fold $k$
Details of K-Fold CV

- Let $n_k$ be the number of test observations in fold $k$, where $n_k = N/K$
- Cross-Validation Error for fold $k$:

\[
CV\{k\} = \sum_{k=1}^{K} \frac{n_k}{N} MSE_k
\]

where $MSE_k = \sum_{i \in C_k} (y_i - \hat{y})^2 / n_k$ is the mean squared error of fold $k$

- Setting $k = N$ is referred to as leave-one-out cross-validation (LOOCV)
Classical frequentist model selection

Akaike information criterion (AIC)

\[ AIC = -\frac{2}{N} \cdot \text{loglik} + 2 \cdot \frac{d}{N} \]

where \( d \) is the number of parameters in our model

Bayesian information criterion (BIC)

\[ BIC = -2 \cdot \text{loglik} + (\log N) \cdot d \]
Classical frequentist model selection

Akaike information criterion (AIC)

$$AIC = -\frac{2}{N} \cdot \text{loglik} + 2 \cdot \frac{d}{N}$$

where $d$ is the number of parameters in our model

Bayesian information criterion (BIC)

$$BIC = -2 \cdot \text{loglik} + (\log N) \cdot d$$

- Both penalize models with more parameters in a somewhat arbitrary fashion
- This usually helps with model selection, but still does not answer the important question of model assessment
What is the Optimal Number of Folds, K?

Do you want big or a small training folds?
What is the Optimal Number of Folds, K?

- Do you want big or a small training folds?
- Because training set is only \((K - 1)/K\) as big as the full dataset, the estimates of the prediction error will be biased upward.
- Bias is minimized when \(K = N\) (LOOCV)
- But LOOCV has higher variance!
What is the Optimal Number of Folds, $K$?

- Do you want big or a small training folds?
- Because training set is only $(K - 1)/K$ as big as the full dataset, the estimates of the prediction error will be biased upward.
- Bias is minimized when $K = N$ (LOOCV)
- But LOOCV has higher variance!
- No clear statistical rules for how to set $k$
- Convention is to set $K = 5$ or $10$ – in practice is a good trade-off between bias and variance for most problems
Plan

1. Introduction
2. Preliminary Terminology
3. Bias-Variance Trade-off
4. Cross-Validation
5. Conclusion
Conclusion

- Econometric models are at risk for overfitting
- But what we want are theories that extend beyond the dataset we have on our computer
- Cross-validation is a key tool that allows us to adjust models so that they more closely match reality