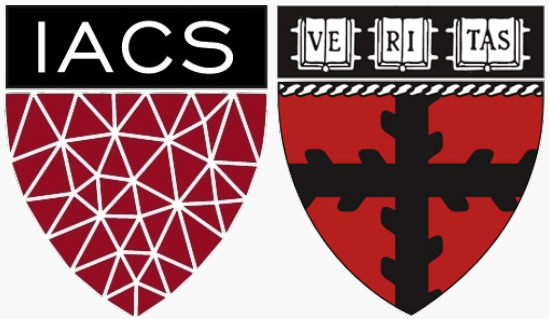
Lecture #6: Model Selection & Cross Validation

Data Science 1
CS 109A, STAT 121A, AC 209A, E-109A

Pavlos Protopapas Kevin Rader
Margo Levine Rahul Dave



Lecture Outline

Review

Multiple Regression with Interaction Terms

Model Selection: Overview

Stepwise Variable Selection

Cross Validation

Applications of Model Selection

Review

Multiple Linear and Polynomial Regression

Last time, we saw that we can build a linear model for multiple predictors, $\{X_1, \ldots, X_J\}$,

$$y = \beta_0 + \beta_1 x_1 + \ldots + \beta_J x_J + \epsilon.$$

Using vector notation,

$$\mathbf{Y} = \begin{pmatrix} y_1 \\ \vdots \\ y_y \end{pmatrix}, \quad \mathbf{X} = \begin{pmatrix} 1 & x_{1,1} & \ldots & x_{1,J} \\ 1 & x_{2,1} & \ldots & x_{2,J} \\ \vdots & \vdots & \ddots & \vdots \\ 1 & x_{n,1} & \ldots & x_{n,J} \end{pmatrix}, \quad \boldsymbol{\beta} = \begin{pmatrix} \beta_0 \\ \beta_1 \\ \vdots \\ \beta_J \end{pmatrix},$$

We can express the regression coefficients as

$$\widehat{\boldsymbol{\beta}} = \underset{\boldsymbol{\beta}}{\operatorname{argmin}}\ \mathsf{MSE}(\boldsymbol{\beta}) = \left(\mathbf{X}^\top \mathbf{X}\right)^{-1} \mathbf{X}^\top \mathbf{Y}.$$

Multiple Linear and Polynomial Regression

We also saw that there are ways to generalize multiple linear regression:

- Polynomial regression

$$y = \beta_0 + \beta_1 x + \ldots + \beta_M x^M + \epsilon.$$

- Polynomial regression with multiple predictors

In each case, we treat each polynomial term x_j^m as an unique predictor and perform multiple linear regression.

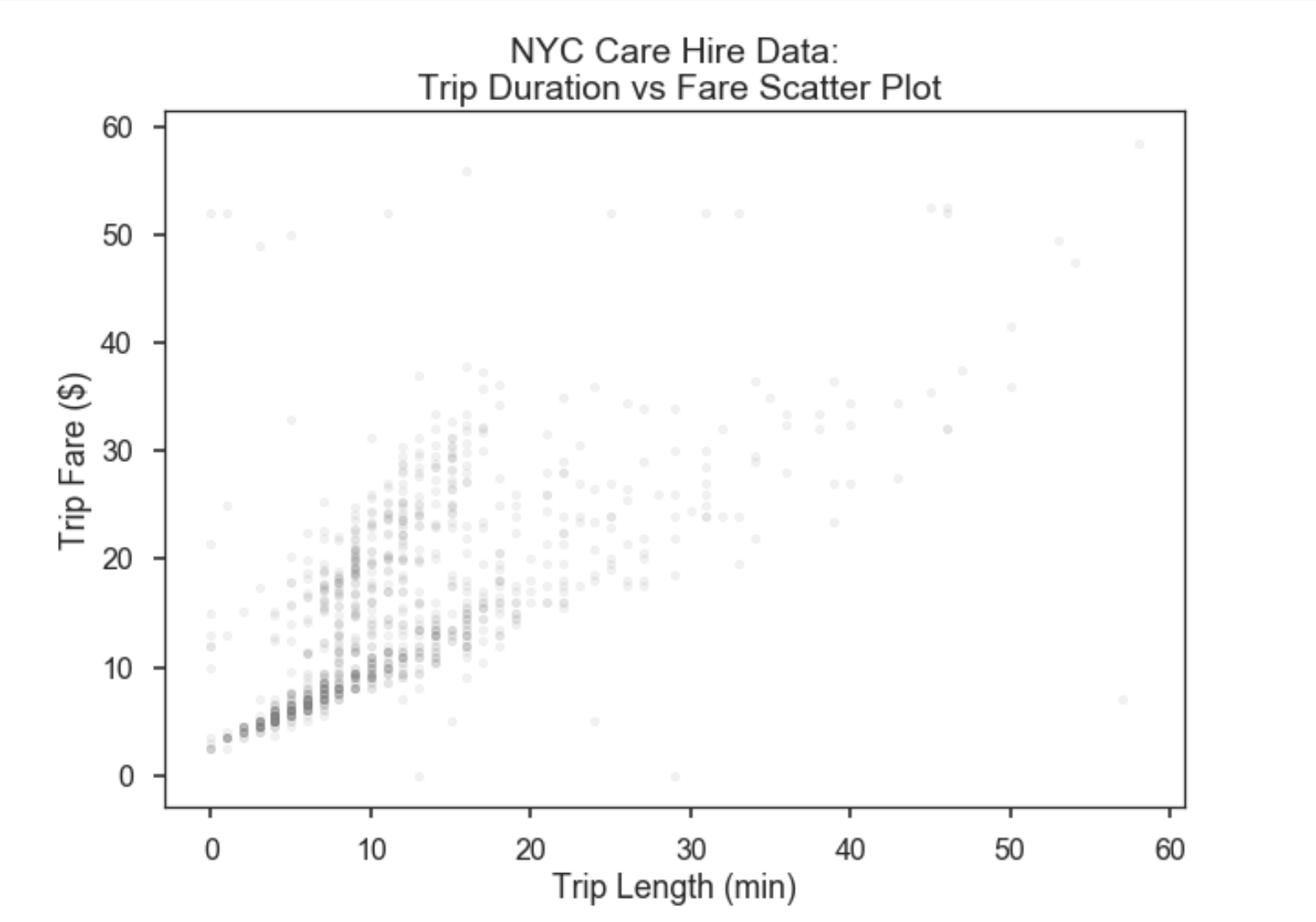
Selecting Significant Predictors

When modeling with multiple predictors, we are interested in which predictor or sets of predictors have a significant effect on the response.

Significance of predictors can be measured in multiple ways:

- **Hypothesis testing:**
  - Subsets of predictors with higher F-stats higher than 1 may be significant.
  - Individual predictors with p-values smaller than established threshold (e.g. 0.05) may be significant.
- **Evaluating model fitness:**
  - Subsets of predictors with higher model R^2 should be more significant.
  - Subsets of predictors with lower model AIC or BIC should be more significant.

Example


NYC Care Hire Data:
Trip Duration vs Fare Scatter Plot
Trip Fare ($)
0
10
20
30
40
50
60
Trip Length (min)
0
10
20
30
40
50
60

Multiple Regression with Interaction Terms

Interacting Predictors

In our multiple linear regression model for the NYC taxi data, we considered two predictors, rush hour indicator x_1 (in 0 or 1) and trip length x_2 (in minutes),

$$y = \beta_0 + \beta_1 x_1 + \beta_2 x_2.$$

This model assumes that each predictor has an independent effect on the response, e.g. regardless of the time of day, the fare depends on the length of the trip in the same way.

In reality, we know that a 30 minute trip covers a shorter distance during rush hour than in normal traffic.

Interacting Predictors

A better model considers how the interactions between the two predictors impact the response,

$$y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \beta_3 x_1 x_2.$$

The term $\beta_3 x_1 x_2$ is called the ***interaction term***. It determines the effect on the response when we consider the predictors jointly.

For example, the effect of trip length on cab fare in the absence of rush hour is $\beta_2 x_2$. When combined with rush hour traffic ($x_1 = 1$), the effect of trip length is $(\beta_2 + \beta_3) x_2$.

Multiple Linear Regression with Interaction Terms

Multiple linear regression with interaction terms can be treated like a special form of multiple linear regression - we simply treat the cross terms (e.g. $x_1 x_2$) as additional predictors.

Given a set of observations $\{(x_{1,1}, x_{1,2}, y_1), \dots (x_{n,1}, x_{n,2}, y_n)\}$, the data and the model can be expressed in vector notation,

$$\mathbf{Y} = \begin{pmatrix} y_1 \\ \vdots \\ y_n \end{pmatrix}, \quad \mathbf{X} = \begin{pmatrix} 1 & x_{1,1} & x_{1,2} & x_{1,1}x_{1,2} \\ 1 & x_{2,1} & x_{2,2} & x_{2,1}x_{2,2} \\ \vdots & \vdots & \vdots & \vdots \\ 1 & x_{n,1} & x_{n,2} & x_{n,1}x_{n,2} \end{pmatrix}, \quad \boldsymbol{\beta} = \begin{pmatrix} \beta_0 \\ \beta_1 \\ \beta_2 \\ \beta_3 \end{pmatrix},$$

Again, minimizing the MSE using vector calculus yields,

$$\widehat{\boldsymbol{\beta}} = \underset{\boldsymbol{\beta}}{\operatorname{argmin}} \operatorname{MSE}(\boldsymbol{\beta}) = \left(\mathbf{X}^\top \mathbf{X}\right)^{-1} \mathbf{X}^\top \mathbf{Y}.$$

Generalized Polynomial Regression

We can generalize polynomial models:

1. considering polynomial models with multiple predictors $\{X_1, \ldots, X_J\}$:

$$\begin{aligned} y = &\beta_0 + \beta_1 x_1 + \ldots + \beta_M x_1^M \\ &+ \ldots \\ &+ \beta_{1+MJ} x_J + \ldots + \beta_{M+MJ} x_J^M \end{aligned}$$

2. consider polynomial models with multiple predictors $\{X_1, X_2\}$ and cross terms:

$$\begin{aligned} y = &\beta_0 + \beta_1 x_1 + \ldots + \beta_M x_1^M \\ &+ \beta_{1+M} x_2 + \ldots + \beta_{2M} x_2^M \\ &+ \beta_{1+2M} (x_1 x_2) + \ldots + \beta_{3M} (x_1 x_2)^M \end{aligned}$$

In each case, we consider each term x_j^m and each cross term $x_1 x_2$ an unique predictor and apply linear regression.

Model Selection: Overview

Overfitting: Another Motivation for Model Selection

Finding subsets of significant predictors is an important for model interpretation. But there is another strong reason to model using the smaller set of significant predictors: to avoid overfitting.

Definition

Overfitting is the phenomenon where the model is unnecessarily complex, in the sense that portions of the model captures the random noise in the observation, rather than the relationship between predictor(s) and response.

Overfitting causes the model to lose predictive power on new data.

An Example

Causes of Overfitting

As we saw, overfitting can happen when

- there are too many predictors:
  - the feature space has high dimensionality
  - the polynomial degree is too high
  - too many cross terms are considered
- the coefficients values are too extreme

A sign of overfitting may be a high training R^2 or low MSE and unexpectedly poor testing performance.

Note: There is no 100% accurate test for overfitting and there is not a 100% effective way to prevent it. Rather, we may use multiple techniques in combination to prevent overfitting and various methods to detect it.

Model Selection

Model selection is the application of a principled method to determine the complexity of the model, e.g. choosing a subset of predictors, choosing the degree of the polynomial model etc.

Model selection typically consists of the following steps:

1. split the training set into two subsets: training and ***validation***
2. multiple models (e.g. polynomial models with different degrees) are fitted on the training set; each model is evaluated on the validation set
3. the model with the best validation performance is selected
4. the selected model is evaluated one last time on the testing set

Stepwise Variable Selection

Exhaustive Selection

To find the optimal subset of predictors for modeling a response variable, we can

- compute all possible subsets of $\{X_1, \ldots, X_J\}$,
- evaluate all the models constructed from the subsets of $\{X_1, \ldots, X_J\}$,
- find the model that optimizes some metric.

While straightforward, ***exhaustive selection*** is computationally infeasible, since $\{X_1, \ldots, X_J\}$ has 2^J number of possible subsets.

Instead, we will consider methods that iteratively build the optimal set of predictors.

Variable Selection: Forward

In ***forward selection***, we find an 'optimal' set of predictors by iterative building up our set.

1. Start with the empty set P_0, construct the null model M_0.
2. For $k = 1, \ldots, J$:
   - 2.1 Let M_{k-1} be the model constructed from the best set of $k-1$ predictors, P_{k-1}.
   - 2.2 Select the predictor X_{n_k}, not in P_{k-1}, so that the model constructed from $P_k = X_{n_k} \cup P_{k-1}$ optimizes a fixed metric (this can be p-value, F-stat; validation MSE, R^2; or AIC/BIC on training set).
   - 2.3 Let M_k denote the model constructed from the optimal P_k.
3. Select the model M amongst $\{M_0, M_1, \ldots, M_J\}$ that optimizes a fixed metric (this can be validation MSE, R^2; or AIC/BIC on training set).
4. Evaluate the final model M on the testing set.

Variable Selection: Backward

In ***backward selection***, we find an 'optimal' set of predictors by iterative eliminating predictors.

1. Start with all the predictors P_J, construct the full model M_J.
2. For $k = 1, \ldots, J$:
   - 2.1 Let M_k be the model constructed from the best set of $k-1$ predictors, P_k.
   - 2.2 Select the predictor X_{n_k} in P_k so that the model constructed from $P_{k-1} = P_{k-1} - \{X_{n_k}\}$ optimizes a fixed metric (this can be p-value, F-stat; validation MSE, R^2; or AIC/BIC on training set).
   - 2.3 Let M_{k-1} denote the model constructed from the optimal P_{k-1}.
3. Select the model M amongst $\{M_0, M_1, \ldots, M_J\}$ that optimizes a fixed metric (this can be validation MSE, R^2; or AIC/BIC on training set).
4. Evaluate the final model M on the testing set.

An Example

Cross Validation

Cross Validation: Motivation

Using a single validation set to select amongst multiple models can be problematic - there is the possibility of overfitting to the validation set.

One solution to the problems raised by using a single validation set is to evaluate each model multiple validation sets and average the validation performance.

One can randomly split the training set into training and validation multiple times, but randomly creating these sets can create the scenario where important features of the data never appear in our random draws.

Leave-One-Out

Given a data set $\{\mathbf{X}_1, \ldots, \mathbf{X}_n\}$, where each $\mathbf{X}_i = (x_{i,1}, \ldots, x_{i,J})$ contains J number of features.

To ensure that every observation in the dataset is included in at least one training set and at least one validation set, we create training/validation splits using the ***leave one out*** method:

- validation set: $\{\mathbf{X}_i\}$
- training set: $\mathbf{X}_{-i} := \{\mathbf{X}_1, \ldots, \mathbf{X}_{i-1}, \mathbf{X}_{i+1}, \ldots, \mathbf{X}_n\}$

for $i = 1, \ldots, n$. We fit the model on each training set, denoted $\widehat{f}_{\mathbf{X}_{-i}}$, and evaluate it on the corresponding validation set, $\widehat{f}_{\mathbf{X}_{-i}}(\mathbf{X}_i)$. The ***cross validation score*** is the performance of the model averaged across all validation sets:

$$CV(\text{Model}) = \frac{1}{n} \sum_{i=1}^{n} L\left(\widehat{f}_{\mathbf{X}_{-i}}(\mathbf{X}_i)\right),$$

where L is a loss function.

K-Fold Cross Validation

Rather than creating n number of training/validation splits, each time leaving one data point for the validation set, we can include more data in the validation set using **K-fold validation**:

- split the data into K uniformly sized chunks, $\{C_1, \ldots, C_K\}$
- we create K number of training/validation splits, using one of the K chunks for validation and the rest for training.

We fit the model on each training set, denoted $\widehat{f}_{C_{-i}}$, and evaluate it on the corresponding validation set, $\widehat{f}_{C_{-i}}(C_i)$. The ***cross validation score*** is the performance of the model averaged across all validation sets:

$$CV(\text{Model}) = \frac{1}{n} \sum_{i=1}^{K} L\left(\widehat{f}_{C_{-i}}(C_i)\right),$$

where L is a loss function.

Applications of Model Selection

Predictor Selection: Cross Validation

Rather than choosing a subset of significant predictors using stepwise selection, we can use K-fold cross validation:

- create a collection of different subsets of the predictors
- for each subset of predictors, compute the cross validation score for the model created using only that subset
- select the subset (and the corresponding model) with the best cross validation score
- evaluate the model one last time on the test set

Degree Selection: Stepwise

We can frame the problem of degree selection for polynomial models as a predictor selection problem: which of the predictors $\{x, x^2, \ldots, x^M\}$ should we select for modeling?

We can apply stepwise selection to determine the optimal subset of predictors.

Degree Selection: Cross Validation

We can also select the degree of a polynomial model using K-fold cross validation.

- consider a number of different degrees
- for each degree, compute the cross validation score for a polynomial model of that degree
- select the degree, and the corresponding model, with the best cross validation score
- evaluate the model one last time on the test set

kNN Revisited

Recall our first simple, intuitive, non-parametric model for regression - the kNN model. We saw that it is vitally important to select an appropriate k for the data.

If the k is too small then the model is very sensitive to noise (since a new prediction is based on very few observed neighbors), and if the k is too large, the model tends towards making constant predictions.

A principled way to choose k is through K-fold cross validation.

A Simple Example

Bibliography

1. Bolelli, L., Ertekin, S., and Giles, C. L. ***Topic and trend detection in text collections using latent dirichlet allocation***. In European Conference on Information Retrieval (2009), Springer, pp. 776-780.
2. Chen, W., Wang, Y., and Yang, S. ***Efficient influence maximization in social networks. In Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining (2009)***, ACM, pp. 199-208.
3. Chong, W., Blei, D., and Li, F.-F. ***Simultaneous image classification and annotation. In Computer Vision and Pattern Recognition***, 2009. CVPR 2009. IEEE Conference on (2009), IEEE, pp. 1903-1910.
4. Du, L., Ren, L., Carin, L., and Dunson, D. B. ***A bayesian model for simultaneous image clustering, annotation and object segmentation***. In Advances in neural information processing systems (2009), pp. 486-494.
5. Elango, P. K., and Jayaraman, K. ***Clustering images using the latent dirichlet allocation model.***
6. Feng, Y., and Lapata, M. ***Topic models for image annotation and text illustration***. In Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics (2010), Association for Computational Linguistics, pp. 831-839.
7. Hannah, L. A., and Wallach, H. M. ***Summarizing topics: From word lists to phrases.***
8. Lu, R., and Yang, Q. ***Trend analysis of news topics on twitter***. International Journal of Machine Learning and Computing 2, 3 (2012), 327.