Lecture #5: Multiple Linear Regression

Data Science 1

CS 109A, STAT 121A, AC 209A, E-109A

Pavlos Protopapas Kevin Rader

Margo Levine Rahul Dave

Lecture Outline

Review

More on Model Evaluation

Multiple Linear Regression

Evaluating Significance of Predictors

Comparison of Two Models

Multiple Regression with Interaction Terms

Polynomial Regression

Review

Statistical Models

We will assume that the response variable, Y, relates to the predictors, X, through some unknown function expressed generally as:

$$Y = f(X) + \epsilon,$$

where ϵ is a random variable representing measurement noise.

A ***statistical model*** is any algorithm that estimates the function f. We denote the estimated function as $\widehat{f}$ and the predicted value of Y given $X = x_i$ as $\widehat{y}_i$.

When performing ***inference***, we compute parameters of $\widehat{f}$ that minimizes the error of our model, where error is measured by a choice of ***loss function***.

Simple Linear Regression

A ***simple linear regression model*** assumes that our statistical model is

$$Y = f(X) + \epsilon = \beta_1^{\text{true}} X + \beta_0^{\text{true}} + \epsilon,$$

then it follows that $\widehat{f}$ must look like

$$\widehat{f}(X) = \widehat{\beta}_1 X + \widehat{\beta}_0.$$

When fitting our model, we find $\widehat{\beta}_0, \widehat{\beta}_1$ to minimize the loss function, for example,

$$\widehat{\beta}_0, \widehat{\beta}_1 = \underset{\beta_0, \beta_1}{\text{argmin}} L(\beta_0, \beta_1).$$

The line $\widehat{Y} = \widehat{\beta}_1 X + \widehat{\beta}_0$ is called the ***regression line***.

Loss Functions Revisited

Recall that there are multiple ways to measure the fitness of a model, i.e. there are multiple ***loss functions***.

1. (**Max absolute deviation**) Count only the biggest ‘error’

$$\max_i |y_i - \widehat{y}_i|$$

2. (**Sum of absolute deviations**) Add up the ‘errors’

$$\sum_i |y_i - \widehat{y}_i| \quad \text{or} \quad \frac{1}{n}\sum_i |y_i - \widehat{y}_i|$$

3. (**Sum of squared errors**) Add up the squared ‘errors’

$$\sum_i |y_i - \widehat{y}_i|^2 \quad \text{or} \quad \frac{1}{n}\sum_i |y_i - \widehat{y}_i|^2$$

The average squared error is the **Mean Squared Error**.

More on Model Evaluation

Evaluating Model Things to Consider

- How well do we know $\hat{f}$?

  How well do we know $\hat{\beta}_0$ and $\hat{\beta}_1$?

- Model Fitness.

  How well can the model predict?

- Evaluating Significance of Predictors.

  Does the outcome depend on the predictors?

- Comparison of Two Models.

  Which model is *better*?

Understanding Model Uncertainty

We interpret the ϵ term in our model

$$Y = f(X) + \epsilon$$

to be noise introduced by random variations in natural systems or imprecisions of our scientific instruments.

We call ϵ the measurement error or ***irreducible error***. Since even predictions made with the actual function f will not match observed values of Y.

Due to ϵ, every time we measure the response Y for a fix value of X we will obtain a different observation, and hence a different estimate of β_0 and β_1.

Uncertainty In $\hat{\beta}_0$ and $\hat{\beta}_1$

Again due to ϵ, if we make only a few observations, the noise in the observed values of Y will have a large impact on our estimate of β_0 and β_1.

If we make many observations, the noise in the observed values of Y will ‘cancel out’; noise that biases some observations towards higher values will be canceled by the noise that biases other observations towards lower values.

This feels intuitively true but requires some assumptions on ϵ and a formal justification - or at least an example.

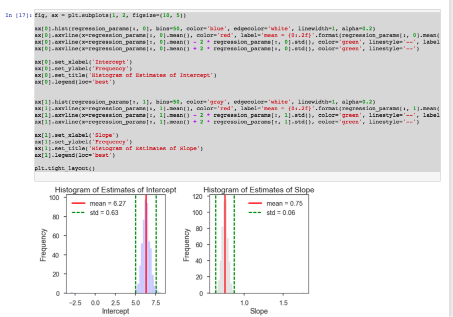
Uncertainty In $\hat{\beta}_0$ and $\hat{\beta}_1$

In summary, the variations in $\widehat{\beta}_0$, $\widehat{\beta}_1$ (estimates of β_0 and β_1 respectively) are affected by

- (**Measurement**) Var$[\epsilon]$, the variance (the scale of the variation) in the noise, ϵ
- (**Sampling**) n, the number of observations we make

The square root of the variances of $\widehat{\beta}_0$, $\widehat{\beta}_1$ are also called ***standard errors***, which we will see later.

A Simple Example



```
In [17]: fig, ax = plt.subplots(1, 2, figsize=(10, 5))

         ax[0].hist(regression_params[:, 0], bins=50, color='blue', edgecolor='white', linewidth=1, alpha=0.2)
         ax[0].axvline(x=regression_params[:, 0].mean(), color='red', label='mean = {0:.2f}'.format(regression_params[:, 0].mean(
         ax[0].axvline(x=regression_params[:, 0].mean() - 2 * regression_params[:, 0].std(), color='green', linestyle='--', label
         ax[0].axvline(x=regression_params[:, 0].mean() + 2 * regression_params[:, 0].std(), color='green', linestyle='--')

         ax[0].set_xlabel('Intercept')
         ax[0].set_ylabel('Frequency')
         ax[0].set_title('Histogram of Estimates of Intercept')
         ax[0].legend(loc='best')


         ax[1].hist(regression_params[:, 1], bins=50, color='gray', edgecolor='white', linewidth=1, alpha=0.2)
         ax[1].axvline(x=regression_params[:, 1].mean(), color='red', label='mean = {0:.2f}'.format(regression_params[:, 1].mean(
         ax[1].axvline(x=regression_params[:, 1].mean() - 2 * regression_params[:, 1].std(), color='green', linestyle='--', label
         ax[1].axvline(x=regression_params[:, 1].mean() + 2 * regression_params[:, 1].std(), color='green', linestyle='--')

         ax[1].set_xlabel('Slope')
         ax[1].set_ylabel('Frequency')
         ax[1].set_title('Histogram of Estimates of Slope')
         ax[1].legend(loc='best')

         plt.tight_layout()
```

Bootstrapping for Estimating Sampling Error

With some assumptions on ϵ, we can compute the variances or standard errors of $\widehat{\beta}_0$ and $\widehat{\beta}_1$ analytically.

The standard errors can also be estimated empirically through ***bootstrapping***.

Definition

Bootstrapping is the practice of estimating properties of an estimator by measuring those properties by, for example, sampling from the observed data.

For example, we can compute $\widehat{\beta}_0$ and $\widehat{\beta}_1$ multiple times by randomly sampling from our data set. We then use the variance of our multiple estimates to approximate the true variance of $\widehat{\beta}_0$ and $\widehat{\beta}_1$.

Model Evaluation: Standard Errors

Recall that our estimates $\widehat{\beta}_0, \widehat{\beta}_1$ will vary depending on the observed data. Thus, the variance of $\widehat{\beta}_0, \widehat{\beta}_1$ indicates the extend to which we can rely on any given estimate of these parameters.

The square root of variance of $\widehat{\beta}_0, \widehat{\beta}_1$ are also called their ***standard errors***.

Model Evaluation: Standard Errors

If our data is drawn from a larger set of observations then we can empirically estimate the standard errors of $\widehat{\beta}_0, \widehat{\beta}_1$ through sampling (bootstrapping).

If we know the variance σ^2 of the noise ϵ, we can compute $SE\left(\widehat{\beta}_0\right), SE\left(\widehat{\beta}_1\right)$ analytically, using the following formulae:

$$SE\left(\widehat{\beta}_0\right) = \sigma\sqrt{\frac{1}{n} + \frac{\overline{x}^2}{\sum_i \left(x_i - \overline{x}\right)^2}}$$

$$SE\left(\widehat{\beta}_1\right) = \frac{\sigma}{\sqrt{\sum_i \left(x_i - \overline{x}\right)^2}}$$

Model Evaluation: Standard Errors

In practice, we do not know the theoretical value of σ^2, since we do not know the exact distribution of the noise ϵ. However, if we make the following assumptions,

- the errors $\epsilon_i = y_i - \widehat{y}_i$ and $\epsilon_j = y_j - \widehat{y}_j$ are uncorrelated, for $i \neq j$,
- each ϵ_i is normally distributed with mean 0 and variance σ^2,

then, we can empirically estimate σ^2 from the data and our regression line:

$$\sigma \approx \sqrt{\frac{n \cdot MSE}{n-2}} = \sqrt{\frac{\sum_i (y_i - \widehat{y}_i)^2}{n-2}}.$$

Model Evaluation: Confidence Intervals

Definition

A c% ***confidence interval*** of an estimate $\widehat{X}$ is the range of values such that the true value of X is contained in this interval with c percent probability.

For linear regression, the 95% confidence interval for $\widehat{\beta}_0, \widehat{\beta}_1$ can be approximated using their standard errors:

$$CI_{95\%}(\beta_k) = \widehat{\beta}_k \pm 2SE\left(\widehat{\beta}_k\right)$$

for $k = 0, 1$. Thus, with approximately 95% probability, the true value of β_k is contained in the interval $\left[\widehat{\beta}_k - 2SE\left(\widehat{\beta}_k\right), \widehat{\beta}_k + 2SE\left(\widehat{\beta}_k\right)\right]$.

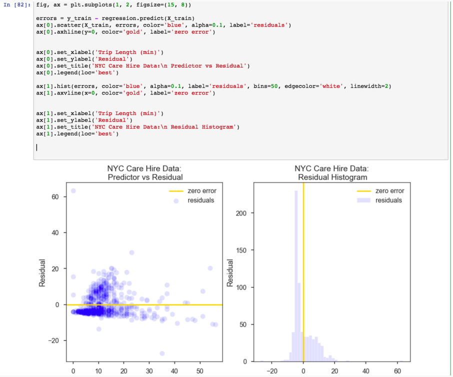
Model Evaluation: Residual Analysis

When we estimated the variance of ϵ, we assumed that the residuals $\epsilon_i = y_i - \widehat{y}_i$ were uncorrelated and normally distributed with mean 0 and fixed variance.

These assumptions need to be verified using the data. In residual analysis, we typically create two types of plots:

1. a plot of ϵ_i with respect to x_i. This allows us to compare the distribution of the noise at different values of x_i.
2. a histogram of ϵ_i. This allows us to explore the distribution of the noise independent of x_i.

A Simple Example

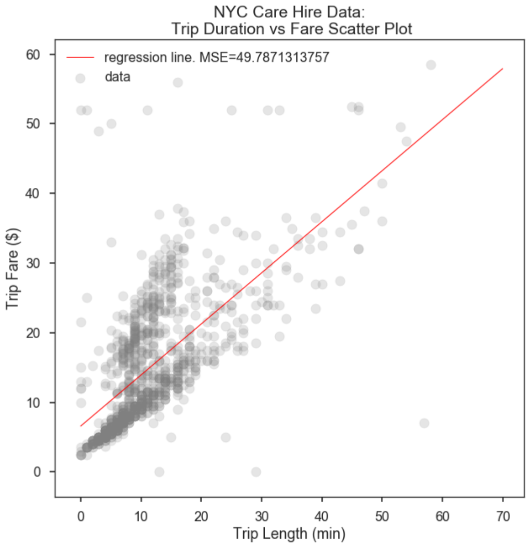


```
In [82]: fig, ax = plt.subplots(1, 2, figsize=(15, 8))

         errors = y_train - regression.predict(X_train)
         ax[0].scatter(X_train, errors, color='blue', alpha=0.1, label='residuals')
         ax[0].axhline(y=0, color='gold', label='zero error')


         ax[0].set_xlabel('Trip Length (min)')
         ax[0].set_ylabel('Residual')
         ax[0].set_title('NYC Care Hire Data:\n Predictor vs Residual')
         ax[0].legend(loc='best')

         ax[1].hist(errors, color='blue', alpha=0.1, label='residuals', bins=50, edgecolor='white', linewidth=2)
         ax[1].axvline(x=0, color='gold', label='zero error')


         ax[1].set_xlabel('Trip Length (min)')
         ax[1].set_ylabel('Residual')
         ax[1].set_title('NYC Care Hire Data:\n Residual Histogram')
         ax[1].legend(loc='best')
```

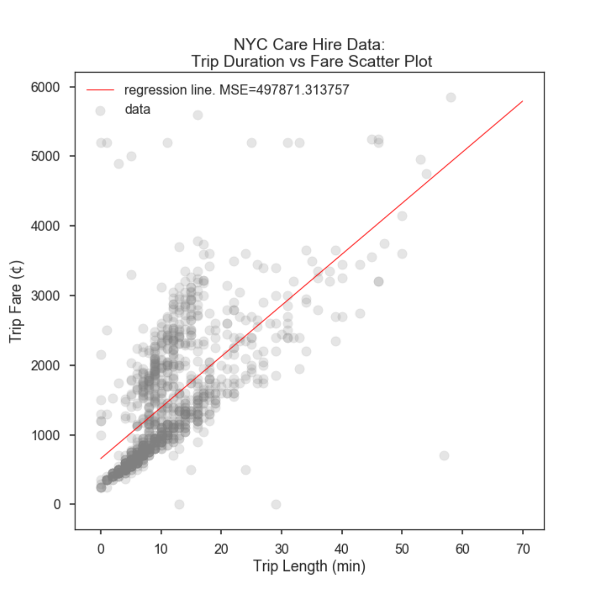
Evaluating Model Things to Consider

- How well do we know $\hat{f}$?
  How well do we know $\hat{\beta}_0$ and $\hat{\beta}_1$?
- **Model Fitness.**
  How well can the model predict?
- Evaluating Significance of Predictors.
  Does the outcome depend on the predictors?
- Comparison of Two Models.
  Which model is *better*?

Model Fitness: R^2



Model Fitness: R^2



Model Fitness: R^2

While loss functions measure the predictive errors made by a model, we are also interested in the ability of our models to capture interesting features or variations in the data.

We compute the ***explained variance*** or R^2, the ratio of the variation of the model and the variation in the data. The explained variance of a regression line is given by

$$R^2 = 1 - \frac{\sum_{i=1}^{n} \left|y_i - \hat{y}_i\right|^2}{\sum_{i=1}^{n} \left|y_i - \overline{y}\right|^2}$$

For a regression line, we have that

$$0 \leq R^2 \leq 1$$

Can you see why?

Model Fitness: Training vs. Testing Sets

One more way to evaluate our model is to use it to predict the responses for predictors that we did not use to build our model.

Typically, after collecting a set of observations of predictor and response, we split the data into a ***training set*** and a ***testing set***.

We use the training set to build a model and use the testing set to perform a final evaluation of the model, simulating model performance in real-time usage.

Caution: In order to maintain the integrity of the final test, you should use your test data once and must not use the results to inform changes you make to the model.

A Simple Example

Evaluating Model Things to Consider

- How well do we know $\hat{f}$?

  How well do we know $\hat{\beta}_0$ and $\hat{\beta}_1$?

- Model Fitness.

  How well can the model predict?

- **Evaluating Significance of Predictors.**

  Does the outcome depend on the predictors?

- Comparison of Two Models.

  Which model is *better*?

Multiple Linear Regression

Multilinear Models

In practice, it is unlikely that any response variable Y depends solely on one predictor x. Rather, we expect that Y is a function of multiple predictors $f(X_1, \dots, X_J)$.

In this case, we can still assume a simple form for f - a multilinear form:

$$y = f(X_1, \dots, X_J) + \epsilon = \beta_0 + \beta_1 x_1 + \ldots + \beta_J x_J + \epsilon.$$

Hence, $\widehat{f}$ has the form

$$\widehat{y} = \widehat{f}(X_1, \dots, X_J) = \widehat{\beta}_0 + \widehat{\beta}_1 x_1 + \ldots + \widehat{\beta}_J x_J.$$

Again, to fit this model means to compute $\widehat{\beta}_0, \dots, \widehat{\beta}_J$ to minimize a loss function; we will again choose the MSE as our loss function.

Multiple Linear Regression

Given a set of observations

$$\{(x_{1,1},\ldots,x_{1,J},y_1),\ldots(x_{n,1},\ldots,x_{n,J},y_n)\},$$

the data and the model can be expressed in vector notation,

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

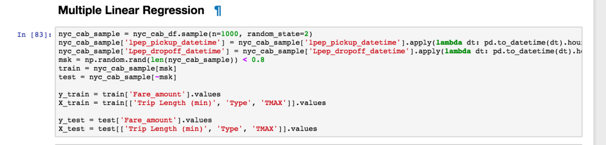
Thus, the MSE can be expressed in vector notation as

$$\text{MSE}(\boldsymbol{\beta}) = \frac{1}{n}\|\boldsymbol{Y} - \boldsymbol{X}\boldsymbol{\beta}\|^2.$$

Minimizing the MSE using vector calculus yields,

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

A Simple Example



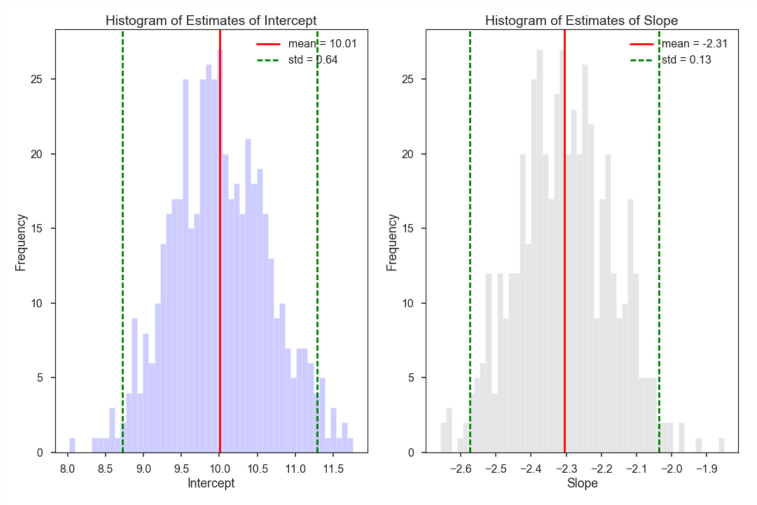
Multiple Linear Regression ¶

In [83]:

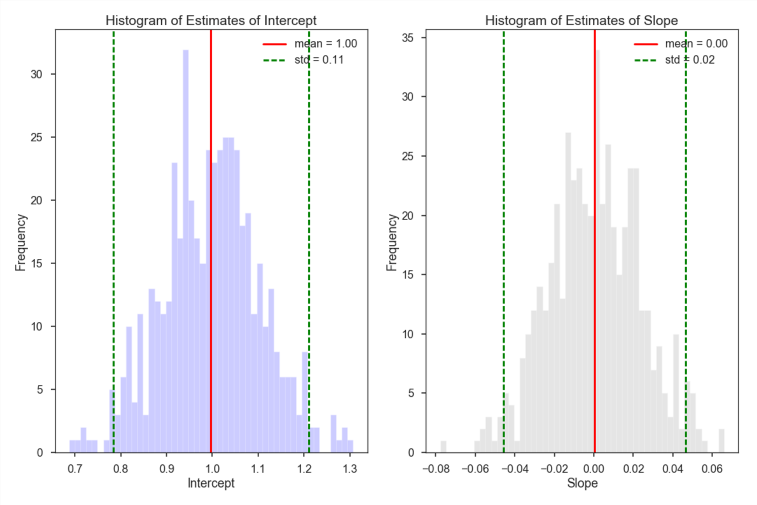
```
nyc_cab_sample = nyc_cab_df.sample(n=1000, random_state=2)
nyc_cab_sample['lpep_pickup_datetime'] = nyc_cab_sample['lpep_pickup_datetime'].apply(lambda dt: pd.to_datetime(dt).hou
nyc_cab_sample['Lpep_dropoff_datetime'] = nyc_cab_sample['Lpep_dropoff_datetime'].apply(lambda dt: pd.to_datetime(dt).h
msk = np.random.rand(len(nyc_cab_sample)) < 0.8
train = nyc_cab_sample[msk]
test = nyc_cab_sample[~msk]

y_train = train['Fare_amount'].values
X_train = train[['Trip Length (min)', 'Type', 'TMAX']].values

y_test = test['Fare_amount'].values
X_test = test[['Trip Length (min)', 'Type', 'TMAX']].values
```

Evaluating Significance of Predictors

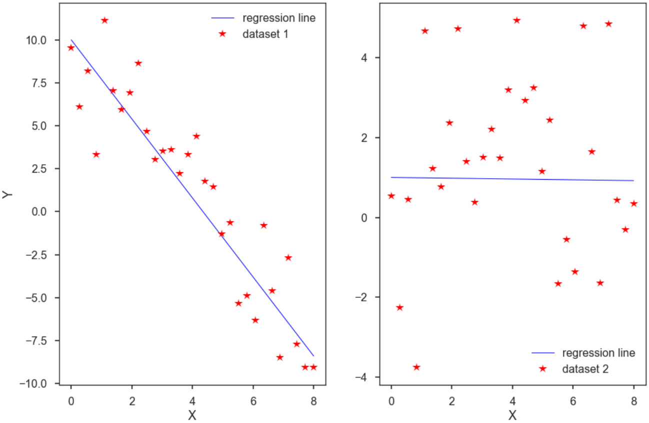
Finding Significant Predictors



Finding Significant Predictors



Finding Significant Predictors



Finding Significant Predictors: Hypothesis Testing

With multiple predictors, an obvious analysis is to check which predictor or group of predictors has a ‘significant’ impact on the response variable.

One way to do this is to analyze the ‘likelihood’ that any one or any set of regression coefficient is zero. Significant predictors will have coefficients that are deemed less ‘likely’ to be zero.

Unfortunately, since the regression coefficients vary depending on the data, we cannot simply pick out non-zero coefficients from our estimate β.

Finding Significant Predictors: Hypothesis Testing

Hypothesis Testing

Hypothesis testing is a formal process through which we evaluate the validity of a statistical hypothesis by considering evidence for or against the hypothesis gathered by random sampling of the data.

1. State the hypotheses, typically a ***null hypothesis***, H_0, and an ***alternative hypothesis***, H_1, that is the negation of the former.
2. Choose a type of analysis, i.e. use sample data to evaluate the null hypothesis. Typically this involves choosing a single test statistic.
3. Sample data and compute the test statistic.
4. Use the value of the test statistic to either reject or not reject the null hypothesis.

Finding Significant Predictors: Hypothesis Testing

For checking the significance of linear regression coefficients:

1. We set up our hypotheses

$$H_0 : \beta_0 = \beta_1 = \ldots = \beta_J = 0 \qquad \textbf{(Null)}$$
$$H_1 : \beta_j \neq 0, \text{ for at least one } j \qquad \textbf{(Alternative)}$$

2. we choose the F-stat to evaluate the null hypothesis,

$$F = \frac{\text{explained variance}}{\text{unexplained variance}}$$

3. we can compute the F-stat for linear regression models by

$$F = \frac{(\text{TSS} - \text{RSS})/J}{\text{RSS}/(n - J - 1)}, \quad \text{TSS} = \sum_i (y_i - \overline{y})^2, \quad \text{RSS} = \sum_i (y_i - \widehat{y}_i)^2$$

4. If $F = 1$ we consider this evidence for H_0; if $F > 1$, we consider this evidence against H_0.

More on Hypothesis Testing

Applying the F-stat test to $\{X_1, \ldots, X_J\}$ determines if any of the predictors have a significant relationship with the response.

We can also apply the test to a subset of predictors to determine if a smaller group of predictors have a significant relationship with the response.

Note: There is not a fixed threshold for rejecting the null hypothesis based on the F-stat.

For n and J that are large, F values that are slightly above 1 are considered to be strong evidence against H_0.

More on Hypothesis Testing

To determine if any single predictor has a significant relationship with the response, we can again perform hypothesis testing. In this case, the test statistics we use is typically the p-value.

Definition

The p***-value*** is the probability that, when the null hypothesis is true, the statistical summary of a given model would be the same as or more extreme than the observed results.

Smaller p-values are interpreted to be evidence against the null hypothesis. A standard p-value threshold for rejecting the null hypothesis is 0.05 (or 5%).

Finding Significant Predictors: R^2

We can compare the ‘significance’ of two specific groups of predictors $\{X_{j_1}, \ldots, X_{j_k}\}$ and $\{X_{j'_1}, \ldots, X_{j'_{k'}}\}$, by comparing the R^2 values of the two models constructed using each set

$$R^2\left(\widehat{f}(X_{j_1}, \ldots, X_{j_k})\right) \quad \text{v.s.} \quad R^2\left(\widehat{f}(X_{j'_1}, \ldots, X_{j'_{k'}})\right)$$

We may conclude that a higher R^2 (i.e. a model that fits the observation better) is evidence that one set of predictors impacts the response more significantly than the other.

Question: Do you think this is a valid statement?

Finding Significant Predictors: Information Criteria

Yet another way to evaluate the explanatory power of different sets of predictors is to use ***information criteria***. These are a set of metrics that measures the fit of the model to observations given the number of parameters used in the model.

Below are two different such criteria, ***Akaike's Information Criterion*** and ***Bayes Information Criterion***

$$\text{AIC} \approx n \cdot \ln(\text{MSE}) + 2J$$
$$\text{BIC} \approx n \cdot \ln(\text{MSE}) + J \cdot \ln(n)$$

where J is the number of parameters.

From the above, we can see that the smaller the AIC or BIC, the better the model.

Evaluating Model Things to Consider

- How well do we know $\hat{f}$?
  How well do we know $\hat{\beta}_0$ and $\hat{\beta}_1$?
- Model Fitness.
  How well can the model predict?
- Evaluating Significance of Predictors.
  Does the outcome depend on the predictors?
- **Comparison of Two Models.**
  Which model is *better*?

Comparison of Two Models

Finding Significant Predictors: Information Criteria

We can compare the ‘significance’ of two specific groups of predictors $\{X_{j_1}, \ldots, X_{j_k}\}$ and $\{X_{j'_1}, \ldots, X_{j'_{k'}}\}$, by comparing the AIC or BIC values of the two models constructed using each set

$$\text{AIC/BIC}\left(\widehat{f}(X_{j_1}, \ldots, X_{j_k})\right) \text{ v.s. AIC/BIC}\left(\widehat{f}(X_{j'_1}, \ldots, X_{j'_{k'}})\right)$$

We may conclude that a lower AIC or BIC (i.e. a model that fits the observation better) is evidence that one set of predictors impacts the response more significantly than the other.

Parametric vs. Non-parametric Models

Linear regression is an example of ***a parametric model***, that is, it is a model with a fixed form and a fixed number of parameters that does not depend on the number of observations in the training set.

kNN is an example of a ***non-parametric model***, that is, it is a model whose structure depends on the data and no assumptions are made about the underlying probability distributions. The set of parameters of the kNN model is the entire training set.

In particular, the number of parameters in kNN depends on the number of observations in the training set.

kNN vs. Linear Regression

So which model is better? Rather than answer this question, let's define 'better'.

To compare two models, we can consider any combination of the following criteria (and possibly more):

- Which model gives more predictive error, with respect to a loss function?
- Which model takes less space to store?
- Which model takes less time to train (perform inference)?
- Which model takes less time to make a prediction?

Which Metric of Significance Should We Use?

The procedure of systematically choosing a set of predictors that have a significant relationship with the response variable is called ***variable selection*** (or feature selection).

But which metric (F-stats, p-values, R^2, AIC/BIC) should we use to determine the significance of a set of predictors?

In later lectures, we will see that each metric has its strengths and draw-backs. Rather than relying on a single metric, we should use multiple metrics in conjunction and double check with common sense!

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, 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_1x_2) as additional predictors.

Given a set of observations $\{(x_{1,1}, x_{1,2}, y_1), \ldots (x_{n,1}, x_{n,2}, y_n)\}$, 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,

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

Polynomial Regression

Polynomial Regression as Linear Regression

The simplest non-linear model we can consider, for a response Y and a predictor X, is a polynomial model of degree M,

$$y = \beta_0 + \beta_1 x + \beta_2 x^2 + \ldots + \beta_M x^M + \epsilon.$$

Just as in the case of linear regression with cross terms, polynomial regression is a special case of linear regression - we treat each x^m as a separate predictor. Thus, we can write

$$\mathbf{Y} = \begin{pmatrix} y_1 \\ \vdots \\ y_n \end{pmatrix}, \quad \mathbf{X} = \begin{pmatrix} 1 & x_1^1 & \ldots & x_1^M \\ 1 & x_2^1 & \ldots & x_2^M \\ \vdots & \vdots & \ddots & \vdots \\ 1 & x_n & \ldots & x_n^M \end{pmatrix}, \quad \boldsymbol{\beta} = \begin{pmatrix} \beta_0 \\ \beta_1 \\ \vdots \\ \beta_M \end{pmatrix}.$$

Again, minimizing the MSE using vector calculus yields,

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

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.

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.