An Introduction to Predictive Modelling

UBCO MDS — DATA 570

Course Description

Introduction to regression for Data Science. Simple linear regression, multiple linear regression, interactions, mixed variable types, model assessment, simple variable selection, k-nearest-neighbours regression. Credit will be granted for only one of DATA 311 or DATA 570. Restricted to students in the MDS program.

Course Schedule

Lecture: Mon/Wed 9:30 AM to 11:00 AM in EME-1153
Lab: Thu 1:30 PM to 3:30 PM in EME-1153

Office Hours: Tue/Thu 4:00-4:50? SCI-108

Marking and Evaluation

This module will be structured as follows:

- 7 lectures
- 4 labs
- 2 quizzes

Item	**Weighting**
Assignments	40%
Quizes	60%

Textbook

ISLR We will be following the Springer textbook: An Introduction to Statistical Learning *with Applications in R* by Gareth James, Daniela Witten, Trevor Hastie and Robert Tibshirani.

ESL A more advanced treatment of these topics can be found in the The Elements of Statistical Learning: *Data Mining, Inference, and Prediction*

What is Predictive Modeling

What is the difference between Predictive Modelling, Statistical Learning, Machine Learning, etc.?

Statistical learning refers to a set of tools for modeling and understanding complex datasets.

- Blends statistics with computer science (in particular machine learning)
- Involves forming a hypothesis before building a model
- Examples: regression, classification, support vector machines

What is Predictive Modeling

Machine learning uses statistical tools that have the ability to "learn" from data without being explicitly programmed.

- Driven by data more-so than our hypothesis
- Examples: supervised learning, unsupervised learning, clustering,

What is Predictive Modeling

Predictive Modeling is a mathematical technique which uses statistics for forecasting (i.e. predicting) future outcomes.

- An advanced form of descriptive analytics on current data to provide and outcome.
- Examples: Decision trees, regression (linear and logistic), neural networks

What is Predictive Modeling

- At its core, all of predictive methods involve three elements:
  1. Data
  2. Statistical Model
  3. Assumptions

Motivating Examples

- The following data sets were taken from the ISLR supported files.
- They will serve as motivating examples throughout this course.
- The easiest way to access them in through the ISLR pacakge available for R (we will demonstrate how to do this in the first lab in case you don't know already)
- I will use `typewritter text` to denote objects that can be referenced in R.

Wage

- The `Wage` data set comprise the wages from 3000 males from the Atlantic regions of the United States.
- There are 11 variables for this data set.
  - `year, age, maritl, race, education, region, jobclass, health, health_ins, logwage, wage`
- We might use this data to understand the relationship between an employee's `age` and `wage`, for example.
- We might also/instead use this data to predict the wage of an employee based on `age` and `eduction`, for example.
- We will refer to `wage` as our *quantitative* output value.

Stock Market Data

- The `Smarket` (stock market) data set comprise 1250 observations of the daily percentage returns for the Standard & Poor's 500 (S&P) stock index between 2001 and 2005.
- There are 11 variables for this data set.
  - `Year, Lag1, Lag2, Lag3, Lag4, Lag5, Volume, Today, Direction`
- We might also/instead use this data to predict whether the index will increase or decrease on a given day using information from past 5 days.
- Notice that our output variable is *categorical* or *qualitative*.

Gene Expression Data

- The `Khan` data set comprise 63 tissue samples from cell tumours.
- There are 2308 gene expression measurements (i.e. variables) for this data set
- In this example, there is no natural *output* variables.
- We might be interested in finding similar groups within this data.

Notation

Notation is not standard across different disciplines/courses/textbooks.

We adopt the same notational conventions used in ISLR.

- n – the number of distinct data points/observations in our sample
- p – the number of variables available for making preditions

Example: there are n=3000 observations (i.e. male workers in the Mid-Atlantic region) that comprise the `Wage` data set. There are p - 11 variables which include `year`, `age`, `race`, `education`, ...

Matrices, Vectors, and scalars

- Let $\mathbf{X}$ define an $n \times p$ matrix whose (i, j)th element is x_{ij}
- Some may find it helpful to think of $\mathbf{X}$ as a spreadsheet of numbers with n rows and p columns.

$$\mathbf{X} = \begin{pmatrix} x_{11} & x_{12} & \cdots & x_{1p} \\ x_{21} & x_{22} & \cdots & x_{2p} \\ \vdots & \vdots & \ddots & \vdots \\ x_{n1} & x_{n2} & \cdots & x_{np} \end{pmatrix}$$

Matrices, Vectors, and scalars

- We refer to the rows of $\mathbf{X}$ using x_i
- Hence, $\mathbf{X}$ is comprised of the n row vectors $x_1, x_2, \ldots, x_n$ where x_i is a vector of length p.
- Typically, x_i stores the variable measurements for the ith observation
- Vectors are by default represented as columns:

$$x_i = \begin{pmatrix} x_{i1} \\ x_{i2} \\ \vdots \\ x_{i2} \end{pmatrix}$$

Matrices, Vectors, and scalars

- We refer to the columns of $\mathbf{X}$ using $\mathbf{x}_j$
- Hence, $\mathbf{X}$ is comprised of the p column vectors $\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_n$ where $\mathbf{x}_j$ is a vector of length n.

$$\mathbf{x}_j = \begin{pmatrix} x_{1j} \\ x_{2j} \\ \vdots \\ x_{nj} \end{pmatrix}$$

Matrices, Vectors, and scalars

- Using this notation, the matrix $\mathbf{X}$ can be written:

$$\mathbf{X} = (\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_p)$$

or:

$$\mathbf{X} = \begin{pmatrix} x_1^T \\ x_2^T \\ \vdots \\ x_n^T \end{pmatrix}$$

or

$$\mathbf{X} = (x_1, x_2, \ldots, x_n)^T$$

- The ^{T} notation denotes the *transpose* of a matrix or vector

Matrices, Vectors, and scalars

- We often use $\mathbf{X}$ to denote our input variable(s) and $\mathbf{y}$ to denote our output variable.
- For instance, y_i may refer to the `wage` of the ith observation in the `Wage` data set.
- The collection of all n observations form the vector

$$\mathbf{y} = \begin{pmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{pmatrix}$$

Our observed data consists of $\{(x_1, y_1), (x_2, y_2), \ldots, (x_n, y_n)\}$, where each x_i is a vector of length p.

Matrix algebra

- A * B Element-wise multiplication
- A %*% B Matrix multiplication
- t(A) Transpose
- diag(x) Creates diagonal matrix with elements of x in the principal diagonal
- diag(A) Returns a vector containing the elements of the principal diagonal
- diag(k) If k is a scalar, this creates a k x k identity matrix.
- solve(A, b) Returns vector x in the equation b = Ax (i.e., $A^{-1}b$)
- solve(A) Inverse of A where A is a square matrix.
- y¡-eigen(A) y$val are the eigenvalues of A; y$vec are the eigenvectors of A
- cbind(A,B,...) Combine matrices(vectors) horizontally.
- rbind(A,B,...) Combine matrices(vectors) vertically.

Matrices, Vectors, and scalars

Notation	Description
n	number of samples
$\mathbf{y}$	lower case bold for vectors of length n
x	lower case normal font for vectors of length $\neq n$
$\mathbf{X}$	capital bold for matrices
a	lower case normal font for scalars (beginning of alphabet)

It may also be useful to brush up on some (basic) matrix algebra...

Statistical Learning

- Statistical learning is the set of methods by which we pursue the underlying model of a data set.
- We assume that data arises by

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

where

- $X = (X_1, X_2, \ldots, X_p)$ are inputs/predictors/features/variables
- Y is the output/response/dependent variable
- ϵ is the error term (independent of X and with mean 0)

- Statistical learning is concerned with estimating f.
- The function f is used to map the input variables to an output variable.
- For ease of notation in this section, we use X to denote an input variables which we distinguish using subscripts, eg. X_1= year, X_2= age, etc. for the `Wage` example.
- Let's simulate an example in R...

Prediction and Inference

- Our reasons for finding f fall into two primary categories: prediction and inference.

  Prediction With inputs X available, our concern is predicting the output Y.

  Inference We want to understand the relationship between X and Y.

- Often, we will be interested in both, perhaps to varying extents.

Prediction

- Motivation for prediction problems often stem from the situation where X is cheap but Y is "expensive".
- We denote our estimate of f by $\hat{f}$, and $\hat{Y}$ will represents the resulting prediction for Y
- f is might be treated as a *black box*
- We are not particularly concerned with the exact form of $\hat{f}$, instead we provide accurate predictions for Y.

Prediction

- The accuracy of $\hat{Y}$ depends on:
  reducible error error from estimating f with $\hat{f}$
  irreducible error associated with a natural variability in a system
- Reducible error can be minimized by selecting the most appropriate model for the data at hand, e.g. choosing an exponential model over a linear model in our simulation.
- Irreducible error may stem from variables that are useful in predicting $\mathbf{y}$ that we have not measured, e.g. perhaps hours working per week is a significant unmeasured variable for wage.

Inference

- Answers how is Y affected by X?
- Since $\hat{f}$ is used to model this relationship, we no longer want to treat it as a black box.
- Some related question may include:
  - Which input variables (predictors) are associated with output variables (response)?
  - What is the relationship between the the response and each predictor? e.g. positive, negative, linear

Prediction and Inference

- Again, in many cases we are interested in both prediction and inference with our model.
- Therefore, we often have competing interests.
- Complicated models are often (there are caveats here) better at prediction, but also generally harder to understand.

Estimating f

- We will use the data on hand to fit models under varying assumptions, e.g. linear, non-linear.
- The data used to fit the model will often be referred to as training data.
- Sometimes, we will split a data set (randomly) into training and testing data sets, in order to investigate how well the model might predict for future values not used at the fitting stage.
- Goal: find a function $\hat{f}$ such that $Y \approx \hat{f}(X)$ for any observation (X, Y).

Parametric vs Non-parametric

- Parametric methods assume an explicit model for f and use the data to estimate the parameters of f.
  - Linear regression, for example, is a parametric method — the parameters are the intercept and slopes.
- Non-parametric methods do not make an assumption for the form of f.
- Parametric methods generally offer advantages toward inference, non-parametric methods generally offer advantages toward prediction.

Parametric vs Non-parametric

- There are caveats to that last bullet point.
  1. We can assume as complicated a parametric model as we can dream up, and thereby lose our ability to make meaningful inferences.
  2. We can fit a non-parametric model that fits our observed data perfectly, yet is terrible at prediction!
- Give an example for #2...

Parametric

- There are two steps involved in a parametric approach
  1. Make an assumption on the functional form of f, e.g. linear, gaussian.
  2. "Fit" or "train" the model using our training data, i.e. estimating the set of parameters used in f
- In the case of a linear model, this step would estimate the slope β_1 and intercept β_0.
- We will discuss a number of possible ways step 2. is done later in the course.

Parametric

- For practical applications, f is unknown.
- We there run into the danger of picking the wrong f. Remember:
  All models are wrong but some are useful.

  George Edward Pelham Box
- If the model is too far from the true underlying model, then our estimate will be poor.
- We can combat this issue by choosing *flexible* models (these tend to have more parameters), but we run the risk of *overfitting*.

Non-parametric

- Non-parametric approaches do not assume a particular form of f (i.e. distribution/parameter free).
- This method is useful for data that are not real-valued (e.g. ordinal, intervals)
- As non-parametric methods make fewer assumptions, they have the potential to be applied to a wider range of problems.
- They tend to require a large amount of data in order to be useful.

Supervised vs Unsupervised

- The distinction between supervised and unsupervised learning comes down to one thing: presence or absence of Y.
- If you have measurements on the response variable Y, then you can perform supervised learning.
- Unsupervised learning, predicting Y blindly, is (obviously?) very challenging. In most cases, the only form of Y we can look for are categorical variables (groups).

Supervised vs. Unsupervised

- In the supervised learning domain, we fit models using predictor variables X AND response variables Y in an effort to understand the relationship between them (i.e. inference) or for predicting the value of a new observation.
- In the unsupervised learning domain, Y is not measured, and in some contexts, a natural response is not known.
- In this module, we will be concerned with supervised approaches.


THE UNIVERSITY OF BRITISH COLUMBIA