Lecture 01 – Data Mining

AMOD-5440

Kat Clark

Trent University

2026-09-11

Welcome Information

Contact Details

- **Me**: Dr. Kat Clark
- **Email**: katclark@trentu.ca
- **Office**: ENW 332

Digital Tech & Links

- **Blackboard**: official grades, class-wide communications, slides and video links, ~~assignment code submission~~.
- **Crowdmark**: Assignment release and PDF submissions.
- **RStudio**: learning to **do** statistics and data analysis (5 assignments) - preferably your own computer, but https://sage.trentu.ca/ if not

RStudio

The **R** programming language and interface is **the** language of statistics in the 21st century.

- You will be learning (more about how) to do **data analysis** using **R** in this class
- Many ways to use RStudio - recommend you bite the bullet and install it on your own computer if at all possible (instructions on Blackboard). If not, our server is an option.

Crowdmark

Assignments will conducted with Crowdmark.

- You will get an email from Crowdmark when assignments are released.
- PDFs must be uploaded to Crowdmark.
- You will also be submitting your R code on Crowdmark.

Course Overview

Posted Material

- Lectures: First 2 hours of class on Friday afternoons. Slides posted before class.
- Live Coding Labs: Third hour of class. We will code together and the files will be posted after class.

Text

- The textbook we are using is a Springer book, and is considered **the** standard introductory reference for data mining algorithms.
- Thankfully, while you can buy a physical copy for $65-85 on Amazon, there's a completely free PDF available from the authors' website.
- If you want a physical copy, have at it! Otherwise, save your money.

Things Worth Marks

- Assignments: 5 x 10% = 50%
- In-class quizzes: 5 x 2% = 10%
- Midterm: 90 minutes in class on October 23rd. 15%. Covers material from weeks 1-6
- Final Exam: During exam period. 25%. Cumulative

R Assignments (50%, 5 x 10%)

- Due on Thursdays in weeks 3, 5, 7, 10, and 12.
- The R assignments are designed to assess your learning of the material from the previous two weeks.
- Each one will examine some algorithms, and analyze some real data.
- The reports will be written in Quarto, and turned in as a PDF on Crowdmark.
- The Quarto file will be submitted on Crowdmark.

Paired Assignment Quizzes (10%, 5 x 2%)

- During class the day following the assignment
- Will be short ~5-10 minutes.
- Hand-written, closed-book
- Very straightforward
- Tests if you know your own submission
- While each quiz is worth only 2%, it will cap your assignment grade.

Quiz Cap

Quiz Score	Maximum Assignment Report Grade
80–100%	No cap
70–79%	85%
60–69%	75%
Below 60%	65% and subject to instructor review

Midterm (15%)

- The midterm will take place in class on October 23rd
- It is expected to take 90 minutes.
- It will be hand-written.
- Crib sheet allowed (details given later)
- The midterm will cover material from weeks 1-6

Final Exam (25%)

- The final exam will be scheduled by the registrar
- Same rules as the midterm
- The exam is cumulative

How to Get Help

- Read the textbook
- Google is surprisingly helpful for learning R stuff - there's a huge wealth of materials out there for beginners
- Lots of data mining / stat learning blogs out there - **warning** can be hard to wade through if you don't have experience
- Email me
- Schedule an office hour

Generative AI Tools: Overview

- Policies vary by course & instructor
- Examples: ChatGPT, Copilot, Claude, Gemini
- Use to *troubleshoot*, not do your work for you.
- Always ensure work submitted reflects **your own** decisions and understanding

Appropriate and Inappropriate Uses

Appropriate Uses:

- Debugging & syntax correction
- Identifying errors in code
- For example:
  - You keep encountering an error when rendering your document. You try to fix the problem but come up empty.
  - You paste the error code in Copilot and it tells you that you have an extra comma. You fix it.
  - You make sure to write a note in your assignment about which AI you used and how.

Appropriate and Inappropriate Uses

Inappropriate uses:

- Direct file-modifying AI tools (e.g., Copilot, Gemini Code, Claude Code, Posit Assistant)
- Copying/pasting AI-generated solutions directly
- Using AI to complete graded work
- Asking AI to interpret output or suggest what to do next
- “Consulting” AI to make decisions, e.g. for which parameters or model to choose

Violations: Academic Integrity

- Improper AI use.
- Copying code, solutions, or write-ups without attribution
- Uploading course materials to:
  - AI tools (e.g., ChatGPT)
  - Chegg, CourseHero, or similar
- If unsure, **ask your instructor**
- A first offence will result in a **zero** on the relevant assessment. Subsequent offences will result in failure of the course.

Key Takeaways

1. Use AI to **troubleshoot**, not to replace your own work and/or thinking
2. Always **verify** AI information with trusted sources
3. Attribute help from AI & online resources
4. Protect course materials & respect instructor permissions

Data Mining

Statistical Learning

What IS Data Mining?

Data mining, a.k.a. *statistical learning* refers to a set of tools for **understanding data**. These tools can be classified as:

- **supervised**: involves building a statistical model for predicting, or estimating, an output based on one or more inputs.
- **unsupervised**: there are inputs but no supervising output; nevertheless we can learn relationships and structure from such data.

Linear Models

- In previous courses, you may have seen classes of **linear models**, often just called *regression* or *multiple regression*, when the inputs and outputs are numeric.
- You may also have seen **ANOVA**, which is a specific class of linear model where the inputs are categorical.
- Both of these fall under the classification of **supervised statistical learning**.

Some Language

Almost all models we will cover this semester share some common language.

- **input variable**: also called predictors, *independent* variables, features, or just variables
  - typically denoted by X
  - if you have more than one, X_1, X_2, through X_p
- **output variable**: also called the response, or *dependent* variable
  - typically denoted by Y

Mathematical Notation

- We will do our best to keep the mathematical notation to a minimum, but sometimes you'll want to read the textbook, and puzzle out what it is trying to say.
- If we assume that we observe some Y which is quantitative, and have p different predictors, $X_1, X_2, \ldots, X_p$, then what we are interested in is a **relationship** between these.

Some Notation

- Generically, we might write:

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

  where the unknown function f represents the relationship between the X predictors and the response Y.

- We assume there is some error in the **observations**, so that we cannot capture $f(\cdot)$ perfectly: this is ε.

Multiple Linear Regression

In multiple linear regression, the unknown function is just summation:

$$Y = \beta_0 + \beta_1 X_1 + \cdots + \beta_p X_p + \varepsilon,$$

f(x)

such that Y is related to the Xs via a linear combination, the sum of each of the Xs multiplied by a scale factor β_i, for $i = 1, 2, \cdots, p$.

Note: you will never have to do mathematical derivations in your assignments. Each of these Xs corresponds to a vector of observations in R, as does the Y.

Example, R

$$Y = 0.5 + 1.0X_1 + 2.5X_2 - 1.3X_3 + \varepsilon$$

```r
library(dplyr)
dat <- data.frame(x1 = rexp(20),
                  x2 = rexp(20) + 2,
                  x3 = rexp(20) - 1)

dat <- dat |>
  mutate(y = 0.5 + 1.0 * x1 + 2.5 * x2 - 1.3 * x3 + rnorm(20))

mod <- lm(y ~ x1 + x2 + x3, data = dat)

mod |> coefficients()
```

$\varepsilon \sim N(0,1)$

```
(Intercept)          x1          x2          x3
   1.281539    1.244548    2.221301   -1.335892
```

Example, R

- We generated some fake data, built a relationship between X and Y, then asked a **supervised learning** algorithm to estimate the relationship.
- You can see that we **estimated** the coefficients on the previous slide, in the R code provided.
  - $\beta_0 = 0.5$, we estimated 1.282
  - $\beta_1 = 1.0$, we estimated 1.245
  - $\beta_2 = 2.5$, we estimated 2.221
  - $\beta_3 = -1.3$, we estimated -1.336

Estimating

Why Estimate f?

- Recall that our relationship between the X predictors and the response Y is given by:

$$Y = f(X) + \varepsilon.$$

- **Why** would we want to estimate f?
- There are two main reasons: for **prediction**, and for **inference**.

Prediction

When the outcome is known for past observations, we can use those data to build a model and make predictions for new observations.

Example: Use information from previous patients and their surgical outcomes to predict the probability of a favourable outcome for a new patient.

$$\text{Past Data} \longrightarrow \text{Model} \longrightarrow \text{Prediction for New Data}$$

Goal: Accurately predict the response variable Y for future observations.

Inference

When the goal is inference, we are interested in understanding the **relationship** between the predictors X and the response Y, not just making accurate predictions.

Example: Suppose we want to understand how age, smoking status, and exercise habits affect the probability of a favourable surgical outcome. In this case, we care about *which variables matter* and *how they affect the outcome*.

$$X \longrightarrow \text{Model} \longrightarrow Y$$

Goal: Understand and explain the relationship between X and Y, rather than treating the model as a black box.

In many scientific studies, explanation and understanding are more important than prediction.

Prediction

Notation for Prediction

- Let's start with the first goal: prediction.
- We use "hat" notation to refer to predictors and prediction. So, we would say that our prediction of Y is $\hat{Y}$:

$$\hat{Y} = \hat{f}(X),$$

where $\hat{f}$ is our prediction ("estimate") of the unknown function f that links the inputs X to the output Y.

Examples of Settings for Prediction

- blood chemistry & patient risk
- rental price for an apartment with specific characteristics (“market value”)
- grade you should have gotten on your final exam, if you’d had the chance to write it (aegrotat standing)
- potential income based on education, seniority, field of work
- fitness of a species under constraints in the environment

Accuracy

- We're estimating an unknown function, but *estimating* could be rephrased as *educated guessing*.
- Technically speaking, **any** guess is an estimate! And not necessarily a **good** one!

Example: the "stopped clock" estimator. If you ask me what time it is, and I always respond 11 o'clock, I will almost always be wrong. But it **is** an estimate, and it **will** be correct … sometimes. Namely, at 11am and 11pm!

- Some estimators are clearly better than others.
- Ideally, would like to estimate *close* to the true value, *most of the time*.
- However, some error is unavoidable.

Errors in Prediction

Even with the best possible model, predictions are not perfect. Prediction error is often divided into two components: ε

Reducible Error

Error that can be decreased by improving the model, collecting more data, or choosing better predictors.

Example: If I estimate the current time by always saying “11:00”, my predictions will be poor. Looking at a clock provides a much more accurate estimate.

Errors in Prediction

Irreducible Error

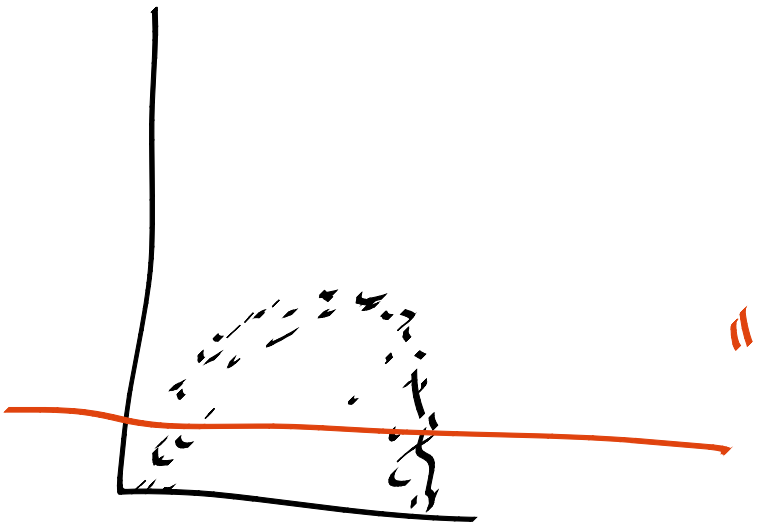
Error that remains even when the best possible model is used.

Example: A clock may be slightly inaccurate, or it may not display seconds. Even with access to the clock, some uncertainty remains.

Key idea: Reducible error comes from our estimate of f, while irreducible error comes from the random error term ε, which cannot be eliminated.

General Points

- Because this is a survey course of data mining methods, we will cover a lot of different approaches.
- Each approach/algorithm has its strengths, and its weaknesses.
- Always consider more than one factor when being told something is “the best at X” - there’s almost always more to the story.



General Points (Continued)

- For example, certain Machine Learning algorithms are very strong at prediction, as measured by the accuracy of the method on a test set (a strength).
- However, once you go outside the realm of the test/train data, the methods can perform very poorly.
- This is a weakness. It does not mean they are not useful, simply that they are limited, as all things are.

Our Objective for Prediction in General

- Our objective, any time we are predicting things, is to reduce the Reducible Error as much as is feasible.
- That means we will have the best prediction we can, under the constraints of the data, and our available skills.
- This course will hopefully broaden your skill-base so that you have access to significantly more skills, which should make your scientific predictions significantly more capable.

A Small R Example

Using the data from before, X_1, X_2, X_3 which are linearly combined (with noise) to make Y. Let's try to predict the value of Y at a particular combination of Xs.

```
1 predict(mod, newdata = data.frame(x1 = 2, x2 = 1, x3 = -1))
```

```
       1
7.327827
```

What is the true, average value?

β_0 $\beta_1 x_1$ $\beta_2 x_2$ $\beta_3 x_3$

$$\begin{aligned} Y = f(2, 1, -1) &= 0.5 + 1.0(2) + 2.5(1) - 1.3(-1) \\ &= 6.3 \end{aligned}$$

A Small R Example (Continued)

- We know we used the “right model” (multiple linear regression is the perfect fit for this data).
- The difference between the true value of 6.3 and our predicted value of 7.33 is **irreducible error**.
- This is due to the ε.

Inference

Inference

What are some of the questions we might ask about the function $f(\cdot)$?

- which predictors are associated with the response?
- what is the relationship between the response and each predictor?
- can the relationship between Y and each predictor be adequately summarized using a linear equation, or is the relationship more complicated?

How Do We Do It?

- Estimating f, the goal of inference, is **hard**.
- There are many methods available that do some aspect of this, but all are subject to **constraints**.
- This kind of problem is sometimes referred to as **underdetermined** - this means there just isn't enough information available to estimate $f(\cdot)$ without making assumptions and simplifications.

What Kinds of Approaches?

- Primarily, we divide our approaches to estimating $f(\cdot)$ into two classes: **parametric** and **non-parametric**.
- Both approaches obtain, from the available data, a **training data** set.
- We use this data to **train** our method on how to estimate $f(\cdot)$.
- In doing this, we apply the method to find an educated guess $\hat{f}$ such that $Y \approx \hat{f}(X)$ for any observation (X, Y).

Parametric Methods

These are what you think of when you think of statistical methods, especially the family of linear models.

1. Make an assumption about the form of $f(\cdot)$, e.g., assume f is linear in X, e.g. linear regression– this is a tremendous simplification.
2. Use a procedure that takes the training data, and trains (or "fits") the model. Typically, we use software for this.

- The **parametric** part of the name comes from the process whereby we reduce the problem from estimating f down to only estimating a set of parameters.
- In linear regression, the parameters are the $\beta_0, \ldots, \beta_p$.

Non-Parametric Methods

- Non-parametric methods do not make explicit assumptions about the functional form of $f(\cdot)$.
- They still have some sort of framework in place (remember: it's impossible to estimate f without something!), but instead of assuming things about the functional form, they seek an estimate of f that is as close to the data as possible without being either too wiggly or too flat.
- The typical restriction we require for non-parametric fits to data is that the resulting fit (estimate) be "smooth".
- This is somewhat vague, and deliberately so. We'll come back to this topic in a few weeks.

How to Choose …

- This semester, we will be looking at a variety of methods.
- Some are parametric, some are non-parametric.
- Some are highly restrictive and have strong assumptions;
- Others are very free, and have only weak or limited assumptions.
- How do you decide what is appropriate for your data?

Trade-Offs: Accuracy and Interpretability

There are two, or maybe three, considerations when choosing a method.

1. Accuracy (or Flexibility):

- how restrictive are the assumptions to the possible outputs of the model?
- Can the chosen model represent a wide variety of possible relationships?
- High accuracy/flexibility examples: deep learning, neural networks, SVMs;
- Low accuracy/flexibility examples: Lasso, least squares, GAMS

Summarized as “how accurately can the model reproduce the given data?”.

Trade-Offs: Accuracy and Interpretability

2. Interpretability:

- understanding the structure and implications of the model framework is important for inference and scientific understanding.
- How well can the model fit be interpreted to give scientific information?
- High interpretability examples: Lasso, least squares;
- Low interpretability examples: deep learning, neural networks

Summarized as “is there a direct interpretation of the coefficients of your model which can be translated into scientific language?”

Trade-Offs

- When **prediction** is your goal, and you don't actually care about the interpretability of the model itself, complex and uninterpretable models such as neural nets can be highly powerful.
- When **inference** is your goal, you do care about interpretability of your model, so less-complex and more inflexible statistical models have advantages.
- We will spend most of this course talking about these trade-offs, especially once we have a few models developed and can discuss how and why you might want to use them.

Assessing Model Accuracy

Evaluate Performance & Make Decisions

- If we want to compare some methods, one to another, or even evaluate the performance of a single method, we need **metrics**: ways of measuring how well predictions actually match observed data.
- Formally, we want to **quantify** how close the predicted response value for a given observation is to the actual, true, response value for that observation.
- In math, we want to somehow measure $y - \hat{y}$: the difference between the true y and the predicted $\hat{y}$.

Choosing a Metric

- We will talk about this a lot, actually - the way of choosing how to measure that difference varies tremendously across methods.
- Choosing a different way of measuring it often gives an entirely different method!
- Some people's careers have been made on coming up with a clever choice…

Squared Error

- One of, if not the, oldest methods available is the **squared error**.
- We want our metric to quantify how close, overall - that is, across the n data points you have.
- So if you have some predictions that are + and some that are -, they can balance out, and indicate the difference is 0, when it is not!

$$\frac{\sum (y - \hat{y})}{n}$$ usually 0

$$\frac{1}{n} \sum (y - \hat{y})^2$$

MSE

So, instead, we square the differences. This leads to **mean squared error** (MSE):

$$\text{MSE} = \frac{1}{n}\sum_{i=1}^{n}\left(y_i - \hat{f}(x_i)\right)^2$$

$\hat{y}_i$

This is the **mean** (average) across the i data points of the squared difference between y_i and $\hat{f}(x_i) = \hat{y}_i$.

This is what is used to give us regression!

How to Apply This?

- There is a key difference between how well a model works on training data (data you've "seen before"), and how well a model works on test data ("new data").
- We don't usually care **that** much about how well a model fits to training data, because … we have all of that data!
- Instead, what we care about is how well a model fits to data we've never seen, because predicting those situations is why we modelled in the first place.

How to Apply This? (Continued)

- In practice: you may not have test data. There are ways around this.
- We can split the data we **do** have into two sets, train on the “training” one, and then compute the MSE or similar metrics on the “test” one. This is the traditional approach in machine learning.
- Or, we can get fancy … more on this later in the term!

Course Set Up for Next Week

First Major Topic

- Our first major topic will be a revisit of linear models.
- We will consider simple linear regression, multiple linear regression, and the concept of transformation of variables.
- It is intended to be a refresher, and give you two weeks to get your feet under you and ready to learn new ideas.

Readings

- The material of this first week is Chapter 2 of ISLR, inclusive.
- Some parts were skipped or only skimmed (esp. 2.1.5, 2.2.2 and 2.2.3), and will be covered “as needed” (i.e., in a couple of weeks).
- Chapter 1 was also mentioned in part, but if you read it, just skim and look for high points.
- The mathematical notation at the end of Chapter 1 is important if you want to understand the equations we will use, although that isn’t strictly required for success.
- The material of the second week, and the first major topic, is Chapter 3, sections 3.1-3.3.3 inclusive, if you would like to (and I recommend it!) read ahead on the material.

Today's Lab

- We will be working through a refresher on R and Quarto, and ensuring everyone is set up to be able to do their work in a good environment.
- We will use some data from ISLR that was mentioned in Chapters 1 and 2, and recreate some of the figures!