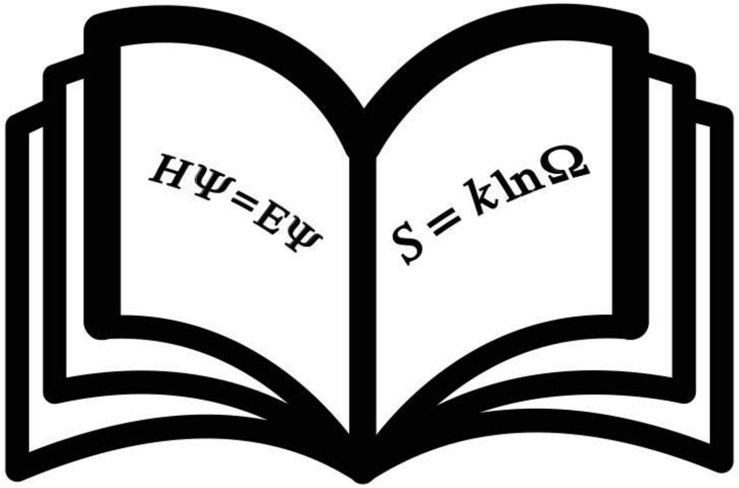
iCOMSE: Machine Learning in Molecular Science

Professor Camille Bilodeau

University of Virginia

April 28th 2025



Today: Introduction to Machine Learning

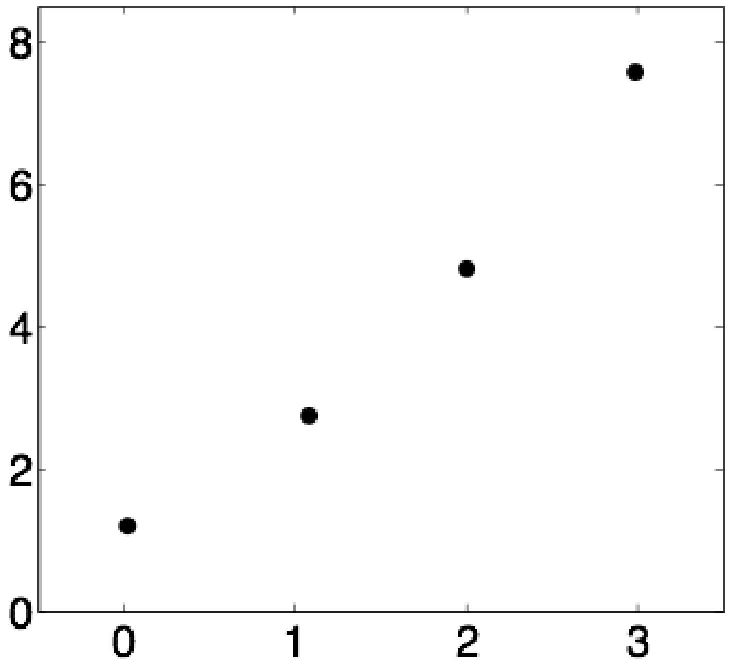
1. Classes of Machine Learning Models
2. Regression Models
3. Classification Models

Types of machine learning:

- Supervised Learning- given a set of input and output data, learn the function that maps the input data to the output data
- Unsupervised Learning- given a set of input data, learn useful data transformations of the input data, often for the purposes of visualization, compression, or generation
- Reinforcement Learning- given a training environment, create an "agent" that maximizes the rewards and minimizes the penalties from that environment

Within supervised learning:

- Regression task- any task where the output is a continuous variable
- Binary classification task- any task where the output can take on one of two categorical values
- Multi-class classification- any task where the output can take on one of three or more categorical values
- Ordinal Classification- any task where the output can take on one of three or more categorical values and those values have a numerical order associated with them



Y	X
1.21	0.02
2.75	1.08
2.80	2.01
7.56	2.99

Regression: Mathematical problem of fitting to data

- Let’s take some measurements that we assume should be in a straight line, but we don’t know *which* straight line.
- $y = ax + b$

Linear Least Squares Regression:

- mathematical equation for a line: $y = \underbrace{a_0}_{} + \underbrace{a_1}_{} x + \varepsilon$ → error

coefficients

↓ rearrange:

$$\varepsilon = y - a_0 - a_1 x$$

- the "best" fit line for a dataset is one that minimizes error

$$\sum_{i=1}^{n} \varepsilon_i = \sum_{i=1}^{n} (y_i - a_0 - a_1 x_i)$$

← n total datapoints

↑ i^{th} datapoint

! Positive and negative errors will cancel out!

Instead:

$$\sum_{i=1}^{n} e_i^2 = \sum_{i=1}^{n} (y_i - a_0 - a_1 x_i)^2 = S_r = \text{sum of residuals}$$

↑ we want to minimize... how?

$$\sum_{i=1}^{n} e_i^2 = \sum_{i=1}^{n} (y_i - a_0 - a_1 x_i)^2 = S_r = \text{sum of residuals}$$

we want to minimize... how?

$$\frac{\partial S_r}{\partial a_0} = -2\sum_{i=1}^{n} (y_i - a_0 - a_1 x_i) = 0$$

$$\frac{\partial S_r}{\partial a_1} = -2\sum_{i=1}^{n} [(y_i - a_0 - a_1 x_i)x_i] = 0$$

rearrange:

$$0 = \sum y_i - \sum a_0 - \sum a_1 x_i$$

$$0 = \sum x_i y_i - \sum a_0 x_i - \sum a_1 x_i^2$$

each $\sum$ is $\sum_{i=1}^{n}$

$\sum a_0 = n a_0$, so:

$$\sum y_i = n a_0 + a_1 \sum x_i$$

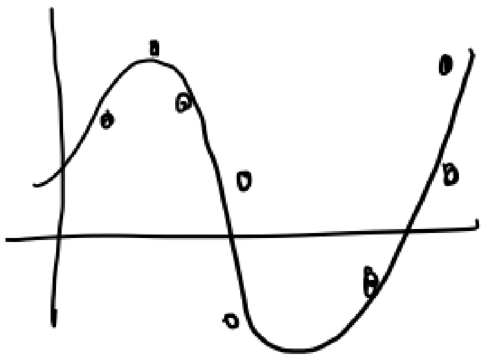
rearrange

$$a_0 = \bar{y} - a_1 \bar{x}$$

$$\sum x_i y_i = a_0 \sum x_i + a_1 \sum x_i^2 \rightarrow a_1 = \frac{\sum x_i y_i - a_0 \sum x_i}{\sum x_i^2}$$

☆ Remember: x_i and y_i are known! a_0 and a_1 are unknown!

$$a_1 = \frac{n\sum x_i y_i - \sum x_i \sum y_i}{n\sum x_i^2 - (\sum x_i)^2}$$



Polynomial Regression

$$y = a_0 + a_1 x + a_2 x^2 + \varepsilon$$

$$S_r = \text{sum of residuals} = \sum_{i=1}^{n} (y_i - a_0 - a_1 x_i - a_2 x_i^2)^2$$

$$\frac{dS_r}{da_0} = -2 \sum (y_i - a_0 - a_1 x_i - a_2 x_i^2) = 0$$

$$\frac{dS_r}{da_1} = -2 \sum x_i (y_i - a_0 - a_1 x_i - a_2 x_i^2) = 0$$

$$\frac{dS_r}{da_2} = -2 \sum x_i^2 (y_i - a_0 - a_1 x_i - a_2 x_i^2) = 0$$

rearrange

$$\sum y_i = n a_0 + a_1 \sum x_i + a_2 \sum x_i^2$$

$$\sum x_i y_i = a_0 \sum x_i + a_1 \sum x_i^2 + a_2 \sum x_i^3$$

$$\sum x_i^2 y_i = a_0 \sum x_i^2 + a_1 \sum x_i^3 + a_2 \sum x_i^4$$

* note: all equations are linear in unknowns (a_0, a_1, a_2) (system of linear equations we can solve)

Recast our problem:

$$y = a_0 \underbrace{z_0}_{1} + a_1 \underbrace{z_1}_{x} + a_2 \underbrace{z_2}_{x^2}$$

where for our problem:

$$\vec{y} = [Z]\vec{a} + \vec{e}$$

where $Z = \begin{bmatrix} 1 & x_1 & x_1^2 \\ 1 & x_2 & x_2^2 \\ 1 & x_3 & x_3^2 \\ 1 & x_4 & x_4^2 \\ \vdots & \vdots & \vdots \end{bmatrix}$ and $\vec{a} = \begin{bmatrix} a_0 \\ a_1 \\ a_2 \end{bmatrix}$

Recall that our data looks like:

y	x
10	101
20	178
30	289
40	372
⋮	⋮

$\vec{y}$ $\vec{x}$

we can rewrite our previous linear equations as:

$$[Z]^T\vec{y} = [Z]^T[Z]\vec{a}$$

$$\times (Z^T Z)^{-1} \qquad \times (Z^T Z)^{-1}$$

$$\boxed{\vec{a} = (Z^T Z)^{-1} Z^T \vec{y}}$$

General Least Squares Regression

Special Cases of General Least Squares Regression:

① polynomial regression:

$$Z = \begin{bmatrix} 1 & x_1 & x_1^2 & \cdots & x_1^m \\ 1 & x_2 & x_2^2 & \cdots & x_2^m \\ \vdots & & & & \vdots \\ 1 & & & & x_n^m \end{bmatrix}$$

② Fourier Analysis:

$$Z = \begin{bmatrix} 1 & \cos(\omega x_1) & \sin(\omega x_1) \\ 1 & \cos(\omega x_2) & \sin(\omega x_2) \\ \vdots & & \vdots \\ 1 & & \sin(\omega x_n) \end{bmatrix}$$

③ Multiple Linear Regression:

y	x_1	x_2
4.3	323	5.2
3.2	348	7.3
2.9	373	8.1
⋮		

$$S_r = \sum_{i=1}^{n} (y_i - a_0 - a_1 x_{1,i} - a_2 x_{2,i})^2$$

$$\begin{cases} \dfrac{\partial S_r}{\partial a_0} = -2\sum (y_i - a_0 - a_1 x_{1,i} - a_2 x_{2,i}) = 0 \\ \dfrac{\partial S_r}{\partial a_1} = -2\sum x_{1,i} (y_i - a_1 x_{1,i} - a_2 x_{2,i}) = 0 \\ \dfrac{\partial S_r}{\partial a_2} = -2\sum x_{2,i} (y_i - a_1 x_{1,i} - a_2 x_{2,i}) = 0 \end{cases}$$

$$[Z]^T \vec{y} = [Z]^T [Z] \vec{a} \quad \text{where } Z = \begin{bmatrix} 1 & x_{11} & x_{12} \\ 1 & x_{12} & x_{22} \\ 1 & x_{13} & x_{23} \\ \vdots & \vdots & \vdots \\ 1 & x_{1n} & x_{2n} \end{bmatrix}$$

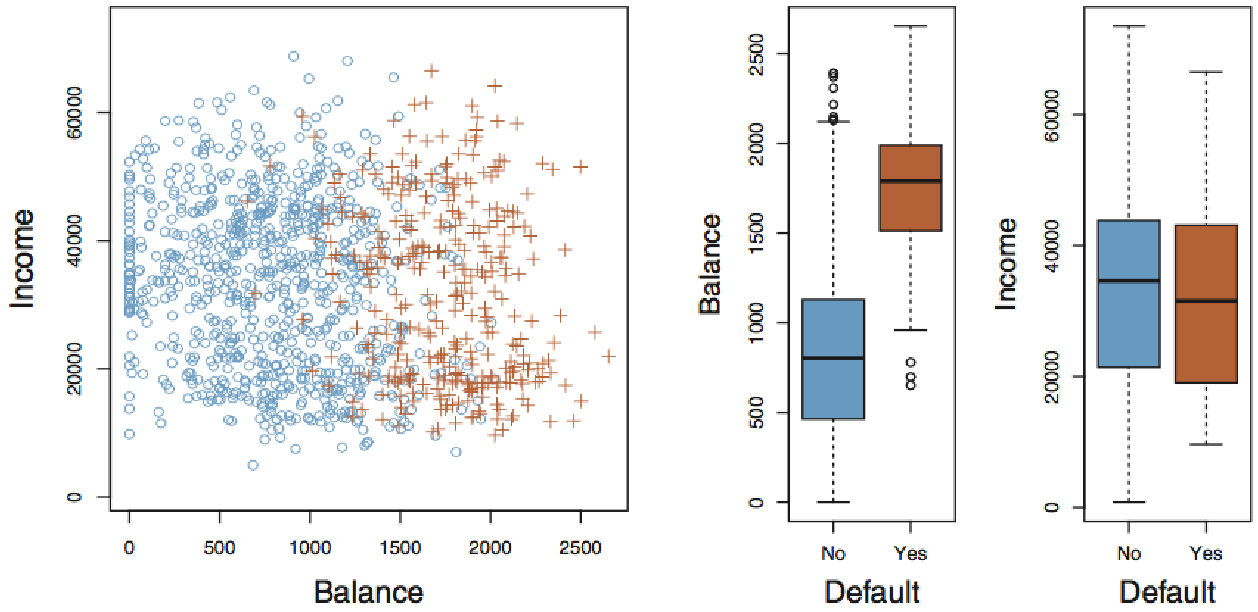
$$\vec{a} = (Z^T Z)^{-1} Z^T \vec{y}$$

$\longleftrightarrow$ # of features

To the Notebook!

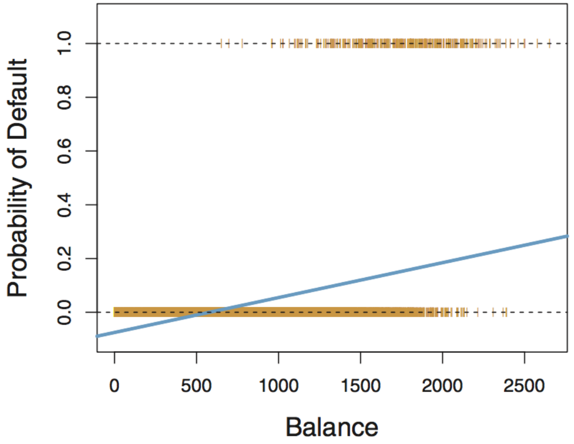
Classification with the logistic model

- Consider the data below:
- We are trying to make a guess as to whether someone will default on a loan on the basis of their bank account balance and income level
- We have a two choice classification: default or not default



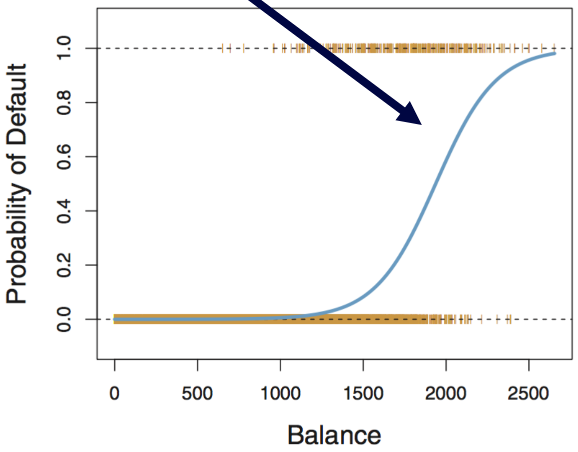
Why logistic regression?

- Let's say we trained a linear model such that the output was either 1 or 0.
- This is what our predictions might look like



$$\Pr(\texttt{default} = \texttt{Yes} | \texttt{balance})$$

- We would like something more like this:



The logistic function

$$p(X) = \beta_0 + \beta_1 X.$$

simple linear model

$$p(X) = \frac{e^{\beta_0+\beta_1 X}}{1 + e^{\beta_0+\beta_1 X}}.$$

logistic equation

$$\frac{p(X)}{1 - p(X)} = e^{\beta_0+\beta_1 X}.$$

Solve for $e^{\beta_0+\beta_1 X}$

The logistic function

$$\frac{p(X)}{1 - p(X)} = e^{\beta_0 + \beta_1 X}.$$

$$\log\left(\frac{p(X)}{1 - p(X)}\right) = \beta_0 + \beta_1 X.$$

The left side of the equation is log of the odds
Called the "logit",
Equation elates the change in log of odds of
success w/change in X

The logit is a linear function

Probability goes from 0 to 1
Log probability goes from -∞ to 1
Logit (log odds) goes from -∞ to ∞

Multiple ways to solve it

- Use linear regression to solve:

$$\log\left(\frac{p(X)}{1-p(X)}\right) = \beta_0 + \beta_1 X.$$

- Use nonlinear regression of $$p(X) = \frac{e^{\beta_0+\beta_1 X}}{1+e^{\beta_0+\beta_1 X}}.$$

- Predict $\log\left(\frac{p(X)}{1-p(X)}\right)$ > 0, yes, if < 0. No.

Is the logistic model a classifier?

- Yes!
- Given a two class situation “e.g., default vs. non-default”,
- The logistic model can take a set of training data
- And gives a function that makes a prediction about what class a new or different input would be in
- P < 0.5 = false vs. P >= 0.5 true

To the Notebook!