Introduction to Machine Learning

What is Machine Learning?

Machine Learning (ML) is a subset of artificial intelligence that allows systems to learn and improve from experience without being explicitly programmed. It focuses on developing algorithms that can access data and use it to learn for themselves.

For example, a spam email filter learns to classify emails as spam or not spam based on past email data.

Supervised Learning

In supervised learning, the model is trained on a labeled dataset. That means each training example is paired with an output label. The model learns to map inputs to the correct output.

Example: Predicting house prices based on features such as size, number of bedrooms, and location. The training data contains past house sales with these features and the corresponding prices.

Unsupervised Learning

In unsupervised learning, the model is given data without explicit instructions on what to do with it. The system tries to learn the patterns and the structure from the data.

Example: Customer segmentation in marketing, where the algorithm groups customers based on purchasing behavior without prior labels.

Regression

Regression

- It is a statistical approach used to analyze the relationship between a dependent variable and one or more independent variables.
- The objective is to determine the most suitable function that characterizes the connection between these variables.
- Data for regression problems comes as a set of N input/output observation pairs

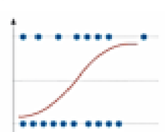
$$\{(\boldsymbol{x}_n, y_n)\}_{n=1}^{N} \triangleq (\boldsymbol{x}_1, y_1), (\boldsymbol{x}_2, y_2), \ldots, (\boldsymbol{x}_N, y_N)$$

  - $\boldsymbol{x}_n$: independent variable known as input/ regressor/ predictors/ covariates.
  - Each dimension of the input is referred to as a *feature/explanatory variable.*
  - y_n : dependent variable known as output/ response/prediction/estimation.
- In classification, the output values y_n are called *labels*, and all points sharing the same label value are referred to as a *class* of data.

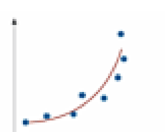
Regression

- Regression is performed to
  - produce a trend line or a curve that can be used to help visually summarize.
  - drive home a particular point about the data under study.
  - learn a model to precise predictions can be made regarding output values in the future.
- Types of regression

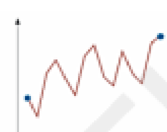
Linear regression: Predicts continuous output by modeling a straight-line relationship



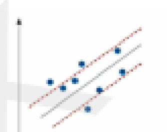
Logistic regression: Models the probability of binary outcomes



Polynomial regression: Capture nonlinear relationship by fitting polynomial curve



Time series regression: Predicts future values in a time-dependent data set.



Support vector regression: Approx. continuous function using a hyperplane that best fits the data.

Linear Regression

- Linear regression is a type of supervised learning algorithm.
- It computes the linear relationship between the dependent variable and one or more independent features by fitting a linear equation to observed data.

Linear Regression

- **Simple Linear Regression** (only one dependent variable or one dimension)
- **Multiple Linear Regression** (more than one dependent variable or multi dimension)

Simple Linear Regression

- Fitting of a representative line to a set of input/output data points.
- For an input $\boldsymbol{x}_n \in \mathbb{R}^N$, it is a column vector of length N, given by

$$\boldsymbol{x}_n = \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_N \end{bmatrix}$$

- The relationship between input and output is

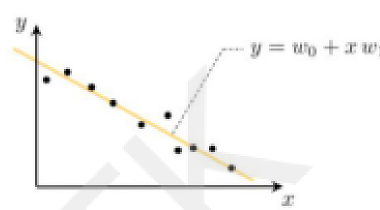
$$\boldsymbol{y_n} \approx w_0 + w_1 \boldsymbol{x_n}$$

- The linear equation is called the model $f_{w_0,w_1}(\boldsymbol{x_n})$

$$f_{w_0,w_1}(\boldsymbol{x_n}) = w_0 + w_1 \boldsymbol{x_n}$$

Where w_0, w_1 are the parameters known as

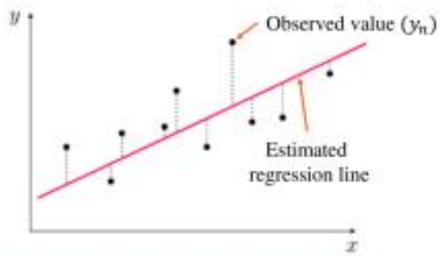
w_1: regression coefficient; w_0: bias/ intercept



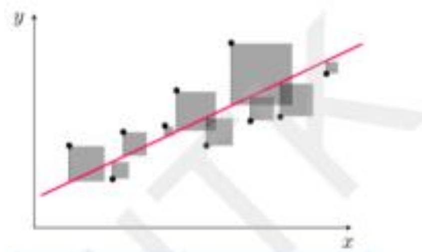
Simple Linear regression one independent variable $\boldsymbol{x_n}$

Linear Regression: Least Squares Cost Function

- Find a weight vector $\boldsymbol{w}$ that
  - tightly holds each of N approximate equalities
  - Or the error between $\dot{\boldsymbol{x}}_n^T\boldsymbol{w}$ and y_n to be small
  - Or deviation of the observations from the true regression line is small



The best line and its errors



The best line and its squared errors

Linear Regression: Least Squares Cost Function

- Point-wise cost that measures the error/ residual (ϵ_n) of a model for the point $\{\boldsymbol{x}_n, y_n\}$ is given by

$$g_n(\boldsymbol{w}) = \epsilon_n^2 = \left(\dot{\boldsymbol{x}}_n^T\boldsymbol{w} - y_n\right)^2$$

- For all N such values to be small, cost is averaged over dataset, forming a *least squares cost function* $g(\boldsymbol{w})$

$$g(\boldsymbol{w}) = \frac{1}{N}\sum_{n=1}^{N} g_n(\boldsymbol{w}) = \frac{1}{N}\sum_{n=1}^{N}\left(\dot{\boldsymbol{x}}_n^T\boldsymbol{w} - y_n\right)^2$$

- It is a function of the weights $\boldsymbol{w}$ and data as well.
- The best fitting hyperplane is the one whose parameters minimize this error.

$$\min_{\boldsymbol{w}} g(\boldsymbol{w}) = \min_{\boldsymbol{w}} \frac{1}{N}\sum_{n=1}^{N}\left(\dot{\boldsymbol{x}}_n^T\boldsymbol{w} - y_n\right)^2$$

Objective: Find the parameters 'w' that minimize the cost function.

Least Squares Cost Function

Linear Regression: Least Squares Estimators

Recollect : $y_n \approx w_0 + x_{1,n}w_1 + x_{2,n}w_2 + \cdots + x_{P,n}w_P$, $n = 1, \ldots, N$

- Consider $P = 1$; then $g(\boldsymbol{w}) = \frac{1}{N}\sum_{n=1}^{N}\left(\dot{\boldsymbol{x}}_n^T\boldsymbol{w} - y_n\right)^2 = \frac{1}{N}\sum_{n=1}^{N}(\overbrace{w_0 + w_1\boldsymbol{x_n}}^{a} - \overbrace{y_n}^{b})^2$ $(a-b)^2$
- The least square estimators of w_0 and w_1, say $\widehat{w}_0$ and $\widehat{w}_1$ must satisfy

$$\left.\frac{\partial g(\boldsymbol{w})}{\partial w_0}\right|_{\widehat{w}_0,\widehat{w}_1} = 0$$

$$\Rightarrow \frac{2}{N}\sum_{n=1}^{N}(w_0 + w_1x_n - y_n) = 0$$

$$\Rightarrow N\widehat{w}_0 + \widehat{w}_1\sum_{n=1}^{N}x_n = \sum_{n=1}^{N}y_n$$

$$\Rightarrow \boxed{\widehat{w}_0 = \bar{y} - \widehat{w}_1\bar{x}}$$

$$\left.\frac{\partial g(\boldsymbol{w})}{\partial w_1}\right|_{\widehat{w}_0,\widehat{w}_1} = 0$$

$$\Rightarrow \frac{2}{N}\sum_{n=1}^{N}(w_0 + w_1x_n - y_n)x_n = 0$$

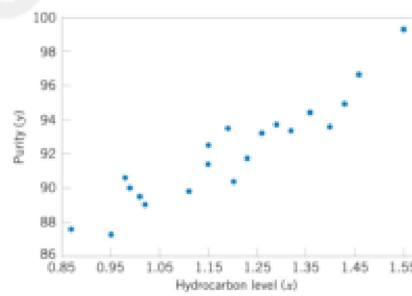
$$\Rightarrow \widehat{w}_0\sum_{n=1}^{N}x_n + \widehat{w}_1\sum_{n=1}^{N}x_n^2 = \sum_{n=1}^{N}y_nx_n$$

Substitute $\widehat{w}_0$

$$\Rightarrow \boxed{\widehat{w}_1 = \frac{\sum_{n=1}^{N}y_nx_n - N\bar{x}\bar{y}}{\sum_{n=1}^{N}x_n^2 - \frac{(N\bar{x})^2}{N}} = \frac{S_{xy}}{S_{xx}}}$$

Case Study: Oxygen Purity Dataset

Observation Number	Hydrocarbon Level x (%)	Purity y (%)
1	0.99	90.01
2	1.02	89.05
3	1.15	91.43
4	1.29	93.74
5	1.46	96.73
6	1.36	94.45
7	0.87	87.59
8	1.23	91.77
9	1.55	99.42
10	1.40	93.65
11	1.19	93.54
12	1.15	92.52
13	0.98	90.56
14	1.01	89.54
15	1.11	89.85
16	1.20	90.39
17	1.26	93.25
18	1.32	93.41
19	1.43	94.98
20	0.95	87.33



Scatter diagram of oxygen purity versus hydrocarbon level

Objective: Fit a simple linear regression model to the oxygen purity data in Table

Least Squares Estimators and Case Study

Case Study: Oxygen Purity Dataset

- Solution: From Table

$$N = 20, \quad \sum_{n=1}^{20} x_n = 23.92, \quad \bar{x} = 1.196, \quad \sum_{n=1}^{20} y_n = 1843.21, \quad \bar{y} = 92.1605$$

$$\sum_{n=1}^{20} x_n^2 = 29.2892, \quad \sum_{n=1}^{20} y_n^2 = 170{,}044.5321, \quad \sum_{n=1}^{20} x_n y_n = 2214.6566$$

$$S_{xx} = 0.68088, \quad S_{xy} = 10.17744$$

The least square estimates:

$$\hat{w}_1 = \frac{S_{xy}}{S_{xx}} = 14.94748, \quad \hat{w}_0 = \bar{y} - \hat{w}_1 \bar{x} = 74.2833$$

The fitted simple linear regression model:

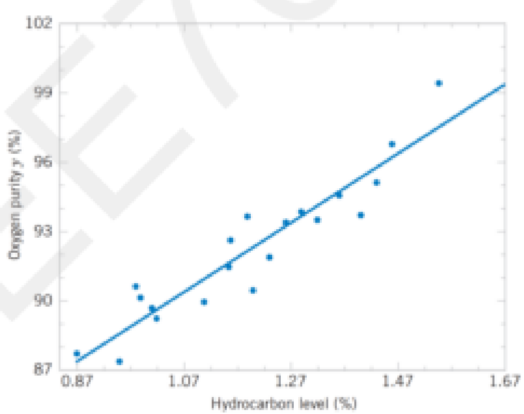
$$\boxed{\hat{y} = \hat{w}_0 + \hat{w}_1 x = 74.283 + 14.947x}$$

Case Study: Oxygen Purity Dataset

Linear Reg.

$y = w_0 + w_1 x$

we found this



Scatter plot of oxygen purity (y) versus hydrocarbon level (x) and regression model $\hat{y} = 74.283 + 14.947x$.

Case Study: Oxygen Purity Dataset

Linear Regression: Gradient Descent along the Cost Function

- Objective: Find the value of w_0 and w_1 that minimizes the cost/error function $g(\boldsymbol{w})$.
- Outline:
  - Start with some w_0, w_1
  - Keep changing w_0, w_1 to reduce $g(\boldsymbol{w})$
  - Settle at or near a minimum

- Model: $f_{w_0,w_1}(\boldsymbol{x_n}) = w_0 + w_1\boldsymbol{x_n}$
- *Cost Function* : $g(\boldsymbol{w}) = \frac{1}{N}\sum_{n=1}^{N}\left(f_{w_0,w_1}(\boldsymbol{x_n}) - y_n\right)^2$

Weight/Parameter Updating

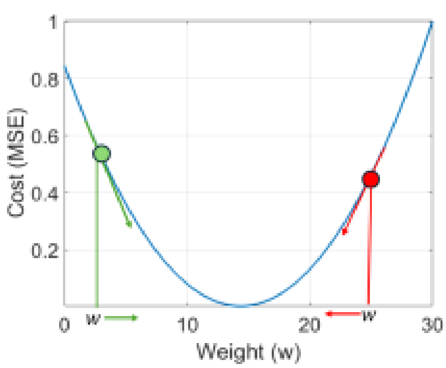
$$w_0 = w_0 - \alpha\frac{\partial}{\partial w_0}g(\boldsymbol{w})$$
$$w_1 = w_1 - \alpha\frac{\partial}{\partial w_1}g(\boldsymbol{w})$$

Simultaneously update the weights
Repeat until convergence
α: Learning rate

EE708 Module 3 18

29-05-20

Linear Regression: Gradient Descent along the Cost Function



$$w = w - \alpha\frac{\partial}{\partial w}g(\boldsymbol{w})$$

- $\frac{\partial}{\partial w}g(\boldsymbol{w}) < 0$ (negative number)

  $w = w - \alpha(negative\ number)$: Increase w

- $\frac{\partial}{\partial w}g(\boldsymbol{w}) > 0$ (positive number)

  $w = w - \alpha(positive\ number)$: Decrease w

EE708 Module 3 19

Gradient Descent along the Cost Function