Artificial Intelligence Algorithms and Mathematics

CSCN 8000

Statistics

- Continue on statistics:
  - Data preprocessing
  - Feature normalization
  - Data Encoding
  - Feature Engineering
- Linear Regression
- Regression Evaluation Metrics

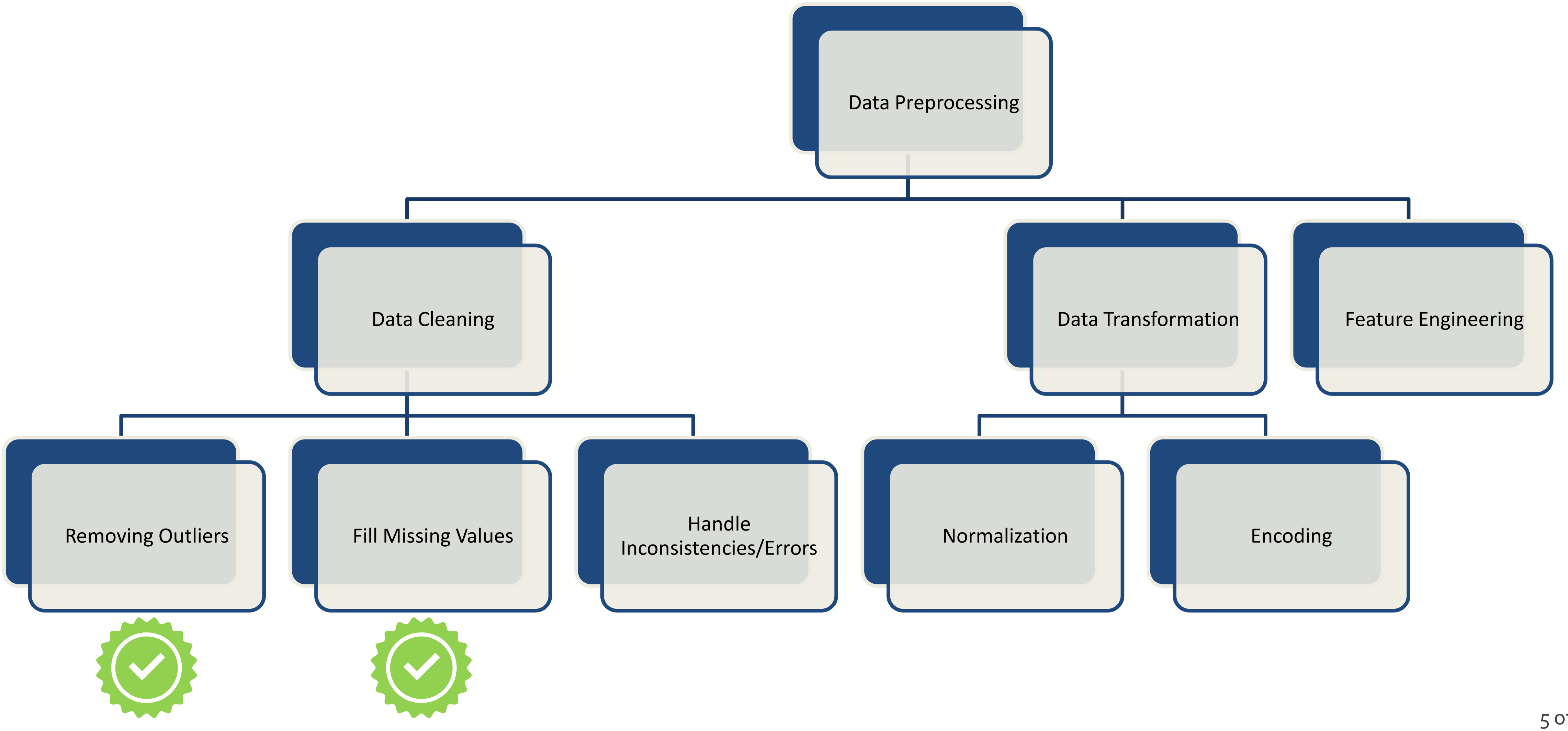
Data Preprocessing

Garbage in, Garbage out

Low-quality data will lead to low-quality and misleading analysis results

(No matter how sophisticated the model is!)

Data Preprocessing



Handling Inconsistencies

- A crucial step to correct inconsistencies in the data via fuzzy joins, regular expressions or other methods.

patientCity	Value Count
Guelph	1662
Kitchener	1247
Waterloo	793
Cambridge	330
Fergus	204
KITCHENER	10
KITTChener	21
Geulph	12
GUELPH	23
WATERLO	9
FRGUS	13

Before

patientCity	Value Count
Guelph	1697
Kitchener	1278
Waterloo	802
Cambridge	330
Fergus	217

After

Data Normalization

- A crucial step in preparing data for machine learning algorithms. It helps to ensure that features are on a similar scale, which can lead to more stable and faster convergence during training.
- Allows us to make sure that no variable dominates the other variable.
- There are several ways to perform feature scaling in machine learning.

Min-Max Scaling

- Definition:
  - Values are shifted and rescaled to range from 0 to 1.
- Formulation:
  - $X_{norm} = \frac{X - X_{min}}{X_{max} - X_{min}}$
- Implementation:
  - Sklearn provides MinMaxScaler for this.

Pros	Cons
• Keeps variable relationships intact • Suitable for algorithms requiring similar scales	• Sensitive to outliers

House	Cost ($)	Size (sq. ft)	Cost Scaled	Size Scaled
1	250000	2000	0.25	0.142
2	300000	2200	0.375	0.21
3	200000	1800	0.0	0.0
4	400000	2500	0.75	0.375
5	150000	1500	0.125	0.071
6	450000	2800	1.0	1.0
7	350000	2100	0.625	0.25
8	275000	1900	0.3125	0.118
9	325000	2300	0.5	0.429
10	275000	1600	0.3125	0.071

Z-score Normalization

- Definition:
  - Scales the data to have a mean of 0 and a standard deviation of 1.
- Formulation:
  - $X_{norm} = \frac{X-\mu}{\sigma}$
- Implementation:
  - Sklearn provides StandardScaler for this.

House	Cost ($)	Size (sq. ft)	Cost Scaled	Size Scaled
1	250000	2000	-0.588	-0.226
2	300000	2200	0.117	0.159
3	200000	1800	-1.293	-0.543
4	400000	2500	1.822	1.063
5	150000	1500	-1.949	-1.351
6	450000	2800	2.117	1.866
7	350000	2100	0.823	0.523
8	275000	1900	-0.411	-0.098
9	325000	2300	0.529	0.764
10	275000	1600	-0.411	-1.072

Pros	Cons
• Maintains shape of original distribution. • Less sensitive to outliers.	• Not suitable if needs to maintain the mean and standard deviation.

Robust Scaling

- Definition:
  - Uses the median and the interquartile range (IQR) instead of the mean and standard deviation.
- Formulation:
  - $X_{norm} = \frac{X - median}{IQR}$
- Implementation:
  - Sklearn provides RobustScaler for this.

Pros	Cons
• Least sensitive to outliers compared to Min-Max/Z-score	• Still influenced by extreme outliers.

House	Cost ($)	Size (sq. ft)	Cost Scaled	Size Scaled
1	250000	2000	-0.25	0.0
2	300000	2200	0.25	0.2857
3	200000	1800	-0.75	-0.2857
4	400000	2500	1.25	0.5714
5	150000	1500	-1.25	-0.5714
6	450000	2800	1.5	1.1429
7	350000	2100	0.75	0.4286
8	275000	1900	0.0	0.1429
9	325000	2300	0.5	0.8571
10	275000	1600	0.0	-0.4286

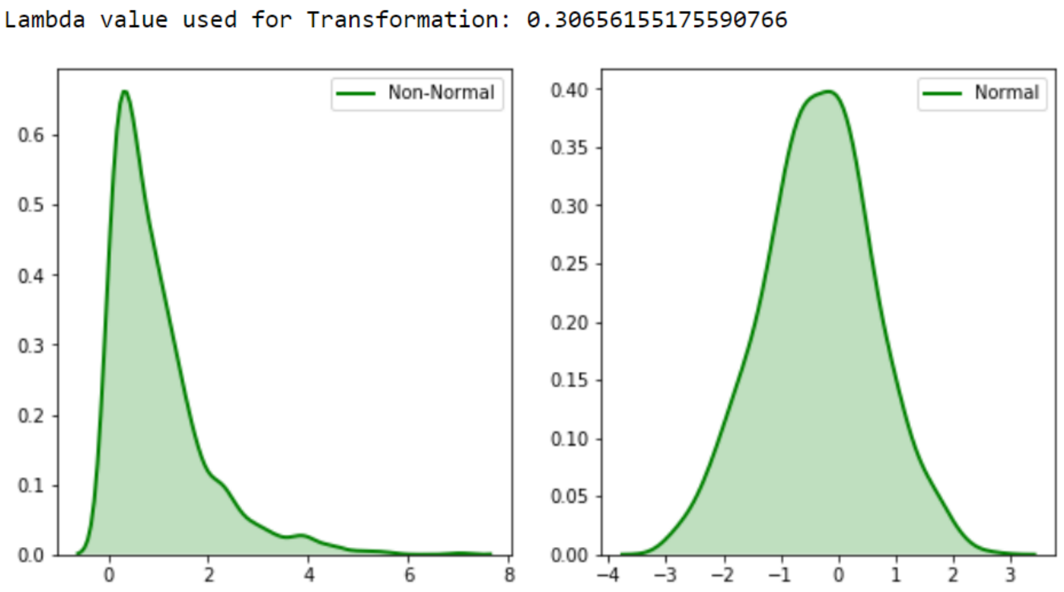
Box Cox Transformation

- Definition:
  - Used to stabilize the variance and make the data more normally distributed.
- Formulation:

- $$X_{norm} = \begin{cases} \log(X)\,, \ \ if\ \lambda = 0 \\ \frac{X^{\lambda}-1}{\lambda}, otherwise \end{cases}$$

- Implementation:
  - Scipy.stats provides boxcox() for this.

Pros	Cons
• Best if the algorithm requires normal distributions	• Assumes that all values are strictly positive



Data Encoding

- Its primary purpose is to transform categorical variables, which represent qualitative attributes, into a numerical format that can be effectively utilized by mathematical models.
- This conversion is imperative because most machine learning algorithms are designed to operate on numerical data
- There are several ways to perform data encoding in machine learning.

Label Encoding

- Assigns a unique integer to each category.
- Suitable for ordinal categorical variables with a clear order.
- May introduce unintended ordinal relationships.

Sample	Education Level
1	High School
2	Bachelor's Degree
3	Master's Degree
4	High School
5	PhD
6	Bachelor's Degree
7	High School
8	Master's Degree
9	Bachelor's Degree
10	High School

Sample	Encoded Education Level
1	0
2	1
3	2
4	0
5	3
6	1
7	0
8	2
9	1
10	0

One-Hot Encoding

- Represents each category as a binary vector.
- Suitable for nominal categorical variables.
- Avoids the assumption of ordinality between categories.
- Can lead to high-dimensional data if there are many categories.

Sample	Favorite Color
1	Red
2	Blue
3	Green
4	Red
5	Blue
6	Green
7	Red
8	Blue
9	Green
10	Red

Sample	Red	Blue	Green
1	1	0	0
2	0	1	0
3	0	0	1
4	1	0	0
5	0	1	0
6	0	0	1
7	1	0	0
8	0	1	0
9	0	0	1
10	1	0	0

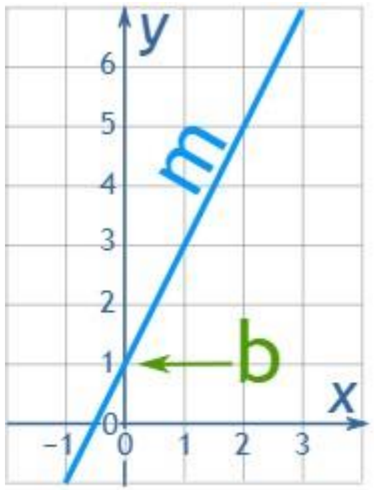
Feature Engineering

- Involves creating new features or modifying existing ones to improve the performance of machine learning models.
- Example:
  - Combine two or more existing features to create new ones
  - Create summary statistics (i.e. fill with mean, median per another categorical feature)

Height (cm)	Weight (kg)	BMI
165	70	25.71
170	68	23.53
155	60	24.97
180	75	23.15
160	65	25.39
175	72	23.51

Machine Learning Algorithms

Recall Equation of Line



$$y = mx + b$$

Slope or Gradient (m)

y value when **x=0** (*see Y Intercept*) (b)

y = how far up

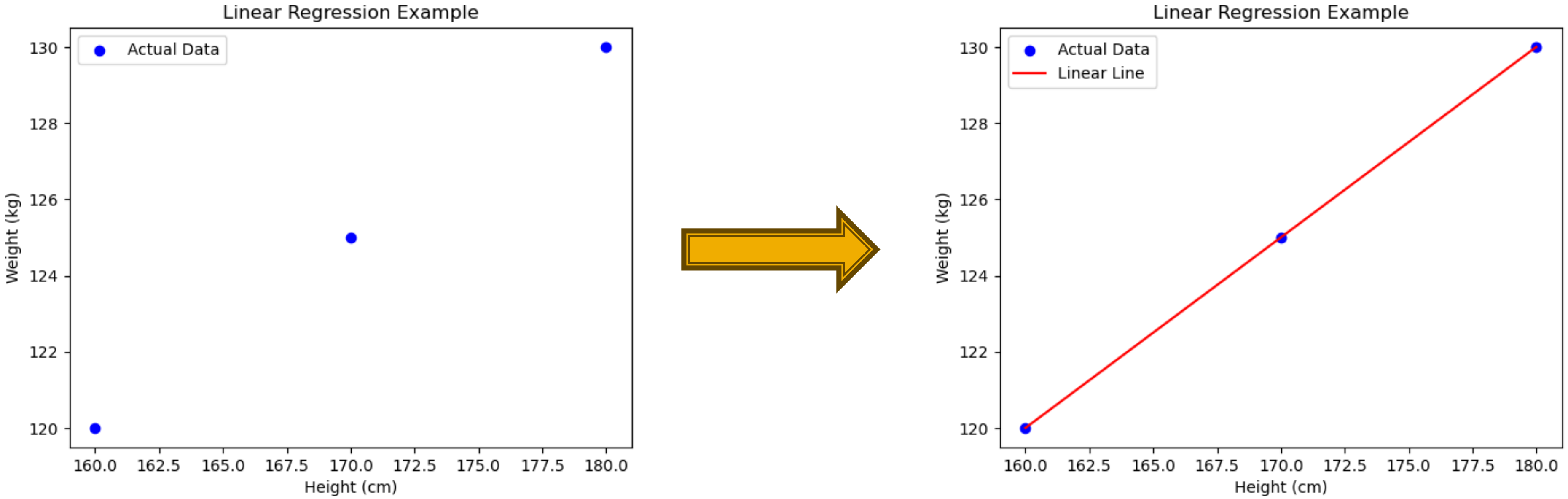
x = how far along

m = Slope or Gradient (how steep the line is)

b = value of **y** when **x=0**

Linear Regression: Formulation

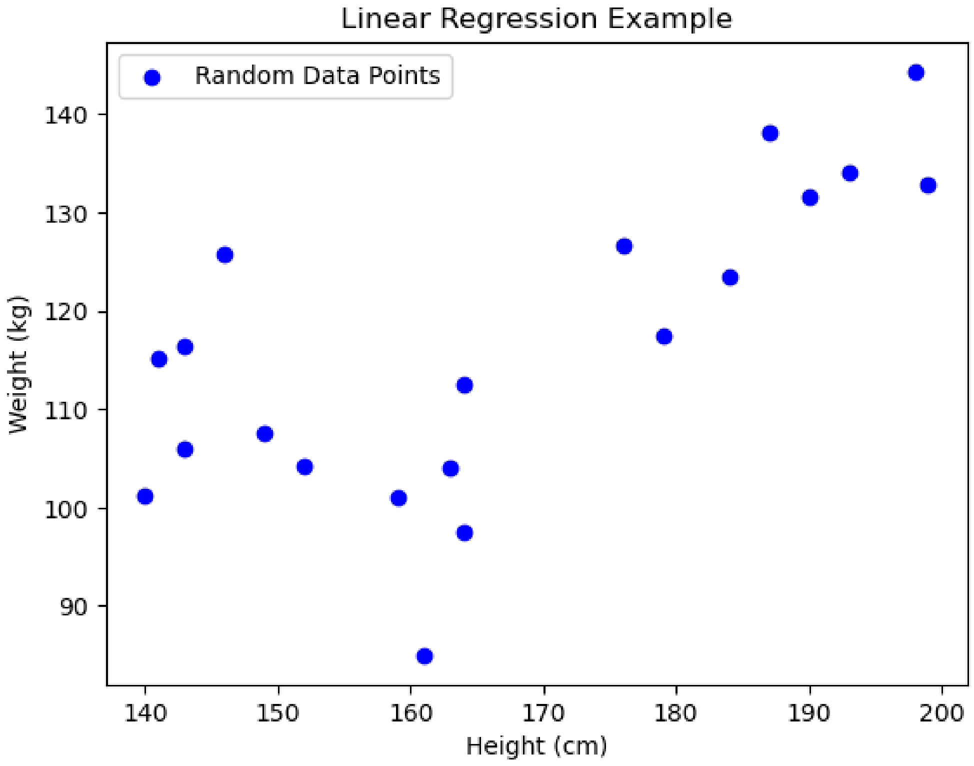
- Assume we have a set of three 2D points where the x-axis represents Height and y-axis represents Weight, such that [(160,120), (170,125), (180,130)). Can we directly compute the equation of line passing through the three points?



$$y = \frac{1}{2}x + 40$$

Linear Regression: Formulation

- Assume we have a set of 20 randomly scattered 2D points where the x-axis represents Height and y-axis represents Weight. Can we directly compute the equation of line passing through **all the points**?



$$y = m\,x + c$$

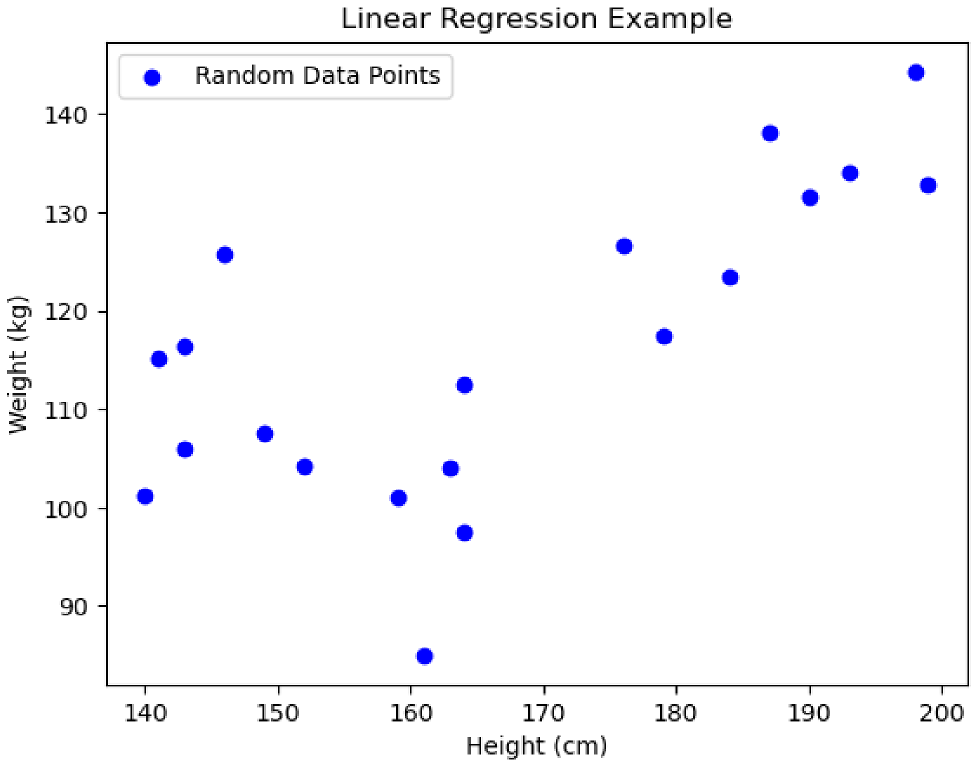
$$m = ?\,,\, c = ?$$

Linear Regression: Formulation

- Linear regression is a supervised learning algorithm which allows us to find the **best fit** line/hyperplane passing through the set of available data points.
- The predicted **best fit line** equation corresponds to predicting a continuous variable $\hat{\boldsymbol{y}}$ given input features $\boldsymbol{x}$, such that:

$$\hat{\boldsymbol{y}} = \vec{\boldsymbol{w}}\,\vec{\boldsymbol{x}} + \boldsymbol{b}$$

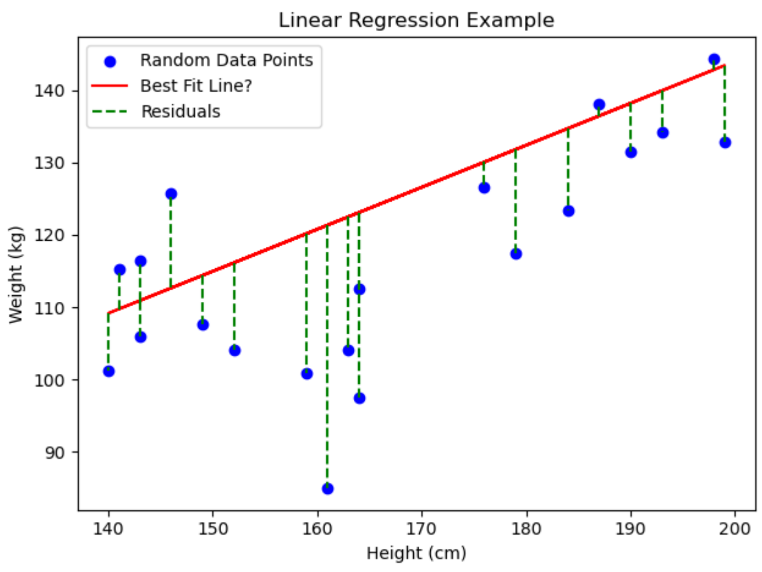
- $\vec{\boldsymbol{w}}, \boldsymbol{b}$ are the missing parameters that need to be estimated to get the best fit line equation.

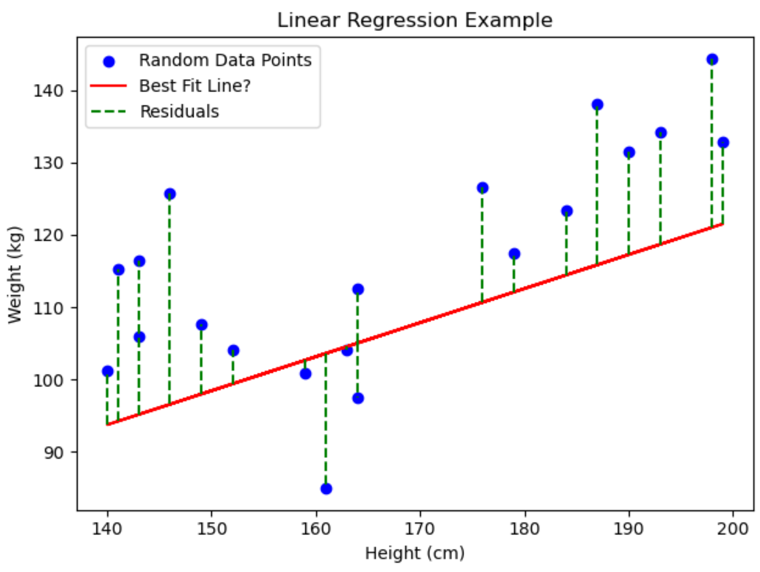


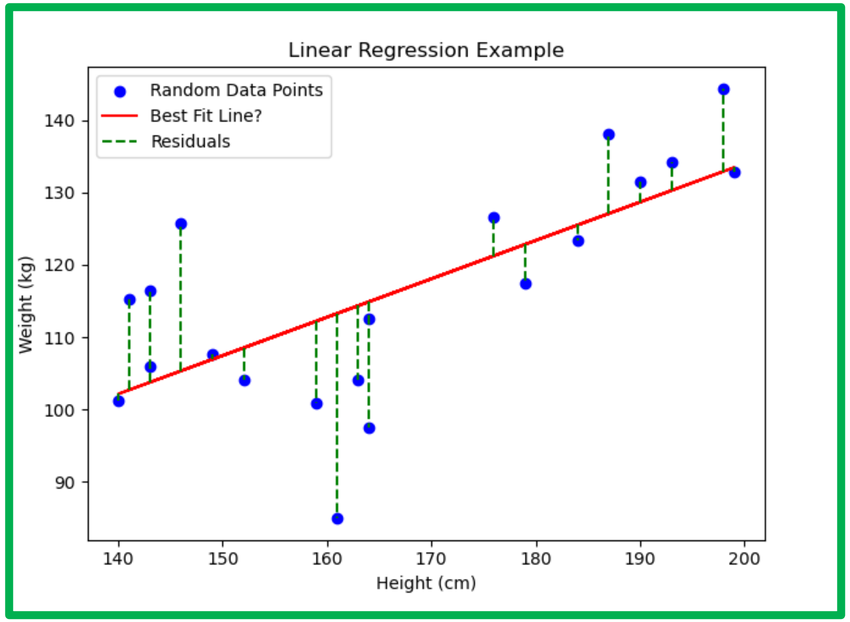
$$\boldsymbol{w} = ?\,,\ \boldsymbol{b} = ?$$

Linear Regression: Cost Function

- By definition, the **best fit** line is one that has the minimum distances (residuals) between itself and all the data points available.
- Which of these could be the **best fit** line?







Linear Regression: Cost Function

- By definition, the **best fit** line is one that has the minimum distances (residuals) between itself and all the data points available.
- To find the best fit line, we need to **minimize** the average of the squared distances between the predictions $\hat{\boldsymbol{y}}$ and the actual output $\boldsymbol{y}$, such that:

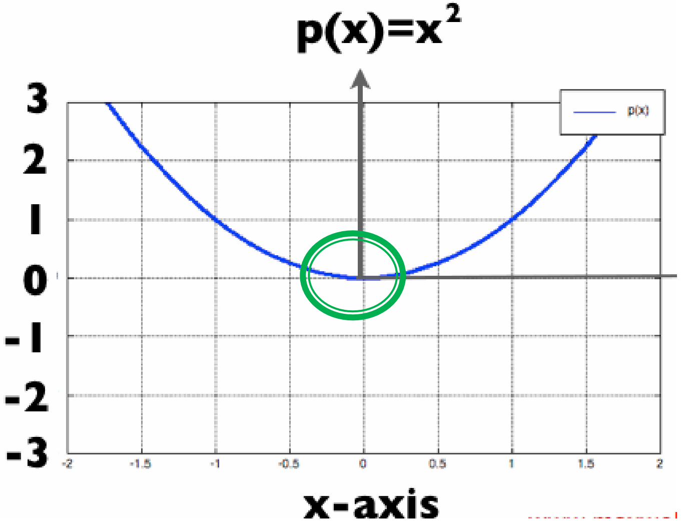
$$L(\vec{\boldsymbol{w}}, \boldsymbol{b}) = \frac{1}{N}\sum_{i=1}^{N}(\boldsymbol{y}_i - \hat{\boldsymbol{y}}_i)^2 = \frac{1}{N}\sum_{i=1}^{N}(\boldsymbol{y}_i - (\vec{\boldsymbol{w}}\,\vec{\boldsymbol{x}_i} + \boldsymbol{b}))^2$$

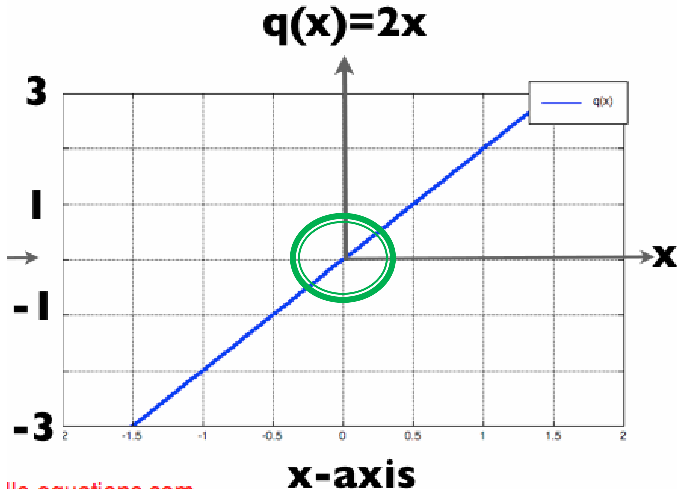
- L is usually referred to as "Loss Function" or "Cost Function".
- Our target is to find the value of $\vec{\boldsymbol{w}}$ and $\boldsymbol{b}$ at which L is **minimum.**

Linear Regression: Loss Minimization

- Given a function $f(x)$, how to get the x value at which $f(x)$ is **minimum** ?
- In general, one should get the x-value at which the derivative (differentiation) of $f(x)$ with respect to x is **equal to zero**, such that,

$$\frac{d(f(x))}{dx} = 0$$





Linear Regression: Loss Minimization

$$L(\vec{\boldsymbol{w}}, \boldsymbol{b}) = \frac{1}{N}\sum_{i=1}^{N}(\boldsymbol{y}_i - (\vec{\boldsymbol{w}}\,\vec{\boldsymbol{x}_i} + \boldsymbol{b}))^2$$

- Get the values of $\vec{\boldsymbol{w}}, \boldsymbol{b}$ at which $L(\vec{\boldsymbol{w}}, \boldsymbol{b})$ is minimum → Get the values of $\vec{\boldsymbol{w}}, \boldsymbol{b}$ at which $\frac{d(L)}{d\vec{\boldsymbol{w}}} = 0$, and $\frac{d(L)}{d\boldsymbol{b}} = 0$.
- For linear regression, $\frac{d(L)}{d\vec{\boldsymbol{w}}} = 0$ and $\frac{d(L)}{d\boldsymbol{b}} = 0$ both have **closed-form** solutions that can be derived by making $\vec{\boldsymbol{w}}$ and $\boldsymbol{b}$ the subjects of their equations.
- In this case, the closed-form solution corresponds to:

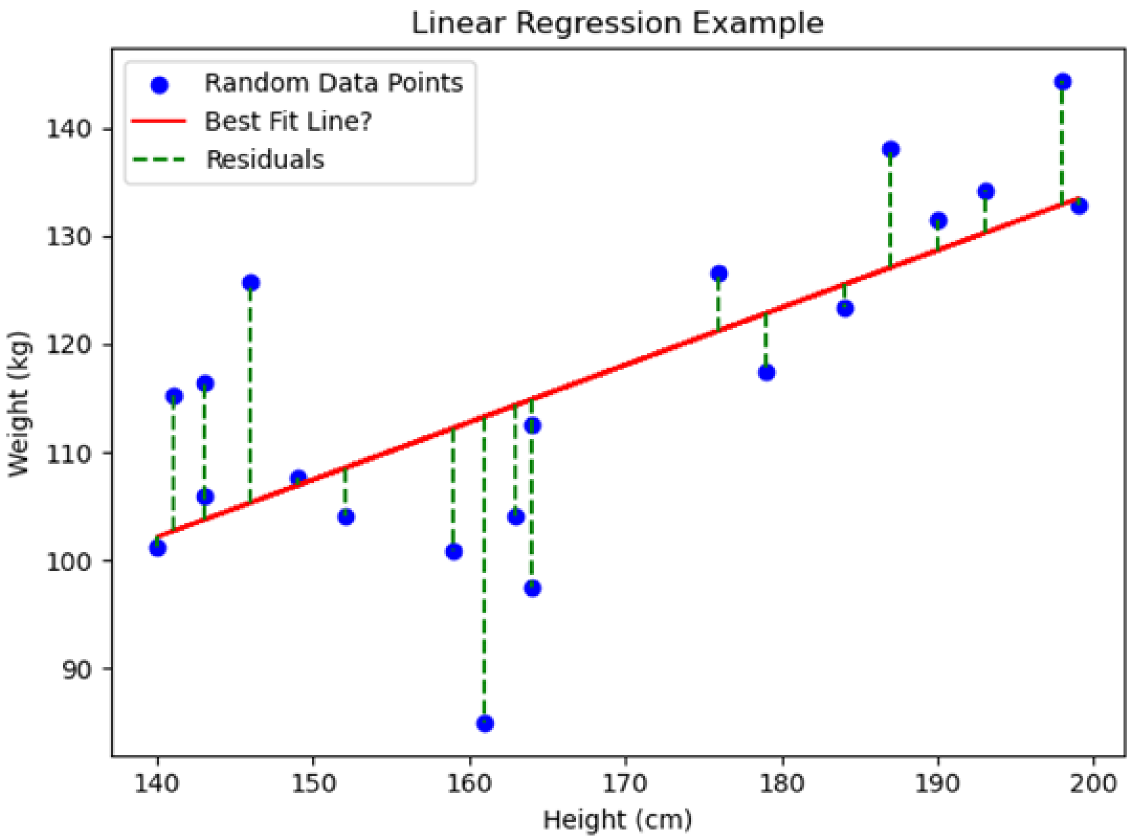
$$\begin{bmatrix} \vec{\boldsymbol{w}} \\ \boldsymbol{b} \end{bmatrix} = \left(X^T X\right)^{-1} X^T y$$

- X is a matrix where each row represents a data point and each column represents a feature, y is a vector of needed output.

Linear Regression: Loss Minimization

- There are other machine learning algorithms for which their cost functions don't have a closed-form solution.
- In other words, we cannot set $\vec{\boldsymbol{w}}$ and $\boldsymbol{b}$ the subject of their equations $\frac{d(L)}{d\vec{\boldsymbol{w}}} = 0$ and $\frac{d(L)}{d\boldsymbol{b}} = 0$, respectively.
- For that reason, we utilize iterative optimization approaches like the famous ***Gradient Descent*** algorithm.

Linear Regression: Solution



$$\hat{y} = w_1\, x + b$$

$$w = 0.533\,, b = 27.94$$

Linear Regression: Single vs Multiple Variables

Linear Regression: Single Variable

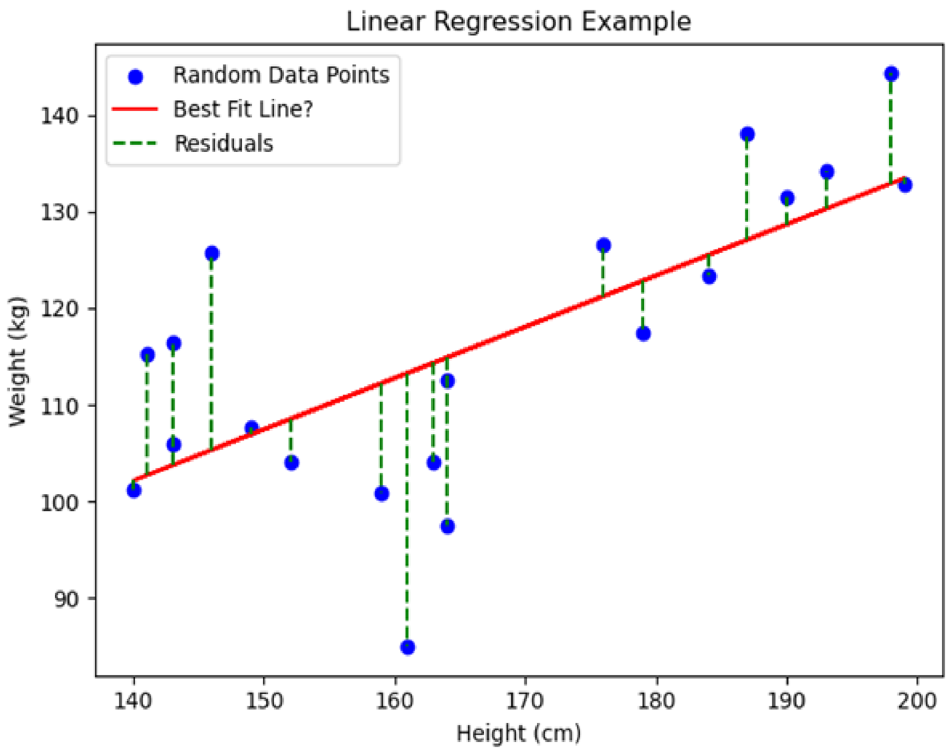
$$\underbrace{\boxed{\hat{y}}}_{\text{Predicted output}} = \underbrace{\beta_0 + \beta_1}_{\text{Coefficients}} \underbrace{\boxed{x}}_{\text{Input}}$$

Linear Regression: Multiple Variables

$$\boxed{\hat{y}} = \beta_0 + \beta_1 \boxed{x_1} + \ldots + \beta_p \boxed{x_p}$$

Evaluation Metrics for Regression Models

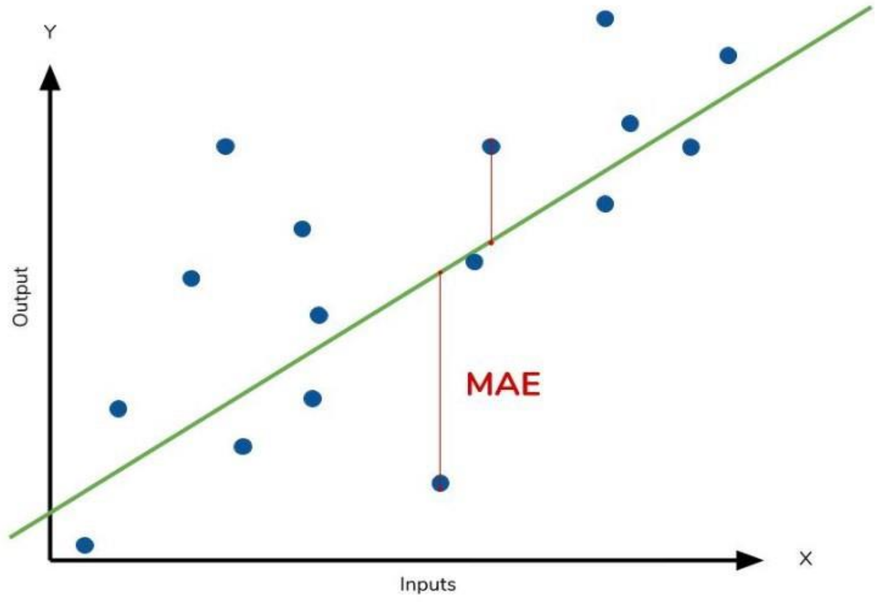
- Difference between the actual value and the model's estimate a **residual or error.**
- Evaluation metrics are measurements that take our collection of residuals and condense them into a *single* value that represents the predictive ability of our model.
  - Mean Absolute Error (MAE)
  - Mean Square Error (MSE)
  - Mean Absolute Percentage Error (MAPE)
  - Mean Percentage Error (MPE)



Mean Absolute Error

- Formulation:
  - $MAE = \frac{1}{N}\sum_{i=1}^{N}|y_i - \hat{y}_i|$

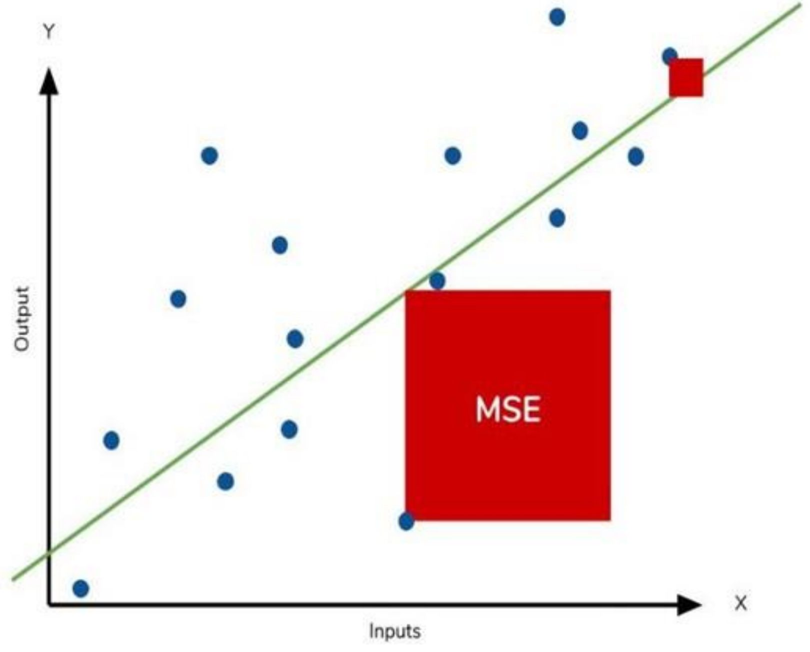
Pros	Cons
• Easy to understand and interpret • Not sensitive to outliers, as it treats all errors equally	• Doesn't punish large errors as much as MSE, which may be a drawback if you want to heavily penalize outliers.



Mean Squared Error

- Formulation:
  - $MSE = \frac{1}{N}\sum_{i=1}^{N}(y_i - \hat{y}_i)^2$

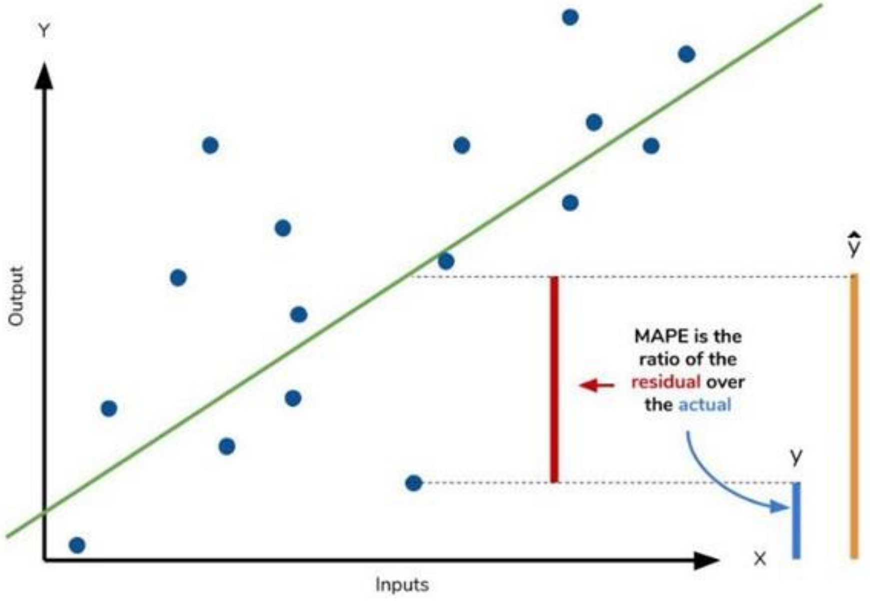
Pros	Cons
• It is differentiable, making it possible to reach closed-form solutions.	• Sensitive to outliers and gives more weight to larger errors.



Mean Absolute Percentage Error

- Formulation:
  - $MAPE = \frac{1}{N}\sum_{i=1}^{N}\left|\frac{y_i - \hat{y}_i}{y_i}\right| * 100$

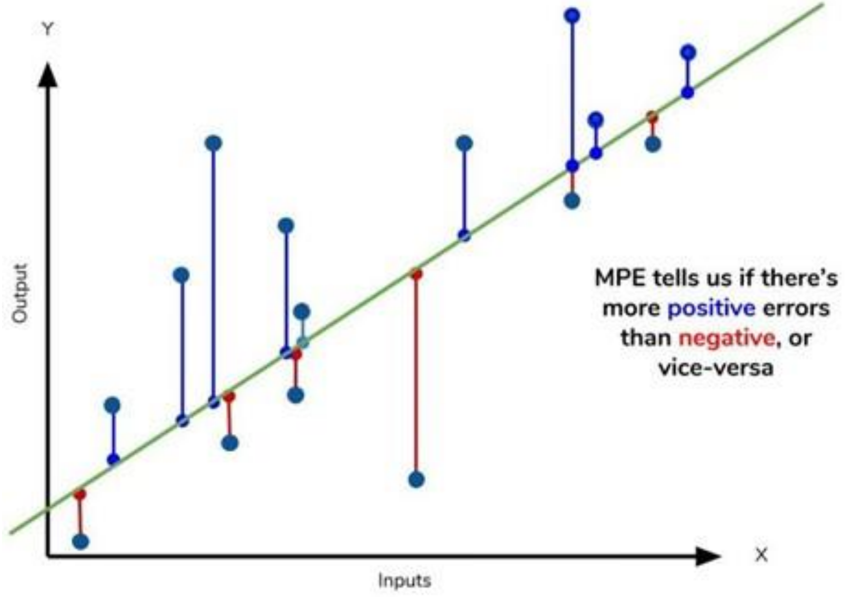
Pros	Cons
• Expresses errors as a percentage of the actual values, which can be more intuitive • Gives an idea of the relative size of the error.	• Problematic when actual values are close to zero



Mean Percentage Error

- Formulation:
  - $MPE = \frac{1}{N}\sum_{i=1}^{N}\frac{y_i - \hat{y}_i}{y_i} * 100$

Pros	Cons
• It gives a sense of the direction (overestimation or underestimation) of the errors.	• Problematic when actual values are close to zero



Summary

Acroynm	Full Name	Residual Operation?	Robust To Outliers?
MAE	Mean Absolute Error	Absolute Value	Yes
MSE	Mean Squared Error	Square	No
RMSE	Root Mean Squared Error	Square	No
MAPE	Mean Absolute Percentage Error	Absolute Value	Yes
MPE	Mean Percentage Error	N/A	Yes

References

- https://learning.oreilly.com/library/view/practical-statistics-for/9781491952955/ch06.html
- https://www.mathsisfun.com/data/standard-deviation.html

Thank you!

- Any questions?

Disclaimer

Due to nature of the course, various materials have compiled from different open source resources with some moderation. I sincerely acknowledge their hard work and contribution

Thank You

Youssef Abdelkareem

yabdelkareem@conestogac.on.ca

** Credits for Dr. Anasuya Bhima for assisting in the slides structure*