



Supervised learning – Regression(2)



Parcours Progis

Etudes, Medias, communication, Marketing

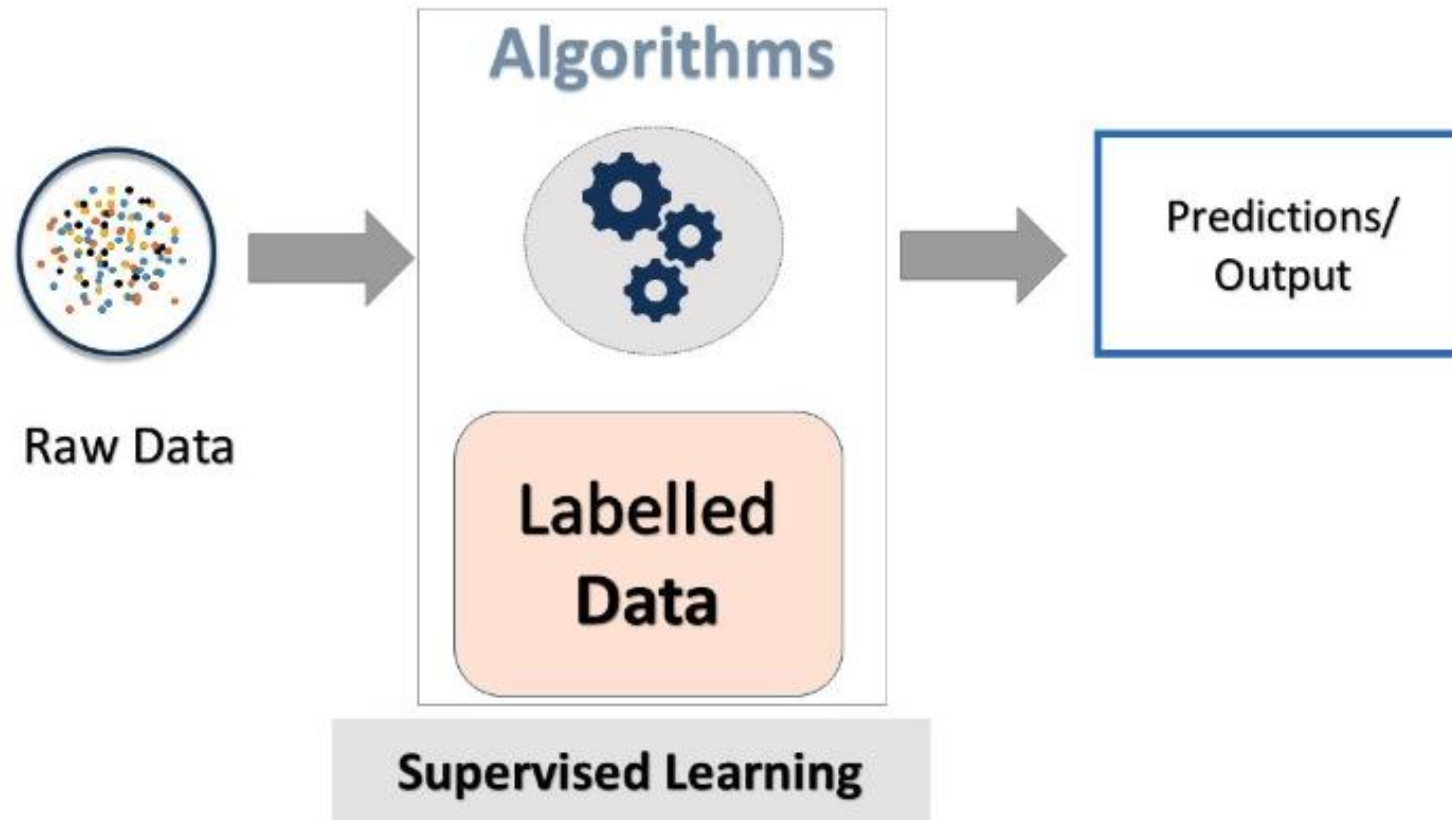
Bahareh Afshinpour

18.11.2024

References

- **Book:** Hwang, Yoon. 2019. *Hands-On Data Science for Marketing*. Packt Publishing, Ltd. Chapter 9, 473–475.
- <https://www.freecodecamp.org/news/machine-learning-mean-squared-error-regression-line-c7dde9a26b93/>
- <https://www.youtube.com/watch?v=pR-Of1ua6Dc>

Supervised learning



Supervised Learning



Regression



What will be the temperature tomorrow?

84°



Fahrenheit

Classification



Will it be hot or cold tomorrow?

COLD

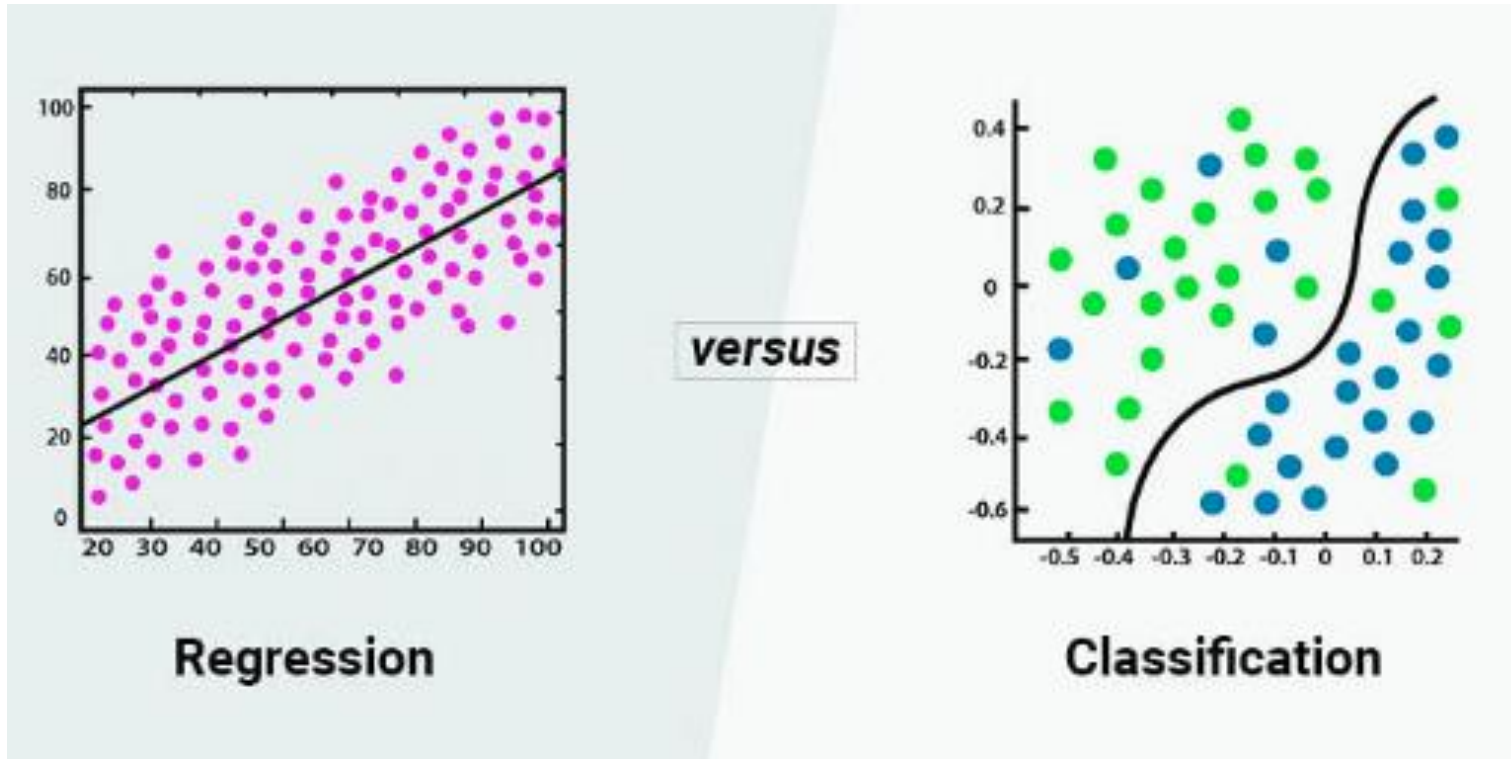
HOT



Fahrenheit

<https://www.enjoyalgorithms.com/blogs/classification-and-regression-in-machine-learning>

Supervised learning



<https://www.simplilearn.com/regression-vs-classification-in-machine-learning-article>

What is a machine learning Model?

- A machine learning model is a program that can find patterns or make decisions from a previously unseen dataset.



Regression

- If you are faced with a prediction problem linear regression should really be the first thing that you need to try.

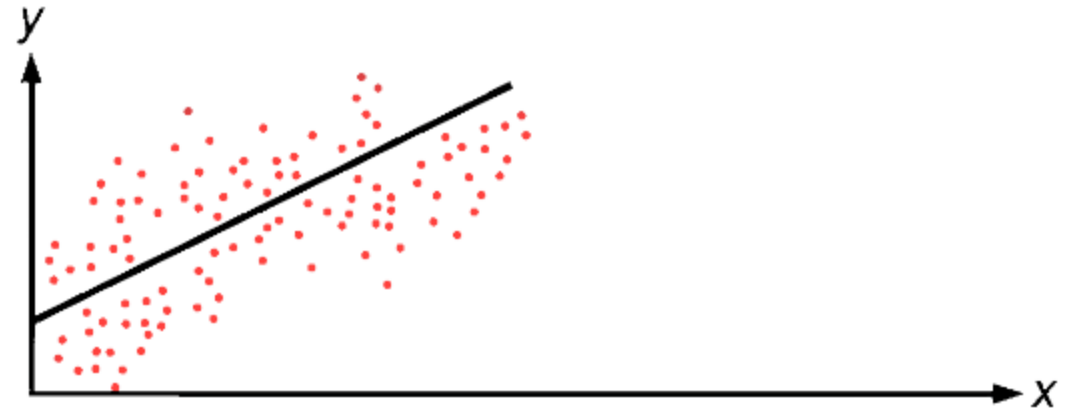
We need to find the right line(function g)

$$g(x)=Ax+B$$

A is slop

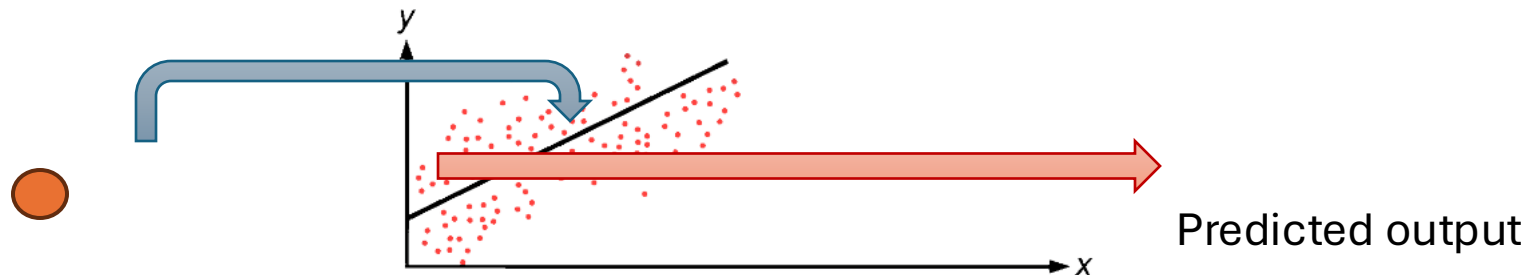
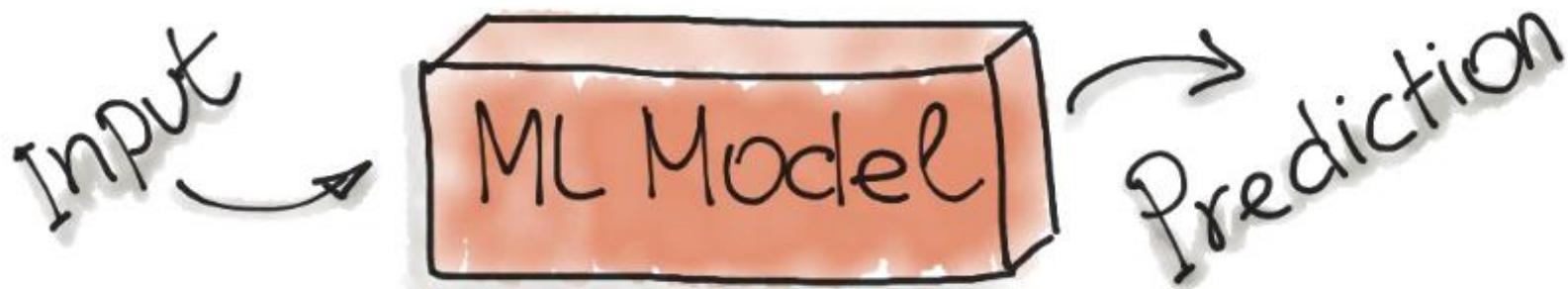
B is intercept

The unknown parameters A and B must be estimated based on historical data



- Import sklearn
- Declare regression model
- Fit it to training data (fit function)
- To get prediction we should use predict function

What is a machine learning Model?



New point(new data/unseen data)

Evaluating Regression Models

- This makes it easy to show your model to other people and helps you understand how well your model works.
- Unlike classification, accuracy in a regression model is slightly harder to illustrate. (output of a regression model is continuous value)
 - It is impossible for you to predict the exact value
 - You can find how close your prediction is against the real value
- we are going to discuss four commonly used methodologies to evaluate the models:
 - mean square error, mean absolute error, R square, and predicted versus actual scatter plot.

Mean squared error

A function that measures how well a predicted value \hat{Y} matches some ground-truth value Y .

Sum of error of all points

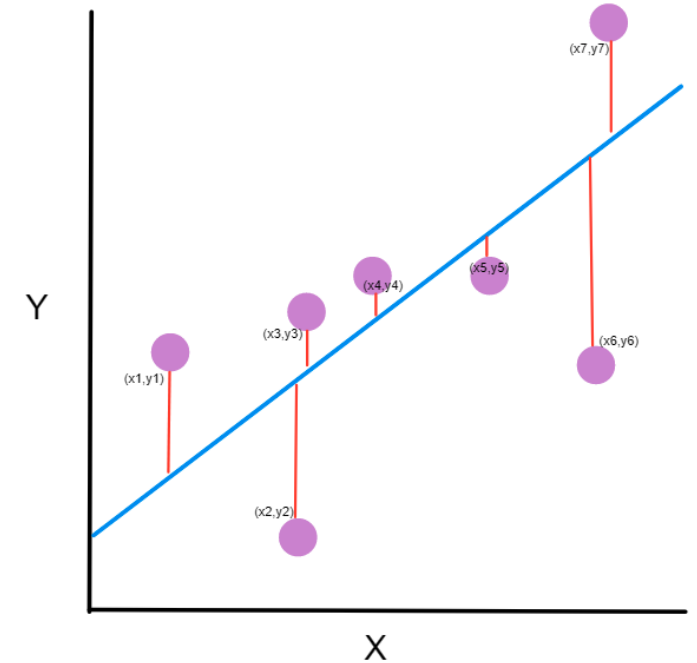
$$MSE = \frac{1}{n} \sum_{i=1}^n (y_i - \tilde{y}_i)^2$$

the mean

summation

Real value= The y-coordinate is our purple dot

Predicted value= The y' point sits on the line we created.



Our goal is to minimize this mean, which will provide us with the best line that goes through all the points.

Mean absolute error (MAE)

- Sometimes MSE values can be too big to compare easily.
- Mae is less sensitive to outliers .

$$\text{MAE} = \frac{\sum_{i=1}^n |y - \hat{y}_i|}{n}$$

summation of all values
(with i ranging from 1
to n)

this operator gives the
absolute value of a
number

No. of data
points

y = actual value, \hat{y} = predicted value

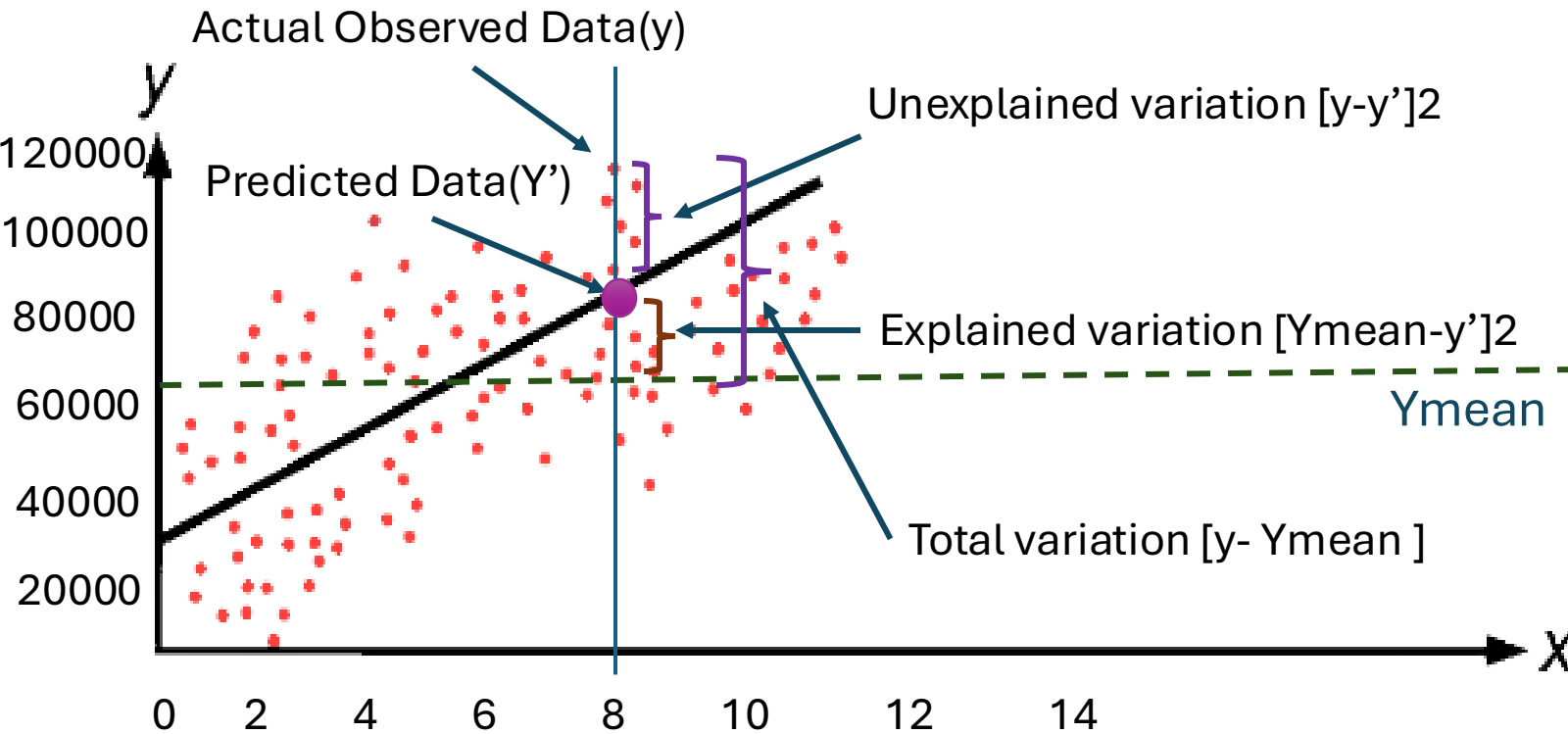
<https://www.mage.ai/blog/model-evaluation-mean-absolute-error-MAE>

Coefficient of determination r^2

- R square measures the goodness of fit, in other words,
 - how well a regression model is fitted to the data.
- R square typically ranges between zero and one.
 - $R=0$, the model does not explain or capture the target variability at all
 - $R=1$, model is a perfect fit to the data.
- The closer to one the R square value is, the better the model fitted.

Coefficient of determination (R^2)	Interpretation
0	The model does not predict the outcome.
Between 0 and 1	The model partially predicts the outcome.
1	The model perfectly predicts the outcome.

Coefficient of determination r^2



$$R^2 = \frac{SSR}{SST} = \frac{[y_{mean} - \hat{y}]^2}{[y - y_{mean}]^2}$$

Coefficient of determination r^2

```
from sklearn.linear_model import LinearRegression

#initiate linear regression model
model = LinearRegression()

#define predictor and response variables
X, y = df[["hours", "prep_exams"]], df.score

#fit regression model
model.fit(X, y)

#calculate R-squared of regression model
r_squared = model.score(X, y)

#view R-squared value
print(r_squared)

0.7175541714105901
```

Advantages and Disadvantages of Linear Regression

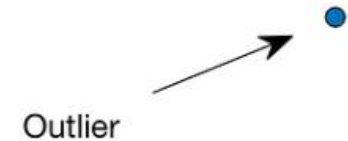
Advantages	Disadvantages
Simple Implementation	Sensitive to Outliers
Regression models are easy to understand	linear or near linear relationship
	As the number of variables increases the reliability of the regression models decreases. The regression models work better if you have a small number of variables.

Linear regression model has two primary limitations.

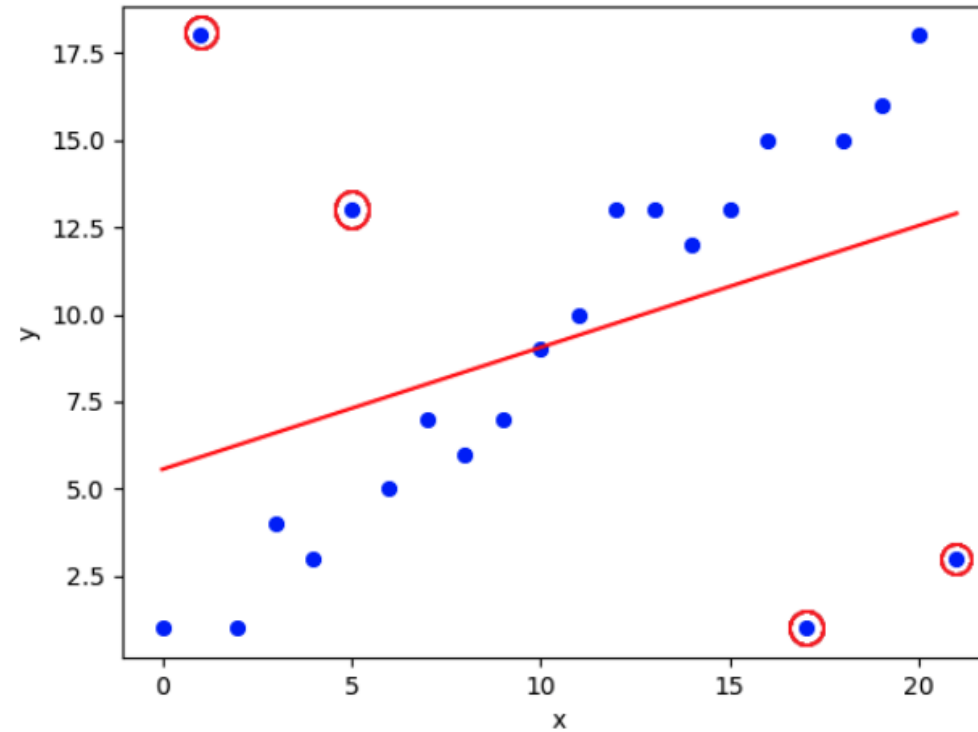
- **First, it assumes that there is a *linear* relationship between inputs and outputs, which is not always the case in real-world data.**
- **Second, the model's performance is significantly impacted by the presence of *outliers***

What is a outlier?

- An observation that differs significantly from the other observations in a dataset is considered an outlier.
- An outlier is therefore considerably larger or smaller than the other values in the collection.
- **Reasons for the occurrence of Outliers**
 - Error in data entry.
 - Errors caused during measurement....



What is a outlier?



How do you handle outliers in linear regression?

- **When to drop an outlier?**
 - When we know for sure that the outlier is completely wrong.
 - When we have large amount of data.
- **What to do with the undroppable outliers?**
 - **Imputation:** We can replace the outlier values with the mean, median or mode value based on the use case.
 - **Quantile-based Flooring and Capping:** In this technique, we can do the flooring (e.g., replacing with the 10th percentile) for the lower values and capping (e.g., replacing with the 90th percentile) for the higher values.

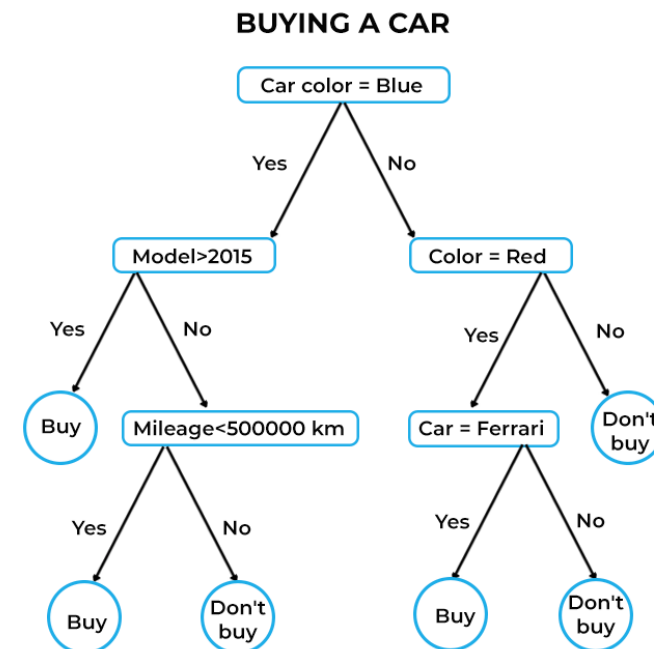
How do you handle outliers in linear regression?

- Outliers can be handled by **removing** them from the dataset, transforming the data, or using robust regression methods that are less sensitive to outliers.

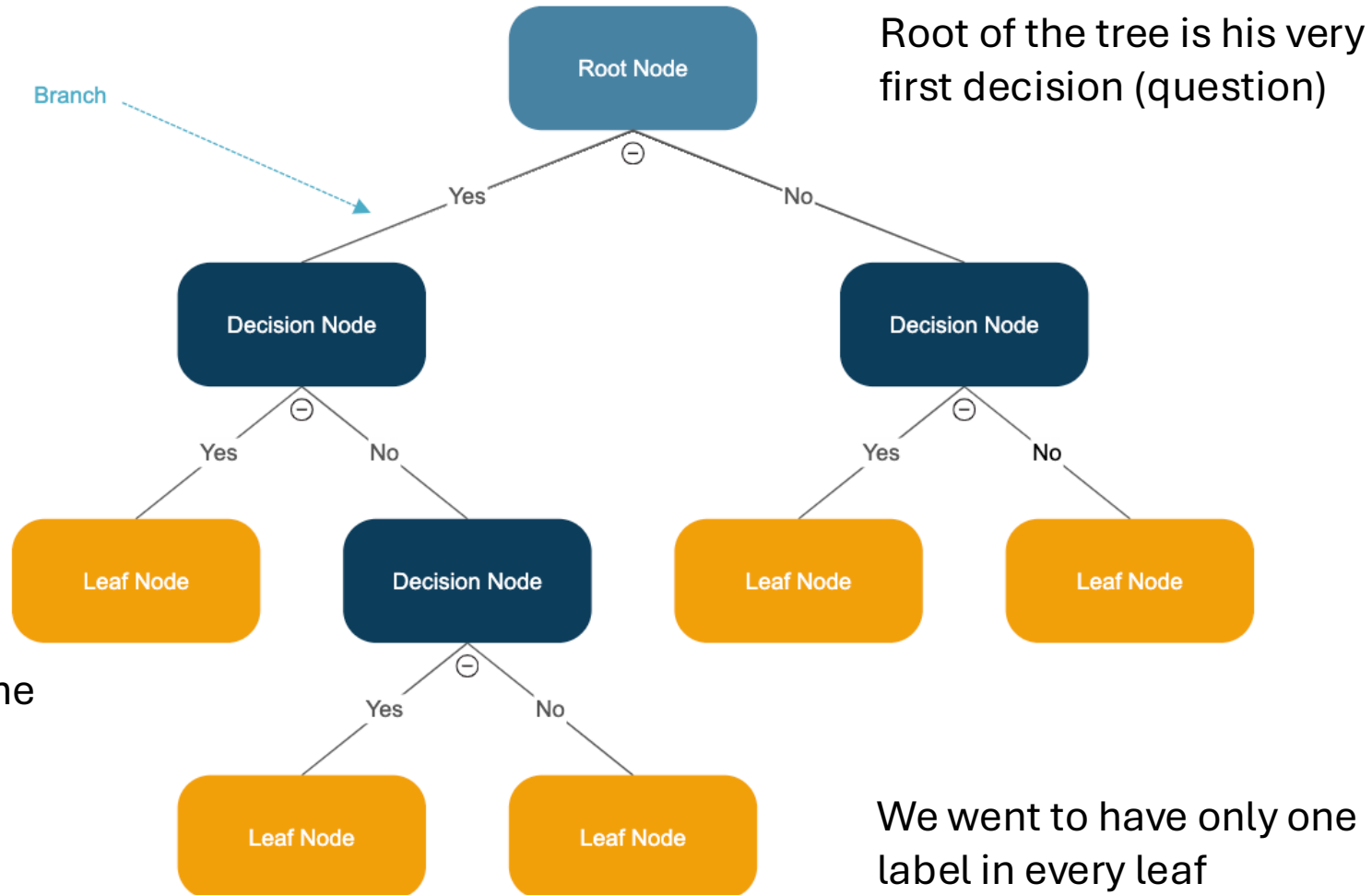
Decision Trees

- Decision trees are easy to learn as they are powerful
- A fundamental supervised learning algorithm
- How decision trees learn from data
 - They recursively split data into the largest, **purest** groups (all examples have the same label)

We ask a question, then based on the answer to the question, we move down to the tree.



Decision Tree



Decision Tree

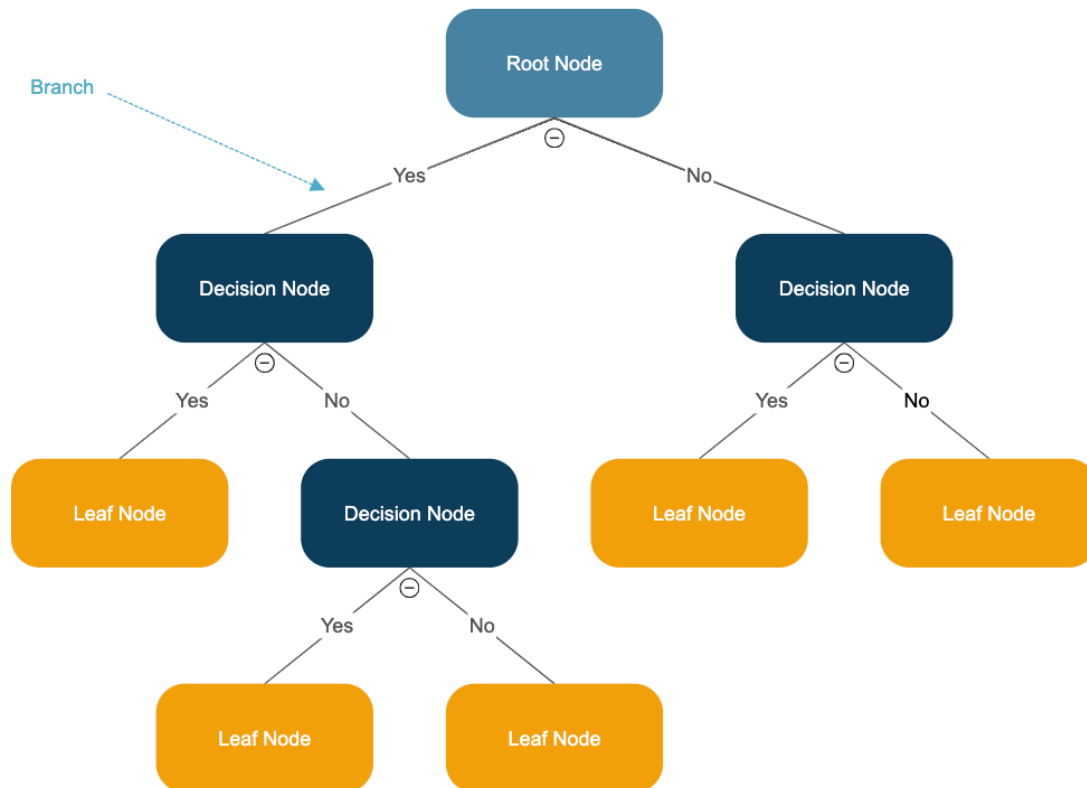
- A leaf node is 100% pure :
 - every row of data that flows down the tree and ends up in a particular node have a same label
- decision tree models learn the data by partitioning the data points based on certain criteria.
- Using either of these measures, decision trees can grow until all of the nodes are pure or until the stopping criteria are met.

Decision Tree Regression

- The goal of using a decision tree as you create a training model that can be used to predict the class or value of the target variable by learning simple decision rules inferred from prior data commonly referred to as the training data.
- we start from the root of the tree. We compare the values of the root attribute with the records attribute. On the basis of this comparison, we follow the branch corresponding to that value and jump to the next node.

Decision Tree- Parent and child node

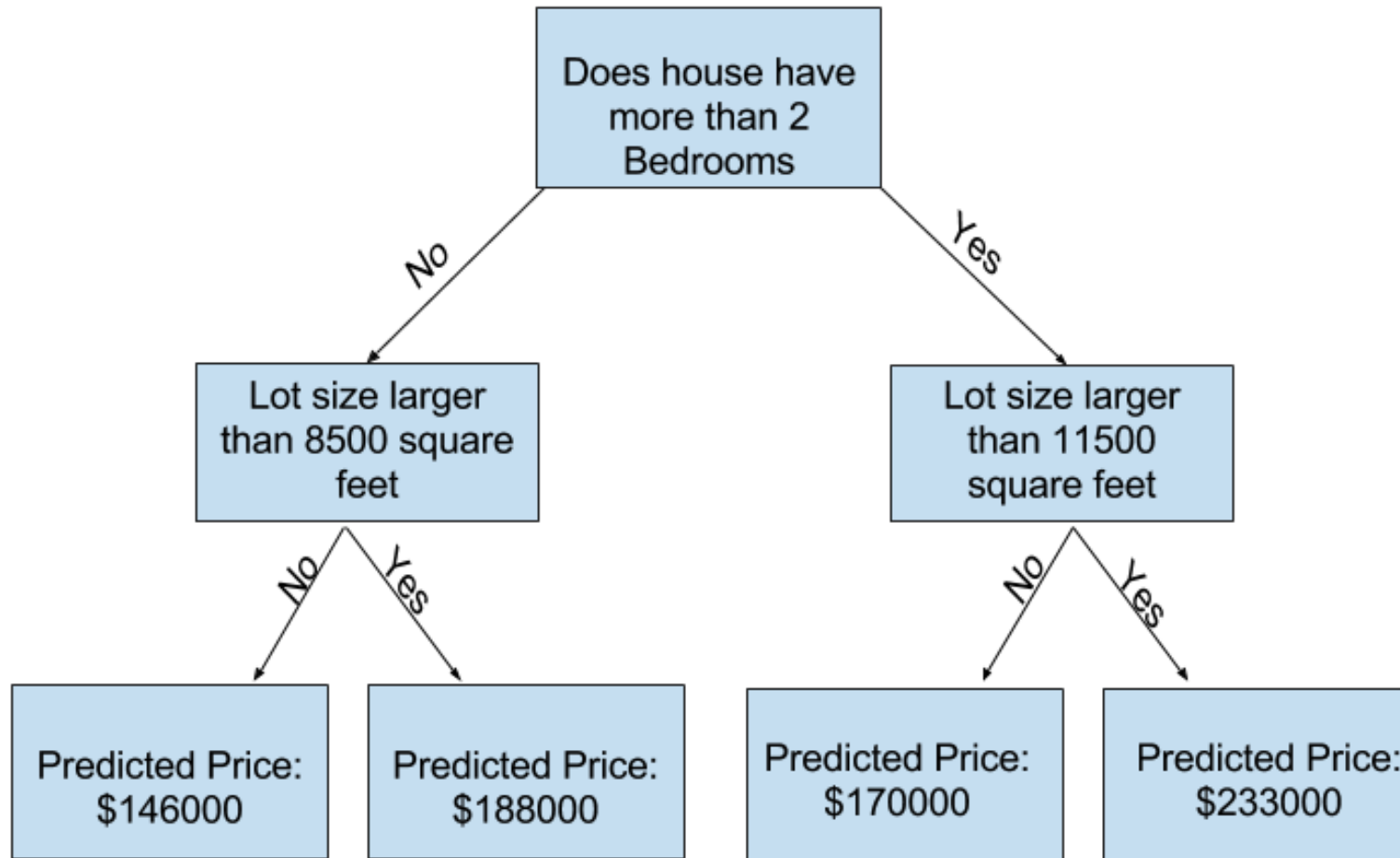
- A node which is divided into sub nodes is called a **parent** node of the sub. Nodes whereas the sub nodes are the **child** node of that particular parent.



Decision Tree Regression

- Decision trees as the name suggests learn from the data by growing a tree.
- The main difference between the logistic regression and decision tree model is the fact that logistic regression algorithms search for a single best linear boundary in the feature set, whereas the decision tree algorithm **partitions** the data to **find the subgroups** of data that have **high likelihood** of an event occurring.
- Decision tree models perform better for **non-linear** datasets.

Example of Decision Tree



Working of the algorithm

- **Building the Tree:**

- Imagine you have a dataset with various attributes (features) and their corresponding target values. The algorithm begins by creating a root node that represents the entire dataset.

- **Splitting the Data:**

- The algorithm analyzes each feature to determine the best way to divide the data into distinct groups based on their target values. It does this by setting specific conditions or thresholds on the feature values.

- **Recursive Splitting:**

- The algorithm then repeats this process for each child node, recursively splitting the data further based on the best features and conditions.

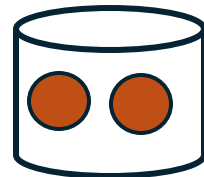
Example of Decision Tree-visual representation

Target variable

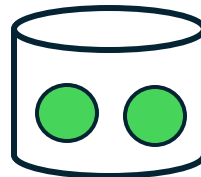
Age	Education	Marital status	Race	Sex	Hours Per Week	Label
61	master	married	White	Male	40	<=50k
48	PhD	divorse	White	Female	16	<=50
55	PhD	married	Black	Male	45	>50 k
30	master	Never married	Black	Female	50	>50 k

Which of these columns(features) best splits these labels into the largest purest buckets?

We have two rows less that 50k and two more than 50k



No



Yes





Feature x

<=50k 

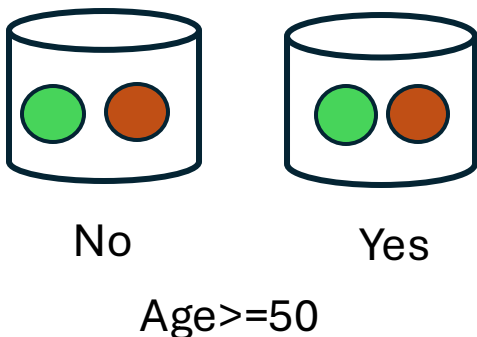
>50 k 

Example of Decision Tree-visual representation

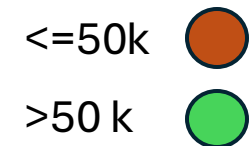
Target variable

Age	Education	Marital status	Race	Sex	Hours Per Week	Label
61	master	married	White	Male	40	<=50k 
48	PhD	divorse	White	Female	16	<=50k 
55	PhD	married	Black	Male	45	>50 k 
30	master	Never married	Black	Female	50	>50 k 

First we look at age feature:



We have impurity.
 Age creates a 50/50 split.
 We are completely uncertain of its effect on salary.
 This feature does not really help us in making a prediction



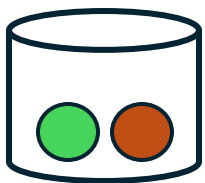
Example of Decision Tree-visual representation

Target variable

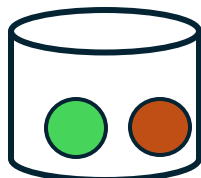
Age	Education	Marital status	Race	Sex	Hours Per Week	Label
61	master	married	White	Male	40	<=50k
48	PhD	divorse	White	Female	16	<=50k
55	PhD	married	Black	Male	45	>50 k
30	master	Never married	Black	Female	50	>50 k

move on the education columns

50/50. Education won't help us make prediction.

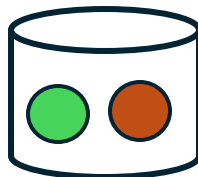


No

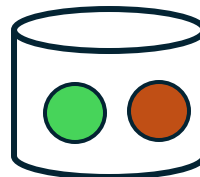


Yes

Age >= 50



No



Yes

Education = PhD

<=50k 

>50 k 

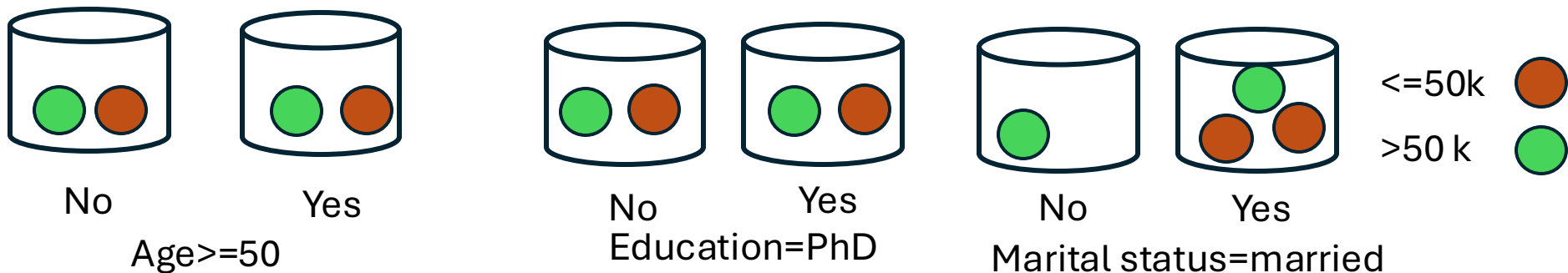
Example of Decision Tree-visual representation

Target variable

Age	Education	Marital status	Race	Sex	Hours Per Week	Label
61	master	married	White	Male	40	<=50k
48	PhD	divorse	White	Female	16	<=50k
55	PhD	married	Black	Male	45	>50 k
30	master	Never married	Black	Female	50	>50 k

move on the Marital status

At least one is pure. It does not offer a clean split either



Example of Decision Tree-visual representation

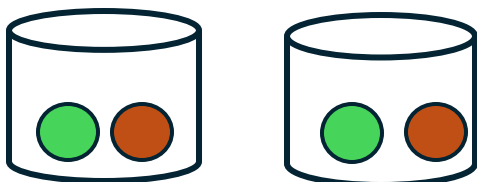
Target variable

Age	Education	Marital status	Race	Sex	Hours Per Week	Label
61	master	married	White	Male	40	<=50k
48	PhD	divorse	White	Female	16	<=50k
55	PhD	married	Black	Male	45	>50 k
30	master	Never married	Black	Female	50	>50 k



move on Race

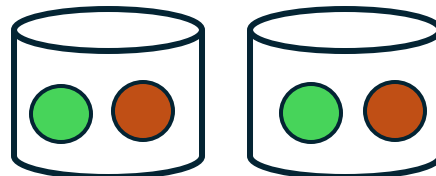
100% pure



No

Yes

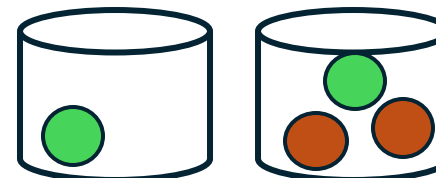
Age >= 50



No

Yes

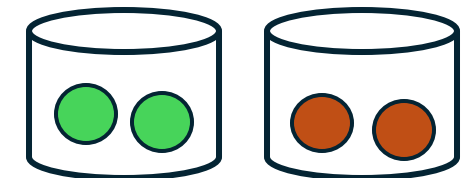
Education = PhD



No

Yes

Marital status = married



Race = Black

Decision Tree

- We can continue to check other features.
- In this example you can see the feature hours per week is also 100% pure.
- But between race and hours per week, which one is the best one.
- Decision trees is known as a **greedy algorithm**. And it picks the very **first feature** that it finds that is the best.
- So, in this example, at the top of the tree should use race=black as a root node. (first decision in the tree)

End