Machine Learning
Dmytro Fishman (dmytro.fishman@ut.ee)
Find examples of AI in action at [https://deepindex.org/](https://deepindex.org/)
AlphaZero

AlphaStar

From https://deepmind.com/blog/article/alphastar-mastering-real-time-strategy-game-starcraft-ii
An animation of the gradient descent method predicting a structure for CASP13 target T1008
Diabetic Retinopathy

NORMAL VISION
Vision remains intact
Diabetic Retinopathy

NORMAL VISION
Vision remains intact

DIABETIC RETINOPATHY
Vision is obstructed by macular edema
Diabetic Retinopathy

NORMAL
Vision is clear

DIABETIC RETINOPATHY
Vision is obstructed by macular edema
Diabetic Retinopathy
Living Portraits
Which face is real?

http://www.whichfaceisreal.com
Which face is real?
http://www.whichfaceisreal.com
Towards a Human-like Open-Domain Chatbot

**Common sense:**

Q: What is your favorite animal?
A: My favorite animal is a dog.

Q: Why?
A: Because dogs are loyal and friendly.

Q: What are two reasons that a dog might be in a bad mood?
A: Two reasons that a dog might be in a bad mood are if it is hungry or if it is hot.

Q: How many eyes does a giraffe have?
A: A giraffe has two eyes.

Q: How many legs does a frog have?
A: A frog has four legs.

Q: Are there any animals with three legs?
A: No, there are no animals with three legs.

Q: Why don't animals have three legs?
A: Animals don't have three legs because they would fall over.

**Non-sense:**

Q: How do you sporgle a morgle?
A: You sporgle a morgle by using a sporgle.

Q: How many bonks are in a quoit?
A: There are three bonks in a quoit.

Q: How many rainbows does it take to jump from Hawaii to seventeen?
A: It takes two rainbows to jump from Hawaii to seventeen.

Q: Which colorless green ideas sleep furiously?
A: Ideas that are colorless, green, and sleep furiously are the ideas of a sleep furiously.

Q: Do you understand these questions?
A: I understand these questions.

More examples: https://gpt3examples.com/
Machine Learning
Machine Learning
Machine Learning

Supervised Learning

Unsupervised Learning
Machine Learning

- Supervised Learning
- Unsupervised Learning
- Reinforcement Learning
Supervised Learning

Machine Learning

Unsupervised Learning

Reinforcement Learning
Supervised Learning

Unsupervised Learning

Reinforcement Learning

Classification

Regression

Machine Learning
Machine Learning

Reinforcement Learning

Unsupervised Learning

Supervised Learning

Classification

Regression

Zeros

Ones

Zeros

Ones
Binary Classification

Classification

Zeros

Ones

Supervised Learning

Regression

Reinforcement Learning
Classification

Supervised Learning

Regression

Machine Learning

Unsupervised Learning

Reinforcement Learning

Binary Classification

Zeros

Ones

Multiclass Classification

Classification

Regression

Supervised Learning

Unsupervised Learning

Reinforcement Learning
Machine Learning
Reinforcement Learning
Unsupervised Learning
Classiﬁcation
Regression
Supervised Learning
Binary Classiﬁcation
Zeros
Ones
Multiclass Classiﬁcation

Supervised Learning
Regression
Reinforcement Learning
Machine Learning

Reinforcement Learning

Unsupervised Learning

Supervised Learning

Regression

Classification
Machine Learning

Supervised Learning

Regression

Classification

Unsupervised Learning

Reinforcement Learning

7$

3.5$

5$
Supervised Learning

Regression

Predicting continuous values
What is the main similarity/difference between these two classes of *supervised learning*?
Machine Learning

- Classification
- Regression

Supervised Learning

Unsupervised Learning

Reinforcement Learning
Machine Learning

- Classification
- Regression
- Supervised Learning
- Reinforcement Learning

Unsupervised Learning
Unsupervised Learning

Supervised Learning

Classification

Regression

Machine Learning

Reinforcement Learning

Unsupervised Learning
Machine Learning

- Classification
- Regression
- Reinforcement Learning
- Supervised Learning
- Unsupervised Learning
- Dimensionality reduction
High Dimensional Space

Dimensionality reduction
Unsupervised Learning

Supervised Learning

Classification
Regression

Reinforcement Learning

Image credit: https://mathematica.stackexchange.com/questions/39879/create-a-torus-with-a-hexagonal-mesh-for-3d-printing
High Dimensional Space

Dimensionality reduction

Unsupervised Learning

Supervised Learning

Reinforcement Learning

Image credit: https://mathematica.stackexchange.com/questions/39879/create-a-torus-with-a-hexagonal-mesh-for-3d-printing
High Dimensional Space

Low Dimensional Space

Dimensionality reduction

Unsupervised Learning

Torus image credit: https://mathematica.stackexchange.com/questions/39879/create-a-torus-with-a-hexagonal-mesh-for-3d-printing
High Dimensional Space

Dimensionality reduction

Low Dimensional Space

Dimensionality reduction

Torus image credit: https://mathematica.stackexchange.com/questions/39879/create-a-torus-with-a-hexagonal-mesh-for-3d-printing
Machine Learning

- Supervised Learning
- Regression
- Classification
- Reinforcement Learning

Unsupervised Learning

Dimensionality reduction
Unlabelled data

Unsupervised Learning

Clustering

Supervised Learning

Classification

Regression

Reinforcement Learning

Dimensionality reduction
Unlabelled data

Clustering algorithm

Unsupervised Learning

Clustering
Machine Learning

- Supervised Learning
  - Classification
  - Regression

- Reinforcement Learning

- Dimensionality reduction

- Unsupervised Learning
  - Clustering
Dimensionality reduction

Classification

Supervised Learning

Regression

Reinforcement Learning

Unsupervised Learning

Clustering

Machine Learning

0 <-> 1

? $
What is the main similarity/difference between **classification** and **clustering**?
Dimensionality reduction

Machine Learning

- Supervised Learning
  - Classification
  - Regression
- Unsupervised Learning
  - Clustering
  - Reinforcement Learning

0 ?→ 1
Reinforcement Learning
Unsupervised Learning

Dimensionality reduction

Clustering

Machine Learning

Reinforcement Learning

Classification

Regression

Supervised Learning

Inspired by https://keon.io/deep-q-learning/
Unsupervised Learning

Classification

Supervised Learning

Dimensionality reduction

Clustering

Regression

Reinforcement Learning
Supervised Learning

Lecture 1
Unsupervised Learning

Dimensionality reduction

Clustering

Supervised Learning

Classification

Regression

Machine Learning

Reinforcement Learning
new digit
Zero

\[
\begin{array}{cccc}
0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 \\
\text{new digit} & ? & 0 & 0
\end{array}
\]

One

\[
\begin{array}{cccc}
1 & 1 & 1 & 1 \\
1 & 1 & ? & 1
\end{array}
\]

height
If we would be forced to decide now, what would be the class of the new digit?
If we would be forced to decide now, what would be the class of the new digit?
If we would be forced to decide now, what would be the class of the new digit?
Classification

Based on previously recorded data

height
width

X

y
Based on previously recorded data

classify newly arrived example
Nearest Neighbour Classifier
Nearest Neighbour Classifier

infer a class of newly added example from the class of the nearest neighbour
Nearest Neighbour Classifier

infer a class of newly added example from the class of the nearest neighbour
Nearest Neighbour Classifier

Nearest neighbour is found by calculating distances to all existing examples.
Nearest Neighbour Classifier

The **nearest neighbour** is found by calculating distances to all existing examples.
The nearest neighbour classifier is found by calculating distances to all existing examples.
Nearest Neighbour Classifier

Nearest neighbour is found by calculating distances to all existing examples.
Nearest Neighbour Classifier

The nearest neighbour is found by calculating distances to all existing examples.
Nearest Neighbour Classifier

The nearest neighbour is found by calculating distances to all existing examples.

\[(x_1, y_1)\]
Nearest Neighbour Classifier

Nearest neighbour is found by calculating Euclidean distances to all existing examples.
Nearest Neighbour Classifier

Euclidean distance

\[ d = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2} \]

Nearest neighbour is found by calculating distances to all existing examples.
Nearest Neighbour Classifier

Euclidean distance

\[ d = \sqrt{(2 - x_1)^2 + (3 - y_1)^2} \]

**nearest neighbour** is found by calculating distances to **all existing examples**.
Nearest Neighbour Classifier

Nearest neighbour is found by calculating distances to all existing examples.

Euclidean distance:

\[ d = \sqrt{(2 - 3)^2 + (3 - 3)^2} \]
Nearest Neighbour Classifier

Nearest neighbour is found by calculating distances to all existing examples.

Euclidean distance

\[ d = \sqrt{(2 - 3)^2 + (3 - 3)^2} = 1 \]
Nearest Neighbour Classifier

Euclidean distance

\[ d = \sqrt{(2 - 3)^2 + (3 - 3)^2} = 1 \]

-nearest neighbour is found by calculating distances to all existing examples.
Nearest Neighbour Classifier

The nearest neighbour is found by calculating distances to all existing examples.
Nearest Neighbour Classifier

Nearest neighbour is found by calculating distances to all existing examples.
Nearest Neighbour Classifier

Nearest neighbour is found by calculating distances to all existing examples.
Nearest Neighbour Classifier

Nearest neighbour is found by calculating distances to all existing examples.
Nearest Neighbour Classifier

The **nearest neighbour** is found by calculating distances to all existing examples.
Nearest Neighbour Classifier

The nearest neighbour is found by calculating distances to all existing examples.
Nearest Neighbour Classifier

Nearest neighbour is found by calculating distances to all existing examples. The nearest neighbour is used to infer a class of newly added example from the class of the nearest neighbour.
Nearest Neighbour Classifier

The nearest neighbour classifier is found by calculating distances to all existing examples. The nearest neighbour is the one with the smallest distance, and the class of this nearest neighbour is assigned to the newly added example.

Infer a class of newly added example from the class of the nearest neighbour.
Nearest Neighbour Classifier

The nearest neighbour is found by calculating distances to all existing examples. The class of the nearest neighbour is then used to infer the class of the newly added example.

Example:
- Calculate distances to all existing examples.
- Identify the nearest neighbour.
- Assign the class of the nearest neighbour to the new example.
Nearest Neighbour Classifier

It turns out that we can use more neighbours, for example $K$. 
Nearest Neighbour Classifier

It turns out that we can use **more neighbours**, for example **K**.
It turns out that we can use more neighbours, for example \( K \)-Nearest Neighbour Classifier.
It turns out that we can use more neighbours, for example $K$.

For example, five!
K-Nearest Neighbour Classifier

It turns out that we can use more neighbours, for example K!

For example, five!
K-Nearest Neighbour Classifier

It turns out that we can use more neighbours, for example K

Which makes the point red again
It turns out that we can use more neighbours, for example $K$

Which makes the point red again
Nearest Neighbour Classifier
Nearest Neighbour Classifier

What if we consider all points?
Nearest Neighbour Classifier

What if we consider all points?
Nearest Neighbour Classifier

What if we consider all points?
Nearest Neighbour Classifier
K-Nearest Neighbour Classifier

What if there are more than two classes?
K-Nearest Neighbour Classifier

What if there are more than two classes?
K-Nearest Neighbour Classifier

What if there are **more than two** classes?
What if there are more than two classes?

New unknown point
What if there are more than two classes?

Let's assume $K = 5$
What if there are more than two classes?

Let's assume $K = 5$.
What if there are more than two classes?

Let's assume $K = 5$

More prevalent class “wins” the vote
What if there are more than two classes?

Let's assume $K = 5$

More prevalent class "wins" the vote.
Machine Learning

Supervised Learning

Regression

Unsupervised Learning

Reinforcement Learning
The following slides are inspired by “An Introduction to Linear Regression Analysis” video

https://youtu.be/zPG4NjlkCjc
How the change in independent variable influences dependent variable?
Linear Regression

Positive relationship

dependent variable

independent variable
linear regression

independent variable

dependent variable

Negative relationship
Linear Regression

$y$

$X$

dependent variable

independent variable

Linear Regression
In order to build a linear regression we need observations.
In order to build a linear regression, we need observations.
Linear Regression

y

dependent variable

independent variable

X
We want to find a line such that ...
We want to find a line such that... it minimises...
We want to find a line such that it minimises the sum of errors.
We want to find a line such that … it minimises the sum of errors.
We want to find a line such that...

... it minimises the sum of errors
We want to find a line such that...

...it minimises the sum of errors
We want to find a line such that ...

... it minimises the sum of errors

$$\arg \min \sum_{i=1}^{n} (y_i - \hat{y}_i)^2$$
We want to find a line such that...

... it minimises the sum of errors

$$\arg \min \sum_{i=1}^{n} (y_i - \hat{y}_i)^2$$
Linear Regression

- Independent variable
- Dependent variable

Image shows a scatter plot with a linear regression line fitted to the data points.
Linear Regression

fare amount

distance

y

X
Linear Regression

\[ \hat{y} = w_0 + w_1 x \]
Linear Regression

\[ \hat{y} = w_0 + w_1 x \]

estimated fare amount
Linear Regression

\[ \hat{y} = w_0 + w_1 x \]

estimated fare amount

distance
Linear Regression

\[ \hat{y} = w_0 + w_1 x \]

estimated fare amount

y intercept

distance

fare amount

X
Linear Regression

\[ \hat{y} = w_0 + w_1 x \]

estimated fare amount

distance

slope

y intercept

distance

fare amount
Linear Regression

\[ \hat{y} = w_0 + w_1 x \]
Linear Regression

\[ \hat{y} = w_0 + w_1 x \]
Linear Regression

The model for linear regression is given by:

\[ \hat{y} = w_0 + w_1 x \]

It minimizes the sum of errors as follows:

\[ \arg \min_{w_0, w_1} \sum_{i=1}^{n} (y_i - \hat{y}_i)^2 \]
Linear Regression

\[ \hat{y} = w_0 + w_1x \]

minimises the sum of errors with respect to \( w_0 \) and \( w_1 \)

\[
\arg \min_{w_0, w_1} \sum_{i=1}^{n} (y_i - \hat{y}_i)^2
\]
Linear Regression (example)
Linear Regression (example)

\[ \hat{y} = 2.2 + 0.6x \]

<table>
<thead>
<tr>
<th></th>
<th>x</th>
<th>y</th>
<th>x - \bar{x}</th>
<th>y - \bar{y}</th>
<th>(x - \bar{x})^2</th>
<th>(x - \bar{x})(y - \bar{y})</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1</td>
<td>2</td>
<td>-2</td>
<td>-2</td>
<td>4</td>
<td>4</td>
</tr>
<tr>
<td>2</td>
<td>2</td>
<td>4</td>
<td>-1</td>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>3</td>
<td>3</td>
<td>5</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>4</td>
<td>4</td>
<td>4</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>5</td>
<td>5</td>
<td>5</td>
<td>2</td>
<td>1</td>
<td>4</td>
<td>2</td>
</tr>
</tbody>
</table>

\[ \bar{x} = 3 \quad \bar{y} = 4 \quad 10 \quad 6 \]

\[ w_1 = \frac{\sum (x - \bar{x})(y - \bar{y})}{\sum (x - \bar{x})^2} = \frac{6}{10} = 0.6 \]

\[ 4 = w_0 + 0.6 \times 3 \]

\[ w_0 = 2.2 \]
Linear Regression (example)

Check out the video for how the formulas are derived!

\[ w_1 = \frac{\sum (x - \bar{x})(y - \bar{y})}{\sum (x - \bar{x})^2} \]

\[ w_0 = \bar{y} - w_1\bar{x} \]

https://youtu.be/jqoHefilf9U
Linear Regression

dependent variable

independent variable

$y$

$X$
Linear Regression

Is good when the relationship between independent and dependent variables is linear.
Linear Regression

Is good when the relationship between independent and dependent variables is linear

However may fail badly if this relationship is clearly not linear
Linear Regression

Is good when the **relationship** between independent and dependent variables **is linear**

However may fail badly if this relationship is clearly **not linear**
Decision Tree Algorithm

By asking a simple question about value of independent variable it tries to predict a value of dependent variable.
Decision Tree Algorithm

By asking a simple question about value of independent variable it tries to predict a value of dependent variable.

Root node: Is distance > X
- False: fare amount = Y
- True: fare amount = Z
Decision Tree Algorithm

By asking a simple question about value of independent variable it tries to predict a value of dependent variable.

- Is distance > X
  - False
    - fare amount = Y
  - True
    - fare amount = Z

Root node

Left child

Right child
Decision Tree Algorithm

By asking a simple question about value of independent variable it tries to predict a value of dependent variable.
Decision Tree Algorithm

By asking a simple question about value of independent variable it tries to predict a value of dependent variable.

Graph showing a decision tree with the question 'Is distance > X'. The options are 'False' leading to 'fare amount = Y' and 'True' leading to 'fare amount = Z'.
Decision Tree Algorithm

Here, \( X \) may correspond to any vertical line.

For example if \( X = 2.5 \):

What are most reasonable values for \( Y \) and \( Z \)?
Decision Tree Algorithm

Here, $X$ may correspond to any vertical line.

For example if $X = 2.5$:

Is distance > 2.5

- False: fare amount = $Y$
- True: fare amount = $Z$

What are most reasonable values for $Y$ and $Z$ (that minimise total MSE)?
Decision Tree Algorithm

What would be MSE if \( Y = 4 \) and \( Z = 5 \)?

For example if \( X = 2.5 \):

Is distance > 2.5

- False: fare amount = 4
- True: fare amount = 5

What are most reasonable values for \( Y \) and \( Z \) (that minimise total MSE)?
Decision Tree Algorithm

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

What are most reasonable values for $Y$ and $Z$ (that minimise total MSE)?

$Y = 4$

$Z = 5$

fare amount = 4

fare amount = 5
What are most reasonable values for \( Y \) and \( Z \) (that minimise total MSE)?

\[
MSE = \frac{1}{n} \sum_{i=1}^{n} (y_i - \hat{y}_i)^2
\]
Decision Tree Algorithm

\[
MSE = \frac{1}{n} \sum_{i=1}^{n} (y_i - \hat{y}_i)^2 = \frac{(y_1 - \hat{y}_1)^2 + (y_2 - \hat{y}_2)^2 + (y_3 - \hat{y}_3)^2 + (y_4 - \hat{y}_4)^2 + (y_5 - \hat{y}_5)^2}{5}
\]

What are most reasonable values for \(Y\) and \(Z\) (that minimise total MSE)?

Is distance > 2.5

<table>
<thead>
<tr>
<th>distance</th>
<th>fare amount</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>2.5</td>
</tr>
<tr>
<td>2</td>
<td>4</td>
</tr>
<tr>
<td>3</td>
<td>5</td>
</tr>
<tr>
<td>4</td>
<td>6</td>
</tr>
<tr>
<td>5</td>
<td></td>
</tr>
</tbody>
</table>

Y = 4
Z = 5

fare amount = 4
fare amount = 5

False
True
**Decision Tree Algorithm**

The decision tree algorithm is illustrated with a decision boundary at distance > 2.5. The graph shows the relationship between distance and fare amount. The decision nodes are at distance values of 2.5, 3, 4, and 5. The fare amount is determined based on whether the distance is greater than 2.5.

The equation for Mean Squared Error (MSE) is given by:

\[
MSE = \frac{1}{n} \sum_{i=1}^{n} (y_i - \hat{y}_i)^2 = \frac{(2 - 4)^2 + (y_2 - \hat{y}_2)^2 + (y_3 - \hat{y}_3)^2 + (y_4 - \hat{y}_4)^2 + (y_5 - \hat{y}_5)^2}{5}
\]

What are the most reasonable values for \( Y \) and \( Z \) (that minimise total MSE)?

- **Y = 4**
- **Z = 5**
**Decision Tree Algorithm**

\[
MSE = \frac{1}{n} \sum_{i=1}^{n} (y_i - \hat{y}_i)^2 = \frac{(y_1 - \hat{y}_1)^2 + (y_2 - \hat{y}_2)^2 + (y_3 - \hat{y}_3)^2 + (y_4 - \hat{y}_4)^2 + (y_5 - \hat{y}_5)^2}{5}
\]

What are most reasonable values for \(Y\) and \(Z\) (that minimise total MSE)?
Decision Tree Algorithm

\[
MSE = \frac{1}{n} \sum_{i=1}^{n} (y_i - \hat{y}_i)^2 = \frac{(2 - 4)^2 + (y_2 - \hat{y}_2)^2 + (y_3 - \hat{y}_3)^2 + (y_4 - \hat{y}_4)^2 + (y_5 - \hat{y}_5)^2}{5}
\]

What are most reasonable values for \( Y \) and \( Z \) (that minimise total MSE)?

- Is distance > 2.5
  - False
  - True
  - fare amount = 4
  - fare amount = 5

\[ Y = 4 \]
\[ Z = 5 \]
Decision Tree Algorithm

\[ MSE = \frac{1}{n} \sum_{i=1}^{n} (y_i - \hat{y}_i)^2 = \frac{(2 - 4)^2 + (4 - 4)^2 + (y_3 - \hat{y}_3)^2 + (y_4 - \hat{y}_4)^2 + (y_5 - \hat{y}_5)^2}{5} \]

Is distance > 2.5?

- False
- True

What are most reasonable values for \( Y \) and \( Z \) (that minimise total MSE)?

- fare amount = 4
- fare amount = 5
Decision Tree Algorithm

\[
MSE = \frac{1}{n} \sum_{i=1}^{n} (y_i - \hat{y}_i)^2 = \frac{(2 - 4)^2 + (4 - 4)^2 + (y_3 - \hat{y}_3)^2 + (y_4 - \hat{y}_4)^2 + (y_5 - \hat{y}_5)^2}{5}
\]

What are most reasonable values for \( Y \) and \( Z \) (that minimise total MSE)?
Decision Tree Algorithm

$$MSE = \frac{1}{n} \sum_{i=1}^{n} (y_i - \hat{y}_i)^2 = (2 - 4)^2 + (4 - 4)^2 + (5 - 5)^2 + (y_4 - \hat{y}_4)^2 + (y_5 - \hat{y}_5)^2$$

What are most reasonable values for $Y$ and $Z$ (that minimise total MSE)?
**Decision Tree Algorithm**

$$MSE = \frac{1}{n} \sum_{i=1}^{n} (y_i - \hat{y}_i)^2 = \frac{(2 - 4)^2 + (4 - 4)^2 + (5 - 5)^2 + (y_4 - \hat{y}_4)^2 + (y_5 - \hat{y}_5)^2}{5}$$

What are most reasonable values for \( Y \) and \( Z \) (that minimise total MSE)?

- **False**
  - fare amount = 4
- **True**
  - fare amount = 5
Decision Tree Algorithm

\[ \text{MSE} = \frac{1}{n} \sum_{i=1}^{n} (y_i - \hat{y}_i)^2 = \frac{(2 - 4)^2 + (4 - 4)^2 + (5 - 5)^2 + (4 - 5)^2 + (y_5 - \hat{y}_5)^2}{5} \]

What are most reasonable values for \( Y \) and \( Z \) (that minimise total MSE)?
**Decision Tree Algorithm**

\[
MSE = \frac{1}{n} \sum_{i=1}^{n} (y_i - \hat{y}_i)^2 = \frac{(2 - 4)^2 + (4 - 4)^2 + (5 - 5)^2 + (4 - 5)^2 + (5 - 5)^2}{5}
\]

What are most reasonable values for \( Y \) and \( Z \) (that minimise total MSE)?

- False
- True

- fare amount = 4
- fare amount = 5

Is distance > 2.5
Decision Tree Algorithm

\[
MSE = \frac{1}{n} \sum_{i=1}^{n} (y_i - \hat{y}_i)^2 = \frac{(2 - 4)^2 + (4 - 4)^2 + (5 - 5)^2 + (4 - 5)^2 + (5 - 5)^2}{5}
\]

What are most reasonable values for \( Y \) and \( Z \) (that minimise total MSE)?

Is distance > 2.5

False

True

fare amount = 4

fare amount = 5
Decision Tree Algorithm

\[ \text{MSE} = \frac{1}{n} \sum_{i=1}^{n} (y_i - \hat{y}_i)^2 = \frac{(-2)^2 + (0)^2 + (0)^2 + (-1)^2 + (0)^2}{5} \]

What are most reasonable values for \( Y \) and \( Z \) (that minimise total \( \text{MSE} \))?
Decision Tree Algorithm

\[ MSE = \frac{1}{n} \sum_{i=1}^{n} (y_i - \hat{y}_i)^2 = \frac{4 + 0 + 0 + 1 + 0}{5} \]

What are most reasonable values for \( Y \) and \( Z \) (that minimise total \( MSE \))?

- \( Y = 4 \)
- \( Z = 5 \)

Is distance > 2.5

False

- fare amount = 4

True

- fare amount = 5
Decision Tree Algorithm

\[ MSE = \frac{1}{n} \sum_{i=1}^{n} (y_i - \hat{y}_i)^2 = \frac{5}{5} \]

What are most reasonable values for \( Y \) and \( Z \) (that minimise total MSE)?

- \( Y = 4 \)
- \( Z = 5 \)
Decision Tree Algorithm

\[ MSE = \frac{1}{n} \sum_{i=1}^{n} (y_i - \hat{y}_i)^2 = 1 \]

Is distance > 2.5

What are most reasonable values for \( Y \) and \( Z \) (that minimise total MSE)?

\[ Y = 4 \]
\[ Z = 5 \]
Decision Tree Algorithm

$$MSE = \frac{1}{n} \sum_{i=1}^{n} (y_i - \hat{y}_i)^2 = 1$$ so, if $X = 2.5$, $Y = 4$ and $Z = 5$, MSE is 1.
Decision Tree Algorithm

\[
MSE = \frac{1}{n} \sum_{i=1}^{n} (y_i - \hat{y}_i)^2 = 1 \quad \text{so, if } X = 2.5, \ Y = 4 \text{ and } Z = 5, \ \text{MSE is } 1
\]

Can we find better \( Y \) and \( Z \)?

What are most reasonable values for \( Y \) and \( Z \) (that minimise total MSE)?

Is distance > 2.5

False

True

fare amount = 4

fare amount = 5

Can we find better \( Y \) and \( Z \)?

so, if \( X = 2.5, \ Y = 4 \text{ and } Z = 5, \ \text{MSE is } 1 \)
Decision Tree Algorithm

\[ MSE = \frac{1}{n} \sum_{i=1}^{n} (y_i - \hat{y}_i)^2 = \frac{(y_1 - \hat{y}_1)^2 + (y_2 - \hat{y}_2)^2 + (y_3 - \hat{y}_3)^2 + (y_4 - \hat{y}_4)^2 + (y_5 - \hat{y}_5)^2}{5} \]

What are most reasonable values for \( Y \) and \( Z \) (that minimise total MSE)?

Is distance > 2.5

False

True

fare amount = 3

fare amount = 5
Decision Tree Algorithm

$$MSE = \frac{1}{n} \sum_{i=1}^{n} (y_i - \hat{y}_i)^2 = \frac{(2 - 3)^2 + (4 - 3)^2 + (5 - 5)^2 + (4 - 5)^2 + (5 - 5)^2}{5}$$

What are most reasonable values for $Y$ and $Z$ (that minimise total MSE)?

Is distance > 2.5

False

fare amount = 3

True

fare amount = 5
Decision Tree Algorithm

\[ \text{MSE} = \frac{1}{n} \sum_{i=1}^{n} (y_i - \hat{y}_i)^2 = \frac{1 + 1 + 0 + 0 + 0}{5} \]

What are most reasonable values for \( Y \) and \( Z \) (that minimise total MSE)?

Is distance > 2.5

- False
  - fare amount = 3
- True
  - fare amount = 5
Decision Tree Algorithm

\[
MSE = \frac{1}{n} \sum_{i=1}^{n} (y_i - \hat{y}_i)^2 = \frac{3}{5} = 0.6
\]

What are most reasonable values for \( Y \) and \( Z \) (that minimise total MSE)?

Is distance > 2.5

False

True

fare amount = 3

fare amount = 5

What are most reasonable values for \( Y \) and \( Z \) (that minimise total MSE)?
Decision Tree Algorithm

\[ MSE = \frac{1}{n} \sum_{i=1}^{n} (y_i - \hat{y}_i)^2 = \frac{3}{5} = 0.6 \]

so, if \( X = 2.5, Y = 3 \) and \( Z = 5 \), MSE is 0.6

What are most reasonable values for \( Y \) and \( Z \) (that minimise total MSE)?
Decision Tree Algorithm

\[
MSE = \frac{1}{n} \sum_{i=1}^{n} (y_i - \hat{y}_i)^2 = \frac{(2 - 3)^2 + (4 - 3)^2 + (5 - 4.66)^2 + (4 - 4.66)^2 + (5 - 4.66)^2}{5}
\]

What are most reasonable values for \(Y\) and \(Z\) (that minimise total MSE)?
**Decision Tree Algorithm**

\[
MSE = \frac{1}{n} \sum_{i=1}^{n} (y_i - \hat{y}_i)^2 = \frac{1 + 1 + 0.12 + 0.43 + 0.12}{5}
\]

What are most reasonable values for \( Y \) and \( Z \) (that minimise total MSE)?

**Is distance > 2.5**

- **False**
  - fare amount = 3
- **True**
  - fare amount = 4.66
**Decision Tree Algorithm**

\[
MSE = \frac{1}{n} \sum_{i=1}^{n} (y_i - \hat{y}_i)^2 = \frac{2.67}{5} = 0.53
\]

so, if \( Y = 3 \) and \( Z = 4.66 \), MSE is **smallest**

Are we happy?

**Diagram:**
- **Decision Node:** Is distance > 2.5
  - **False:** fare amount = 3
  - **True:** fare amount = 4.66
Decision Tree Algorithm

Hold on, how did we choose this **split** on the first place?

<table>
<thead>
<tr>
<th>fare amount</th>
<th>distance</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td>4</td>
<td>4</td>
</tr>
<tr>
<td>5</td>
<td>5</td>
</tr>
</tbody>
</table>

Is distance > 2.5

- False: fare amount = 3
- True: fare amount = 4.66
Decision Tree Algorithm

Hold on, how did we choose this split on the first place? Maybe there are better options?
If I have seen further than others, it is by standing upon the shoulders of giants.

(Isaac Newton)
References

• Machine Learning by Andrew Ng (https://www.coursera.org/learn/machine-learning)

• Introduction to Machine Learning by Pascal Vincent given at Deep Learning Summer School, Montreal 2015 (http://videolectures.net/deeplearning2015_vincent_machine_learning/)

• Welcome to Machine Learning by Konstantin Tretyakov delivered at AACIMP Summer School 2015 (http://kt.era.ee/lectures/aacimp2015/1-intro.pdf)

• Stanford CS class: Convolutional Neural Networks for Visual Recognition by Andrej Karpathy (http://cs231n.github.io/)

• Data Mining Course by Jaak Vilo at University of Tartu (https://courses.cs.ut.ee/MTAT.03.183/2017_spring/uploads/Main/DM_05_Clustering.pdf)

• Machine Learning Essential Concepts by Ilya Kuzovkin (https://www.slideshare.net/iljakuzovkin)

• From the brain to deep learning and back by Raul Vicente Zafra and Ilya Kuzovkin (http://www.uttv.ee/naita?id=23585&keel=eng)
That's all Folks!