



มหาวิทยาลัยธรรมศาสตร์
THAMMASAT UNIVERSITY

รายงานประกอบโครงงานการเรียนรู้ของเครื่อง

จัดทำโดย

1. 6510742098 ธนกฤต แม่นผล
2. 6510742510 มณทิระ อินทร์น้อย
3. 6510742254 สรยุทธ อิงบุญ

นำเสนอ

รศ.ดร.วิรัช อโณทัยไพบูรณ์

รายงานฉบับนี้เป็นส่วนหนึ่งของวิชา
CN340 การเรียนรู้ของเครื่อง
ภาคเรียนที่ 2 ปีการศึกษา 2567

สารบัญ

ชื่อเรื่อง	หน้า
บทคัดย่อ (Abstract)	1
บทที่ 1 บทนำ (Introduction)	2
1.1 ที่มาและความสำคัญ	2
1.2 วัตถุประสงค์	2
บทที่ 2 งานวิจัยที่เกี่ยวข้อง (Related Work)	3
บทที่ 3 ชุดข้อมูล (Dataset)	4
3.1 แหล่งที่มาและการได้มาของข้อมูล (Data Source and Acquisition)	4
3.2 คำอธิบายชุดข้อมูลและการสำรวจข้อมูลเบื้องต้น (Dataset Description and EDA)	4
3.3 การประมวลผลข้อมูลเบื้องต้น (Data Preprocessing)	4
บทที่ 4 วิธีการ (Methodology)	6
4.1 ภาพรวมของแนวทาง (Overall Approach)	6
4.2 แบบจำลองโครงข่ายประสาทเทียม (Neural Network Model)	6
4.3 การฝึกสอนโมเดล (Model Training)	7
4.4 ตัวชี้วัดการประเมินผล (Evaluation Metrics)	7
บทที่ 5 ผลลัพธ์ (Results)	8
5.1 ประสิทธิภาพของแบบจำลองโครงข่ายประสาทเทียม	8
5.2 กราฟการเรียนรู้ (Learning Curves)	9
5.3 กราฟการเปรียบเทียบค่าจริงและค่าที่ทำนาย (Actual vs. Predicted Plot)	9
บทที่ 6 อภิปรายผล (Discussion)	11
บทที่ 7 สรุปและข้อเสนอแนะสำหรับงานวิจัยในอนาคต (Conclusion and Future Work)	13
บรรณานุกรม Bibliography/References	14
ภาคผนวก Appendix	16

บทคัดย่อ (Abstract)

โครงการนี้มีวัตถุประสงค์เพื่อพัฒนาแบบจำลองการเรียนรู้ของเครื่องสำหรับทำนายปริมาณแคลอรีที่เผาผลาญ (Calories Burned) จากกิจกรรมการออกกำลังกาย โดยใช้ชุดข้อมูล "Gym Members Exercise Tracking" ซึ่งประกอบด้วยข้อมูลคุณลักษณะต่างๆ ของการออกกำลังกาย เช่น อายุ, เพศ, ระยะเวลา, อัตราการเต้นของหัวใจ, และอุณหภูมิร่างกาย แบบจำลองหลักที่พัฒนาขึ้นคือโครงข่ายประสาทเทียม (Neural Network) แบบหลายชั้น (Multi-layer Perceptron) สร้างขึ้นโดยใช้ไลบรารี Keras ร่วมกับ TensorFlow ข้อมูลดิบผ่านกระบวนการเตรียมข้อมูลหลายขั้นตอน, การแปลงข้อมูลตัวแปรกลุ่ม (Categorical Features) โดยใช้ One-Hot Encoding, และการปรับสเกลข้อมูลตัวแปรเชิงตัวเลข (Numerical Features) โดยใช้ StandardScaler เพื่อให้โมเดลสามารถเรียนรู้ได้อย่างมีประสิทธิภาพ ผลการประเมินโมเดลบนชุดข้อมูลทดสอบ (Test Set) แสดงให้เห็นถึงค่า Mean Squared Error (MSE) และ R-squared (R²) ซึ่งชี้ให้เห็นว่าแบบจำลองโครงข่ายประสาทเทียมที่พัฒนาขึ้นมีความสามารถในการทำนายปริมาณแคลอรีที่เผาผลาญได้ในระดับพอใช้และสามารถเป็นเครื่องมือที่มีประโยชน์ในการประเมินผลการออกกำลังกายส่วนบุคคลได้

บทที่ 1 บทนำ (Introduction)

1. 1 ที่มาและความสำคัญ

การออกกำลังกายเป็นกิจกรรมที่สำคัญในการรักษาสุขภาพและควบคุมน้ำหนัก การทราบปริมาณแคลอรีที่ร่างกายเผาผลาญในระหว่างการออกกำลังกายแต่ละครั้งช่วยให้บุคคลสามารถวางแผนการออกกำลังกายและควบคุมปริมาณอาหารที่บริโภคได้อย่างเหมาะสม อย่างไรก็ตาม การคำนวณแคลอรีที่เผาผลาญได้อย่างแม่นยำมักต้องอาศัยอุปกรณ์เฉพาะทางหรือสูตรคำนวณที่ซับซ้อนและอาจไม่สะดวกสำหรับบุคคลทั่วไป การประยุกต์ใช้เทคนิคการเรียนรู้ของเครื่อง (Machine Learning) จึงเป็นอีกทางเลือกหนึ่งที่น่าสนใจในการสร้างแบบจำลองที่สามารถทำนายปริมาณแคลอรีที่เผาผลาญจากข้อมูลการออกกำลังกายที่บันทึกได้ง่ายกว่า

โครงการนี้จึงมุ่งเน้นไปที่การพัฒนาแบบจำลองโครงข่ายประสาทเทียม (Neural Network) เพื่อทำนายปริมาณแคลอรีที่เผาผลาญ (Calories_Burned) โดยใช้ชุดข้อมูล "Gym Members Exercise Tracking" ที่รวบรวมข้อมูลการออกกำลังกายของสมาชิกยิมหลายราย

1.2 วัตถุประสงค์

วัตถุประสงค์หลักของโครงการนี้ประกอบด้วย:

1. สำรวจและเตรียมชุดข้อมูล "Gym Members Exercise Tracking" ให้พร้อมสำหรับการสร้างแบบจำลองการเรียนรู้ของเครื่อง
2. พัฒนาแบบจำลองโครงข่ายประสาทเทียม (Neural Network) สำหรับทำนายปริมาณแคลอรีที่เผาผลาญ
3. ฝึกสอนและประเมินประสิทธิภาพของแบบจำลองที่พัฒนาขึ้นโดยใช้ตัวชี้วัดที่เหมาะสม
4. วิเคราะห์ผลลัพธ์และอภิปรายถึงความสามารถและข้อจำกัดของแบบจำลอง

ขอบเขตของโครงการนี้จะครอบคลุมตั้งแต่การประมวลผลข้อมูลเบื้องต้น, การสร้างและฝึกสอนแบบจำลองโครงข่ายประสาทเทียมด้วย Keras และ SKLearn และการประเมินผลด้วยตัวชี้วัดเชิงปริมาณ เช่น Mean Squared Error (MSE) และ R-squared (R^2) บนชุดข้อมูลทดสอบ รายงานฉบับนี้จะนำเสนอขั้นตอนการดำเนินงาน, ผลการทดลอง, และบทสรุปตามลำดับ

บทที่ 2 งานวิจัยที่เกี่ยวข้อง

(Related Work)

การทำนายปริมาณการใช้พลังงานหรือแคลอรีที่เผาผลาญระหว่างการทำกิจกรรมทางกายภาพเป็นหัวข้อที่ได้รับความสนใจในงานวิจัยด้านวิทยาศาสตร์การกีฬาและสุขภาพมาอย่างต่อเนื่อง งานวิจัยในระยะแรกมักอาศัยการสร้างสมการถดถอย (Regression Equations) จากข้อมูลที่เก็บในห้องปฏิบัติการโดยใช้อุปกรณ์วัดมาตรฐาน เช่น การวัดปริมาณการใช้ออกซิเจน (Indirect Calorimetry) (Pivarnik et al., 1992; Keytel et al., 2005)

ในช่วงไม่กี่ปีที่ผ่านมา เทคโนโลยีอุปกรณ์สวมใส่ (Wearable Devices) และสมาร์ทโฟนได้ทำให้การเก็บข้อมูลการออกกำลังกายในชีวิตประจำวันเป็นเรื่องง่ายขึ้น เช่น อัตราการเต้นของหัวใจ, จำนวนก้าว, ระยะทาง, และความเร็ว ข้อมูลเหล่านี้ได้ถูกนำมาใช้ร่วมกับเทคนิคการเรียนรู้ของเครื่องเพื่อพัฒนาแบบจำลองการทำนายแคลอรีที่แม่นยำและสะดวกยิ่งขึ้น มีงานวิจัยจำนวนมากที่ประยุกต์ใช้อัลกอริทึมต่างๆ เช่น Support Vector Machines (SVM), Random Forests, Gradient Boosting Machines และโครงข่ายประสาทเทียม (Neural Networks) สำหรับปัญหานี้ (Stisen et al., 2015; Casiraghi et al., 2020)

โครงข่ายประสาทเทียม (Neural Networks) โดยเฉพาะแบบ Multi-layer Perceptrons (MLPs) ได้แสดงให้เห็นถึงศักยภาพในการเรียนรู้ความสัมพันธ์ที่ซับซ้อนและไม่เป็นเชิงเส้นระหว่างตัวแปรนำเข้า (input features) และตัวแปรเป้าหมาย (target variable) ซึ่งเหมาะสมกับปัญหาการทำนายแคลอรีที่อาจขึ้นอยู่กับปัจจัยหลายอย่างที่มีปฏิสัมพันธ์กันอย่างซับซ้อน (Altini et al., 2016) โครงการนี้จะเลือกใช้โครงข่ายประสาทเทียมเพื่อสำรวจความสามารถในการทำนายปริมาณแคลอรีที่เผาผลาญจากชุดข้อมูลการออกกำลังกายที่มีหลายมิติ

บทที่ 3 ชุดข้อมูล (Dataset)

3.1 แหล่งที่มาและการได้มาของข้อมูล (Data Source and Acquisition)

โครงการนี้ใช้ชุดข้อมูลสาธารณะชื่อ "Gym Members Exercise Tracking" ซึ่งเผยแพร่บนแพลตฟอร์ม Kaggle (ValaKhorasani, 2023) ชุดข้อมูลนี้รวบรวมข้อมูลการออกกำลังกายของสมาชิกยิมจำนวน 973 รายการ โดยมี 15 คุณลักษณะที่เกี่ยวข้อง

3.2 คำอธิบายชุดข้อมูลและการสำรวจข้อมูลเบื้องต้น (Dataset Description and EDA)

ชุดข้อมูลประกอบด้วยคุณลักษณะต่างๆ ดังนี้:

- User_ID: รหัสสมาชิก (Categorical/Identifier)
- Gender: เพศ (Categorical - เช่น Male, Female)
- Age: อายุ (Numerical)
- Height: ส่วนสูง (Numerical)
- Weight: น้ำหนัก (Numerical)
- Duration: ระยะเวลาการออกกำลังกาย (นาที) (Numerical)
- Heart_Rate: อัตราการเต้นของหัวใจเฉลี่ย (ครั้งต่อนาที) (Numerical)
- Body_Temp: อุณหภูมิร่างกาย (องศาเซลเซียส) (Numerical)
- Calories_Burned: ปริมาณแคลอรีที่เผาผลาญ (Numerical - Target Variable)

3.3 การประมวลผลข้อมูลเบื้องต้น (Data Preprocessing)

เพื่อให้ชุดข้อมูลมีความเหมาะสมสำหรับการนำไปสร้างแบบจำลองโครงข่ายประสาทเทียม ได้มีการดำเนินการประมวลผลข้อมูลเบื้องต้นดังนี้:

1. การจัดการ Identifier Columns: คอลัมน์ที่ไม่เกี่ยวข้องกับการทำนาย เช่น User_ID ถูกลบออกจากชุดข้อมูล
2. การจัดการ Categorical Features:
 - คุณลักษณะที่เป็นกลุ่ม (Categorical) เช่น Gender ถูกแปลงให้อยู่ในรูปแบบตัวเลขที่โมเดลสามารถประมวลผลได้ โดยในโครงการนี้เลือกใช้เทคนิค One-Hot Encoding

3. การแบ่งข้อมูล (Data Splitting):

- ชุดข้อมูลถูกแบ่งออกเป็นชุดข้อมูลสำหรับฝึกสอน (Training Set) และชุดข้อมูลสำหรับทดสอบ (Test Set) ในสัดส่วน 80:20 โดยมีการกำหนด `random_state` เพื่อให้สามารถทำซ้ำผลการทดลองได้

4. การปรับสเกลข้อมูล (Feature Scaling):

- คุณลักษณะที่เป็นตัวเลข (Numerical Features) ทั้งหมดในชุดข้อมูลฝึกสอนและชุดข้อมูลทดสอบ ถูกปรับสเกลโดยใช้เทคนิค `StandardScaler` จากไลบรารี `scikit-learn` การปรับสเกลนี้ช่วยให้แต่ละ Feature มีค่าเฉลี่ยเป็น 0 และ Standard Deviation เป็น 1 ซึ่งเป็นสิ่งสำคัญสำหรับโมเดลโครงข่ายประสาทเทียมเพื่อให้กระบวนการเรียนรู้ (Optimization) เป็นไปอย่างมีประสิทธิภาพและรวดเร็วขึ้น (`scaler.fit_transform` บน Training Set และ `scaler.transform` บน Test Set)

บทที่ 4 วิธีการ (Methodology)

4.1 ภาพรวมของแนวทาง (Overall Approach)

กระบวนการหลักในโครงงานนี้เริ่มต้นจากการเตรียมข้อมูลตามที่ได้กล่าวไว้ในหัวข้อที่ 3 จากนั้นจึงนำข้อมูลที่ผ่านการประมวลผลแล้วไปใช้ในการสร้าง ฝึกสอน และประเมินแบบจำลองโครงข่ายประสาทเทียมสำหรับการทำนาย Calories_Burned เพื่อเป็นฐานเปรียบเทียบประสิทธิภาพ ได้มีการทดลองใช้แบบจำลอง Linear Regression เบื้องต้นก่อนที่จะพัฒนาแบบจำลองโครงข่ายประสาทเทียม

4.2 แบบจำลองโครงข่ายประสาทเทียม (Neural Network Model)

โครงงานนี้ได้พัฒนาแบบจำลองโครงข่ายประสาทเทียมแบบหลายชั้น (Multi-layer Perceptron - MLP) สำหรับงาน Regression โดยใช้ Keras API ซึ่งเป็นส่วนหนึ่งของ TensorFlow

- สถาปัตยกรรม (Architecture):
 - Input Layer: จำนวนโน้ตใน Input Layer จะเท่ากับจำนวน Features หลังจากผ่านการประมวลผลแล้วเท่ากับ 13
 - Hidden Layers:
 - จำนวน Hidden Layers: 3 ชั้น
 - ชั้นที่ 1: 256 โน้ต พร้อมฟังก์ชันกระตุ้น (Activation Function) แบบ relu
 - ชั้นที่ 2: 128 โน้ต พร้อมฟังก์ชันกระตุ้น (Activation Function) แบบ relu
 - ชั้นที่ 3: 64 โน้ต พร้อมฟังก์ชันกระตุ้น (Activation Function) แบบ relu
 - Output Layer:
 - จำนวนโน้ต: 1 โน้ต (เนื่องจากเป็นการทำนายค่า Calories_Burned ซึ่งเป็นค่าต่อเนื่องค่าเดียว)
 - ฟังก์ชันกระตุ้น: relu
- การคอมไพล์โมเดล (Model Compilation):
 - Optimizer: เลือกใช้ adam พร้อม Learning Rate 0.001
 - Loss Function: เลือกใช้ Mean Square Error ซึ่งเหมาะสมกับงาน Regression และสอดคล้องกับเป้าหมายในการลดค่าความคลาดเคลื่อนกำลังสองเฉลี่ย

4.3 การฝึกสอนโมเดล (Model Training)

แบบจำลองโครงข่ายประสาทเทียมถูกฝึกสอนโดยใช้ชุดข้อมูลฝึกสอนที่ผ่านการปรับสเกลแล้ว (X_{train_scaled} , y_{train}) โดยมีรายละเอียดการตั้งค่าดังนี้:

- Epochs: จำนวนรอบในการฝึกสอนทั้งหมด 100 epochs
- Batch Size: จำนวนตัวอย่างข้อมูลที่ใช้ในการปรับปรุง Weights ในแต่ละครั้ง 32
- Validation Data: 80:20 การใช้ Validation Set ช่วยในการติดตามประสิทธิภาพของโมเดลบนข้อมูลที่โมเดลไม่เคยเห็นระหว่างการฝึกแต่ละ Epoch และสามารถใช้สำหรับเทคนิคอย่าง Early Stopping

4.4 ตัวชี้วัดการประเมินผล (Evaluation Metrics)

ประสิทธิภาพของแบบจำลองสุดท้ายถูกประเมินบนชุดข้อมูลทดสอบ (X_{test_scaled} , y_{test}) โดยใช้ตัวชี้วัดหลักดังต่อไปนี้:

1. Mean Squared Error (MSE): ค่าเฉลี่ยของผลต่างกำลังสองระหว่างค่าที่ทำนายได้กับค่าจริง ยิ่งค่าน้อยยิ่งดี
2. R-squared (R^2 Score หรือ Coefficient of Determination): สัดส่วนของความแปรปรวนใน Target Variable ที่สามารถอธิบายได้โดยโมเดล ค่าอยู่ระหว่าง 0 ถึง 1 (หรืออาจติดลบสำหรับโมเดลที่แย่มาก) ค่ายิ่งใกล้ 1 ยิ่งดี

บทที่ 5 ผลลัพธ์

(Results)

5.1 ประสิทธิภาพของแบบจำลองโครงข่ายประสาทเทียม

หลังจากฝึกสอนแบบจำลองโครงข่ายประสาทเทียมตามสถาปัตยกรรมและพารามิเตอร์ที่ได้ระบุไว้ในหัวข้อที่ 5 แล้ว ได้นำแบบจำลองมาทดสอบกับชุดข้อมูลทดสอบ ผลการประเมินสรุปได้ดังตารางต่อไปนี้:

Keras: Nueral Network

ตัวชี้วัด (Metric)	ค่าที่ได้ (Value)
Mean Squared Error (MSE)	1001.1873046875
R-squared (R ²)	0.9850341581548754

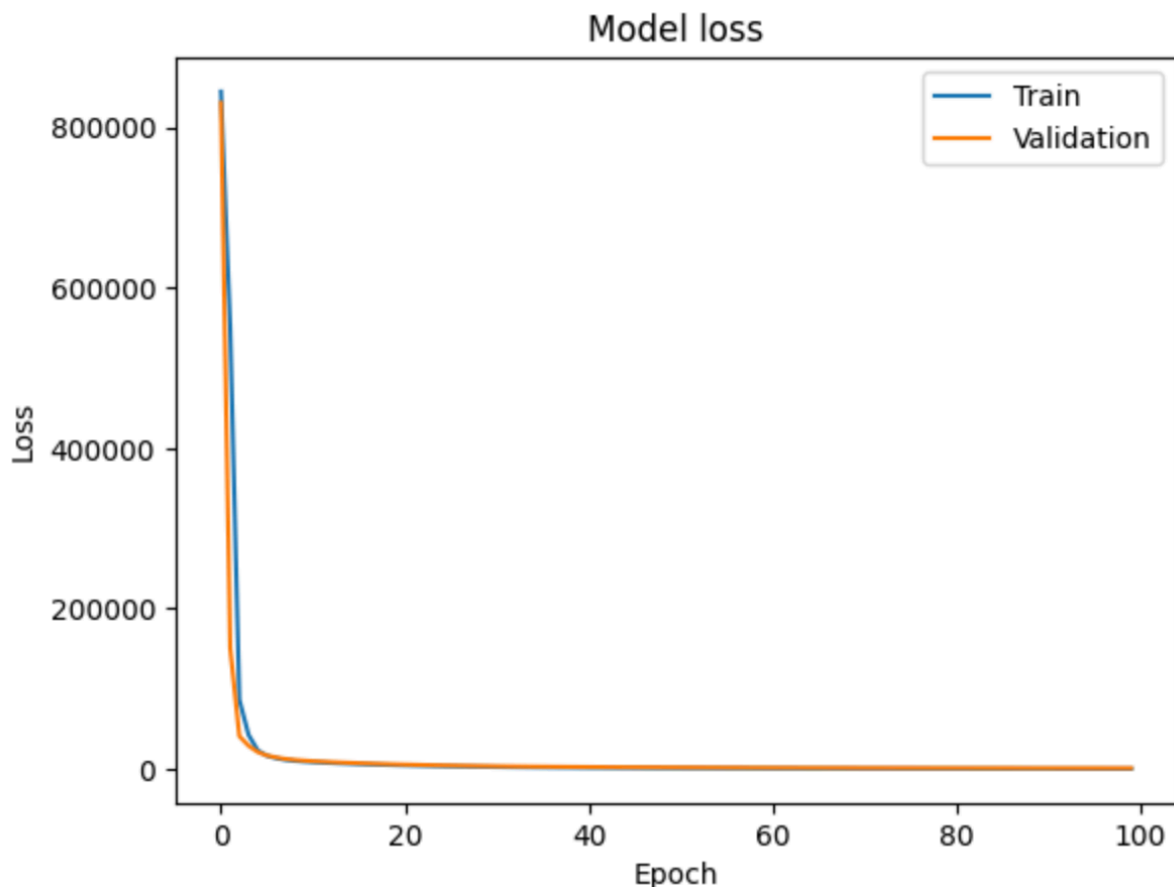
(ตารางที่ 1: ผลการประเมินประสิทธิภาพของแบบจำลอง Neural Network บน Test Set)

Scikit-learn: Linear Regression

ตัวชี้วัด (Metric)	ค่าที่ได้ (Value)
Mean Squared Error (MSE)	1438.379603
R-squared (R ²)	0.979180

5.2 กราฟการเรียนรู้ (Learning Curves)

กราฟแสดงค่า Loss (เช่น Mean Squared Error) ของชุดข้อมูลฝึกสอน (Training Loss) และชุดข้อมูลตรวจสอบ (Validation Loss ถ้ามี) ในแต่ละ Epoch ของการฝึกสอน แสดงในรูปที่ 1 เพื่อประเมินว่าโมเดลมีการเรียนรู้ที่ดีหรือไม่ และมีปัญหา Overfitting หรือ Underfitting หรือไม่

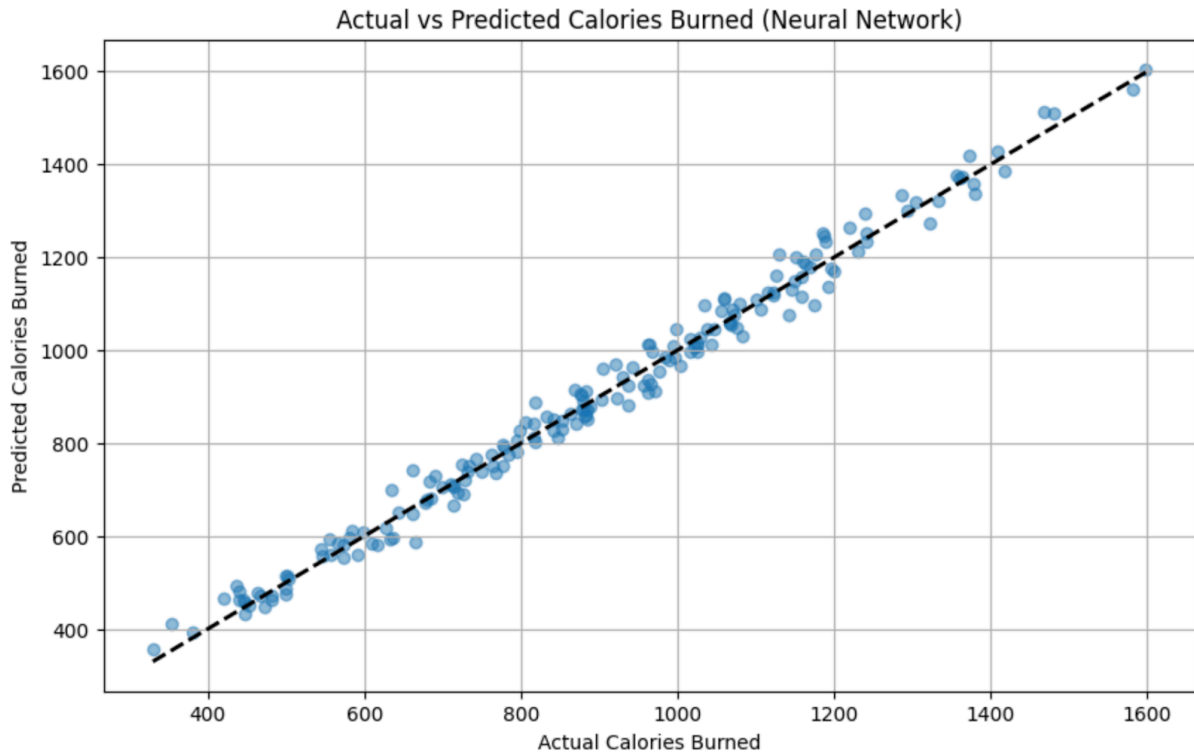


(รูปที่ 1: กราฟ Training and Validation Loss ของแบบจำลอง Neural Network)

- การวิเคราะห์ Loss Curve: โมเดลสามารถเรียนรู้ได้อย่างมีประสิทธิภาพในช่วงเวลาสั้นๆ และมีความเสถียรดีตลอดกระบวนการฝึก โดยไม่มีสัญญาณของการ Overfitting ทั้งนี้สามารถพิจารณานำโมเดลไปใช้งานจริง หรือปรับแต่งเล็กน้อยเพื่อเพิ่มความแม่นยำเพิ่มเติมได้

5.3 กราฟการเปรียบเทียบค่าจริงและค่าที่ทำนาย (Actual vs. Predicted Plot)

เพื่อแสดงให้เห็นภาพการทำนายของโมเดล ได้มีการพล็อตกราฟ Scatter Plot ระหว่างค่า Calories Burned ที่ทำนายได้ (Predicted Values) กับค่าจริง (Actual Values) บนชุดข้อมูลทดสอบ ดังแสดงในรูปที่ 2



(รูปที่ 2: กราฟ Scatter Plot แสดงค่า *Calories_Burned* ที่ทำนายได้เทียบกับค่าจริงบน *Test Set*)

- การวิเคราะห์ Scatter Plot: จากกราฟแสดงความสัมพันธ์ระหว่างค่าจริงและค่าที่โมเดลทำนาย พบว่าจุดข้อมูลกระจายตัวใกล้กับเส้นทแยงมุม ($y = x$) อย่างชัดเจน ซึ่งแสดงให้เห็นว่าโมเดลสามารถทำนายค่าการเผาผลาญแคลอรีได้ค่อนข้างแม่นยำ และไม่มีแนวโน้มที่จะทำนายผิดพลาดหนักในช่วงใดช่วงหนึ่งของข้อมูล ค่าความคลาดเคลื่อน (Error) มีการกระจายตัวอย่างสม่ำเสมอ แสดงถึงคุณภาพของโมเดลที่น่าเชื่อถือในการนำไปใช้งานต่อ

บทที่ 6 อภิปรายผล (Discussion)

จากผลการทดลองในหัวข้อที่ 6 พบว่าแบบจำลองโครงข่ายประสาทเทียมที่พัฒนาขึ้นสามารถทำนายปริมาณแคลอรีที่เผาผลาญได้ด้วยค่า R-squared เท่ากับ 0.985 และค่า MSE เท่ากับ 1001.187 บนชุดข้อมูลทดสอบ ค่า R-squared ที่ได้ จากการประเมินแบบจำลอง พบว่าโมเดลโครงข่ายประสาทเทียม (Neural Network) มีค่า R-squared (R^2) เท่ากับ 0.985 ซึ่ง บ่งชี้ว่าโมเดลสามารถอธิบายความแปรปรวนของข้อมูลได้ถึง 98.5% ถือว่าอยู่ในระดับที่ ดีมาก แสดงว่าโมเดลมีความสามารถสูงในการจับความสัมพันธ์ระหว่างฟีเจอร์และค่าที่ต้องการทำนาย (จำนวนแคลอรีที่เผาผลาญ) ได้อย่างแม่นยำ

เมื่อเปรียบเทียบกับโมเดล Linear Regression ซึ่งมีค่า R^2 เท่ากับ 0.979 และ MSE เท่ากับ 1438.379 จะเห็นว่า Neural Network ให้ผลลัพธ์ที่ดีกว่าอย่างชัดเจน ทั้งในด้านความแม่นยำ และค่าความผิดพลาดในการทำนาย โดยสาเหตุหลักที่ Neural Network มีประสิทธิภาพเหนือกว่า คือความสามารถในการเรียนรู้ ความสัมพันธ์ที่ไม่เป็นเชิงเส้น (non-linear relationships) ระหว่างตัวแปรต่าง ๆ ได้ดีกว่า Linear Regression ที่มีข้อจำกัดในการจับความสัมพันธ์แบบเส้นตรงเท่านั้น

ลักษณะของกราฟ Loss Curve (รูปที่ 1) แสดงให้เห็นว่า กราฟค่า Loss ของชุดข้อมูลฝึก (Train Loss) และชุดข้อมูลตรวจสอบ (Validation Loss) ตลอด 100 รอบการฝึก (Epoch) ของโมเดลโครงข่ายประสาทเทียม (Neural Network)

จะเห็นว่า ค่า Loss ของทั้งสองชุดลดลงอย่างรวดเร็วในช่วงแรกของการฝึก (ประมาณ 0–10 Epoch) และจากนั้นเริ่มเข้าสู่ระยะคงที่ โดย Train Loss และ Validation Loss มีค่าใกล้เคียงกัน ตลอดช่วงการฝึก แสดงว่าโมเดลสามารถ เรียนรู้ได้อย่างมีประสิทธิภาพ (Converge ได้ดี)

นอกจากนี้ ไม่พบสัญญาณของ Overfitting เนื่องจาก Validation Loss ไม่ได้เพิ่มขึ้นในขณะที่ Train Loss ลดลงอย่างต่อเนื่อง ซึ่งเป็นสัญญาณที่ดีว่าโมเดลมีความสามารถในการเรียนรู้โดยไม่จดจำเฉพาะข้อมูลฝึกมากเกินไป

อย่างไรก็ตาม ในกรณีที่เกิด Overfitting ในอนาคต อาจสามารถแก้ไขได้ด้วยวิธีการต่าง ๆ เช่น

- การใช้เทคนิค Dropout เพื่อลดการพึ่งพา neuron บางตัวมากเกินไป
- การใช้ Early Stopping เพื่อลดจำนวนรอบการฝึกเมื่อ Validation Loss เริ่มเพิ่มขึ้น
- การเพิ่มข้อมูล (Data Augmentation) หรือใช้ Regularization (เช่น L2)

กราฟ Scatter Plot (รูปที่ 2) แสดงการกระจายตัวของค่าพลังงานที่เผาผลาญ (Calories Burned) ที่โมเดลโครงข่ายประสาทเทียมทำนายได้ เปรียบเทียบกับค่าจริงจากชุดข้อมูลทดสอบ (Test Set) โดยจุดแต่ละจุดแทนคู่ของค่าจริงและค่าทำนายที่สัมพันธ์กัน

จากกราฟจะเห็นว่า จุดส่วนใหญ่อยู่ใกล้กับเส้นทแยงมุม (เส้น $y = x$) ซึ่งบ่งชี้ว่าโมเดลสามารถทำนายค่าได้ใกล้เคียงกับค่าจริงในหลายช่วงข้อมูล ทั้งช่วงค่าต่ำ (400–800) และช่วงค่าสูง (1200–1600)

การกระจายของค่าความคลาดเคลื่อน (Error) มีลักษณะ กระจายตัวแบบสุ่ม (random distribution) รอบเส้นทแยงมุม โดยไม่ปรากฏลักษณะ bias ที่ชัดเจน เช่น ทำนายค่าสูงเกินในช่วงใดช่วงหนึ่ง หรือทำนายค่าต่ำเกินในอีกช่วงหนึ่ง

อย่างไรก็ตาม มีบางจุดที่เบี่ยงเบนจากเส้นอ้างอิงในระดับปานกลาง โดยเฉพาะในช่วงค่ากลางถึงสูง แต่อัตราการเบี่ยงเบนยังอยู่ในระดับที่ยอมรับได้ และ ไม่ส่งผลกระทบต่อความน่าเชื่อถือโดยรวมของโมเดล

ข้อจำกัดของโครงการ (Limitations of the Study):

1. Features: Features ที่ใช้ในโมเดลอาจจะยังไม่ครอบคลุมปัจจัยทั้งหมดที่มีผลต่อการเผาผลาญแคลอรี การเพิ่ม Features ที่มีความหมาย (เช่น ประเภทการออกกำลังกายอย่างละเอียด, ระดับความฟิตของแต่ละบุคคล) อาจช่วยปรับปรุงประสิทธิภาพได้
2. สถาปัตยกรรมโมเดลและการปรับ Hyperparameter: สถาปัตยกรรมของโครงข่ายประสาทเทียมที่ใช้ (8 ชั้น) และกระบวนการปรับ Hyperparameter ในโครงการนี้อาจจะยังไม่ละเอียดถี่ถ้วนที่สุด การทดลองกับสถาปัตยกรรมที่ซับซ้อนขึ้นหรือการใช้เทคนิคการค้นหา Hyperparameter ที่เป็นระบบมากขึ้น (เช่น Grid Search, Randomized Search, Bayesian Optimization) อาจนำไปสู่ผลลัพธ์ที่ดีขึ้น
3. การตีความโมเดล: โครงข่ายประสาทเทียมมักถูกมองว่าเป็น "Black Box Model" ซึ่งทำให้การตีความว่าเหตุใดโมเดลจึงตัดสินใจทำนายค่านั้นๆ ออกมาเป็นเรื่องยาก แม้ว่าจะมีเทคนิคในการพยายามอธิบาย (เช่น SHAP, LIME) แต่ในโครงการนี้ยังไม่ได้ลงลึกในส่วนนั้น

ผลกระทบที่อาจเกิดขึ้น (Potential Impact): แบบจำลองที่พัฒนาขึ้น แม้จะมีข้อจำกัด สามารถเป็นพื้นฐานในการพัฒนาเครื่องมือที่ช่วยให้ผู้ใช้งานทั่วไปสามารถประมาณการแคลอรีที่เผาผลาญจากการออกกำลังกายได้สะดวกขึ้น ซึ่งจะเป็นประโยชน์ต่อการวางแผนสุขภาพส่วนบุคคล

บทที่ 7 สรุปและข้อเสนอแนะสำหรับงานวิจัยในอนาคต

(Conclusion and Future Work)

โครงการนี้ได้นำเสนอขั้นตอนการพัฒนาแบบจำลองโครงข่ายประสาทเทียมเพื่อทำนายปริมาณแคลอรีที่เผาผลาญจากการออกกำลังกาย โดยใช้ชุดข้อมูล "Gym Members Exercise Tracking" ผลการทดลองแสดงให้เห็นว่าโมเดลที่พัฒนาขึ้นให้ประสิทธิภาพในระดับที่น่าพอใจ และมีความเป็นไปได้ในการพัฒนาต่อด้วยค่า R-squared 0.9850 และ MSE 1001.187 กระบวนการเตรียมข้อมูล เช่น การทำ One-Hot Encoding และ Feature Scaling มีความสำคัญต่อการทำงานของโมเดล

ข้อเสนอแนะสำหรับงานวิจัยในอนาคต:

1. การปรับปรุงสถาปัตยกรรมโมเดล: ทดลองกับสถาปัตยกรรมโครงข่ายประสาทเทียมที่หลากหลายมากขึ้น เช่น การเพิ่มจำนวนชั้น (Deeper Networks), การเปลี่ยนจำนวนโหนดในแต่ละชั้น, การใช้ Activation Function แบบอื่นๆ หรือการเพิ่มเทคนิค Regularization เช่น Dropout เพื่อลด Overfitting
2. การทำ Hyperparameter Optimization อย่างละเอียด: ใช้เทคนิคการค้นหา Hyperparameter ที่เป็นระบบ เช่น K-fold Cross-Validation ร่วมกับ Grid Search หรือ Randomized Search เพื่อหาชุด Hyperparameter ที่ให้ประสิทธิภาพสูงสุด
3. Feature Engineering and Selection: สำรวจการสร้าง Features ใหม่ ๆ ที่อาจมีความสัมพันธ์กับปริมาณแคลอรีที่เผาผลาญ (เช่น อัตราส่วนระหว่างอัตราการเต้นของหัวใจกับระยะเวลา, ดัชนีมวลกาย (BMI) คำนวณจากส่วนสูงและน้ำหนัก) หรือใช้เทคนิค Feature Selection เพื่อเลือกเฉพาะ Features ที่สำคัญที่สุด
4. การใช้ชุดข้อมูลที่ใหญ่ขึ้นและหลากหลายขึ้น: การฝึกสอนโมเดลด้วยชุดข้อมูลที่มีขนาดใหญ่และครอบคลุมกลุ่มประชากรและประเภทการออกกำลังกายที่หลากหลายมากขึ้น จะช่วยเพิ่มความสามารถในการ Generalize ของโมเดล
5. การเปรียบเทียบกับอัลกอริทึมอื่นๆ: ทดลองและเปรียบเทียบประสิทธิภาพกับอัลกอริทึมการเรียนรู้ของเครื่องอื่นๆ ที่ซับซ้อนขึ้น เช่น Gradient Boosting Machines (XGBoost, LightGBM) หรือ Support Vector Regression (SVR) ที่อาจให้ผลลัพธ์ที่ดีในงาน Regression
6. การพัฒนาเพื่อนำไปใช้งานจริง: พิจารณาการพัฒนาโมเดลให้อยู่ในรูปแบบที่สามารถนำไปปรับใช้ในแอปพลิเคชันบนมือถือหรือเว็บแอปพลิเคชัน เพื่อให้ผู้ใช้สามารถป้อนข้อมูลการออกกำลังกายของตนเองและรับผลการทำนายแคลอรีได้

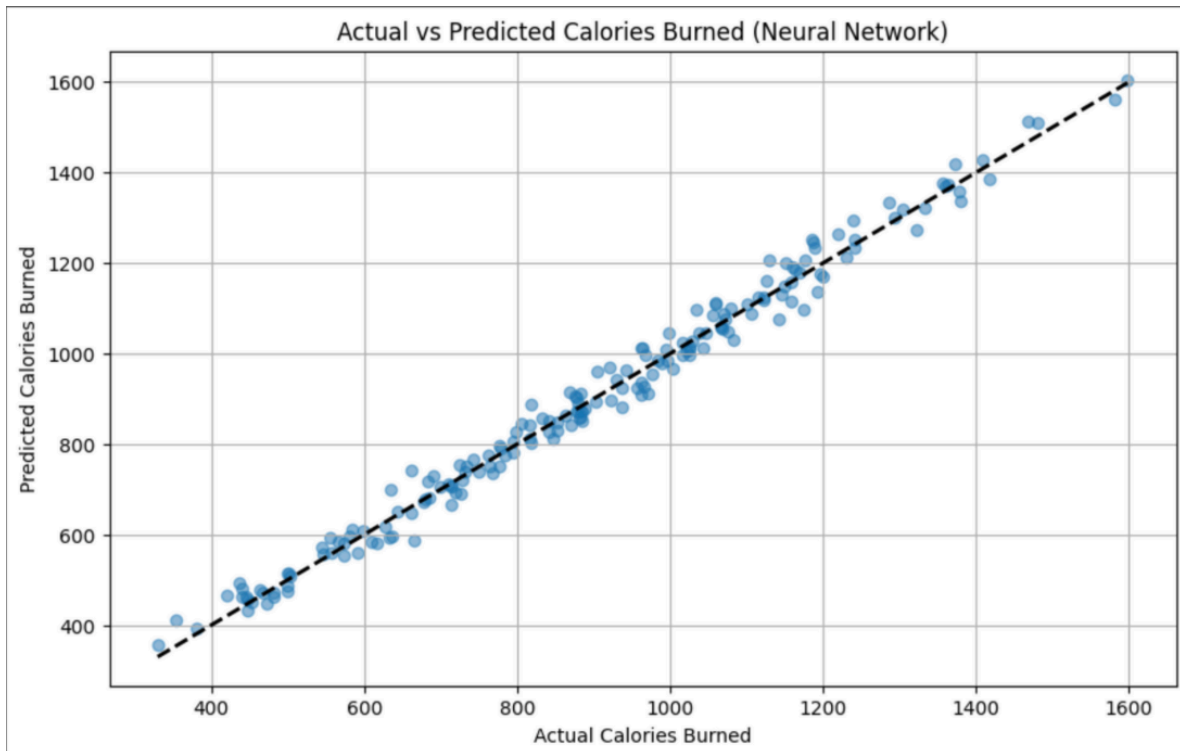
บรรณานุกรม

Bibliography/References

- Abadi, M., Agarwal, A., Barham, P., Brevdo, E., Chen, Z., Citro, C., Corrado, G. S., Davis, A., Dean, J., Devin, M., Ghemawat, S., Goodfellow, I., Harp, A., Irving, G., Isard, M., Jia, Y., Jozefowicz, R., Kaiser, L., Kudlur, M., ... Zheng, X. (2015). *TensorFlow: Large-scale machine learning on heterogeneous systems*. Software available from tensorflow.org. <https://www.tensorflow.org/>
- Altini, M., Penders, J., Vullers, R., & Amft, O. (2016). Deep learning in human activity recognition: A GMM-HMM approach. In *2016 IEEE International Conference on Systems, Man, and Cybernetics (SMC)* (pp. 002243-002248). IEEE. <https://doi.org/10.1109/SMC.2016.7844586>
- Casiraghi, G., Ridi, A., Lazzarini, T., & Ciucci, M. (2020). Machine learning for estimating energy expenditure from wearable sensors: A systematic review. *Sensors*, 20(21), 6130. <https://doi.org/10.3390/s20216130>
- Chollet, F., & others. (2015). *Keras*. GitHub. Retrieved May 20, 2025, from <https://github.com/keras-team/keras>
- Harris, C. R., Millman, K. J., van der Walt, S. J., Gommers, R., Virtanen, P., Cournapeau, D., Wieser, E., Taylor, J., Berg, S., Smith, N. J., Kern, R., Picus, M., Hoyer, S., van Kerkwijk, M. H., Brett, M., Haldane, A., del Río, J. F., Wiebe, M., Peterson, P., Oliphant, T. E. (2020). Array programming with NumPy. *Nature*, 585(7825), 357–362. <https://doi.org/10.1038/s41586-020-2649-2>
- Hunter, J. D. (2007). Matplotlib: A 2D graphics environment. *Computing in Science & Engineering*, 9(3), 90–95. <https://doi.org/10.1109/MCSE.2007.55>
- Keytel, L. R., Goedecke, J. H., Lambert, E. V., Kirschner, V. V., Noakes, T. D., & Lambert, M. I. (2005). Prediction of energy expenditure from heart rate monitoring during submaximal exercise. *Journal of Sports Sciences*, 23(3), 289-297. <https://doi.org/10.1080/02640410470001730089>
- McKinney, W. (2010). Data structures for statistical computing in python. In S. van der Walt & J. Millman (Eds.), *Proceedings of the 9th Python in Science Conference* (pp. 56–61). <https://doi.org/10.25080/Majora-92bf1922-00a>

- Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., & Duchesnay, E. (2011). Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12, 2825-2830. from <http://jmlr.org/papers/v12/pedregosa11a.html>
- Pivarnik, J. M., Montain, S. J., Graves, J. E., & Pollock, M. L. (1992). Prediction of exercise energy expenditure in normal weight and obese women. *Medicine and Science in Sports and Exercise*, 24(12), 1381-1387.
- Plotnine Development Team. (2024). *Plotnine: A grammar of graphics for Python*. GitHub. from <https://github.com/has2k1/plotnine>
- Stisen, A., Henriksen, K., Møller, K., Vestergaard, S., & Grønbæk, K. (2015). Activity classification using a smartphone. In *Proceedings of the 12th European conference on Wireless sensor networks* (p. 16).
- ValaKhorasani. (2023). *Gym Members Exercise Tracking*. Kaggle. from <https://www.kaggle.com/datasets/valakhorasani/gym-members-exercise-dataset>

ภาคผนวก Appendix



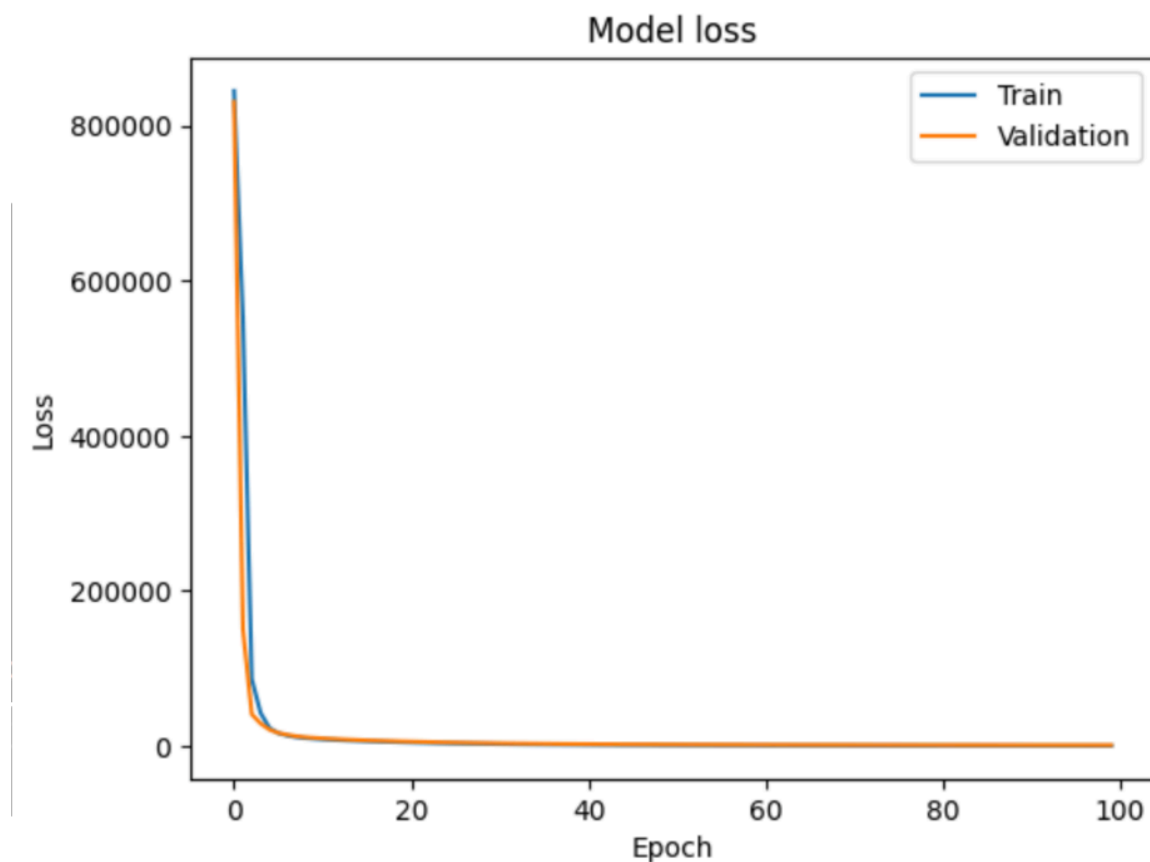
```

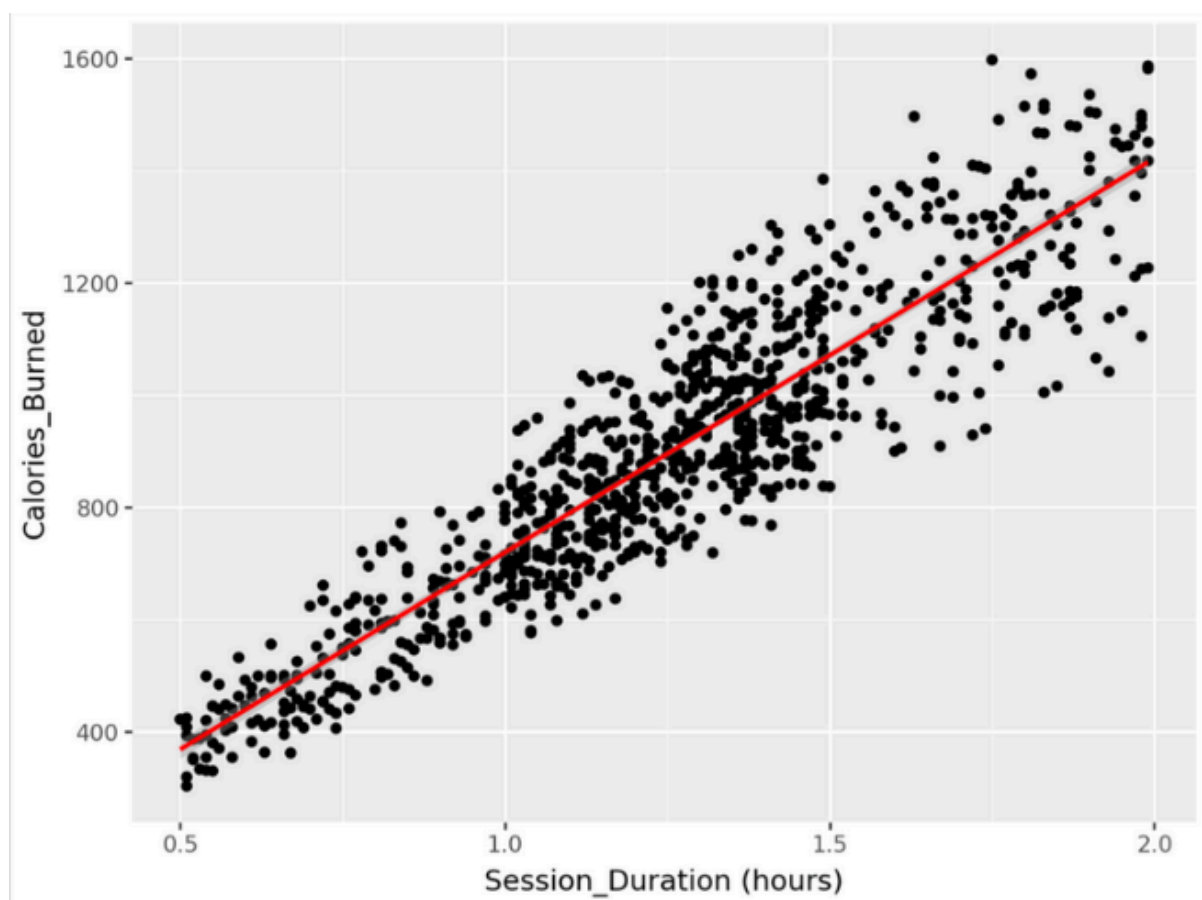
1 final_model = Sequential([
2     Dense(512, activation='relu', input_shape=(X_train_fold_scaled.shape[1],)),
3
4     Dense(256, activation='relu'),
5
6     Dense(128, activation='relu'),
7
8     Dense(64, activation='relu'),
9     Dense(1)
10 ])
11 final_optimizer = tf.keras.optimizers.Adam(learning_rate=0.001)
12 final_model.compile(optimizer=final_optimizer, loss='mean_squared_error', metrics=['mean_squared_error'])

```

24/24 ————— 0s 4ms/step - loss: 224.2202 - mean_squared_error: 224.2202
Average Mean Squared Error (K-Fold): 1001.1873046875

```
6/6 ----- 1s 62ms/step
1242.0 [1251.959] diff : [9.95898438]
777.0 [796.401] diff : [19.40100098]
904.0 [961.6618] diff : [57.6618042]
1107.0 [1087.8983] diff : [19.10168457]
1598.0 [1604.8148] diff : [6.81481934]
420.0 [466.10217] diff : [46.10217285]
1379.0 [1337.6516] diff : [41.34838867]
1231.0 [1211.4873] diff : [19.51269531]
592.0 [560.04956] diff : [31.95043945]
1175.0 [1096.034] diff : [78.96594238]
```





	True Values	Predicted Values
864	1242.0	1278.501688
74	777.0	816.952399
659	904.0	955.797466
528	1107.0	1074.130783
736	1598.0	1466.107738
...
2	677.0	639.282235
871	1025.0	992.989517
769	1082.0	1057.968610
566	1071.0	1061.646107
5	1116.0	1117.652526