# Educational Escape Room Design for Machine Learning: Week 4 - Supervised Learning – Linear Regression

It is meant to be completed by a single student within a shorter time frame (30–45 minutes), integrating **lecture content**, **tutorial exercises**, and **suggested reading**.

**🧩 Escape Room Title: “Forecast Failure at DataCorp”**

**🎯 Learning Goal:**

Apply concepts of **linear regression**, understand **model evaluation** (MSE, R²), and interpret outputs. Reinforce theory from lectures/tutorials and consolidate from reading (e.g. textbook chapter or linked blog post).

**🧠 Narrative:**

You are the new data analyst at *DataCorp*. A high-stakes report on sales prediction was sabotaged. The system is locked until you diagnose and repair the forecasting model. Your only tools: past lectures, your tutorial notebook, and the company’s cryptic internal documentation (a PDF from the reading list).

**⏳ Timing:**

~30–45 minutes, asynchronous  
**Mode:** Individual  
**Progression:** Mostly **linear** but allows brief optional detours to explore links or re-read material

**🔍 Puzzle Structure:**

**🔐 Puzzle 1: The Dataset Decryptor**

**Clue Source:** Lecture slides + tutorial  
**Task:** Identify which feature(s) best correlate with the target (e.g. "Sales") in a mini dataset  
**Tool:** Python + pandas + data.corr()  
**Expected Action:**

import pandas as pd

data = pd.read\_csv("sales\_data.csv")

print(data.corr()['Sales'])

**Clue revealed:** Best predictor feature name (e.g. "MarketingSpend")

**🔐 Puzzle 2: The Sabotaged Model**

**Clue Source:** Suggested reading (blog on overfitting and underfitting)  
**Task:** Fix the code of a faulty regression model (deliberately using too many features, some random noise)  
**Expected Action:**  
Remove unnecessary variables and re-run linear regression

from sklearn.linear\_model import LinearRegression

from sklearn.metrics import mean\_squared\_error, r2\_score

X = data[['MarketingSpend']] # simplified model

y = data['Sales']

model = LinearRegression()

model.fit(X, y)

preds = model.predict(X)

print("MSE:", mean\_squared\_error(y, preds))

print("R²:", r2\_score(y, preds))

**Clue revealed:** A passcode embedded in the printed MSE value (e.g. "MSE = 37.2 → Code: 372")

**🔐 Puzzle 3: Visual Diagnosis Terminal**

**Clue Source:** Tutorial activity on residual plots  
**Task:** Visualise the residuals to confirm if the model is appropriate  
**Expected Action:**

import matplotlib.pyplot as plt

plt.scatter(preds, y - preds)

plt.axhline(y=0, color='r', linestyle='--')

plt.title("Residual Plot")

plt.show()

**Clue revealed:** A message on the plot: “Flat line confirms fix. Use residual variance as key.”

**🔐 Final Puzzle: Unlocking the Report Room**

**Task:** Use all recovered values (best feature, MSE digits, variance estimate) to solve a riddle on a simulated keypad interface (Google Form or in-game script)  
**Expected Final Code:** e.g. Marketing37215

**📚 Integrated Materials**

* **Lecture slides:** basics of linear regression and metrics
* **Tutorials:** hands-on regression model building
* **Reading:** blog article on model complexity and evaluation

**📈 Assessment Possibilities:**

* Submit final model code (Jupyter)
* Auto-marking of final code input
* Reflection prompt: “What did you change and why?”

Would you like one for neural networks or anomaly detection next?

# Modified to get clues from other sources

Certainly. Here is a modified **individual escape room** for **Week 4: Linear Regression**, with **lecture, tutorial, and suggested reading embedded as sources of specific clues**.

**🧩 Escape Room Title: “The Analyst’s Code: A Forecast Gone Wrong”**

**🎯 Learning Goal:**

* Apply **linear regression** in code
* Use evaluation metrics
* Critically read and extract relevant values from lectures/tutorials/articles
* Strengthen model debugging and validation skills

**🧠 Narrative:**

You are the only analyst left at *DataCorp* after a massive systems crash. The forecast model has failed, and your manager has left notes scattered across documents and resources. Recover the model, evaluate its integrity, and submit a clean version before the automatic report dispatch in 45 minutes.

**⏳ Timing:**

30–45 minutes  
**Mode:** Individual  
**Progression:** **Linear**, with clues seeded across resources

**🔍 Puzzle Structure (Modified):**

**🔐 Puzzle 1: The Feature Map**

**Clue Source:** *Lecture slides*  
**Task:** Open the lecture slides, go to slide 7 (where feature correlations were discussed). Identify the strongest predictor for "Sales" from a visual correlation matrix.

**Action:** Match lecture correlation discussion to a small CSV dataset (provided).

# Once student identifies feature, test it:

X = df[['MarketingSpend']]

**Revealed Clue:** "Good choice. First digit of access code: 9"

**🔐 Puzzle 2: The Ghost Parameter**

**Clue Source:** *Suggested Reading (blog or paper)*  
**Task:** The blog on "Best Practices in Regression" includes a worked example using Ridge regression and states:  
*"We found the best alpha value for this case was* ***1.5****, balancing underfitting and overfitting."*

**Action:**  
Student must find that number from the text and apply it:

from sklearn.linear\_model import Ridge

model = Ridge(alpha=1.5)

**Revealed Clue:** Second digit of code = 1

**🔐 Puzzle 3: Residual Reflection**

**Clue Source:** *Tutorial PDF Exercise 3*  
**Task:** In the tutorial, students produced a residual plot using:

plt.scatter(preds, y - preds)

They must recreate this for the current model.

**Clue embedded in tutorial text:**  
*A good model should have residuals randomly scattered around zero. A curved pattern is a sign of trouble.*

**Revealed Clue:** If flat → third digit = 7

**🔐 Puzzle 4: Evaluation Echo**

**Clue Source:** *Lecture slides, slide 10*  
Slide mentions:  
*"Any model with R² above* ***0.85*** *is accepted by the department’s QA system."*

**Task:** Run R² evaluation. If model passes:

from sklearn.metrics import r2\_score

print(r2\_score(y, preds))

If R² ≥ 0.85, clue: final digit = 6  
If < 0.85, must tune or go back.

**🧮 Final Code to Escape:**

Combination of all revealed digits: 9176  
Used to open a mock terminal or unlock the Google Form

**🧠 Integrated Clue Sources Summary:**

| **Clue** | **Source** | **Use** |
| --- | --- | --- |
| Best feature | Lecture slide (correlation matrix) | Select correct column |
| Alpha = 1.5 | Suggested blog | Hyperparameter for model |
| Residual clue | Tutorial PDF | Residual plot pattern |
| R² threshold = 0.85 | Lecture slide | Validate model quality |

**📈 Assessment / Debrief:**

* Submit final notebook
* Brief reflection prompt: “Which clue was most difficult and why?”
* Optional peer discussion: “How did you find the reading-based clue?”

Would you like a similar one for *Support Vector Machines* or *Anomaly Detection*?