CS439: Intro to Data Science Final Project Report
Spring 2025

Decoded Gains: A Data-Driven Approach to Understanding Protein and Training Efficiency

By:
Derick Vega
Darren Bonjour
Burak Serik

Professor Chatevurdi

May 9, 2025

Github Repository: https://github.com/derickv12/git-pull-git-swole.git

Introduction:

For unsupervised learning projects:

What question are you trying to answer

Our primary intention was to find out if we could use unsupervised learning techniques to identify meaningful clusters in muscle-building outcomes based on nutrition and training intervention variables? We weren't trying to predict outcomes—we were interested in understanding if distinct types of interventions exist in the data and whether certain combinations of variables (like protein intake and frequency) align with better lean body mass (LBM) gains. More specifically, we wanted to explore what a "high-efficiency" intervention looks like when you factor in both outcome and nutritional load.

How do you plan to answer it?

We will begin by cleaning a real-world dataset of clinical studies focused on resistance training interventions. Next, we will filter the dataset to retain key variables such as training duration, frequency, energy intake, protein intake, and lean body mass (LBM) change. To better evaluate the quality of each intervention, we will engineer a new metric—LBM gain per gram of protein intake—which will serve as a measure of efficiency. After preprocessing, we will apply KMeans clustering to group the studies based on these features and use silhouette scores to determine the optimal number of clusters. We will then use Principal Component Analysis (PCA) to reduce the dimensionality of the data for visualization and assess the separation between clusters. Finally, we will analyze the characteristics of each cluster using boxplots and descriptive summaries to interpret patterns and assess which intervention profiles appear most efficient. The results of this analysis will help us better understand which combinations of training and nutritional strategies are most effective for promoting muscle growth, offering insights that could inform future research or real-world application in sports and health contexts.

How does this approach relate to the lectures/papers we discussed?

Our project applies several of the core tools and techniques we explored throughout the course. We used Python, Pandas, and NumPy for data cleaning, transformation, and feature engineering—foundational skills for working with real-world datasets. Clustering with KMeans and evaluating the results using silhouette scores tied directly into our study of unsupervised learning, where we learned how to group unlabeled data based on underlying patterns.

We also implemented dimensionality reduction with Principal Component Analysis (PCA) to help visualize and interpret the clusters—an approach emphasized in class for simplifying high-dimensional data without losing meaningful structure. The entire workflow reflects what we were taught about building data pipelines that turn raw inputs into actionable insights. More than anything, this project showed how those tools can be used together to explore a problem, generate new metrics, and arrive at conclusions that actually say something.

Motivation:

Why is your project important?

This project is important because it uses real clinical data to explore how different resistance training interventions affect muscle growth efficiency—not just in terms of raw muscle gained, but how effectively protein intake contributes to those gains. In sports science, fitness, and clinical nutrition, there's constant debate around optimal protein levels, training duration, and frequency. Our work helps cut through that by using clustering to identify patterns that might not be obvious with standard statistical summaries. It offers a data-driven way to compare the *quality* of interventions, which has implications for both research and real-world application.

Why are you excited about it?

This project resonated with each of us because we all care about health, exercise, and muscle-building to some degree—ranging from competing someday to simply improving daily movement. We each have our own goals in fitness and train regularly, whether it's for strength, aesthetics, performance, or overall well-being. Some of us have gone deeper into the research side as well—for example, exploring resources like NASM and doing extensive reading and personal research on muscle hypertrophy and nutrition long before this class. What made this project exciting was getting to merge that real-world passion with the technical tools we've learned in computer science. It was a chance to apply unsupervised learning and data analysis to a topic we genuinely care about and extract insights that could actually inform smarter training or nutrition strategies. The best part was getting a taste of how we can apply our CS knowledge directly to parts of our everyday lives, not just academic work.

What are some existing questions in the area?

Even after years of research, there's still a lack of consensus on what truly defines optimal conditions for muscle growth. A major point of debate is protein intake, how much is actually needed to support hypertrophy. Even the commonly recommended grams of protein per kilogram of body weight vary widely depending on the source,

which is why it's typically presented as a range rather than a fixed number. On top of that, training frequency, duration, total calorie intake,rest, and other variables all interact in complex, non-linear ways. Add genetics into the equation—differences in metabolism, muscle fiber composition, recovery speed—and it becomes clear that no single approach works for everyone. This variability is why projects like ours matter: even if we can't account for every factor, clustering lets us identify broad patterns in the data and draw useful, general insights about what kinds of interventions *tend* to be more efficient for muscle gain across a range of body types and study conditions.

Are there any prior related works? Provide a brief summary.

Absolutely—this topic has been explored for decades, with a large and growing body of research focused on the relationship between resistance training, nutrition, and muscle hypertrophy. One of the earliest and most influential figures in the field was Arthur Jones, the founder of Nautilus. Through controlled experiments in tightly regulated training environments, he emphasized the importance of eccentric (negative) training in stimulating muscle growth—an insight that has since been validated and built upon in modern exercise science. His findings not only reshaped training principles but also led to the creation of some of the most effective, biomechanics-driven gym equipment still widely used today. Since then, research has expanded across countless variables—protein intake ranges, training frequency, volume, recovery—but many open questions remain. Our project adds to this evolving conversation by using modern clustering techniques to surface patterns in intervention efficiency across a diverse set of studies.

Method:

What dataset did you use?

We used a dataset called *"Protein Intake and Muscle Mass Data PMC7727026"* from Kaggle. It's based on a 2020 meta-analysis by Tagawa et al. that looked at how protein intake affects muscle growth. The person who uploaded the dataset pulled the numbers from the study's tables and graphs, but only included the studies that used resistance training. Each row in the dataset is a single group from a study—either experimental or control—so we could look at the results more closely. The uploader also added new columns to make the data easier to work with, like total protein intake and lean body mass change.

	Author and Year	Percent male (%)	Race with largest number	Age (years)	Height (cm)	Weight (kg)	BMI (kg/m2)	Health state	Frequency of exercise before intervention Other	Protein intake before intervention (g/kg/day)	Protein intake before intervention (g/day)
0	Campbell (1995) [1]	83	Caucasian	65.0	173.3	78.0	26.0	Healthy	Unclear		
1	Rozenek (2002) [2]	100	Caucasian	23.2	178.3	76.4	24.1	Healthy	High		
2	Flakoll (2004) [3]	100	Caucasian	18.9	177.0	74.9	23.4	Healthy	High		
3	Rankin (2004) [4]	100	Caucasian	20.8	177.5	78.9	25.1	Healthy	Unclear	1.25	98.7
4	Layman (2005) (with RT) [5]	0	Caucasian	46.7	163.2	87.7	33.0	Healthy	Unclear	0.88	77.0
5	Candow (2006) (Post) [6]	100	Caucasian	64.8	174.0	86.7	28.6	Healthy	Unclear		
6	Candow (2006) (Pre) [6]	100	Caucasian	64.8	174.0	86.7	28.6	Healthy	Unclear		
7	Candow (2006) (Soy) [7]	33	Caucasian	23.2	170.2	70.1	24.2	Healthy	Unclear		
8	Candow (2006) (Whey) [7]	33	Caucasian	23.2	170.2	70.1	24.2	Healthy	Unclear		
9	Kerksick (2006) [8]	100	Caucasian	31.0	179.2	84.0	26.2	Healthy	Unclear	1.85	155.4
10	Cribb (2007) (with Creatine) [9]	100	Others	24.5	181.3	78.5	23.9	Healthy	Unclear	1.7	134.0
11	Cribb (2007) (without Creatine) [9]	100	Others	24.5	181.3	78.5	23.9	Healthy	Unclear	1.7	134.0
12	Hartman (2007) (Milk) [10]	100	Caucasian	24.0	179.0	80.9	25.2	Healthy	High	1.33	111.0
13	Hartman (2007) (Soy) [10]	100	Caucasian	24.0	179.0	80.9	25.2	Healthy	High	1.33	111.0
14	Hoffman (2007) [11]	100	Caucasian	20.7	182.5	95.8	28.8	Healthy	Athlete		

Figure 1: Snippet of raw dataset for context (some columns not seen in image)

What form does this data have? Is it images, raw text, tabular, etc? What are the Features?

The data is tabular, it came as a CSV file. Each row is a group from a study, and the columns are the features. Some of the main features include protein intake (in g/kg/day), energy intake, training frequency, duration of the intervention, and lean body mass (LBM) change. There are also demographic features like age, weight, height, BMI, and whether the group was “experimental” or “control”. Some extra columns were added to calculate things like how much LBM was gained per week, per unit of protein, or relative to body weight. We used a subset of these features for our analysis, mostly the ones directly tied to training and nutrition.

For unsupervised learning projects (Methodology):

What analysis did you do?

What would be your implementation steps?

How will you evaluate your method?

How will you test and measure success?

We used unsupervised learning to find patterns in how different training and nutrition setups influenced muscle growth. First, we cleaned the dataset and selected key features: *Duration (weeks), Frequency (times/week), Energy intake (kcal/kg/day), Protein intake (g/kg/day), and LBM change (kg).* We also created a new feature—*Efficiency*, defined as LBM change divided by protein intake—to measure how

effectively each group turned protein into muscle.
(insert image of the code that calculates the Efficiency column)

```
# Step 3: Create a new efficiency metric: muscle gain per gram of protein intake
df_filtered['Efficiency (kg/g_protein)'] = df_filtered['LBM change (kg)'] / df_filtered['Protein intake (g/kg/day)']
df_filtered['Efficiency (kg/g_protein)'] = df_filtered['Efficiency (kg/g_protein)'].replace([np.inf, -np.inf], np.nan).fillna(0)
```

Figure 2: Code snippet calculating efficiency feature column

After dropping rows with missing data and scaling the selected features with *StandardScaler*, we used *KMeans clustering* to group similar interventions. We tested multiple values of k and evaluated them using the *silhouette score*, which tells us how well the data points fit within their assigned clusters compared to others. The silhouette scores showed that *k=3* had the highest value, so that's what was used for the final clustering output.

```
# Step 4: Normalize data for clustering
scaler = StandardScaler()
scaled_features = scaler.fit_transform(df_filtered.drop(columns=['Efficiency (kg/g_protein)']))

# Step 5: Silhouette score to find optimal k
silhouette_scores = {}

for k in range(2, 8):
    kmeans = KMeans(n_clusters=k, random_state=42)
    labels = kmeans.fit_predict(scaled_features)
    score = silhouette_score(scaled_features, labels)
    silhouette_scores[k] = score

best_k = max(silhouette_scores, key=silhouette_scores.get)
print(f"Best k (based on silhouette score): {best_k}")

# Step 6: Final clustering using best_k
kmeans_final = KMeans(n_clusters=best_k, random_state=42)
df_filtered['Cluster'] = kmeans_final.fit_predict(scaled_features)

# Step 7: Group by cluster and summarize
cluster_summary = df_filtered.groupby('Cluster').agg({
    'Duration (weeks)': 'mean',
    '(times/week)': 'mean',
    'Energy intake (kcal/kg/day)': 'mean',
    'Protein intake (g/kg/day)': 'mean',
    'LBM change (kg)': ['mean', 'max'],
    'Efficiency (kg/g_protein)': ['mean', 'max']
})

# Display cluster summary
print("\nCluster Summary:")
print(cluster_summary)
```

Figure 3: Code showing silhouette score calculation

```
Best k (based on silhouette score): 3

Cluster Summary:
        Duration (weeks) (times/week) Energy intake (kcal/kg/day)  \
                    mean         mean                         mean
Cluster
0              31.750000     2.833333                   28.256667
1              10.790323     3.112903                   28.860000
2               8.736842     4.578947                   38.981053

        Protein intake (g/kg/day) LBM change (kg)       \
                             mean            mean  max
Cluster
0                        1.242500        1.329250  3.3
1                        1.399032        1.209615  4.3
2                        1.915263        1.635421  3.9

        Efficiency (kg/g_protein)
                             mean        max
Cluster
0                        1.051806   2.408759
1                        1.177592  19.545455
2                        0.915827   2.294118
```

Figure 4: Cluster summary output

To further test whether the clusters made sense visually, we used Principal Component Analysis (PCA) to reduce the dimensions of the data and plotted the clusters in 2D. PCA works by compressing several related features (like protein intake, duration, and LBM change) into a smaller number of new variables—called principal components—that still capture the most important patterns in the data. Instead of plotting the original five features separately, we reduced them into two new axes (PC1 and PC2) that represent the strongest directions of variance across all the groups.

- PC1 can be thought of as a mix of input-heavy factors like protein intake and energy.
- PC2 may relate more to outcomes or efficiency, like how much muscle was gained relative to those inputs.

Each dot on the scatterplot represents one group from a study. Its location is based on how it scores along these two new combined axes. Dots that fall near each other had similar overall patterns in the original training and nutrition features—even if those original numbers differed slightly.

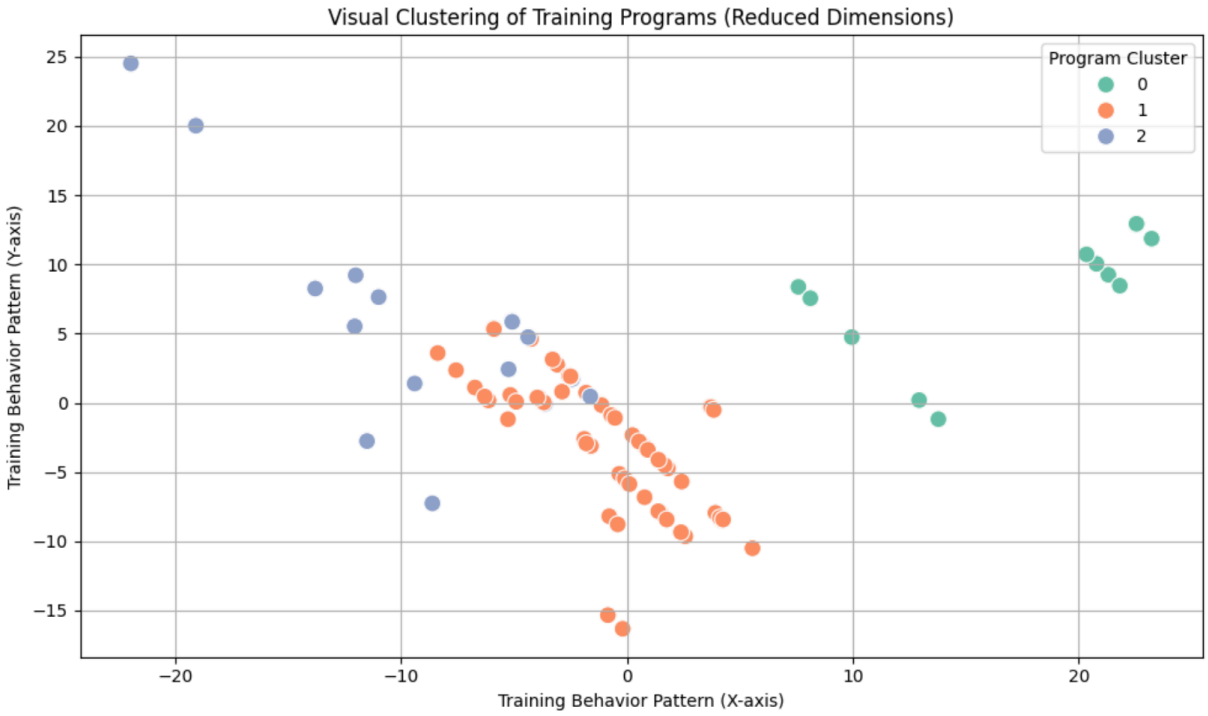

Figure 5: Clustering Scatter plot

This step doesn't tell us exactly *what* the components mean numerically, but it helps us *see* the relationships between groups. If the clusters stayed separated after PCA, that confirmed our unsupervised model was finding meaningful structure. It also let us visually spot outliers, like the top left point in *Figure 5* that performed well despite very low input values

Once the clustering was done, we used boxplots to compare protein intake and LBM change across the clusters. This gave us a clearer picture of how each group behaved and helped validate that the clusters weren't just random splits, but actually reflected different intervention profiles.

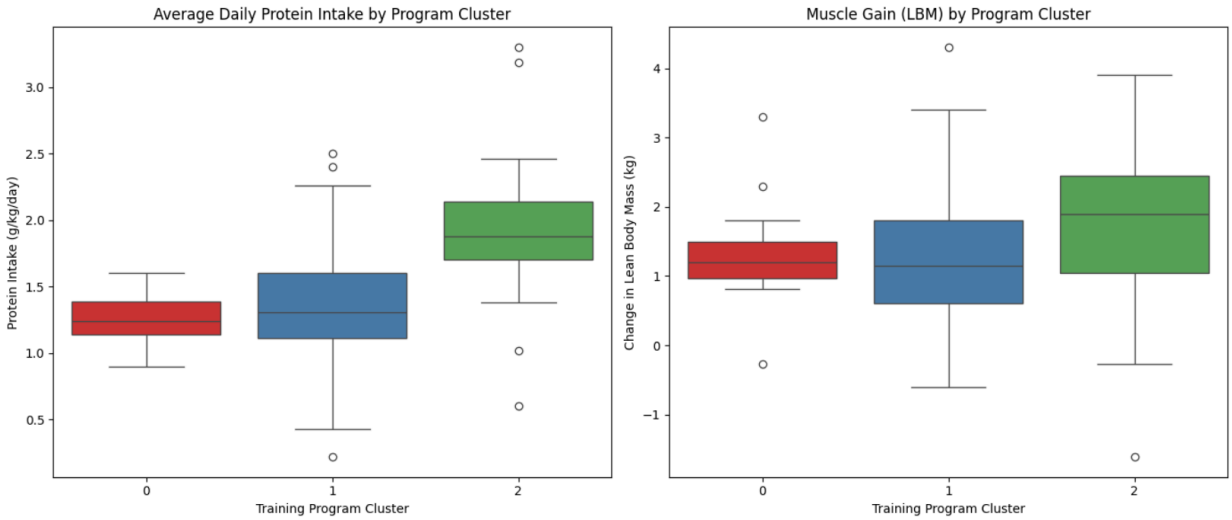

Figure 6: Boxplots showing Protein Intake and LBM Change by cluster

So overall, our success criteria were built into every step: we used silhouette scores to measure clustering quality, PCA to visually test the separation of groups, and the variation in efficiency and intake values across clusters to confirm that the model produced meaningful output. The method was considered successful if it revealed clear, consistent groupings that aligned with training and nutrition variables without needing labeled data.

Results:

For unsupervised learning projects:

What results did your analysis show? Visualize them if possible

What new questions do these results raise, and how can they be addressed by further analysis?

Repeat as necessary

Our clustering revealed three clear profiles of training and nutrition interventions, each associated with different levels of lean body mass (LBM) change and efficiency. The most captivating finding came from Cluster 2, which consistently showed strong muscle gain with moderate protein intake and solid training frequency. This group stood out as the most efficient—getting better results without the highest inputs. That challenges the common assumption that more protein always equals more growth.

In contrast, Cluster 0 featured groups that trained over longer durations, often with average or moderate inputs, but didn't show proportionally greater muscle gain. This

suggests that more time spent training doesn't necessarily equal more results, especially if intake or program structure isn't optimized.

Cluster 1, meanwhile, showed relatively high training frequency but low gains, hinting at inefficiency or possible underfeeding. This group had the lowest average LBM gain, making it the weakest performing cluster overall.

The results naturally lead us to ask *why certain clusters performed better or worse than others.* Specifically, what led Cluster 2 to produce the highest lean body mass gains with only moderate input, and why did Cluster 1 underperform despite frequent training? These questions pushed us to dig deeper into the CSV and examine what variables the groups in each cluster had in common. By identifying patterns such as consistent protein intake ranges, energy balance, or duration thresholds, we gained a clearer sense of what kinds of intervention combinations actually work best.

We also identified one clear outlier in the scatterplot from group 2, a group that gained a notable amount of muscle despite having very low intake and short training duration. It stood apart from all other groups in PCA space, prompting us to trace it back through the data and ask *why this group behaved so differently.* Was it a measurement anomaly, or did it reflect something physiological like newbie gains? Nonetheless, it didn't significantly change the broader trends observed across the rest of the dataset.

Overall, the results show that clustering can meaningfully separate intervention types and expose which combinations of inputs align with better performance and efficiency of LBM gains.

Discussion:

What outcome did you expect from your results?

How did your actual results differ from your expected results?

We didn't go into this with rigid expectations—we were more interested in letting the data speak for itself and seeing what kinds of conclusions we could draw about muscle growth efficiency based on real intervention data. That said, our assumptions loosely mirrored what's generally accepted in the fitness community: that protein intake, training frequency, and duration all contribute to hypertrophy, with most people defaulting to the idea that 1g/kg/day of protein is enough. Our results challenged that a bit. The most effective cluster showed that **~1.5–2.0g/kg/day** of protein, paired with **consistent training**, was associated with greater muscle gain. That outcome aligned with what we

personally believe—that going slightly above the minimum can have meaningful benefits for growth, especially in resistance training contexts.

We weren't surprised by the results in that sense, but it was validating to see the data support that viewpoint in such a clear and structured way. It didn't just confirm some of the theory, it also helped us quantify what better efficiency actually looks like.

If your final report differs from your proposed project, discuss the differences, why you made certain changes, and the bottlenecks that prevented you from proceeding with the proposed project.

Yes, our final report differs significantly from our original proposal. This project idea was actually our original concept during the early brainstorming phase, but we ran into challenges finding a usable dataset. We had also considered using wearable technology to track our own data, scraping websites like MyFitnessPal, or accessing public APIs. However, we quickly realized that while these platforms contain tons of valuable data, most of it is private thus inaccessible to us, and using it would violate user privacy. That made the project much harder to pursue.

As a result, we pivoted to a different idea focused on improving YouTube's Explore algorithm, but it wasn't something we felt as passionate about. Once you mentioned we could script our own dataset, we decided to revisit our original idea, figuring we could fabricate sample numbers just to practice the skills. Fortunately, we ended up finding a perfect real-world dataset on Kaggle that aligned almost exactly with our goals. From there, it was just a matter of diving in and getting to work.