B565 - Data Mining

Project Report On

EXERCISE PERFORMANCE ANALYSIS USING DATA MINING TECHNIQUES

SUBMITTED BY

Satyajeet Sarfare (ssarfare)

Kiran Karandikar (kikarand)

May 2^{nd} , 2022

Abstract

Strength Training techniques have been used in tailored routines during personalized training programs for athletes. During recent times, there has been a shift from tried and tested to evidence-based training routines. But with more advancement in the field of ML and IOT, more companies like HUDL, LongoMatch are leveraging this domain to gain more insights into the field of sports and we think this will be the same in the field of Strength Training.

In future, we think the ML model and a human expert will work together to make decisions productively for realistic short-term goal to long-term fruition. We intend to use historical work out data to gain insights about how to efficiently one can plan his future workout sessions. In our current project we discuss the current data gathering techniques used by athletes, processing methods for this data to be used in the model and proof-of-concept system for strength training based on Progressive Overload.

Key-Words: Strength Training, Progressive Overload, Hypertrophy, proof-of-concept, Volume, Repetitions, Sets, Trend Analysis

Chapter 1

Introduction

The main technique emphasized in our strength training is Progressive Overload. Progressive overload is one of the key factors when it comes to reaching one's strength or hypertrophy goals. Progressive overload takes multiple forms (sets, reps, time under tension, rest periods) but workout volume or exercise volume is one of the major components.

As discussed in the research done on Load Progression Strategy [2], the 3 commonly used techniques in progressive overloading are Volume, Intensity and Density.Where Volume is the number of sets or repetitions, intensity is the weight lifted and density is alternate rest periods. Here, the prediction will be done on feature Volume - a continuous feature. Volume is given by the formula:

$$Volume(V) = Sets \times Reps \times Weight$$

Able to predict, one's progression in terms of workout volume helps in fine tuning the exercise selection and programming. By examining one's historical workout log, we can predict future performances or drop in performance.

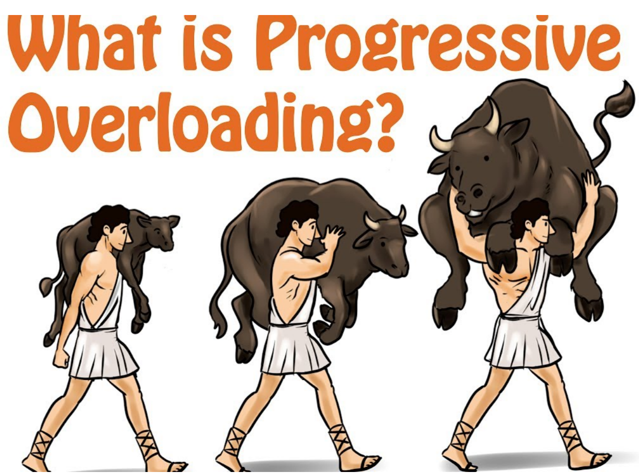

Figure 1.1: Milo Of Croton Story

One such interesting story which demonstrates the true effect of progressive overload is about a wrestler Milo of Croton. His full story can be accessed by clicking on the photo above.

Chapter 2

System Implementation and Methods

The data on which we consider to predict the feature Volume, a continuous target variable, is Time series based, we intend to perform time series and trend analysis to find correlations before training the model.

Since the problem statement involves time series data and prediction of continuous variable, we will be using following regression models.

1. Linear Regression
2. Ridge and lasso
3. SGD Regressor
4. ElasticNet

Hyperparameter tuning using gridsearchCV will be used and cross validation scores will be calculated using "modelselection.TimeSeriesSplit"

These are the following loss functions used to track accuracy :

1. mean absolute error
2. mean squared error
3. mean absolute percentage error

Below workflow has been used in order to efficiently plan and generate the desired results :

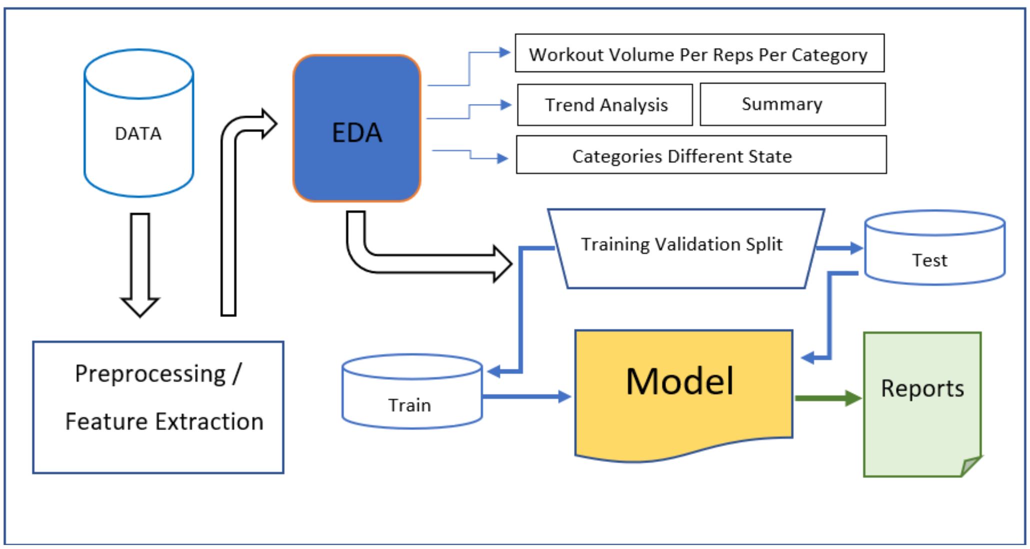

Figure 2.1: Workflow pipeline

System Implementation and its workflow mainly consists following things:

1. Data Assessment and loading :

   The data used for our project consists of personal exercise log which was tracked using Android App Fitnote. The Historical exercise log from 2019-2022. Time series based data with continuous target variable(Volume) with similar characteristics like stock market data. The data consists of 9 features with some columns consisting of null and NaN values. The data was been cleaned and proceeded for modelling purpose in the next step.

2. EDA

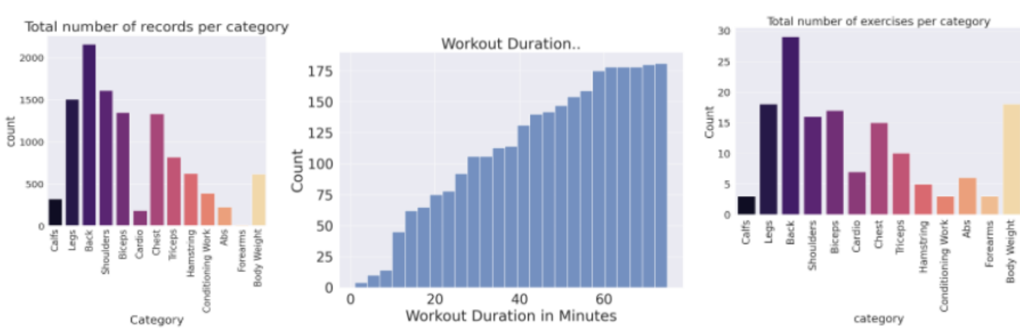

Figure 2.2: Categorical Analysis

Since the data consists of multiple exercise related to different body parts, EDA was performed to find the correlation among different exercise wrt different body-parts (figure 2.2). Since the personal log also contained data pertaining to Cardio exercise, which had features such as distance, time which are not related to strength training. They had NaN values in metric columns which were transformed to zero as part of EDA. Furthermore, QC tests of 'Not Null' are implemented on critical aspects.

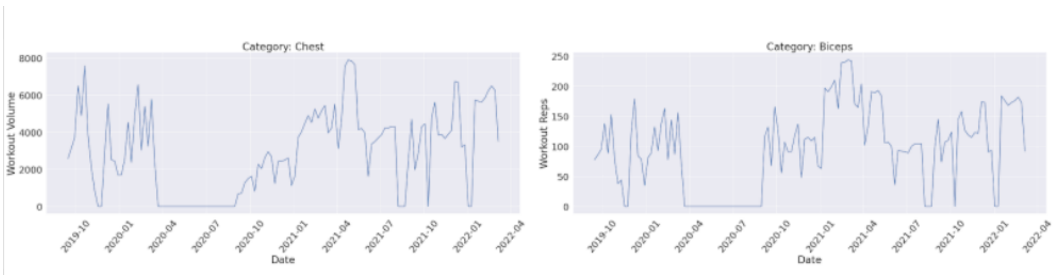

Figure 2.3: Correlation Analysis

Trend analysis was also performed on different muscle category using the library statsmodels tsa.seasonal.seasonal decompose.

Figure 2.4: Trend Analysis

3. Feature Engineering

   Since the data was based according to time, we had to aggregate data according to date and different categories associated with body parts. This was done so that the data could be adapted according to the model requirements. This aggregation had been done to make sure the volume was calculated per week per category type.

```
by_date_cat_resampled_sum = glog.set_index("Date").groupby("Category").resample("W").sum()
```

Figure 2.5: Data Aggeregation

   In the Feature Union Pipeline, we have used Standard Scaler for numerical data and One Hot Encoder for Categorical Data like calf, legs, etc.

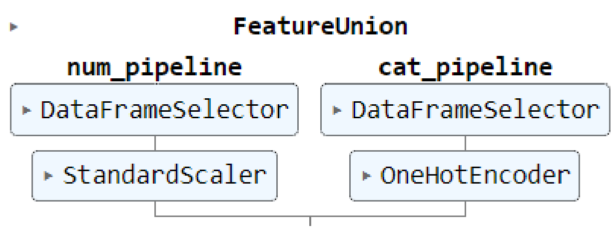

Figure 2.6: Feature Union

4. Gridsearch and Hyperparameter Tuning

   Here we have used baseline model as linear regression. As discussed before, we used the following loss functions to track accuracy such as Mean Square Error, Root Mean Square Error, Mean Absolute Error, Mean Absolute Percentage Error.

During Hyperparameter Tuning, multiple classifiers with different values were tested, namely Linear Regression, Ridge, Lasso, SGD Regressor and Elastic Net . All of them use Standard Scaler for normalization.

5. Deep Learning (Using Regression)

   Since this was a proof of concept, we have added this as an experimental section to implement deep learning model with regression as base. For the model we used Relu as Activation function , Loss function as Mean Square Error and have used Stochastic Gradient Descent as Optimizer.

   Learning Rate for SGD is set to 0.00001

```
model = BaseRegression(in_features=D_in)
loss_fn = torch.nn.MSELoss(reduction='mean')
optimizer = optim.SGD(model.parameters(), lr=0.00001)
```

Figure 2.7: Deep Learning Model Parameters

Chapter 3

Results

Even with this being a Proof of concept project, the idea shows promising results with both the base and deep learning model. The DL model is better optimised for the given data, as the MSE for DL model is much lesser than that of the base model.

After training the model, we find that the best parameters for the base model turn out to be Linear Regression as classifier and Standard Scaler for Scaling. Various Model Scores can be seen in the figure below.

```
Average model scores:
MSE: 912144.806
RMSE: 894.018
MAE: 514.736
MAPE: 1.1306224850851e+17
Explained Variance Score 0.913
```

Figure 3.1: Scores For Base Model

Now coming on to the Deep Learning Model, the model was trained with 5 hidden layers and one output, which in our case would be Volume.

After training the model, we found out that the MSE Loss score was 754921 whereas for Base model, it was 912144. Since the MSE score is considered to be better if the score is less, we can say that the DL model performs better than the base model. The result for the DL model is shown below.

```
--------------------------------------------------
Model:
BaseRegression(
  (fc1): Linear(in_features=7, out_features=5, bias=True)
  (fc2): Linear(in_features=5, out_features=1, bias=True)
)
--------------------------------------------------
Epoch 0,Train MSE loss: 19256650.0
Epoch 50,Train MSE loss: 12704590.0
Epoch 100,Train MSE loss: 833351.125
Epoch 150,Train MSE loss: 727181.438
Epoch 200,Train MSE loss: 698212.312
Epoch 250,Train MSE loss: 683598.312
Epoch 300,Train MSE loss: 674557.688
Epoch 350,Train MSE loss: 668667.438
Epoch 400,Train MSE loss: 664688.125
Epoch 450,Train MSE loss: 662101.375
Epoch 500,Train MSE loss: 660412.438
Epoch 550,Train MSE loss: 659297.562
Epoch 600,Train MSE loss: 658555.938
Epoch 650,Train MSE loss: 658059.375
Epoch 700,Train MSE loss: 657724.625
Epoch 750,Train MSE loss: 657498.188
Epoch 800,Train MSE loss: 657344.125
Epoch 850,Train MSE loss: 657238.375
Epoch 900,Train MSE loss: 657165.625
Epoch 950,Train MSE loss: 657115.25
Finished Training
--------------------------------------------------
 Validation MSE loss: 906978.0
 TEST MSE loss: 754921.0
```

Figure 3.2: Scores For Deep Learning Model

Chapter 4

Future Scope

As discussed before, since this was a proof of concept, our dataset was a personal log of the training done. Even with small dataset, we could see promising results. The current model works for a single user as the training data belongs to that particular user. The future scope for our work would be to build ensemble of models for recommending every lifter. For this we would need to collect data from users/trainees coming from various backgrounds, with multiple features as used by Teikari Petteri and Aleksandra Pietrusz [1]
.

Another promising prospects would be to use multiple IOT sensors to track movement and to validate if they are correct and if how they are affecting one's strength and hypertrophy goals. Data can also be aggregated by time on which time series analysis can be performed. This would help one provide guidance on rest phases and heightened workout phases to maximize the growth.

Similar concepts can also be used in other individual sports like swimming/bike riding to determine the performance metrics.

Chapter 5

References

1. Teikari, Petteri, and Aleksandra Pietrusz. 2021. "Precision Strength Training: Data-driven Artificial Intelligence Approach to Strength and Conditioning." SportRiv. May 20. doi: 10.31236/osf.io/w734a

2. Complexity: A Novel Load Progression Strategy in Strength Training. https://doi.org/10.3389/fphys.2019.00839

3. Weekly Training Frequency Effects on Strength Gain: A Meta-Analysis. https://doi.org/10.1186/s40798-018-0149-9

4. Implementation of the goal-setting components in popular physical activity apps: Review and content analysis. DOI: 10.1177/2055207619862706

5. Progression of volume load and muscular adaptation during resistance exercise. doi:10.1007/s00421-010-1735-9

6. Time Series Analysis: Evaluating performance trends within resistance training exercise sessions.

7. https://silveredgefitness.com/milo-and-progressive-overload