Hyrox Race Performance: Analysis

Contents

1. Executive Summary

This project analysed Hyrox race performance to identify key factors influencing race completion times, providing actionable insights for male athletes in the 'pro' and 'elite' divisions to optimise their strategies. Since Hyrox launched in 2017, literature is limited. One analyst examined variability at each station to improve personal performance (Thaddeussegura, 2020). Other studies focus solely on “elite” athletes, limiting dataset comprehensiveness (Williams, 2022).

The goal was to develop a predictive tool to estimate finish times using intermediate race data, allowing athletes to mentally break down the event and strategically plan their approach. This strategy not only aids in planning for coaches and competitors, but also offers psychological advantages by focusing on shorter, achievable goals.

Correlation analysis highlighted the importance of endurance in the later stages of the race, with Runs 4, 5, 6, 7, and 8 identified as strong predictors of total finish time. Sub-1-hour finishers spend significantly less time on the Sled Push/Pull, Burpee Broad Jump, and Sandbag Lunges, indicating these as "time-saving stations" crucial for achieving faster race times.

Power BI was utilised to create interactive visualisations and dashboards, supporting dynamic analysis and strategic decision-making. This report provides an overview of the methodologies, key insights, and recommendations for future research to enhance performance strategies in Hyrox competitions. It is an accessible application allowing coaches and competitors access to insights and tools to aid their training approach.

2. Data Source and Preparation

The dataset for this project was sourced from publicly available Hyrox race results in a CSV file, offering comprehensive data but potentially subject to collection bias and missing points (LaPine, 2024).

Source: https://www.kaggle.com/datasets/jgug05/hyrox-results?resource=download

Race metrics included total time, individual run times, and workout station times, enabling the exploration of these as predictors of overall completion time.

Data preparation involved importing the data into Power BI, which efficiently handles large datasets and retains history. Headers were promoted, and appropriate data types, such as text and duration, were assigned. Rows with missing data were removed, although may lead to a loss of valuable data (Tamboli, 2021), however with a large volume of data, quality was prioritised over quantity.

The dataset was filtered for male athletes in 'elite' and 'pro' divisions to avoid skewed results due to different weights or rep volumes. Records with a zero time in the final station were excluded as non-finishers. Data was sorted by total race time, indexed, and converted into seconds for easier analysis, forming the "hyrox_times" table.

“Hryox_times” (appendix 8.1), along with “hyrox_results” (appendix 8.2) looking at the proportion of time spent at each station, ensured a robust data foundation, enabling a comprehensive exploration of race performance and the development of predictive models. A separate "StDev" table was created for summary statistics, allowing easier adjustments to standard deviations without cluttering the "hyrox_results" table.

Feature engineering involved creating new metrics like the 'correlation coefficient' and 'R2', which offered insights into how different race segments influenced overall completion time.

Additionally, potential biases due to varying course conditions and event environments were considered, this will be addressed on future iterations to maintain the volume of data for this project. Data was standardised, looking at proportions rather than raw values to ensure comparability across different finish times.

3. Analysis Documentation

3.1 Predicting Finish Time:

To evaluate the relationship between individual race segments and total race completion time, feature engineering was performed by adding correlation coefficients in Power BI (appendix 8.4).

This was then squared for R_2 value to identify segments that were the key drivers for prediction time.

Figure 1: correlation coefficients

RSqd	Name
0.67	Run 4
0.66	Run 5
0.63	Run 7
0.62	Run 6
0.61	Run 8
0.60	Burpee Broad Jumps
0.59	Sandbag Lunges
0.58	Run 3
0.55	Run 2
0.55	Rowing
0.54	Wall Balls
0.52	Sled Pull
0.50	Farmers Carry
0.42	SkiErg
0.38	Sled Push
0.26	Run 1

The R_2 values show Runs 4 and 5 have the strongest correlation with total finish time (0.67 and 0.66), making them key indicators of overall performance. In contrast, Run 1 has the weakest correlation (0.26), making it less predictive.

All segments up to Run 4 were included in a predictive model. Linear regression was chosen for its simplicity in explaining relationships between race variables. The model treated accumulated time until run4 as the independent variable and total race time as the dependent variable. Although adding individual segments improved accuracy, using a single cumulative total was more user-friendly. The cumulative time to Run 4 and final finish times were exported to Excel for linear regression analysis using the analysis tool pack to assess the P-Value only as a reference. Calculations for Intercept and slope were conducted in Power BI to keep the product internal (appendix 8.5).

Figure 2: Summary Output

SUMMARY OUTPUT

Regression Statistics	
Multiple R	0.939465186
R Square	0.882594835
Adjusted R Square	0.882580735
Standard Error	314.2526547
Observations	8329

ANOVA

	df	*SS*	*MS*	*F*	*Significance F*
Regression	1	6181881167	6181881167	62598.32926	0
Residual	8327	822330645	98754.731		
Total	8328	7004211812			

	Coefficients	*Standard Error*	*t Stat*	*P-value*	*Lower 95%*	*Upper 95%*	*Lower 95.0%*	*Upper 95.0%*
Intercept	160.2101261	19.66243949	8.148028942	4.23612E-16	121.6668504	198.7534017	121.6668504	198.7534017
run 4	2.271281964	0.00907799	250.1965812	0	2.253486845	2.289077083	2.253486845	2.289077083

The R-squared value of 0.8826 indicates that 88.26% of the variance in total finish time is explained by "[AccumulatedTime]Run 4," showing it as a strong predictor. The P-Value <0.05 confirms statistical significance.

Ethical considerations included ensuring data privacy and compliance with relevant regulations, even when using publicly available data. Reasons for selecting male competitors in “elite” categories were explained and helps to remove bias.

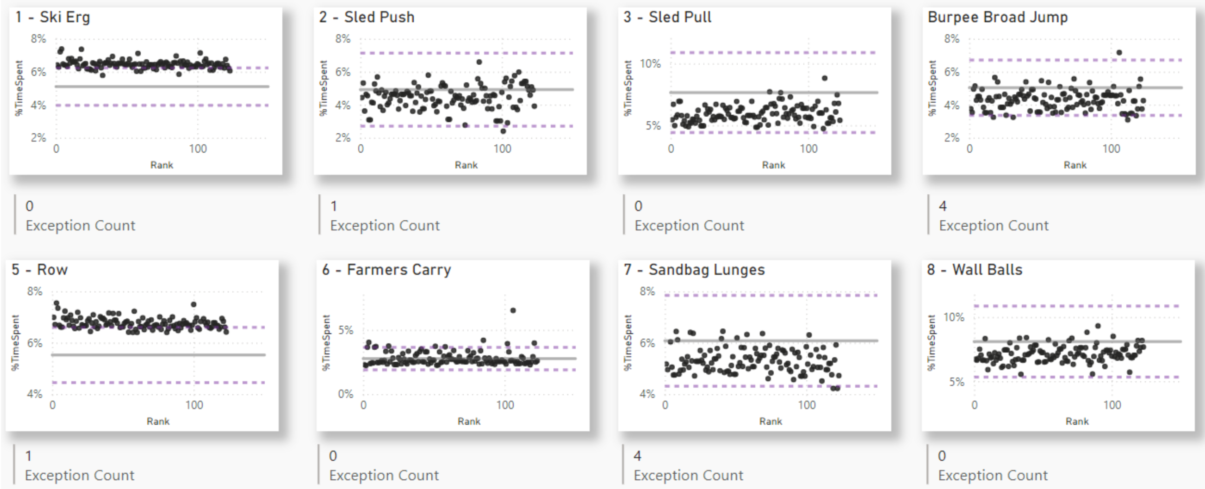
3.2 Sub 1-hour finishers

Standard deviation was calculated on percentage of time allocated to each segment to understand where sub-1-hour finishers perform exceptionally.

Initially 2 standard deviations from the mean (identification of points falling outside of 95% expectancy) was utilised however results were minimal, so this was reduced to 1.645 standard deviations (90% expectancy).

Results were minimal, however reviewing segments where data points were gathered under the average line identified segments where sub-1-hour finishers differed.

Figure 3: Exceptionally low proportion of overall time for sub 1hour finishers



Data points below the average line indicate athletes are spending less time proportionally on these segments compared to their peers. These areas may provide key opportunities for optimising performance and achieving faster overall race times.

4. Visualisations and Dashboards

Power BI was used to visualise race data and communicate key insights. Various visualisation techniques were employed to make the data accessible and understandable:

Figure 4: Landing

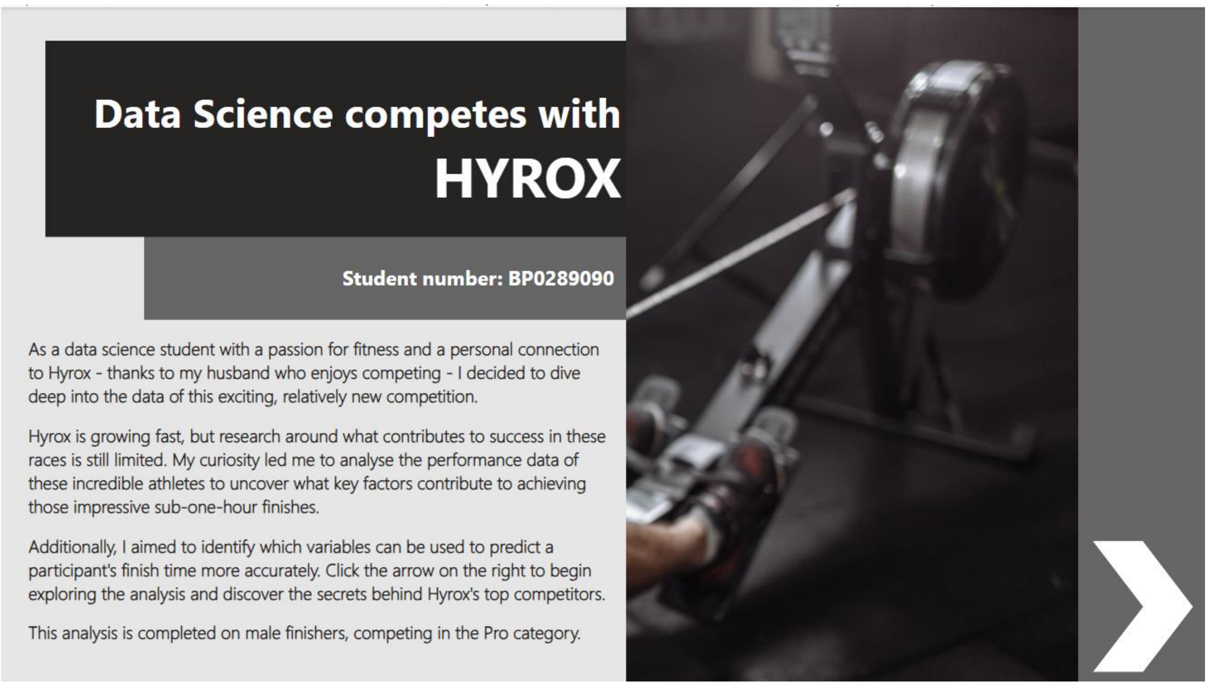


Figure 6: Summary

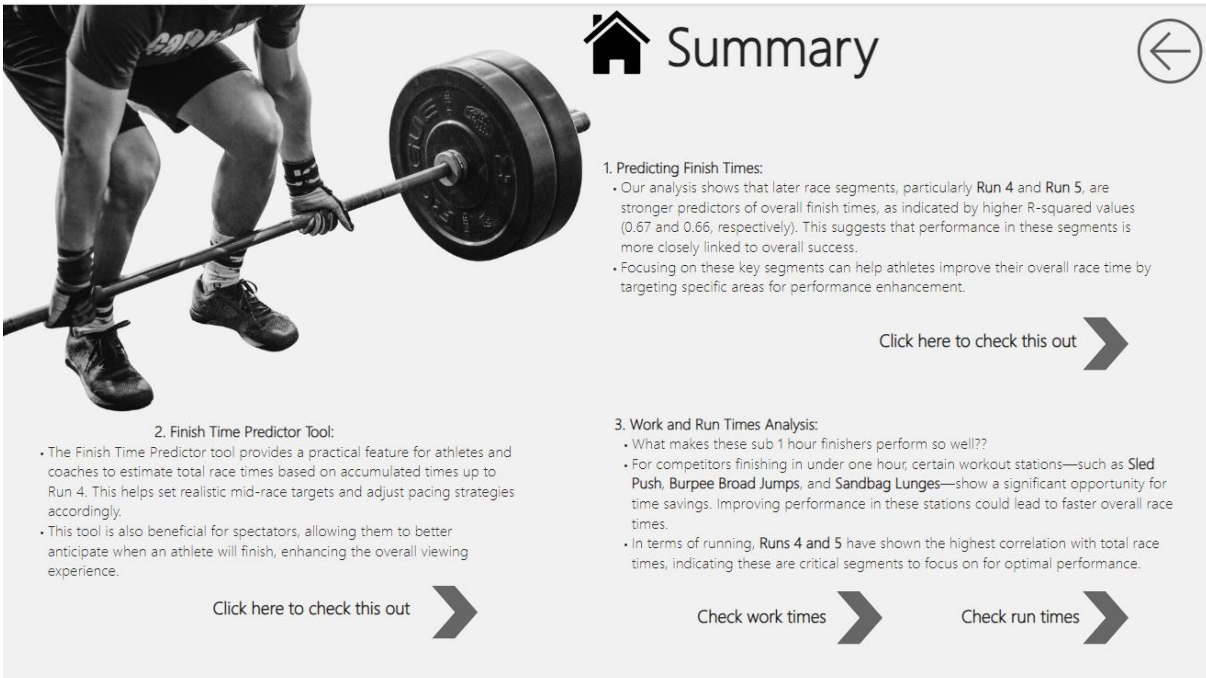


Figure 7: Predicting Times

Figure 8: Prediction Tool

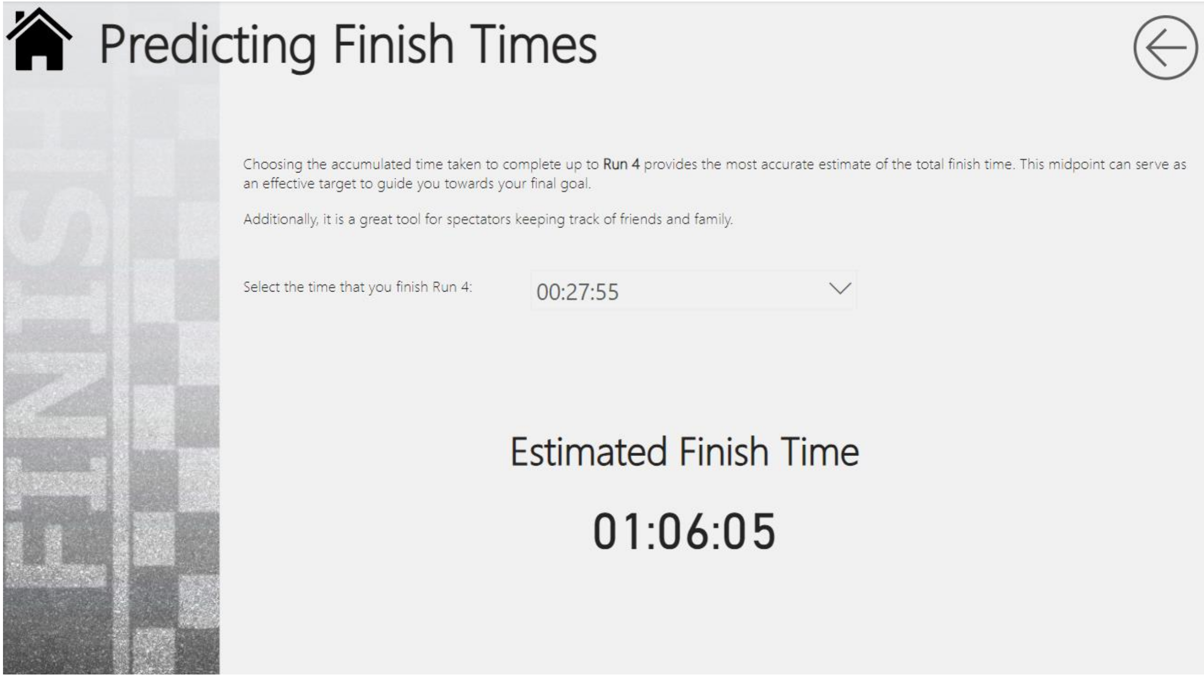
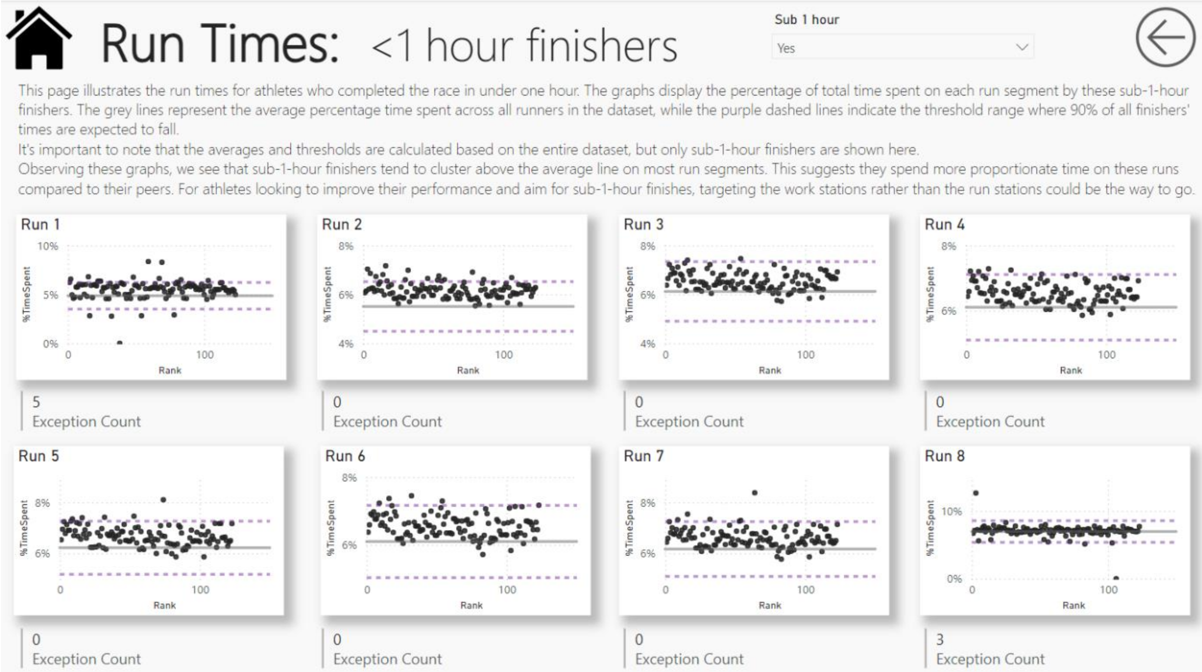


Figure 9: Work Times

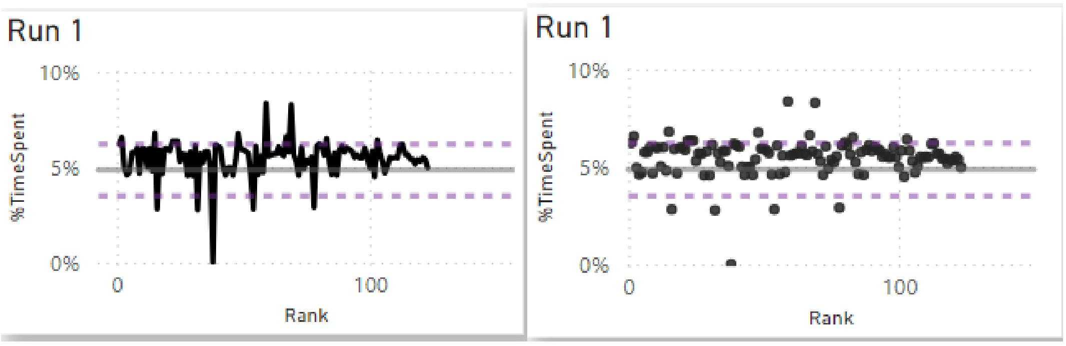
Figure 10: Run Times



Imagery and consistent colour schemes were integrated with written content to enhance storytelling and provide authenticity to the data narrative (Gemignani, 2021). Alignment and clear labelling were used to improve readability and focus. Dashboards were designed with target audience awareness, allowing professional athletes and coaches to extract actionable insights.

Scatter plots were chosen to display the time spent on each station by sub-1-hour finishers, highlighting potential time gains, particularly in stations like Sled Push, Burpee Broad Jumps, and Sandbag Lunges. Scatter plots were preferred over line graphs as they better illustrate clusters above or below average or threshold lines, making data interpretation clearer (West, 2020).

Figure 11: Line versus scatter



This approach facilitates understanding of complex data and identifies key stations for faster race times. Test users provided feedback to ensure visualisations were intuitive and informative.

5. Recommendations for Future Iterations

Demographics: Broaden the analysis to include female athletes and other divisions, offering a more comprehensive understanding of performance across all groups.

Advanced Modelling: Utilise more sophisticated machine learning techniques like random forests or neural networks to improve prediction accuracy and identify complex patterns.

Environmental Factors: Include variables like weather and terrain to better understand their impact on race performance and prevent potential bias due to course conditions

User Feedback: To refine visualisations and tools, ensuring they are intuitive and meet the needs of athletes and coaches effectively.

Continuous model refinement: based on new data extracting through web-scraping.

Stakeholder Engagement: to tailor the predictive tools to their evolving needs and provide more personalised training insights

6. Conclusion

This analysis of Hyrox race performance has provided valuable insights into the key factors that influence race completion times, particularly highlighting the importance of endurance during the latter stages of the race. The study found that Runs 4, 5, 6, 7, and 8 are the strongest predictors of final race performance, suggesting that athletes who can maintain or increase their pace during these critical segments are more likely to achieve better overall results. This underscores the need to focus on building endurance and mental resilience to sustain high performance when fatigue sets in, which could be a decisive factor for success in Hyrox races.

Furthermore, the data showed that sub-1-hour finishers tend to spend less proportionate time on workstations and more on run segments compared to their peers, emphasising that optimising time spent at workstations could be key to achieving faster overall race times. Athletes aiming to improve their performance should consider strategies that target both endurance for the latter runs and efficiency at target workstations.

The Finish Time Predictor tool, developed as part of this analysis, offers a practical application of these findings by enabling athletes and coaches to set realistic mid-race targets and adjust strategies accordingly. Future research should incorporate a more diverse dataset and explore advanced modelling techniques to refine these predictive capabilities, further supporting athletes in optimising their training and competition strategies. The findings from this study not only enhance our understanding of race dynamics but also provide a foundation for developing more targeted training programs that cater to the unique demands of Hyrox competitions.

7. References

Gemignani, Z. (2021). *Launch Customer-Facing Data Products - Juice Analytics*. [online] Launch Customer-Facing Data Products - Juice Analytics. Available at: https://www.juiceanalytics.com/writing/tips-for-using-photos-in-data-stories [Accessed 28 Jul. 2024].

LaPine, T. (2024). *Open-Source Data vs. a Paid Data Provider Like dataplor*. [online] dataplor. Available at: https://www.dataplor.com/resources/blog/pros-and-cons-of-open-source-data/#:~:text=The%20Benefits%20of%20Open%2DSource%20Data&text=Open%2Dsource%20databases%20are%20readily [Accessed 27 Jul. 2024].

Tamboli, N. (2021). *Effective Strategies for Handling Missing Values in Data Analysis*. [online] Analytics Vidhya. Available at: https://www.analyticsvidhya.com/blog/2021/10/handling-missing-value/#:~:text=Here%20are%20effective%20methods%20to [Accessed 27 Jul. 2024].

Thaddeussegura (2020). *Hacking Hyrox: Part 1 - Data Science*. [online] Data Science - One Question at a time. Available at: https://thaddeus-segura.com/hacking-hyrox/ [Accessed 27 Jul. 2024].

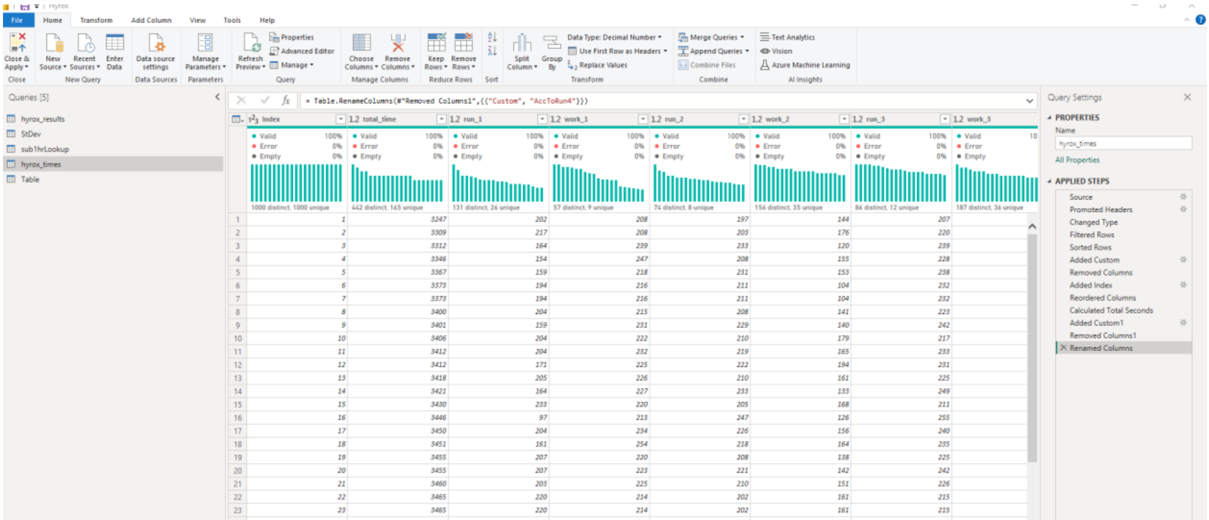
West, C. (2020). *What Is a Scatter Plot and When to Use One*. [online] Visual Learning Center by Visme. Available at: https://visme.co/blog/scatter-plot/ [Accessed 28 Jul. 2024].

Williams, G. (2022). *5 Data Points to Improve your HYROX Performance - Rox Lyfe*. [online] Rox Lyfe. Available at: https://roxlyfe.com/5-interesting-data-results-to-improve-your-hyrox-performance/ [Accessed 27 Jul. 2024].

8. Appendix

8.1 Hyrox_times table – Power BI

8.1.1 Screenshot



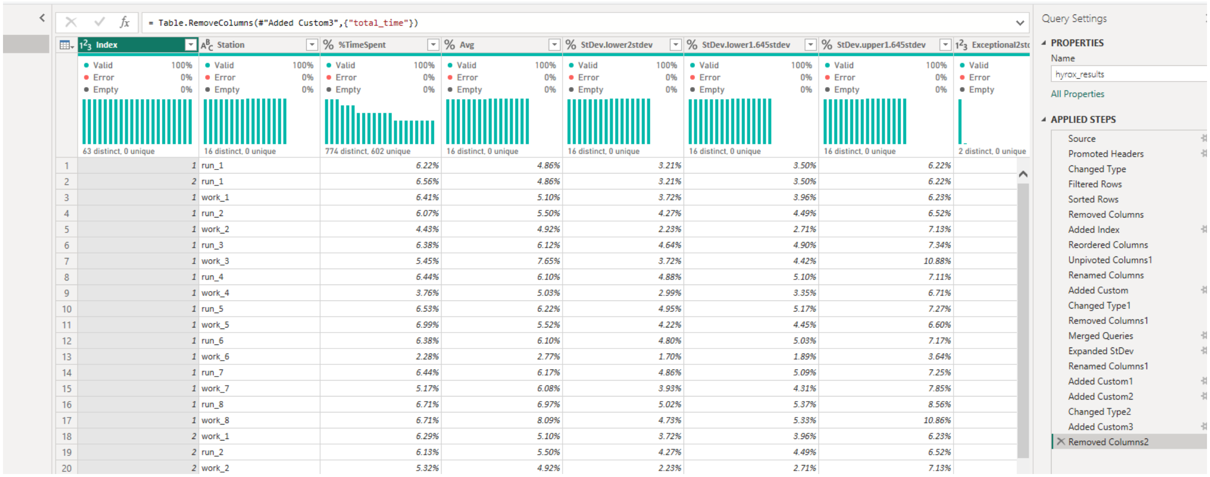
8.1.2 Script

let
  Source = Csv.Document(File.Contents("C:\Users\kiersty.rose\OneDrive - Wiltshire Council\Documents\Apprenticeship\5. Professional Practice\Hyrox\HyroxResults.csv"),[Delimiter=",", Columns=34, Encoding=65001, QuoteStyle=QuoteStyle.None]),
  #"Promoted Headers" = Table.PromoteHeaders(Source, [PromoteAllScalars=true]),
  #"Changed Type" = Table.TransformColumnTypes(#"Promoted Headers",{{"event_id", type text}, {"event_name", type text}, {"gender", type text}, {"nationality", type text}, {"age_group", type text}, {"division", type text}, {"total_time", type duration}, {"work_time", type duration}, {"roxzone_time", type duration}, {"run_time", type duration}, {"run_1", type duration}, {"work_1", type duration}, {"roxzone_1", type duration}, {"run_2", type duration}, {"work_2", type duration}, {"roxzone_2", type duration}, {"run_3", type duration}, {"work_3", type duration}, {"roxzone_3", type duration}, {"run_4", type duration}, {"work_4", type duration}, {"roxzone_4", type duration}, {"run_5", type duration}, {"work_5", type duration}, {"roxzone_5", type duration}, {"run_6", type duration}, {"work_6", type duration}, {"roxzone_6", type duration}, {"run_7", type duration}, {"work_7", type duration}, {"roxzone_7", type duration}, {"run_8", type duration}, {"work_8", type duration}, {"roxzone_8", type duration}}),
  #"Filtered Rows" = Table.SelectRows(#"Changed Type", each ([gender] = "male") and ([division] = "elite" or [division] = "pro") and [work_8] <> #time(0, 0, 0) and [work_8] <> #duration(0, 0, 0, 0)),
  #"Sorted Rows" = Table.Sort(#"Filtered Rows",{{"total_time", Order.Ascending}}),
  #"Added Custom" = Table.AddColumn(#"Sorted Rows", "Rox1-3", each [roxzone_1]+[roxzone_2]+[roxzone_3]),
  #"Removed Columns" = Table.RemoveColumns(#"Added Custom",{"roxzone_8", "roxzone_time", "roxzone_4", "roxzone_5", "roxzone_6", "roxzone_7", "event_id", "event_name", "gender", "nationality", "age_group", "division", "work_time", "run_time", "roxzone_1", "roxzone_2", "roxzone_3"}),
  #"Added Index" = Table.AddIndexColumn(#"Removed Columns", "Index", 1, 1, Int64.Type),

#"Reordered Columns" = Table.ReorderColumns(#"Added Index",{"Index", "total_time", "run_1", "work_1", "run_2", "work_2", "run_3", "work_3", "run_4", "work_4", "run_5", "work_5", "run_6", "work_6", "run_7", "work_7", "run_8", "work_8"}),
#"Calculated Total Seconds" = Table.TransformColumns(#"Reordered Columns",{{"total_time", Duration.TotalSeconds, type number}, {"run_1", Duration.TotalSeconds, type number}, {"work_1", Duration.TotalSeconds, type number}, {"run_2", Duration.TotalSeconds, type number}, {"work_2", Duration.TotalSeconds, type number}, {"run_3", Duration.TotalSeconds, type number}, {"work_3", Duration.TotalSeconds, type number}, {"run_4", Duration.TotalSeconds, type number}, {"work_4", Duration.TotalSeconds, type number}, {"run_5", Duration.TotalSeconds, type number}, {"work_5", Duration.TotalSeconds, type number}, {"run_6", Duration.TotalSeconds, type number}, {"work_6", Duration.TotalSeconds, type number}, {"run_7", Duration.TotalSeconds, type number}, {"work_7", Duration.TotalSeconds, type number}, {"run_8", Duration.TotalSeconds, type number}, {"work_8", Duration.TotalSeconds, type number}, {"Rox1-3", Duration.TotalSeconds, type number}}),
#"Added Custom1" = Table.AddColumn(#"Calculated Total Seconds", "Custom", each [run_1]+[work_1]+[run_2]+[work_2]+[run_3]+[work_3]+[run_4]+[#"Rox1-3"], type number),
#"Removed Columns1" = Table.RemoveColumns(#"Added Custom1",{"Rox1-3"}),
#"Renamed Columns" = Table.RenameColumns(#"Removed Columns1",{{"Custom", "AccToRun4"}})
in
#"Renamed Columns"

8.2 Hyrox-Results table

8.2.1 Screenshot



	Index	Station	%TimeSpent	Avg	StDev.lower2stdev	StDev.lower1.645stdev	StDev.upper1.645stdev	Exceptional2std
1	1	run_1	6.22%	4.86%	3.21%	3.50%	6.22%	
2	2	run_1	6.56%	4.86%	3.21%	3.50%	6.22%	
3	1	work_1	6.41%	5.10%	3.72%	3.96%	6.23%	
4	1	run_2	6.07%	5.50%	4.27%	4.49%	6.52%	
5	1	work_2	4.43%	4.92%	2.23%	2.71%	7.13%	
6	1	run_3	6.38%	6.12%	4.64%	4.90%	7.34%	
7	1	work_3	5.45%	7.65%	3.72%	4.42%	10.88%	
8	1	run_4	6.44%	6.10%	4.88%	5.10%	7.11%	
9	1	work_4	3.76%	5.03%	2.99%	3.35%	6.71%	
10	1	run_5	6.53%	6.22%	4.95%	5.17%	7.27%	
11	1	work_5	6.99%	5.52%	4.22%	4.45%	6.60%	
12	1	run_6	6.38%	6.10%	4.80%	5.03%	7.17%	
13	1	work_6	2.28%	2.77%	1.70%	1.89%	3.64%	
14	1	run_7	6.44%	6.17%	4.86%	5.09%	7.25%	
15	1	work_7	5.17%	6.08%	3.93%	4.31%	7.85%	
16	1	run_8	6.71%	6.97%	5.02%	5.37%	8.56%	
17	1	work_8	6.71%	8.09%	4.73%	5.33%	10.86%	
18	2	work_1	6.29%	5.10%	3.72%	3.96%	6.23%	
19	2	run_2	6.13%	5.50%	4.27%	4.49%	6.52%	
20	2	work_2	5.32%	4.92%	2.23%	2.71%	7.13%	

8.2.2 Script

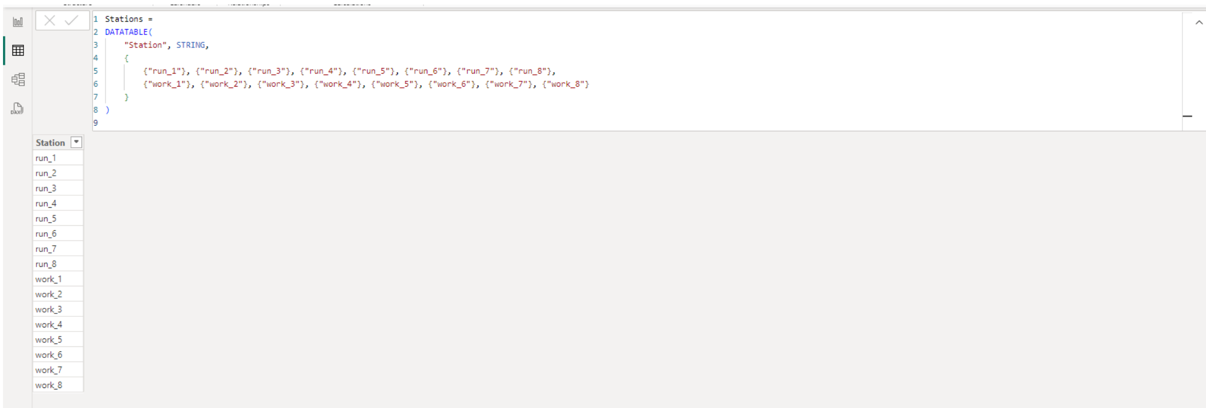
let
Source = Csv.Document(File.Contents("C:\Users\kiersty.rose\OneDrive - Wiltshire Council\Documents\Apprenticeship\5. Professional Practice\Hyrox\HyroxResults.csv"),[Delimiter=",", Columns=34, Encoding=65001, QuoteStyle=QuoteStyle.None]),
#"Promoted Headers" = Table.PromoteHeaders(Source, [PromoteAllScalars=true]),
#"Changed Type" = Table.TransformColumnTypes(#"Promoted Headers",{{"event_id", type text}, {"event_name", type text}, {"gender", type text}, {"nationality", type text}, {"age_group", type text}, {"division", type text}, {"total_time", type duration}, {"work_time", type duration}, {"roxzone_time", type duration}, {"run_time", type duration}, {"run_1", type duration}, {"work_1", type duration}, {"roxzone_1", type duration}, {"run_2", type duration}, {"work_2", type duration}, {"roxzone_2", type

```
duration}, {"run_3", type duration}, {"work_3", type duration}, {"roxzone_3", type duration},
{"run_4", type duration}, {"work_4", type duration}, {"roxzone_4", type duration}, {"run_5", type
duration}, {"work_5", type duration}, {"roxzone_5", type duration}, {"run_6", type duration},
{"work_6", type duration}, {"roxzone_6", type duration}, {"run_7", type duration}, {"work_7", type
duration}, {"roxzone_7", type duration}, {"run_8", type duration}, {"work_8", type duration},
{"roxzone_8", type duration}}),
    #"Filtered Rows" = Table.SelectRows(#"Changed Type", each ([gender] = "male") and ([division] =
"elite" or [division] = "pro") and [work_8] <> #time(0, 0, 0) and [work_8] <> #duration(0, 0, 0, 0)),
    #"Sorted Rows" = Table.Sort(#"Filtered Rows",{{"total_time", Order.Ascending}}),
    #"Removed Columns" = Table.RemoveColumns(#"Sorted Rows",{"roxzone_8","roxzone_time",
"roxzone_1", "roxzone_2", "roxzone_3", "roxzone_4", "roxzone_5", "roxzone_6",
"roxzone_7","event_id", "event_name", "gender", "nationality", "age_group",
"division","work_time","run_time"}),
    #"Added Index" = Table.AddIndexColumn(#"Removed Columns", "Index", 1, 1, Int64.Type),
    #"Reordered Columns" = Table.ReorderColumns(#"Added Index",{"Index", "total_time", "run_1",
"work_1", "run_2", "work_2", "run_3", "work_3", "run_4", "work_4", "run_5", "work_5", "run_6",
"work_6", "run_7", "work_7", "run_8", "work_8"}),
    #"Unpivoted Columns1" = Table.UnpivotOtherColumns(#"Reordered Columns", {"Index",
"total_time"}, "Attribute", "Value"),
    #"Renamed Columns" = Table.RenameColumns(#"Unpivoted Columns1",{{"Attribute", "Station"}}),
    #"Added Custom" = Table.AddColumn(#"Renamed Columns", "%TimeSpent", each
[Value]/[total_time]),
    #"Changed Type1" = Table.TransformColumnTypes(#"Added Custom",{{"%TimeSpent",
Percentage.Type}}),
    #"Removed Columns1" = Table.RemoveColumns(#"Changed Type1",{ "Value"}),
    #"Merged Queries" = Table.NestedJoin(#"Removed Columns1", {"Station"}, StDev, {"Station"},
"StDev", JoinKind.LeftOuter),
    #"Expanded StDev" = Table.ExpandTableColumn(#"Merged Queries", "StDev", {"Avg",
"lower2stdev", "lower1.645stdev", "upper1.645stdev"}, {"StDev.Avg", "StDev.lower2stdev",
"StDev.lower1.645stdev", "StDev.upper1.645stdev"}),
    #"Renamed Columns1" = Table.RenameColumns(#"Expanded StDev",{{"StDev.Avg", "Avg"}}),
    #"Added Custom1" = Table.AddColumn(#"Renamed Columns1", "Exceptional2stdev", each if
[StDev.lower2stdev] = null then null
else if [#"%TimeSpent"] < [StDev.lower2stdev] then 1 else 0),
    #"Added Custom2" = Table.AddColumn(#"Added Custom1", "Exceptional1.65stdev", each if
[StDev.lower1.645stdev] = null then null else if [#"%TimeSpent"] < [StDev.lower1.645stdev] then 1
else 0),
    #"Changed Type2" = Table.TransformColumnTypes(#"Added Custom2",{{"Exceptional1.65stdev",
Int64.Type}, {"Exceptional2stdev", Int64.Type}}),
    #"Added Custom3" = Table.AddColumn(#"Changed Type2", "Sub1hr", each if [total_time] <
#duration(0,1,0,0) then 1 else 0),
    #"Removed Columns2" = Table.RemoveColumns(#"Added Custom3",{"total_time"})
in
    #"Removed Columns2"
```

8.3 St Dev

8.4 Dynamic Correlation Coefficients

8.4.1 Adding “stations” table



8.4.2 Adding DAX to dynamically calculate correlation co-efficient

```
DynamicCorrelation =
VAR SelectedStation = SELECTEDVALUE(Stations[Station])
VAR MeanTotalTime = AVERAGE('hyrox_times'[total_time])
VAR MeanStationTime =
  SWITCH(
    SelectedStation,
    "run_1", AVERAGE('hyrox_times'[run_1]),
    "run_2", AVERAGE('hyrox_times'[run_2]),
    "run_3", AVERAGE('hyrox_times'[run_3]),
    "run_4", AVERAGE('hyrox_times'[run_4]),
    "run_5", AVERAGE('hyrox_times'[run_5]),
    "run_6", AVERAGE('hyrox_times'[run_6]),
    "run_7", AVERAGE('hyrox_times'[run_7]),
    "run_8", AVERAGE('hyrox_times'[run_8]),
    "work_1", AVERAGE('hyrox_times'[work_1]),
    "work_2", AVERAGE('hyrox_times'[work_2]),
    "work_3", AVERAGE('hyrox_times'[work_3]),
    "work_4", AVERAGE('hyrox_times'[work_4]),
    "work_5", AVERAGE('hyrox_times'[work_5]),
    "work_6", AVERAGE('hyrox_times'[work_6]),
    "work_7", AVERAGE('hyrox_times'[work_7]),
    "work_8", AVERAGE('hyrox_times'[work_8]),
    BLANK()
  )
VAR SumNumerator =
  SUMX(
    'hyrox_times',
    ([total_time] - MeanTotalTime) *
    SWITCH(
```

```
            SelectedStation,
            "run_1", [run_1] - MeanStationTime,
            "run_2", [run_2] - MeanStationTime,
            "run_3", [run_3] - MeanStationTime,
            "run_4", [run_4] - MeanStationTime,
            "run_5", [run_5] - MeanStationTime,
            "run_6", [run_6] - MeanStationTime,
            "run_7", [run_7] - MeanStationTime,
            "run_8", [run_8] - MeanStationTime,
            "work_1", [work_1] - MeanStationTime,
            "work_2", [work_2] - MeanStationTime,
            "work_3", [work_3] - MeanStationTime,
            "work_4", [work_4] - MeanStationTime,
            "work_5", [work_5] - MeanStationTime,
            "work_6", [work_6] - MeanStationTime,
            "work_7", [work_7] - MeanStationTime,
            "work_8", [work_8] - MeanStationTime,
            BLANK()
        )
    )
VAR SumDenominator =
    SQRT(
        SUMX(
            'hyrox_times',
            ([total_time] - MeanTotalTime) ^ 2
        ) *
        SUMX(
            'hyrox_times',
            SWITCH(
                SelectedStation,
                "run_1", ([run_1] - MeanStationTime) ^ 2,
                "run_2", ([run_2] - MeanStationTime) ^ 2,
                "run_3", ([run_3] - MeanStationTime) ^ 2,
                "run_4", ([run_4] - MeanStationTime) ^ 2,
                "run_5", ([run_5] - MeanStationTime) ^ 2,
                "run_6", ([run_6] - MeanStationTime) ^ 2,
                "run_7", ([run_7] - MeanStationTime) ^ 2,
                "run_8", ([run_8] - MeanStationTime) ^ 2,
                "work_1", ([work_1] - MeanStationTime) ^ 2,
                "work_2", ([work_2] - MeanStationTime) ^ 2,
                "work_3", ([work_3] - MeanStationTime) ^ 2,
                "work_4", ([work_4] - MeanStationTime) ^ 2,
                "work_5", ([work_5] - MeanStationTime) ^ 2,
                "work_6", ([work_6] - MeanStationTime) ^ 2,
                "work_7", ([work_7] - MeanStationTime) ^ 2,
                "work_8", ([work_8] - MeanStationTime) ^ 2,
                BLANK()
            )
        )
    )
RETURN
```

```
    DIVIDE(SumNumerator, SumDenominator)
```

8.5 y=mx+b DAX

8.5.1 Intercept

```
Intercept =
VAR RowCount = COUNTROWS('hyrox_times')
VAR RowSumX = SUM('hyrox_times'[AccToRun4])
VAR SumY = SUM('hyrox_times'[total_time])
VAR Slope = [Slope] // Reference to the slope measure created above

RETURN
    DIVIDE(SumY - Slope * RowSumX, RowCount)
```

8.5.1 Slope

```
Slope =
VAR Row_Count = COUNTROWS('hyrox_times')
VAR Total_AccToRun4 = SUM('hyrox_times'[AccToRun4])
VAR Total_total_time = SUM('hyrox_times'[total_time])
VAR Sum_Products = SUMX('hyrox_times', 'hyrox_times'[AccToRun4] * 'hyrox_times'[total_time])
VAR Sum_Squares_AccToRun4 = SUMX('hyrox_times', 'hyrox_times'[AccToRun4] *
'hyrox_times'[AccToRun4])

RETURN
    DIVIDE(
        Row_Count * Sum_Products - Total_AccToRun4 * Total_total_time,
        Row_Count * Sum_Squares_AccToRun4 - Total_AccToRun4 * Total_AccToRun4
    )
```

8.5.1 Linear Regression y=mx+b

```
Predicted_total_time =
VAR Slope = [Slope]
VAR Intercept = [Intercept]
VAR Seconds=    Slope * 'Parameter 2'[Parameter Value 2] + Intercept

RETURN
FORMAT(
    TIME(
        INT(Seconds / 3600),
        INT(MOD(Seconds, 3600) / 60),
        MOD(Seconds, 60)
    ),
    "hh:mm:ss"
)
```