# 3 approaches and rationale

- **Optimus**
  - Modify the G4Hunter algorithm to make it applicable for i-motifs.
    - In G4Hunter, one G has a score of 1; in a GG tract, each G is scored 2 and so on. C's get the same score but negated. To make G4Hunter applicable for the CT-based i-motifs, Optimus will be used to find the optimum scoring for each C (positive base, counterpart of G in the case of G-quadruplexes) in a given C-tract length (e.g. the score of each C in a CC tract in an i-motif). Two ways will be tried to score T; (1) T treated as negative base such that it's given the same score as C but negated (2) T treated as the other bases (A, G) such that it's scored 0.
- **Gradient boosting machines**
  - Build a machine learning model predicting the $T_m/pH_t$ of a limited sub-universe of CT-based i-motifs.
  - Get an idea of the importance of chosen features in terms of prediction.
- **Eureqa**
  - Obtain a simple analytical equation expressing $T_m/pH_t$ as a function of the chosen features.
  - Reveal relationship of features.

# Dependent variables

- **Melting temperature ($T_m$)**
- **pH at mid-transition ($pH_t$)**

# Approach A: Optimising G4Hunter Coefficients to deal with i-motifs

$T_m$

- Optimise G4Hunter coefficients for i-motifs
- C – positive base, T – negative (penalised) base

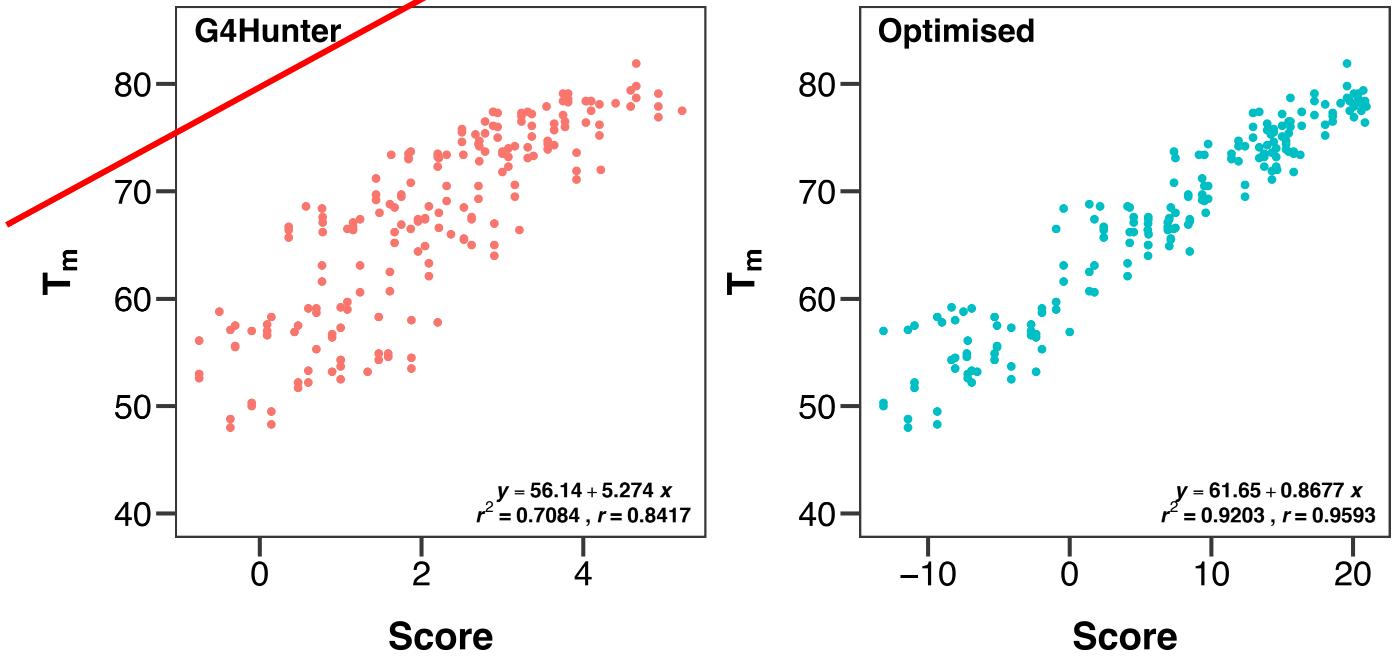
- 3 independent rounds (replicas) of optimisation were done and all arrived at the same set of coefficients/scoring scheme

- Interpreting the table: each C in CC-tract of i-motif (run length=2) is scored 19.6

- Note! May be an artefact of not having (reasonably so) C-tracts shorter than 3, hence for the minimum case only loop lengths were enough.

| | | TRACT/RUN LENGTH | | | | | |
|---|---|---|---|---|---|---|---|
| **Replica** | **Pearson's, *r*** | **1** | **2** | **3** | **4** | **5** | **6** |
| 1 | 0.9593461 | 45.1 | 19.6 | 0.0 | 15.0 | 23.6 | 28.8 |
| 2 | 0.9593461 | 45.1 | 19.6 | 0.0 | 15.0 | 23.6 | 28.8 |
| 3 | 0.9593461 | 45.1 | 19.6 | 0.0 | 15.0 | 23.6 | 28.8 |

Model: Replica 1



# Approach A: Optimising G4Hunter Coefficients to deal with i-motifs

- Optimise G4Hunter coefficients for i-motifs
- C – positive base, rest of bases are given a score of 0.

- 3 independent rounds (replicas) of optimisation were done for each method.

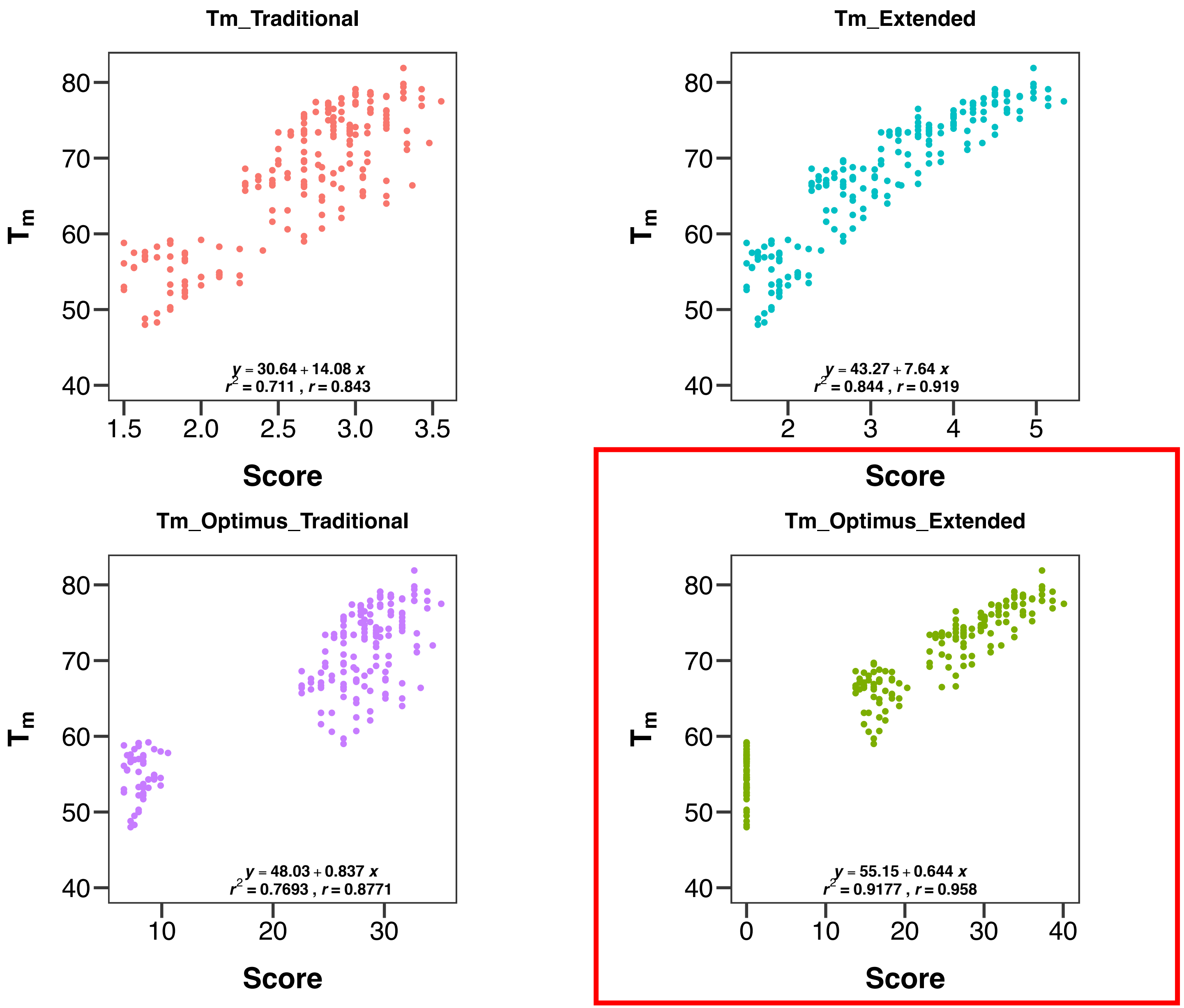
- Interpreting the table: each C in CC-tract of i-motif (run length=4) is scored 24.1.

- Note that in the given sub-universe of the C/T-based i-motifs, C-tract length range from 3 to 6 bases. Therefore, the scores in the table for C-tract lengths less than 3 do not matter in this case.

| | | | SCORE OF POSITIVE BASE PER TRACT LENGTH | | | | | |
|---|---|---|---|---|---|---|---|---|
| **Method** | **Replica** | **Pearson's, *r*** | **1** | **2** | **3** | **4** | **5** | **6** |
| **Trad. G4** | 1 | 0.8770789 | 0.3 | 1 | 13.2 | 39.5 | - | - |
| | 2 | 0.8770789 | 8.8 | 13.7 | 18.4 | 55.1 | - | - |
| | 3 | 0.8770789 | 0.1 | 6.2 | 12.8 | 38.3 | - | - |
| **Ext. G4** | 1 | 0.9579802 | 0 | 0 | 0 | 24.1 | 37.0 | 45.1 |
| | 2 | 0.9579801 | 0 | 0 | 0 | 11.6 | 17.8 | 21.7 |
| | 3 | 0.9579802 | 0 | 0 | 0 | 24.1 | 37.0 | 45.1 |

# Approach A: Optimising G4Hunter Coefficients to deal with i-motifs

$T_m$



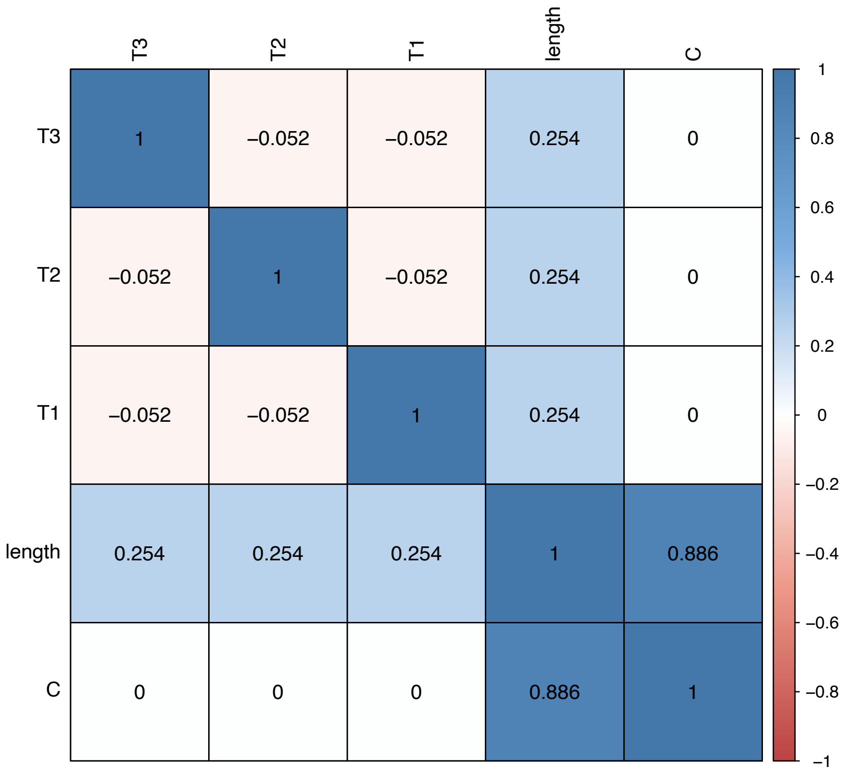
# Approach B: Feature Importance with Gradient Boosting Machines

## Chosen features of i-motif sequences

- The following features were used for the other modelling methods
  - **C –** length of C tract; [3,6]; per sequence the length of the 4 C tracts are equal
  - **T1 –** length of 1st T loop; [1,6]
  - **T2 –** length of 2nd T loop; [1,6]
  - **T3 –** length of 3rd T loop; [1,6]
  - **length** – total sequence length; [15,36]
- Dependent variable: $\mathbf{T_m}$ [48,81.9]

# Approach B: Feature Quality Control and Preprocessing

- Features not extremely correlated.



| | T3 | T2 | T1 | length | C |
|---|---|---|---|---|---|
| T3 | 1 | −0.052 | −0.052 | 0.254 | 0 |
| T2 | −0.052 | 1 | −0.052 | 0.254 | 0 |
| T1 | −0.052 | −0.052 | 1 | 0.254 | 0 |
| length | 0.254 | 0.254 | 0.254 | 1 | 0.886 |
| C | 0 | 0 | 0 | 0.886 | 1 |

- No zero- (predictor have one unique value) and near zero-variance predictors.
- Features centred and scaled.

# Approach B: XGBoost Hyperparameter/Architecture Optimisation

- All 196 sequences were used for this step.
- Features: C, T1, T2, T3, length

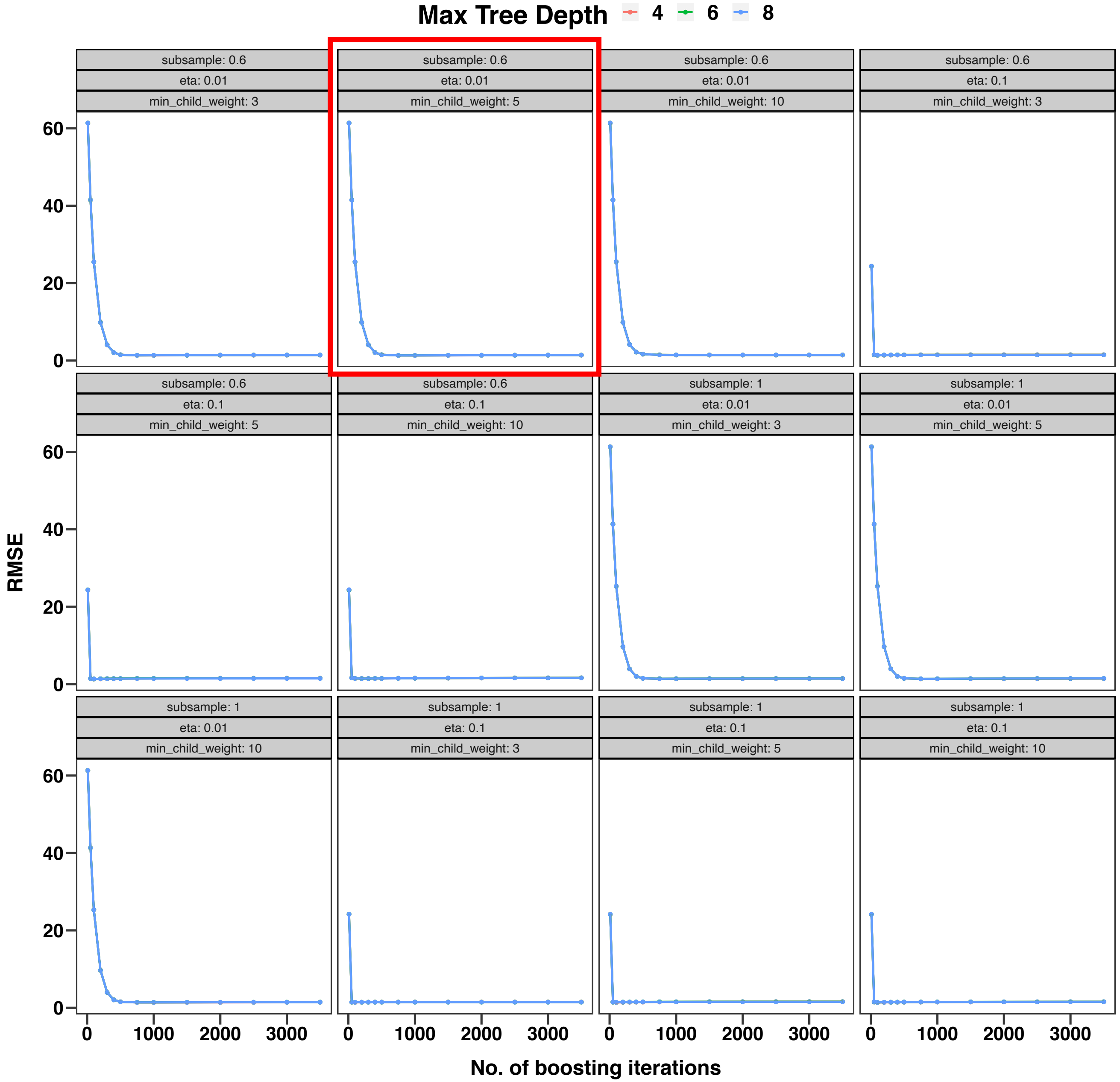
**Hyperparameter of optimal model architecture**

| Hyperparameter | value |
|---|---|
| eta | 0.01 |
| max_depth | 4 |
| Gamma* | 0 |
| colsample_bytree** | 1 |
| min_child_weight | 5 |
| subsample | 0.6 |
| nrounds | 1000 |

*No regularisation (gamma=0, default)

**All features used by every tree (colsample_bytree=1, default)

Same optimal hyperparameters obtained with and without including total length as feature.



# Approach B: XGBoost (Re)training, Performance and Feature Importance

$T_m$

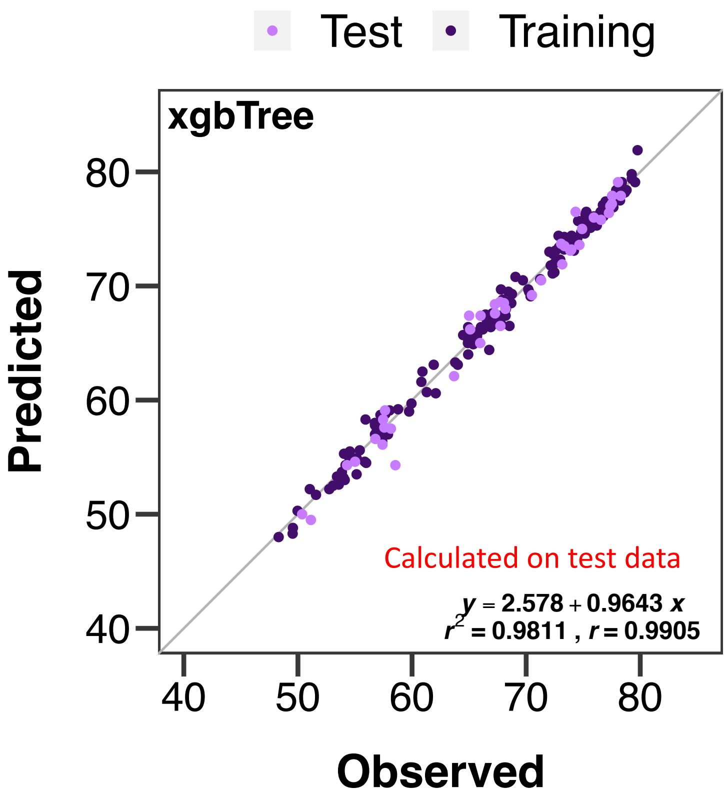
- Using the identified optimal hyperparameters, the model was tuned using 80% of data for training (randomly chosen). The retrained model was then tested on remaining 20%.

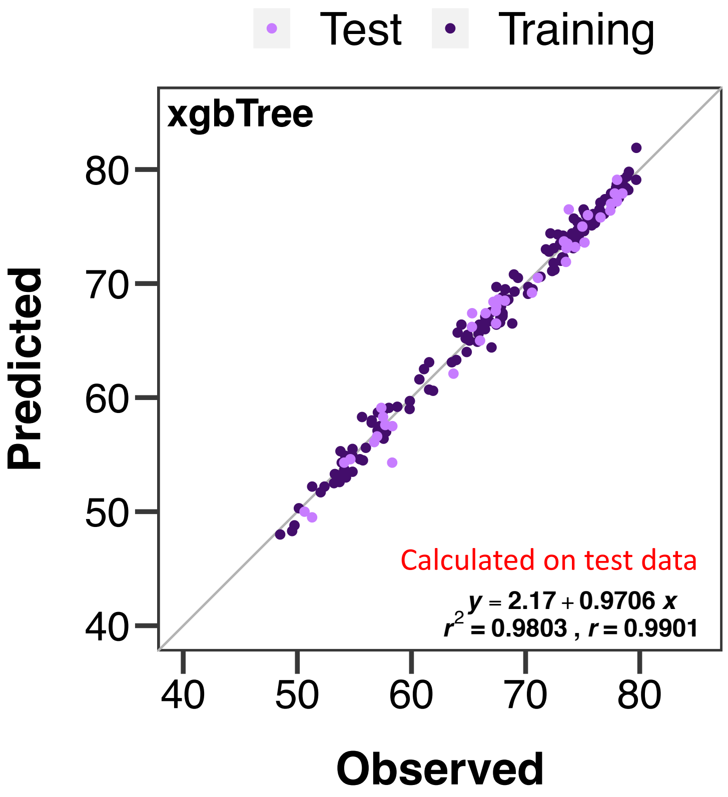
Values based on training with 80% of data:

| Metric | All features | No length |
|---|---|---|
| RMSE | 1.4011 | 1.376196 |
| $r^2$ | 0.9758311 | 0.9767373 |
| MAE | 1.135564 | 1.120195 |
| RMSESD | 0.1692699 | 0.1639031 |
| $r^2$SD | 0.005186407 | 0.005370633 |
| MAESD | 0.1139668 | 0.1081146 |

| Feature | All features |
|---|---|
| C | 100 |
| T3 | 5.123308 |
| length | 4.841326 |
| T1 | 4.253030 |
| T2 | 3.797984 |

| Feature | No length |
|---|---|
| C | 100 |
| T3 | 5.147290 |
| --- | --- |
| T1 | 4.375894 |
| T2 | 4.289519 |





Model with all features > Model without length

# Approach C: Developing Non-Linear Analytical Equation that Explains $T_m$

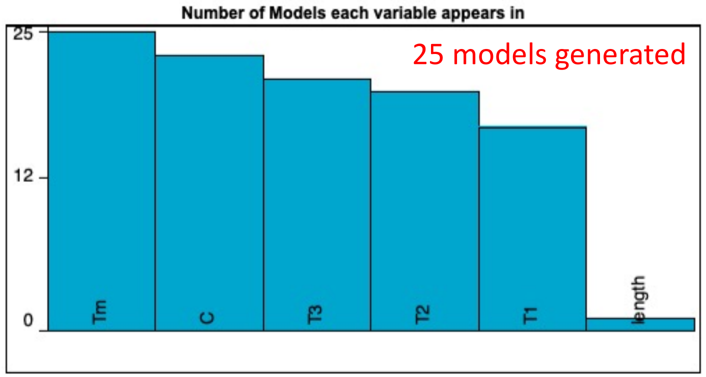
- Allowed forms: Basic (constant, input variable, ±, x, ÷) and Exponential (power, sqrt)
- Error metric is absolute error (default)
- Target expression:

$$T_m = f(length, C, T1, T2, T3)$$

SAMPLE MODEL:

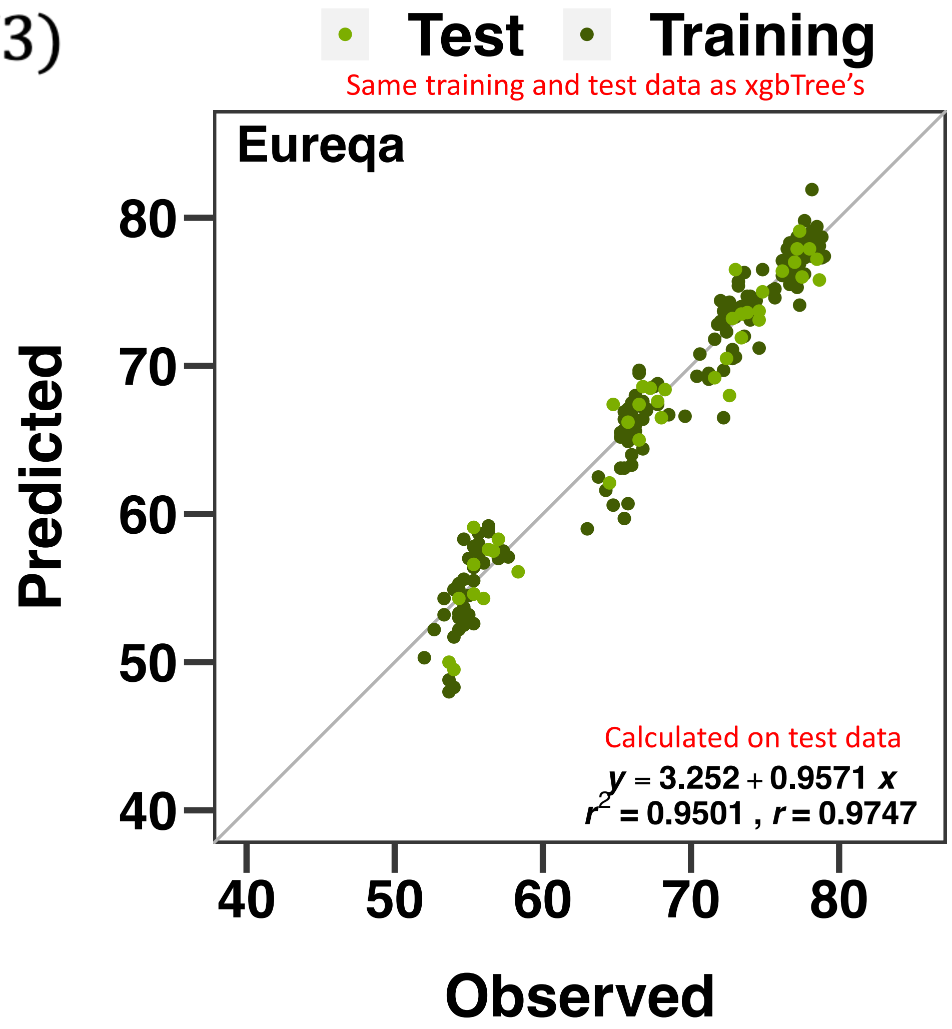
$$T_m = 102 - T3 - \frac{137 - T2*T3 + T1}{C}$$

C-run length (C) appears in almost all models obtained.



25 models generated

With features unscaled as in this case, length only appeared in the 2nd least accurate model ($T_m$=42.3 + length). After centering and scaling, it did appear but only in more complex models, complexity>27 (for reference, sample model has complexity=14).



Same training and test data as xgbTree's

Calculated on test data

# 3 approaches and rationale

- **Optimus**
  - Modify the G4Hunter algorithm to make it applicable for i-motifs.
    - In G4Hunter, one G has a score of 1; in a GG tract, each G is scored 2 and so on. C's get the same score but negated. To make G4Hunter applicable for the CT-based i-motifs,  Optimus will be used to find the optimum scoring for each C (positive base, counterpart of G in the case of G-quadruplexes) in a given C-tract length (e.g. the score of each C in a CC tract in an i-motif). Two ways will be tried to score T; (1) T treated as negative base such that it's given the same score as C but negated (2) T treated as the other bases (A, G) such that it's scored 0.
- **Gradient boosting machines**
  - Build a machine learning model predicting the $T_m/pH_t$ of a limited sub-universe of CT-based i-motifs.
  - Get an idea of the importance of chosen features in terms of prediction.
- **Eureqa**
  - Obtain a simple analytical equation expressing $T_m/pH_t$ as a function of the chosen features.
  - Reveal relationship of features.

# Dependent variables

- **Melting temperature ($T_m$)**
- **pH at mid-transition ($pH_t$)**

# Approach A: Optimising G4Hunter Coefficients to deal with i-motifs

$pH_t$

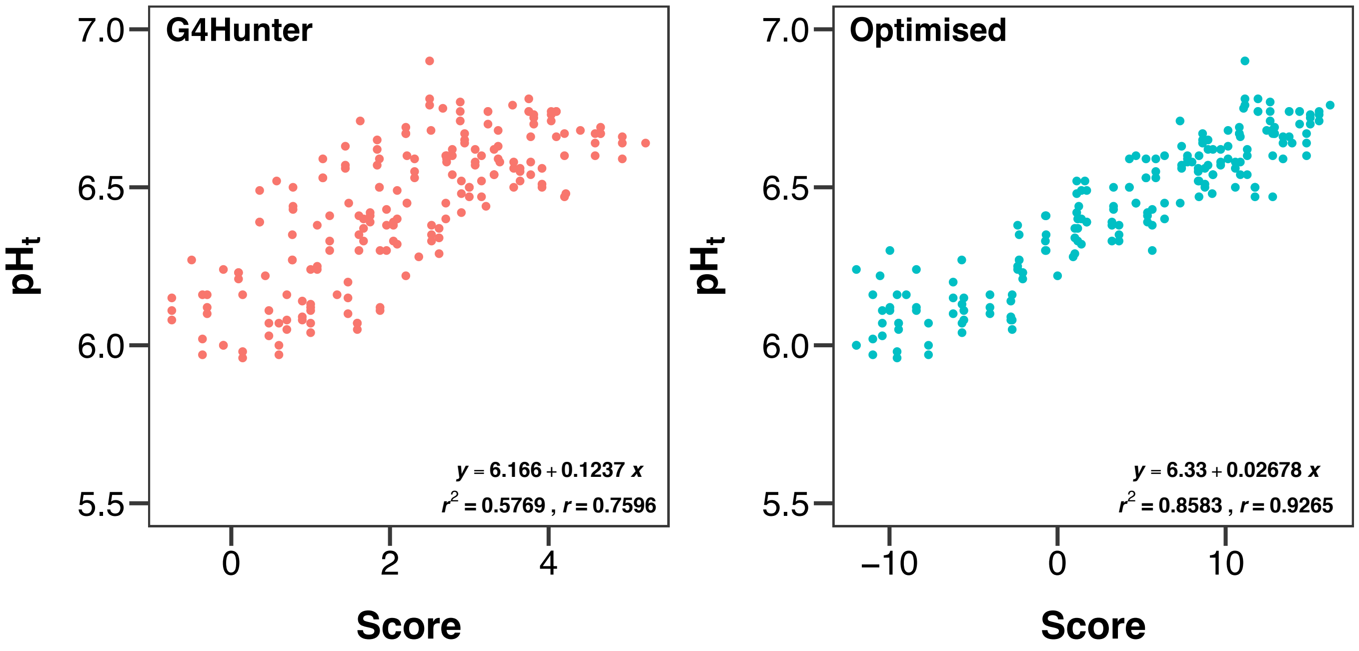
- Optimise G4Hunter coefficients for i-motifs
- C – positive base, T – negative (penalised) base

- 3 independent rounds (replicas) of optimisation were also done. Replicas 1-2 arrived at the same, slightly better set of coefficients/scoring scheme

- Same as in $T_m$’s

| | | TRACT/RUN LENGTH | | | | | |
|---|---|---|---|---|---|---|---|
| **Replica** | **Pearson’s, *r*** | **1** | **2** | **3** | **4** | **5** | **6** |
| 1 | 0.9264508 | 52.8 | 27.0 | 0.0 | 11.4 | 18.5 | 22.3 |
| 2 | 0.9264508 | 52.8 | 27.0 | 0.0 | 11.4 | 18.5 | 22.3 |
| 3 | 0.9264507 | 41.7 | 21.3 | 0.0 | 9.0 | 14.6 | 17.6 |

Model: Replica 1



# Approach A: Optimising G4Hunter Coefficients to deal with i-motifs

- Optimise G4Hunter coefficients for i-motifs
- C – positive base, rest of bases are given a score of 0.

- 3 independent rounds (replicas) of optimisation were done for each method.

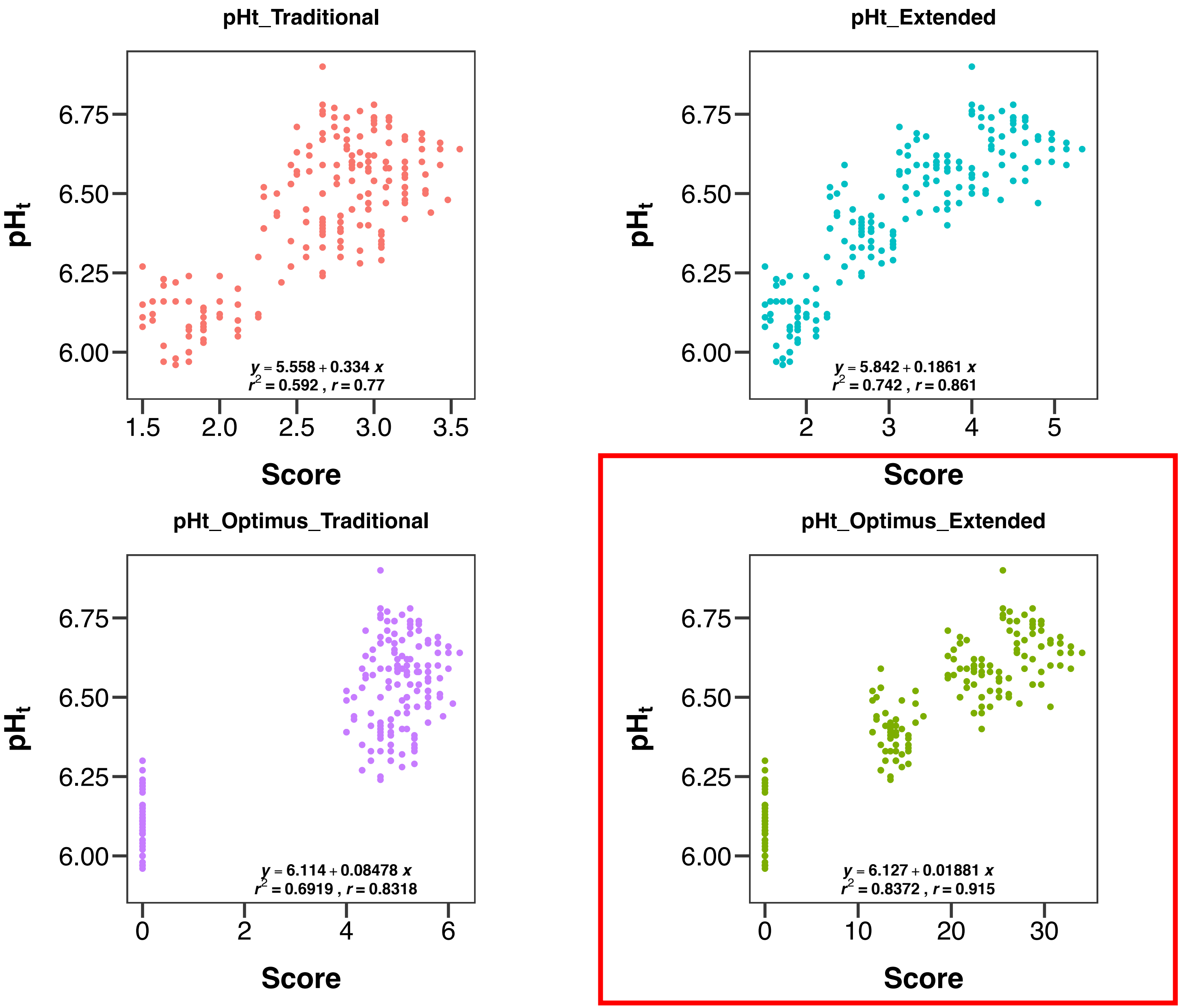
- Interpreting the table: each C in CC-tract of i-motif (run length=4) is scored 24.1.

- Note that in the given sub-universe of the C/T-based i-motifs, C-tract length range from 3 to 6 bases. Therefore, the scores in the table for C-tract lengths less than 3 do not matter in this case.

| | | | SCORE OF POSITIVE BASE PER TRACT LENGTH | | | | | |
|---|---|---|---|---|---|---|---|---|
| **Method** | **Replica** | **Pearson's, *r*** | **1** | **2** | **3** | **4** | **5** | **6** |
| **Trad. G4** | 1 | 0.8317956 | 0 | 0 | 0 | 7 | | |
| | 2 | 0.8317956 | 0 | 0 | 0 | 7 | | |
| | 3 | 0.8317956 | 0 | 0 | 0 | 7 | | |
| **Ext. G4** | 1 | 0.9150095 | 0 | 0 | 0 | 20.2 | 31.4 | 38.3 |
| | 2 | 0.9150094 | 0 | 0 | 0 | 11.7 | 18.2 | 22.2 |
| | 3 | 0.9150095 | 0 | 0 | 0 | 19.3 | 30.0 | 36.6 |

$pH_t$

# Approach A: Optimising G4Hunter Coefficients to deal with i-motifs



pHt_Traditional

pHt_Extended

pHt_Optimus_Traditional

pHt_Optimus_Extended

# Approach B: Feature Importance with Gradient Boosting Machines

## Chosen features of i-motif sequences

- The following features were used for the other modelling methods
  - **C –** length of C tract; [3,6]; per sequence the length of the 4 C tracts are equal
  - **T1 –** length of 1st T loop; [1,6]
  - **T2 –** length of 2nd T loop; [1,6]
  - **T3 –** length of 3rd T loop; [1,6]
  - **length** – total sequence length; [15,36]
- Dependent variable: **pHt** [5.96, 6.9]

# Approach B: XGBoost Hyperparameter/Architecture Optimisation

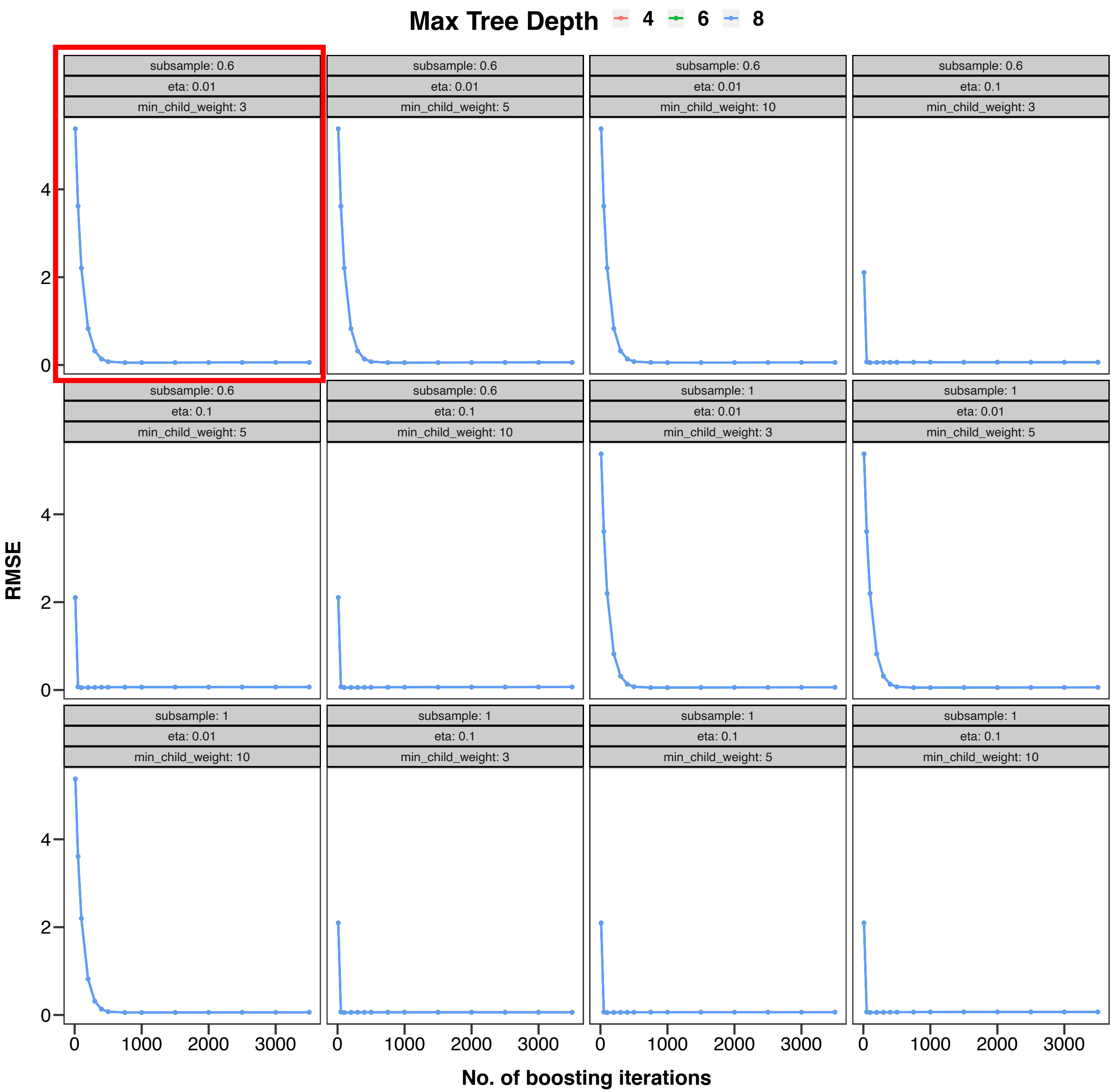
- All 196 sequences were used for this step.
- Features: C, T1, T2, T3, length

**Hyperparameter of optimal model architecture**

| Hyperparameter | All features | No length |
|---|---|---|
| eta | 0.01 | 0.01 |
| max_depth | 8 | 6 |
| Gamma* | 0 | 0 |
| colsample_bytree** | 1 | 1 |
| min_child_weight | 3 | 10 |
| subsample | 0.6 | 0.6 |
| nrounds | 750 | 1500 |

*No regularisation (gamma=0, default)

**All features used by every tree (colsample_bytree=1, default)



# Approach B: XGBoost (Re)training, Performance and Feature Importance

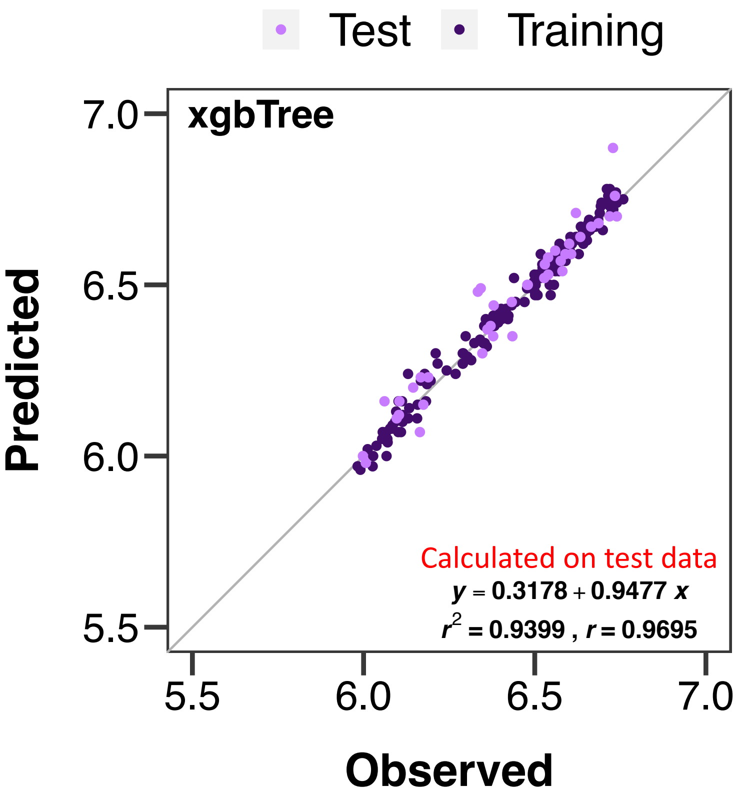
- Using the identified optimal hyperparameters, the model was tuned using 80% of data for training (randomly chosen). The retrained model was then tested on remaining 20%.

Values based on training with 80% of data:

| Metric | All features | No length |
|---|---|---|
| **RMSE** | 0.05632807 | 0.05399737 |
| **$r^2$** | 0.9424578 | 0.9464987 |
| **MAE** | 0.04353706 | 0.04250962 |
| **RMSESD** | 0.008222115 | 0.007408617 |
| **$r^2$SD** | 0.0180082 | 0.01318266 |
| **MAESD** | 0.007648299 | 0.005437334 |

| Feature | All features |
|---|---|
| **C** | 100 |
| **length** | 53.124830 |
| **T1** | 11.133761 |
| **T3** | 10.844397 |
| **T2** | 8.921195 |

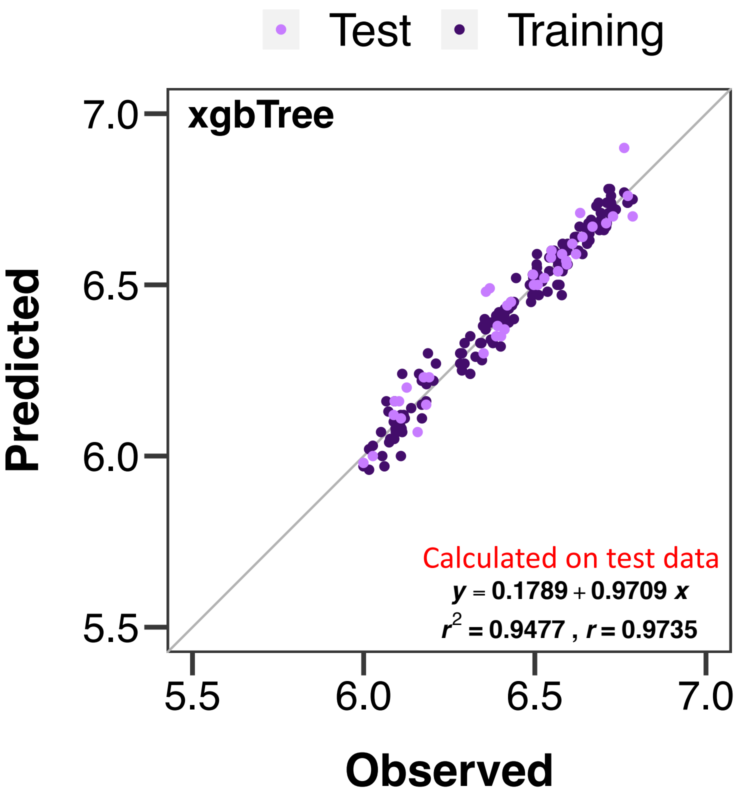
| Feature | No length |
|---|---|
| **C** | 100 |
| --- | --- |
| **T2** | 12.234725 |
| **T1** | 11.778001 |
| **T3** | 9.140414 |



Calculated on test data

$y = 0.3178 + 0.9477\,x$

$r^2 = 0.9399$ , $r = 0.9695$



Calculated on test data

$y = 0.1789 + 0.9709\,x$

$r^2 = 0.9477$ , $r = 0.9735$

Model with all features < Model without length

# Approach C: Developing Non-Linear Analytical Equation that Explains $pH_t$

- Allowed forms: Basic (constant, input variable, ±, x, ÷)
- Error metric is absolute error (default)
- Target expression:

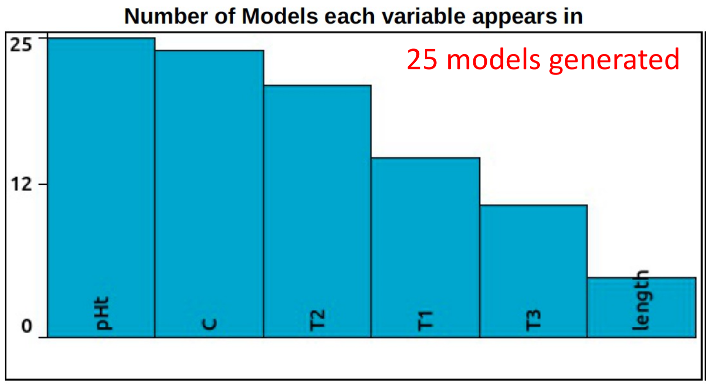
$$pH_t = f(length, C, T1, T2, T3)$$

SAMPLE MODEL:

length = 4*C + T1 + T2 + T3

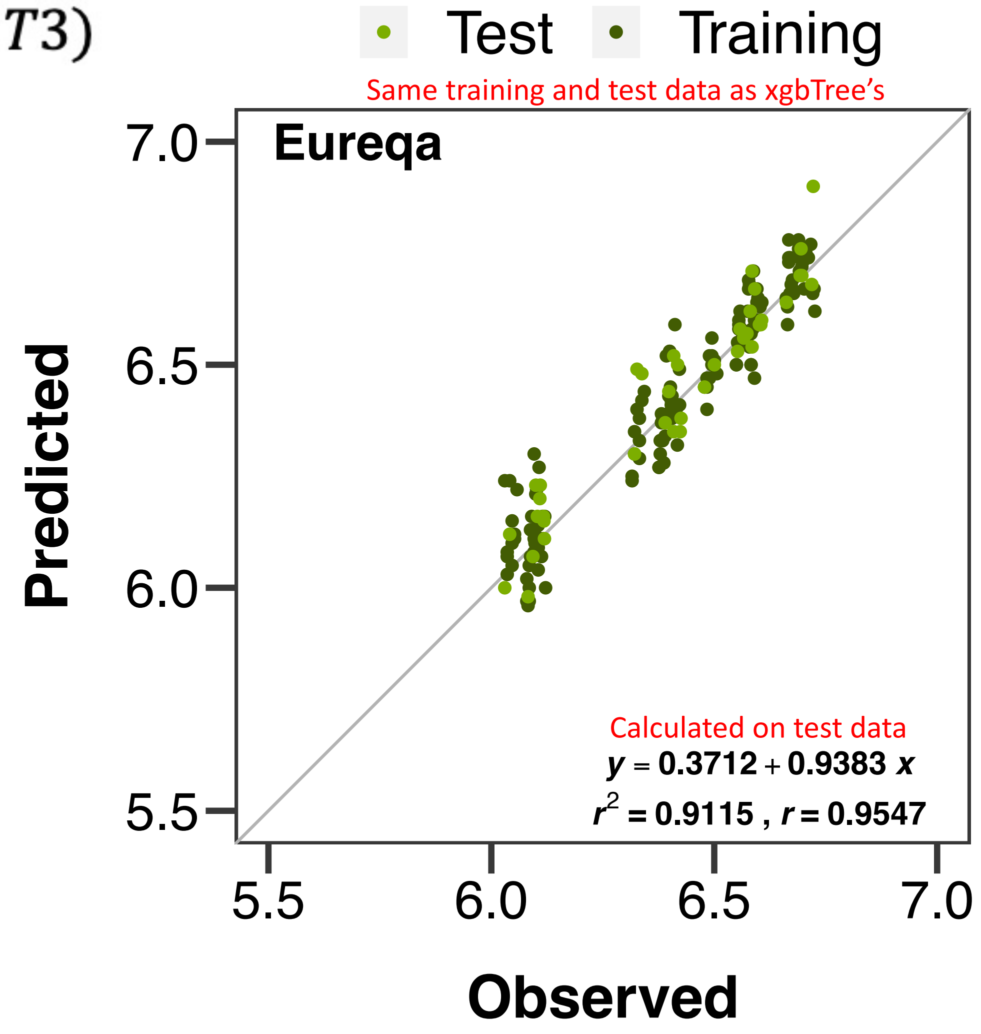
$$pH_t = 7.37 - \frac{3.69}{C} - \frac{0.00549 * length}{T2}$$

C-run length (C) appears in almost all models obtained.



25 models generated

Features are unscaled in this case, sample model complexity=13. More complex equations were also obtained using C-tract and all 3 loop lengths.



Same training and test data as xgbTree's

Calculated on test data

$y = 0.3712 + 0.9383\,x$

$r^2 = 0.9115$ , $r = 0.9547$

# Approach C: Developing Non-Linear Analytical Equation that Explains $pH_t$

- Allowed forms: Basic (constant, input variable, ±, x, ÷)
- Error metric is absolute error (default)
- Target expression:

$$pH_t = f(length, C, T1, T2, T3)$$

SAMPLE MODEL:

$$pH_t = 7.32 - \frac{0.124}{T2} - \frac{3.47}{C}$$

Complexity = 11

$$pH_t = 7.37 - \frac{3.69}{C} - \frac{0.00549*length}{T2}$$

Complexity = 13

$$pH_t = 7.24 + 0.0124T1T2 - 0.0277T1 - \frac{3.48}{C}$$

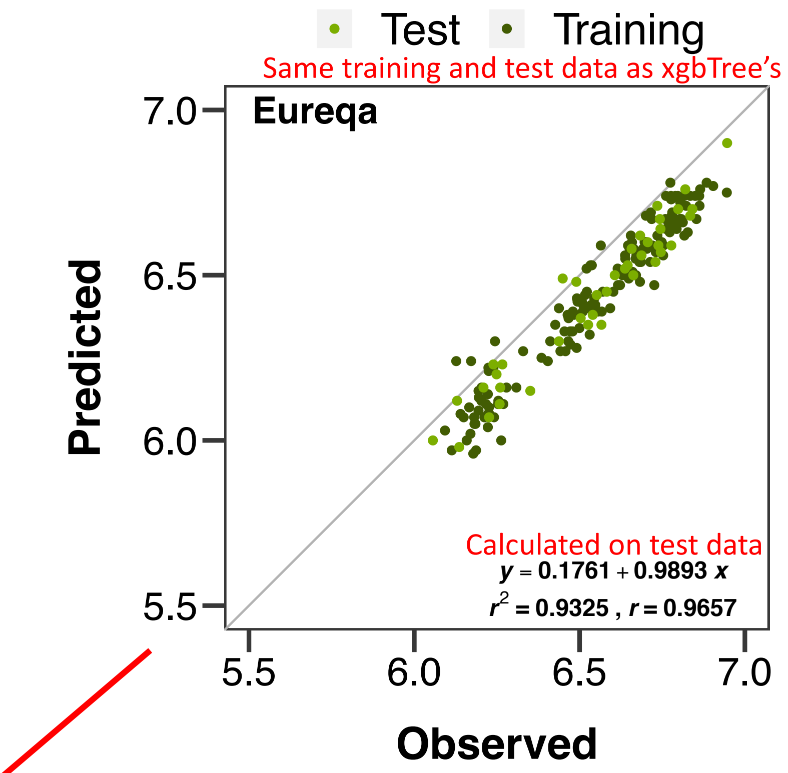
Complexity = 16

$$pH_t = 7.25 + 0.00469T1T2T3 - 0.0272T3 - \frac{3.2+0.124T1}{C}$$

Complexity = 22

Used C and all 3 loop lengths

All equations came from the same pool. For these 4, $r^2$ increases with increasing complexity based on the Eureqa calculations (be wary of overfitting).



Same training and test data as xgbTree's

Eureqa

Calculated on test data

$y = 0.1761 + 0.9893\,x$

$r^2 = 0.9325$, $r = 0.9657$

# Conclusions

- With a restricted sub-universe of the C/T-based i-motifs, all three approaches still resulted to a reasonable quality of $T_m$ and $pH_t$ prediction. For both parameters, the gradient boosting machines performed the best out of the three approaches ($T_m$: $r^2$ = 0.98, RMSE = 1.4 & $pH_t$: $r^2$ =0.95, RMSE = 0.054).

- Out of the 4 suggested methods, in which T is scored 0, to apply the G4Hunter algorithm in predicting i-motif stability, the optimised, extended version performed the best ($T_m$:$r^2$=0.918, $pH_t$:$r^2$=0.837). However, the previous optimised version, in which T is negated, is still slightly better ($T_m$:$r^2$=0.920, $pH_t$:$r^2$=0.858).

- Feature importance analysis from the GBM machine learning approach shows that the most important feature in defining the stability of the i-motifs (in the given sub-universe) both in terms of $T_m$ and $pH_t$ is the length of the C tracts (C). For $T_m$ prediction, the length of the 3rd loop (T3) is slightly more important than that of the other two. For $pH_t$ prediction, this is unclear because the importance ranking of the 3 loops differs whether total sequence length is included or not as a feature.
    - GBM models were built with and without total sequence length because this length can be derived using or is dependent on the other features. This is also why it is not surprising that not specifying length as a feature can result in a model performing comparably (even better in the case of $pH_t$ prediction) than a model with it included.

- The sample model from Eureqa shows that with only a slight compromise in prediction quality, we can have a simple, transparent analytical equation that expresses $T_m$ and $pH_t$ as a function of the chosen features. The $T_m$ equation captures the interplay between the C-tract length and the loop lengths 1-3 in modulating the $T_m$ of i-motifs in the given sub-universe. For $pH_t$, the chosen equation makes use of C, T2 and length (in turn dictated by the C-tract and loop lengths) to define its value. There is another, better-performing, equation using C and all 3 loop lengths but is more complex.
    - With nearly all Eureqa equations using C-tract lengths, Eureqa results agree with GBM's that this feature is very important in predicting the stability in terms of $T_m$ and $pH_t$ of this sub-universe of i-motifs.