Specificity and Methylation sensitivity analysis of CTCF(F1-F9)

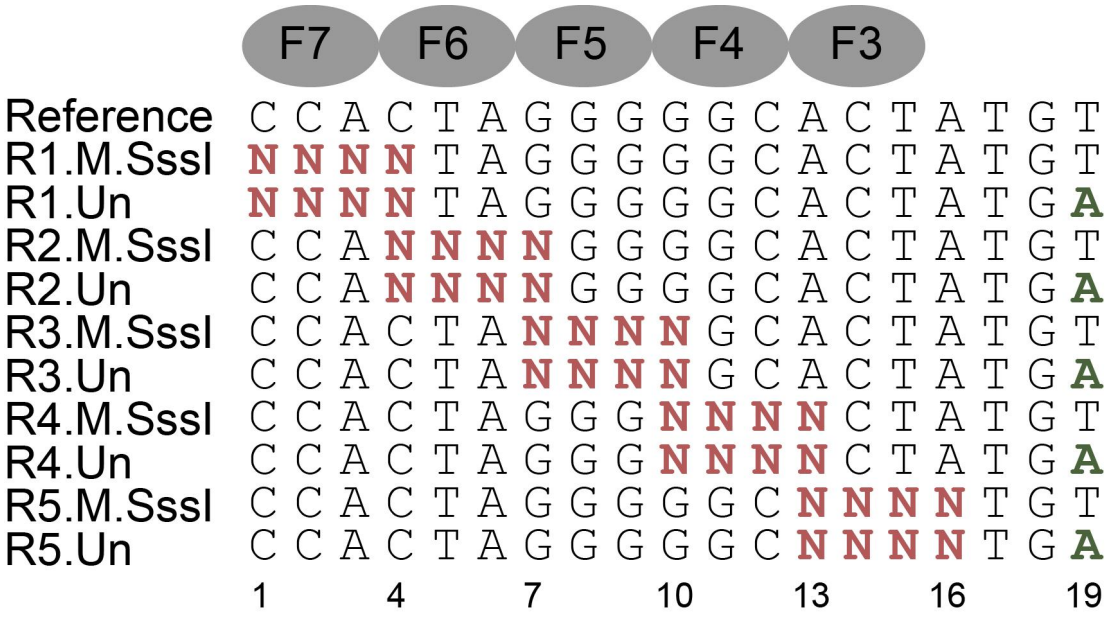
Zheng Zuo

Sep 12, 2020

Table of Contents

Background and Introduction

The CTCF protein, consisting of 11 tandem C2H2-type zinc fingers, is known to be critical insultor for genome architecture in mammals. Its finger 3 to 7 has been shown to recognize underlying reference sequence and motif strongly. Also It was known that CTCF processes some methylation sensitivity, i.e., when some CpG dinucleotide gets methylated within the binding site, the affinity of CTCF to the methylated sites gets compromised. To systematically investigate the specificity and methylation sensitivity of mouse CTCF protein, I designed and synthesized the following dsDNA libraries, covering every single and adjacent double variants of the core binding sites recognized by F3-F7. Position 19 serves as the barcode position to indicate wheather each sequence is treated by M.SssI methylated beforehand or not (For M.SssI treatment, only CpG dinucleotide can get methylated).



Importing and preprocessing data

```
load("../data/CTCF.rda")
(CTCF <- CTCF %>%
  dplyr::mutate(`Bound/Unbound` = Bound/Unbound,
                Energy = -log(Bound/Unbound)))
```

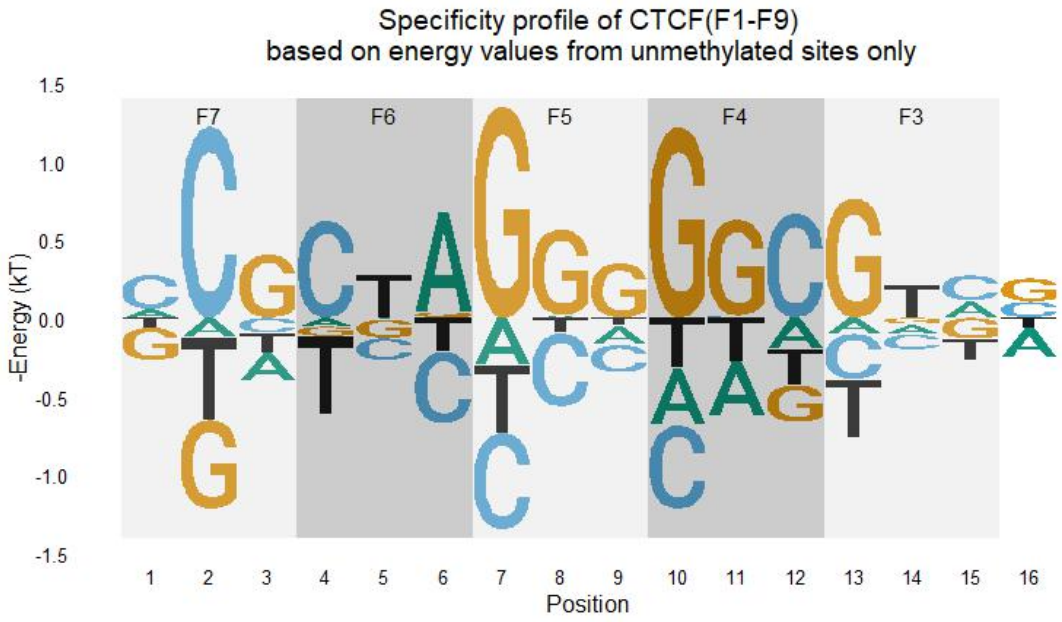
```
## # A tibble: 2,538 x 6
##    Sequence          Property Bound Unbound `Bound/Unbound` Energy
##    <chr>             <chr>    <dbl>   <dbl>           <dbl>  <dbl>
##  1 CCACTAGGGGGCGCTG  me        3714     243            15.3  -2.73
##  2 CCACTAGGGGGCGCTG  un        3184     232            13.7  -2.62
##  3 CCACTAGGGGGCGCTC  un        2232     182            12.3  -2.51
##  4 CCACTAGGGGGCGCAC  un        1494     127            11.8  -2.47
##  5 CCACTAGGGGGCGCTC  me        1711     148            11.6  -2.45
##  6 CCACTAGGGGGCGCTA  me       14185    1236            11.5  -2.44
##  7 CCACTAGGGGGCGCAG  me        2408     210            11.5  -2.44
##  8 CCACTAGGGGGCGCAG  un        2179     191            11.4  -2.43
##  9 CCACTAGGGGGCGCAA  un        1577     139            11.3  -2.43
## 10 CCACTAGGGGGCGCCG  me        2565     230            11.2  -2.41
## # ... with 2,528 more rows
```

Building regular specificity model from unmethylated sites

It is easy to build specificity model and derive motif based on energy values from those unmethylated sites only.

```
CTCF.Model.3Lp1<- CTCF %>%
  dplyr::filter(Property == "un") %>% # Select those untreated sites
  TFCookbook::buildEnergyModel(encoding = "3L+1")

CTCF.Model.3Lp1%>%
```

```
  TFCookbook::getEnergyMatrix() %>%
  TFCookbook::plotEnergyLogo() +
  addFingers(index = 7:3, yMin =  -1.4, yMax = 1.4) +
  labs(title = "Specificity profile of CTCF(F1-F9)\n based on energy va
lues from unmethylated sites only") +
  theme(plot.title = element_text(hjust = 0.5))
```




```
#ggsave("Regular motif.svg", plot = last_plot(), height = 4.5, width =
9)
```

Building Methylation Sensitivity model

Pairwise comparison of unmethylated sites and M.SssI treated sites

Since the designed libraries cover both M.SssI treated and untreated sequences in one-to-one correspondence, it is possible to do pairwise comparison for each site. Note that M.SssI treatment doesn't necessarily mean methylation of cytosine, only those cytosines within CpG dinucleotide can get methylated, therefore those CpG-non-containing sequences can serve as negative control to gauge the intrinsic variances of our Methyl-Spec-seq measurement.

```
CTCF.pairwise <- inner_join(subset(CTCF, Property=="un"),
                            subset(CTCF, Property=="me"), by = "Sequenc
e") %>%
  dplyr::select(Sequence,
                Energy.un = `Energy.x`,
                Energy.me = `Energy.y`) %>%
  dplyr::mutate(CpG.containing = as.integer(stringi::stri_count_fixed(S
equence, "CG")),
                Energy = Energy.me - Energy.un)
```

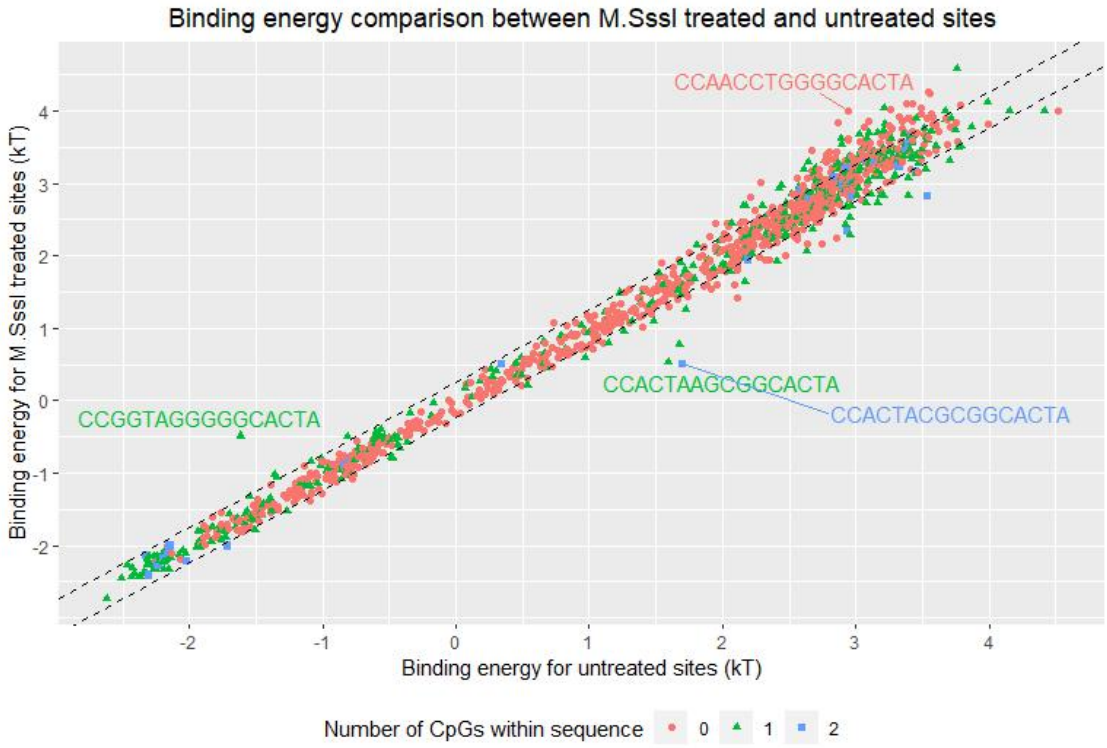
```
CTCF.pairwise %>%
  dplyr::arrange(desc(Energy)) %>%
  dplyr::filter(Energy.un <= 1.5) %>%
  dplyr::rename(`Number of CpG sites` = CpG.containing,
                `Methylation effect (kT)` = Energy)
```

```
## # A tibble: 523 x 5
##    Sequence       Energy.un Energy.me `Number of CpG sit~ `Methylati
on effect (~
##    <chr>              <dbl>     <dbl>               <int>
       <dbl>
##  1 CCGGTAGGGGGCA~    -1.62     -0.488                   1
       1.13
##  2 TCGCTAGGGGGCA~    -1.36     -1.01                    1
       0.343
##  3 CTCCTAGGGGGCA~     0.727     1.07                    0
       0.343
##  4 ACGGTAGGGGGCA~    -0.814    -0.489                   1
       0.325
##  5 CCGATAGGGGGCA~    -1.34     -1.07                    1
       0.273
##  6 CCACTAGGGCCCG~     1.23      1.49                    1
       0.258
##  7 CCACTAGGGGGCC~    -1.14     -0.891                   0
       0.247
##  8 CCACGAGGGGGCA~    -1.08     -0.842                   1
       0.242
##  9 TTACTAGGGGGCA~     1.30      1.53                    0
       0.234
## 10 CCACTAGGGGGCC~    -1.39     -1.16                    0
       0.227
## # ... with 513 more rows
```

Clearly, for those high-affinity, or low-binding energy sequences, CCGGTAGGGGGCACTA deviate significantly from the diagnoal lines by up to ~1kT, whereas almost all non-CpG containing sites fall within the 0.25kT energy deviation bounds, so in our measurement mCpG at position 2 significantly compromise the binding affinity of CTCF, which is consistent with previous literature report. For those weak binding sites, there are certain degree of divergence, very likely because of the alternative recognition mode of CTCF including the nearby barcode position, thus we consider that could be technical artifact.

```
CTCF.pairwise %>%
  dplyr::mutate(Label = if_else(abs(Energy.un - Energy.me)> 1, Sequence,
 ""),
                CpG.containing = as.factor(CpG.containing)) %>%
  ggplot(aes(x = Energy.un, y = Energy.me, shape = CpG.containing, colo
r = CpG.containing, label = Label)) +
  geom_point() +
  geom_abline(intercept = -0.25, slope = 1, linetype="dashed") +
```

```
  geom_abline(intercept = 0.25, slope = 1, linetype="dashed") +
  geom_text_repel(show.legend = FALSE, force = 10) +
  ggtitle("Binding energy comparison between M.SssI treated and untreat
ed sites") +
  scale_x_continuous(breaks = seq(-2, 4, 1)) + scale_y_continuous(break
s = seq(-2, 4, 1)) +
  xlab("Binding energy for untreated sites (kT)") +
  ylab("Binding energy for M.SssI treated sites (kT)") +
  labs(shape = "Number of CpGs within sequence", color = "Number of CpG
s within sequence") +
  theme(legend.position = "bottom", plot.title = element_text(hjust = 0.
5))
```



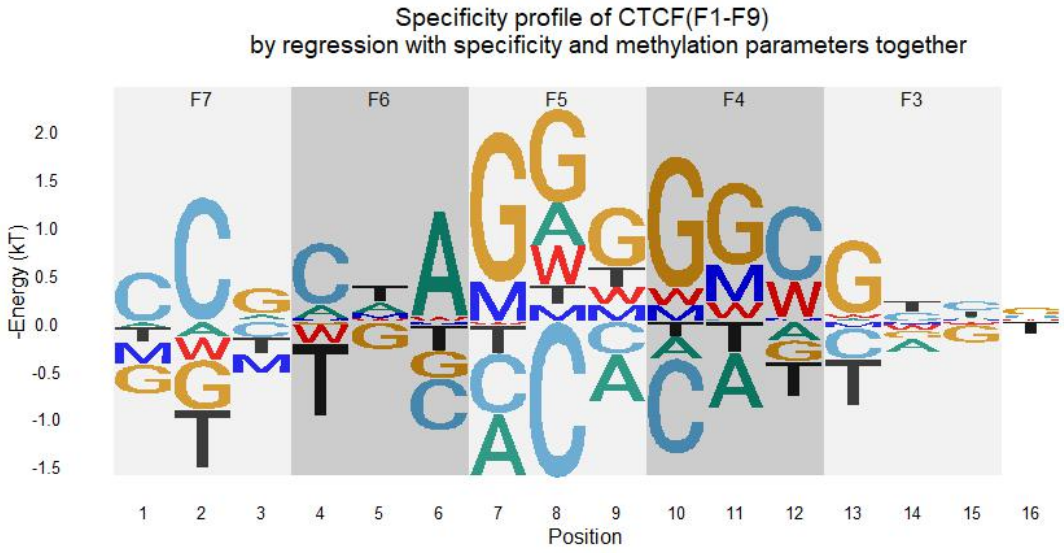

```
#ggsave("Pairwise comparison.svg", plot = last_plot(), height = 5.7, wi
dth = 8)
```

Building Methylation sensitivity model by regression of 4L+1 parameters all together

As discussed in the main text, if we use encoding scheme (3+1)L+1 and do regression analysis with all 4L+1 parameters altegether, we can get some methylation profiles shown below, which is very different from what we expect based on visual inspection, e.g., position 2 didn't show any negative methylation sensitivity at all and other positions show considerable ammount of methyl effect.

```
CTCF.Model.4Lp1 <- CTCF %>%
  dplyr::mutate(Sequence = if_else(Property=="un",
                                   Sequence,
```

```
                                        stringi::stri_replace_all_fixed(Sequ
ence, "CG", "MW"))) %>%
  dplyr::filter(Energy <= 1.7) %>%
  TFCookbook::buildEnergyModel(encoding = "4L+1")

CTCF.Model.4Lp1 %>%
  TFCookbook::getEnergyMatrix() %>%
  TFCookbook::addMethylMatrix(CTCF.Model.4Lp1, encoding = "(3+1)L+1") %
>%
  TFCookbook::plotEnergyLogo() +
  addFingers(index = 7:3, -1.6, 2.45) +
  scale_y_continuous(breaks = seq(-1.5, 2, 0.5)) +
  labs(title = "Specificity profile of CTCF(F1-F9)
       by regression with specificity and methylation parameters togeth
er") +
  theme(plot.title = element_text(hjust = 0.5))
```




```
#ggsave("4L+1 motif.svg", plot = last_plot(), height = 4.5, width = 9)
```

Building Methylation sensitivity model by separate evaluation of methylation effect on each binding site

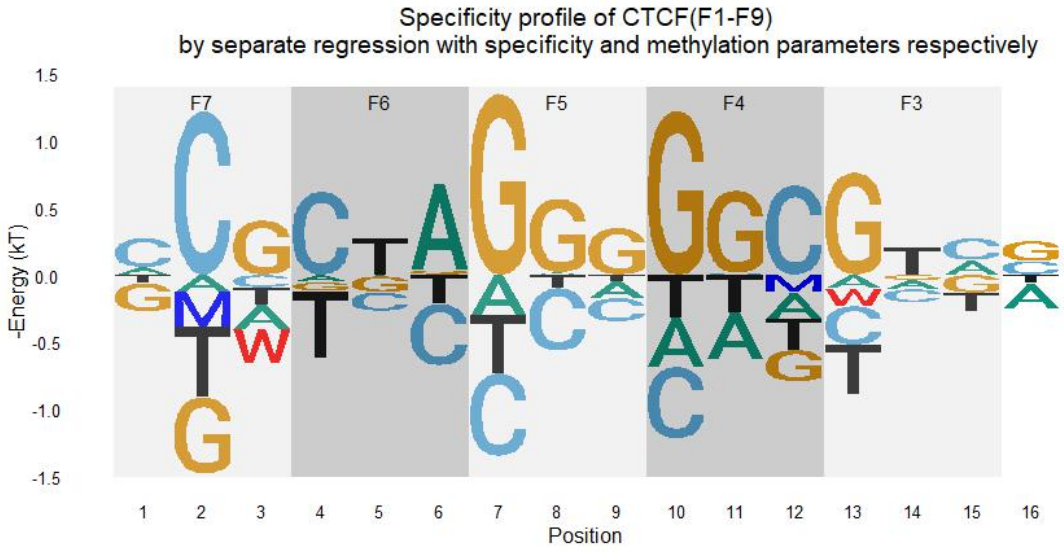
Alternatively, if we perform regression over methylation-related parameters (1MW to 16MW) alone using pairwise comparison data, it is easy to derive a methylation sensitivity model that matches our visual inspection well. Note that our measurement resolution is around 0.25kT, so we can choose 0.25kT as cut-off to analyze those significant methylation effect alone.

```
(CTCF.MethylModel <- CTCF.pairwise %>%
  filter(Energy.un <= 1.5, Energy > 0.25) %>% ## Filtering out low affi
nity binding sites and select only significant observation
  mutate(Sequence = stringi::stri_replace_all_fixed(Sequence, "CG", "MW
")) %>%
```

```
  arrange(desc(Energy)) %>%
  TFCookbook::buildMethylationModel(encoding = "(3+1)L+1", withIntercep
t = FALSE))

##
## Call:
## lm(formula = Energy ~ . - Energy + 0, data = input)
##
## Coefficients:
##  `1MW`   `2MW`   `3MW`   `4MW`   `5MW`   `6MW`   `7MW`   `8MW`   `9M
W`  `10MW`
##     NA  0.5174      NA      NA      NA      NA      NA      NA
NA      NA
## `11MW`  `12MW`  `13MW`  `14MW`  `15MW`  `16MW`
##     NA  0.2582      NA      NA      NA      NA
```

By combining specificity and methylation sensitivity info together, we can produce some motif logo that catches most of information to our expectation. Since we have no information about the strand-specific methylation effect in current experimetal setup, we just plot it as 50/50 by default. It has been shown in other literature that the methylation effect at position (2,3) primarily comes from the upper strand.

```
CTCF.Model.3Lp1%>%
  TFCookbook::getEnergyMatrix() %>%  ## Getting energy matrix for regul
ar specificity model
  TFCookbook::addMethylMatrix(CTCF.MethylModel, encoding = "(3+1)L+1")
%>% ## Adding methylation parameters to the matrix
  TFCookbook::plotEnergyLogo() +
  addFingers(index = 7:3, -1.5, 1.4) +
  scale_y_continuous(breaks = seq(-1.5, 1.5, 0.5)) +
  labs(title = "Specificity profile of CTCF(F1-F9)
       by separate regression with specificity and methylation paramete
rs respectively") +
  theme(plot.title = element_text(hjust = 0.5))
```



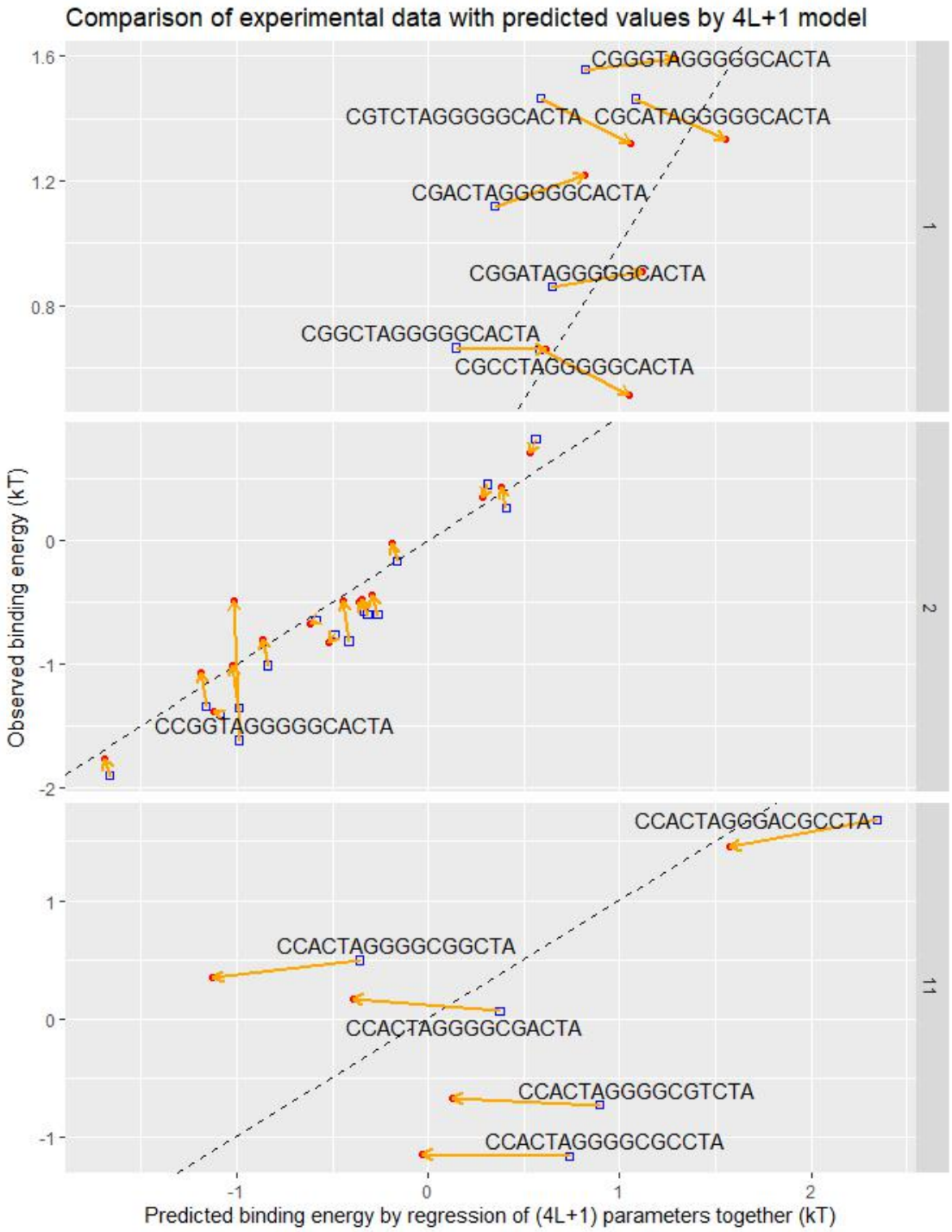

```
#ggsave("Separate regressions motif.svg", plot = last_plot(), height = 4.5, width = 9)
```

Why separate regression over specificity and methylation data performs better than all-in-one regression?

To illustrate the intrinsic limitation of all-in-one regresson of all parameters, we can compare all those predicted binding energy for unmethylated and methylated sites at each position based on 4L+1 model with the observed values as following figure. Clearly, for those sequences with CpG at position 1 and 11, each pair exhibited almost no methylation effect, but we still get non-zero methylation parameters (1MW and 11MW), most likely because non-zero methyl- parameters decreases the overall deviation between observed and predicted values and "push" most pairs closer to the diagnoal lines. On the other hand, for sequences with CpG at position 2, only one sequence (CCGGTAGGGGGCACTA) showed significant methylation effect, which gets "buried" in other insignificant ones, and thus we couldn't get some positive 2MW parameter with 4L+1 regression method.

```
CTCF.4Lp1 <- CTCF %>%
  dplyr::mutate(position.CpG = stringi::stri_locate_first_fixed(Sequence, "CG")[,"start"]) %>%
  dplyr::filter(Energy <= 1.7) %>%
  dplyr::mutate(Predicted.Energy = CTCF.Model.4Lp1$fitted.values) ##Adding predicted values by 4L+1 regression method

CTCF.4Lp1.paired <-
    dplyr::inner_join(subset(CTCF.4Lp1, Property == "un"),
                      subset(CTCF.4Lp1, Property == "me"),
                      by = "Sequence") %>%
    dplyr::select(Sequence,
                  position.CpG = position.CpG.x,
                  Energy.un = Energy.x,
```

```
                  Predicted.Energy.un = Predicted.Energy.x,
                  Energy.me = Energy.y,
                  Predicted.Energy.me = Predicted.Energy.y)

CTCF.4Lp1.paired %>%
    dplyr::filter(position.CpG %in% c(1, 2, 11)) %>%
    ggplot() +
    geom_abline(intercept = 0, slope = 1, linetype = "dashed") +
    geom_point(aes(x = Predicted.Energy.un, y = Energy.un), color = "bl
ue", shape = 0) +
    geom_point(aes(x = Predicted.Energy.me, y = Energy.me), color = "re
d", shape = 19) +
    geom_segment(aes(x = Predicted.Energy.un, y = Energy.un,
                     xend = Predicted.Energy.me, yend = Energy.me),
                 lineend = 'round', linejoin = 'bevel', size = 1, color
 = "orange",
                 arrow = arrow(length = unit(0.03, "npc"))) +
    geom_text_repel(data = function(x) filter(x, ((position.CpG != 2) |
 (Sequence == "CCGGTAGGGGGCACTA"))),
                    aes(x = Predicted.Energy.un, y = Energy.un, label =
 Sequence), force = 1) +
    ggtitle("Comparison of experimental data with predicted values by 4
L+1 model") +
    xlab("Predicted binding energy by regression of (4L+1) parameters t
ogether (kT)") +
    ylab("Observed binding energy (kT)") +
    facet_wrap(~position.CpG,
               scales = "free_y",
               strip.position = "right",
               nrow = 3)
```

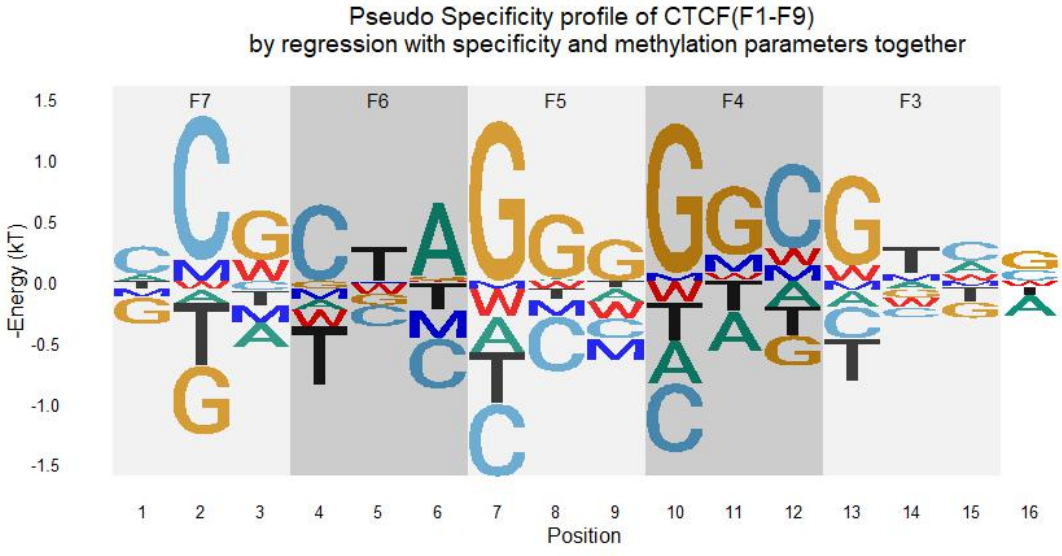



```
#ggsave("Model 4Lp1 comparison.svg", plot = last_plot(), height = 9, wi
dth = 7)
```

4L+1 modeling with artificial data exhibits the intrinsic limitation of all-in-one regression strategy

Besides comparison of experimental data with predicted values by 4L+1 model, we can create an artificial "pseudo" dataset in which each methylated site has exactly the same value as the unmethylated one, and construct 4L+1 model to illustrate the intrinsic limitation of this all-in-one regression strategy.

```
CTCF.pseudoMe <- subset(CTCF, Property == "un") %>%
  dplyr::mutate(Sequence = stringi::stri_replace_all_fixed(Sequence, "C
G", "MW"),
                Property = "me") %>%
  dplyr::filter(stringr::str_detect(Sequence, "MW"))

CTCF.pseudoModel.4Lp1 <-  rbind(subset(CTCF, Property == "un"), CTCF.ps
eudoMe) %>%
  TFCookbook::buildEnergyModel(encoding = "4L+1")

CTCF.pseudoModel.4Lp1 %>%
  TFCookbook::getEnergyMatrix() %>%
  TFCookbook::addMethylMatrix(MethylModel = CTCF.pseudoModel.4Lp1, enco
ding = "(3+1)L+1") %>%
  TFCookbook::plotEnergyLogo() +
  addFingers(index = 7:3, -1.6, 1.6) +
  scale_y_continuous(breaks = seq(-1.5, 1.5, 0.5)) +
  labs(title = "Pseudo Specificity profile of CTCF(F1-F9)
       by regression with specificity and methylation parameters togeth
er") +
  theme(plot.title = element_text(hjust = 0.5))
```

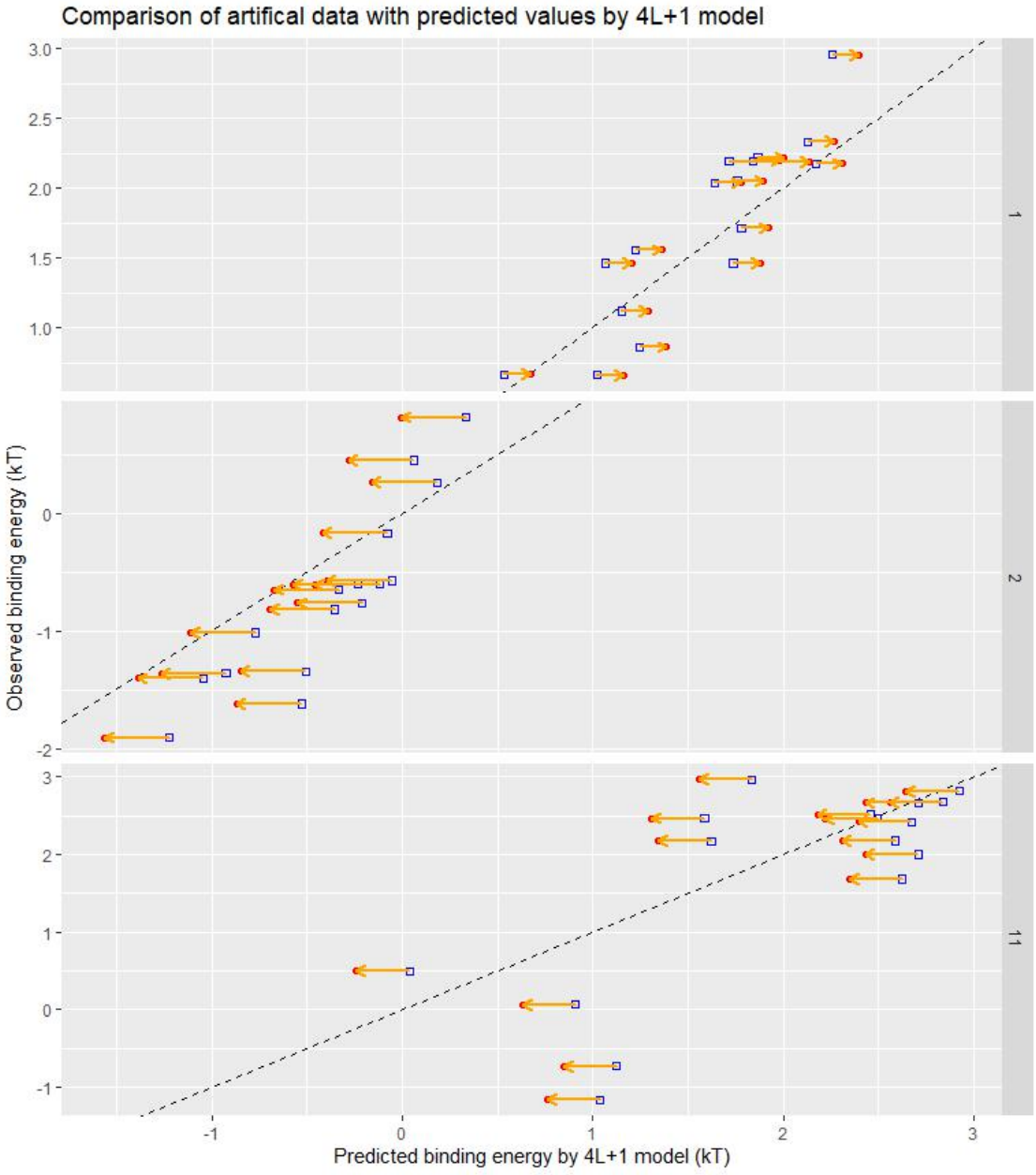


Clearly, under this "absolute no methylation effect" scenario, we still derive some non-zero methylation parameters from 4L+1 model, which shows us it is more

realistic to build separate methylation effect model based on pairwise comparison between each individual site.

```
CTCF.pseudo.paired <- subset(CTCF, Property == "un") %>%
  dplyr::mutate(Sequence.MW = stringi::stri_replace_all_fixed(Sequence,
 "CG", "MW"),
                position.CpG = stringi::stri_locate_first_fixed(Sequenc
e, "CG")[,"start"],
                Predicted.Energy.un = TFCookbook::predictEnergyMW(Seque
nce, CTCF.pseudoModel.4Lp1),
                Predicted.Energy.me = TFCookbook::predictEnergyMW(Seque
nce.MW, CTCF.pseudoModel.4Lp1)) %>%
  dplyr::select(Sequence, Sequence.MW, position.CpG, Energy, Predicted.
Energy.un, Predicted.Energy.me)

## Warning: Expected 64 pieces. Additional pieces discarded in 1266 row
s [1, 2, 3,
## 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, ...].

## Warning in predict.lm(model, newdata = ., type = "response"): predic
tion from a
## rank-deficient fit may be misleading

## Warning: Expected 64 pieces. Additional pieces discarded in 1266 row
s [1, 2, 3,
## 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, ...].

## Warning in predict.lm(model, newdata = ., type = "response"): predic
tion from a
## rank-deficient fit may be misleading

CTCF.pseudo.paired %>%
    dplyr::filter(position.CpG %in% c(1, 2, 11)) %>%
    ggplot() +
    geom_abline(intercept = 0, slope = 1, linetype = "dashed") +
    geom_point(aes(x = Predicted.Energy.un, y = Energy), color = "blue",
 shape = 0) +
    geom_point(aes(x = Predicted.Energy.me, y = Energy), color = "red",
 shape = 19) +
    geom_segment(aes(x = Predicted.Energy.un, y = Energy,
                     xend = Predicted.Energy.me, yend = Energy),
                 lineend = 'round', linejoin = 'bevel', size = 1, color
 = "orange",
                 arrow = arrow(length = unit(0.03, "npc"))) +
   ## geom_text_repel(aes(x = Predicted.Energy.un, y = Energy, label =
Sequence)) +
    ggtitle("Comparison of artifical data with predicted values by 4L+1
 model") +
    xlab("Predicted binding energy by 4L+1 model (kT)") +
    ylab("Observed binding energy (kT)") +
    facet_wrap(~position.CpG,
```

```
                scales = "free_y",
                strip.position = "right",
                nrow = 3)
```




```
#ggsave("Model 4Lp1 comparison.eps", plot = last_plot(), height = 9, wi
dth = 8)
```

Conclusions

Based on above analysis, it is clear that:

1) It is unrealistic to use “all-in-one” regression strategy to construct specificty and methylation effect model for CTCF and potential many other TFs;

2) We can build a composite specificity and methylation effect model for CTCF by doing regression over methylation-irrelevant and methylation parameters separately. Since the measurement resolution of typical Spec-seq experiment is around 0.2 kT, we can perform pairwise comparison on each pair of sequences and drop all those insignificant results below certain threshold (~0.25kT) and highlight those significant positions only.