Supporting Information

Deciphering DNA Nucleotide Sequence and their Rotation Dynamics with Interpretable Machine Learning Integrated C_3N Nanopore

Milan Kumar Jena,[†] Sneha Mittal,[†] Surya Sekhar Manna,[†] Biswarup Pathak[†,*]

[†]Department of Chemistry, Indian Institute of Technology (IIT) Indore, Indore, Madhya Pradesh, 453552, India

*E-mail: biswarup@iiti.ac.in

Contents **Page No.**

1. Computational Details

Text S1: (Machine Learning Details)

Linear Regression (LR): Linear regression is used to find the best line of fit, which describes the linear relation between input (x) and target output (y).

$$y = wx + b$$

Where y is the target output, w is the slope of the line, and b is the intercept of the line.

Kernel Ridge Regression (KRR): Kernel ridge regression is a ridge regression (linear least squares with l2-norm regularization) with the kernel trick. It helps in exploring the nonlinear relations of a regression problem. It has four common kernels: linear, polynomial, RBF, and laplacian.[1]

Gaussian Process Regression (GPR): Gaussian process regression combines the concepts of marginalization and the Bayesian approach to regression. In the normal regression model, y=*f*(x) is evaluated. In GPR, the Gaussian process is placed over the *f*(x). For this, the prior GP needs to be specified.

eXtreme Gradient Boosting Regression (XGBR): XGBR was initially developed and described by Chen et al. in their 2016 paper titled "XGBoost: A Scalable Tree Boosting System".[2] XGBR is one of the ensemble learning algorithms that aggregate multiple tree learners to achieve better prediction results. Here, the term "gradient boosting" originates from the idea of "boosting" or improving a single weak model by combining it with several other weak models (decision trees) to generate a collectively strong model.[3]

Random Forest Regression (RFR): In 2001, Breiman et al.[4] introduced the random forest machine learning algorithm. This is an ensemble technique that uses multiple decision trees and a technique called Bootstrap and Aggregation, commonly known as bagging. The target output

of the ensemble is calculated as the average of the predictions of each tree rather than relying on a single decision tree. The hyperparameters of Random Forest Regression include the number of decision trees in the forest, the maximum depth of each decision tree, and the number of features to consider for each split. Random Forest Regression can handle both categorical and numerical data and is less prone to overfitting than other decision tree-based algorithms due to the randomization of feature subsets and split points. However, it may suffer from high bias due to the randomization process, leading to suboptimal performance in some cases.

Adaptive Boosting Regression (AdaBoost): AdaBoost Regression involves iteratively training a series of decision trees, where each decision tree is trained on a modified version of the training data. The modification involves assigning higher weights to the data points that were not well-predicted by the previous decision tree. The algorithm assigns weight to each data point, which represents its importance in the model. The weights are updated after each iteration based on the performance of the previous decision tree. During prediction, the algorithm passes the input data through each decision tree in the series and obtains the output of each tree. The output of each decision tree is then combined using a weighted sum to obtain the final prediction. The hyperparameters of AdaBoost Regression include the number of decision trees in the series, the learning rate, which controls the contribution of each decision tree, and the maximum depth of each decision tree. AdaBoost Regression can handle both categorical and numerical data and is less prone to overfitting than other decision tree-based algorithms. However, it may be sensitive to noisy data and outliers, which can negatively affect the performance of the model.[5]

Extra Tree Regression (ETR): Extra Trees Regression involves randomly selecting a subset of features from the dataset and choosing random split points for each subset. This process is repeated for a large number of decision trees to create a forest of decision trees. During

prediction, the algorithm passes the input data through each decision tree in the forest and obtains the output of each tree. The output of all decision trees is then combined to obtain the final prediction. The algorithm can handle both categorical and numerical data and is less prone to overfitting than other decision tree-based algorithms due to the randomization of feature subsets and split points.

Statistical Evaluation Metrics

Coefficient of Determination (R^2):

It measures how well the regression model predicts the output and its value range between 0 and 1.

$$R^2 = \frac{\sum_{i=1}^{n}\left(M_i - \bar{m}\right)^2}{\sum_{i=1}^{n}\left(m_i - \bar{m}\right)^2} \quad \text{where} \quad \bar{m} = \frac{\sum_{i=1}^{n} m_i}{n}$$

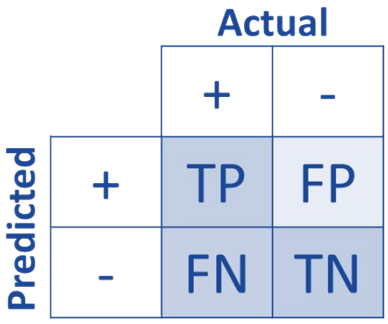
Root Mean Square Error (RMSE):

It determines the standard deviation of the predicted output value from the actual value.

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}\left(M_i - m_i\right)^2}$$

Where M_i represents the predicted output value, m_i represents the actual DFT calculated value, $\bar{m}$ is the mean of the DFT calculated values, and n is the total data points in the data set.

In ML classification, the confusion matrix is considered to be the best evaluation matrix and basic of all other matrices. It is a table with combinations of predicted and actual values.



True positive (TP) = The number of correct positive predictions made by a model.

True negative (TN) = The number of correct negative predictions made by a model.

False positive (FP) = The number of incorrect positive predictions made by a model.

False negative (FN) = The number of incorrect negative predictions made by a model.

Accuracy = Accuracy measures the number of correct predictions done by the model among the total number of predictions.

$$\text{Accuracy} = \frac{TP + TN}{TP + FP + TN + FN}$$

Precision = Precision explains how many of the correctly predicted instances actually turned out to be positive. Precision is helpful in cases where False Positive is a greater concern than False Negatives. It is also known as the true positive rate.

$$\text{Precision} = \frac{TP}{TP + FP}$$

Recall = Recall explains how many of the actual positive instances we were able to predict correctly with our model. It is a useful metric in cases where a False Negative is of greater concern than a False Positive. It is also known as the sensitivity of the model.

$$\text{Recall} = \frac{TP}{TP + FN}$$

F1 Score = It is the harmonic mean of Precision and Recall metrics.

$$\text{F1 Score} = 2 \times \frac{Precision \times Recall}{Precision + Recall}$$

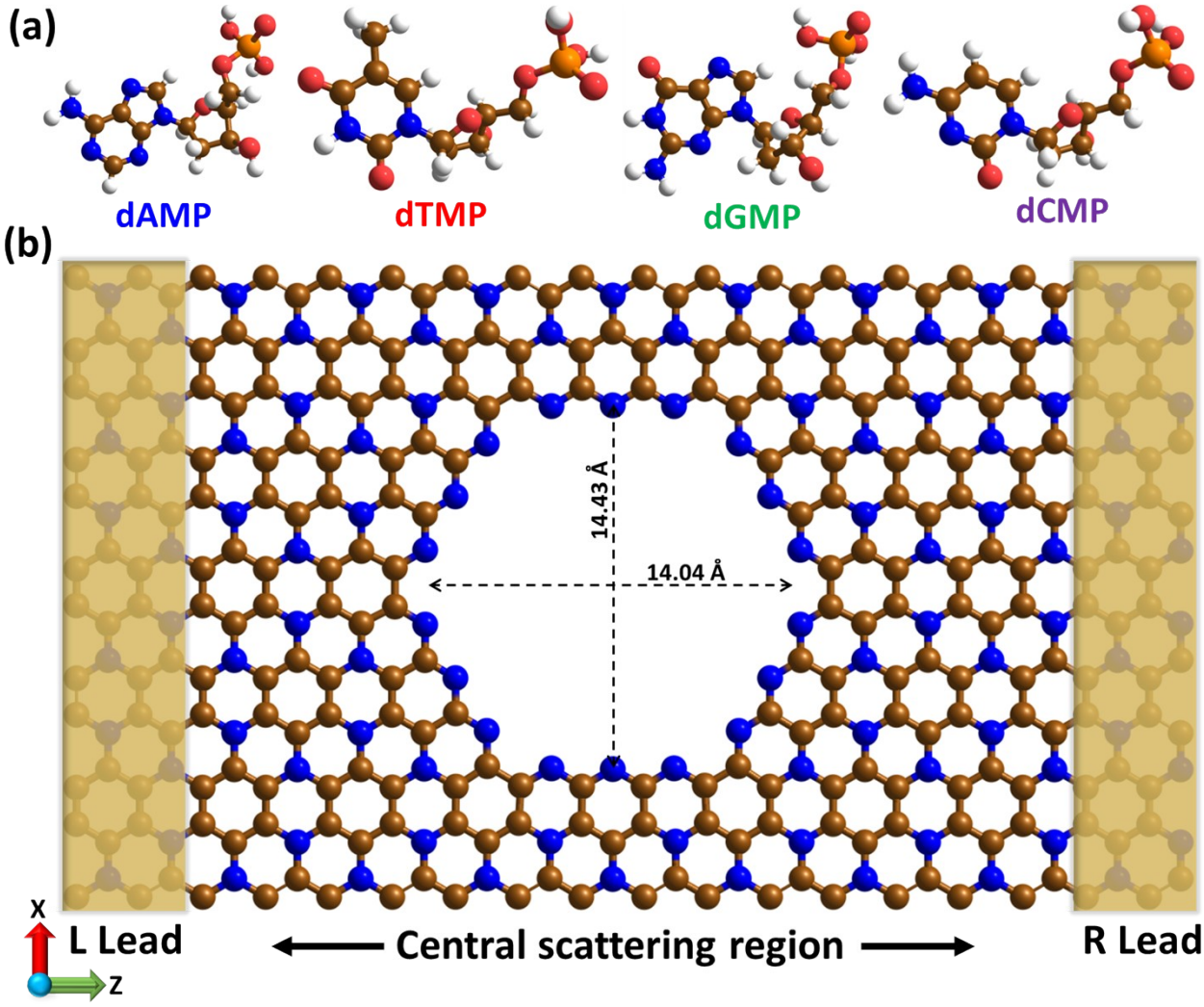


Figure S1. Optimized atomic structures of the four nucleotides (dAMP, dTMP, dGMP, and dCMP), and (b) schematic illustration of C_3N nanopore device considered for quantum transport

studies; showing left (L), right (R) electrode/lead and central scattering region. The quantum transport calculation is performed along the z-direction.

Text S2: (DFT-NEGF Method Details)

In this study, four naturally available DNA nucleotides, deoxyadenosine monophosphate (dAMP), deoxythymidine monophosphate (dTMP), deoxyguanosine monophosphate (dGMP), and deoxycytidine monophosphate (dCMP) have been selected and relaxed using the B3LYP functional and 6-311++G** basis sets with the Gaussian 09 package.[6,7] The bare C_3N nanopore structure optimization is performed separately using the SIESTA (Spanish initiative for electronic simulations with thousands atoms) code.[8,9] After that, the C_3N nanopore+nucleotide setups are fully optimized. All the structural and electronic properties calculation optimizations are performed by using the first-principles-based DFT method, as included in the SIESTA. The modeled nanopore consists of an atomically thick and nanoscale sized C_3N sheet. The vdW-DF-cx functional, norm-conserved Troullier–Martins pseudopotentials, double-zeta polarized (DZP) basis sets, and conjugate gradient (CG) algorithm are employed for all the calculations.[10–12] The quantum transport calculations are done with the non-equilibrium Green's function (NEGF) method with DFT by employing the TranSIESTA code.[13,14] The zero-bias transmission function $T(E)$ is the probability of transport of electron from the left (source) to the right (drain) electrode (sum of all the possible transmission channels) in the z-direction is calculated by the following equation.

$$T(E) = Tr[\Gamma_L(E)G_C(E)\Gamma_R(E)G_C^{\dagger}(E)]$$

where $\Gamma_{L/R}(E)$ symbolizes the left (L) and right (R) lead coupling matrix, and $G_C(E)/G_C^{\dagger}(E)$ are the retarded/advanced Green's functions. According to Landauer's model,[15,16] the molecular conductance, in the limit of zero temperature and zero bias voltage, is expressed as

$$G = \left(\frac{2e^2}{h}\right)T(E_F)$$

Where $\frac{2e^2}{h}$ represents the quantum conductance, $T(E_F)$ is the transmission function at the Fermi level(E_F).[16,17]

2. Relative Energy Table

Table S1. The calculated relative energies (in eV) of the in-plane rotated configuration of all the four nucleotides with respect to the energetically most stable geometry.

Nucleotide	0°	30°	60°	90°	120°	150°	180°
dAMP	0.78	0	0.95	0.14	0.81	0.21	0.79
dTMP	0.92	0.87	0.34	0.43	0.21	0	0.15
dGMP	0.88	0.14	0.02	0.14	0	0.22	0.08
dCMP	0.40	0.44	0.31	0.05	0.26	0.45	0

3. Effect of Rotation on Transmission Function

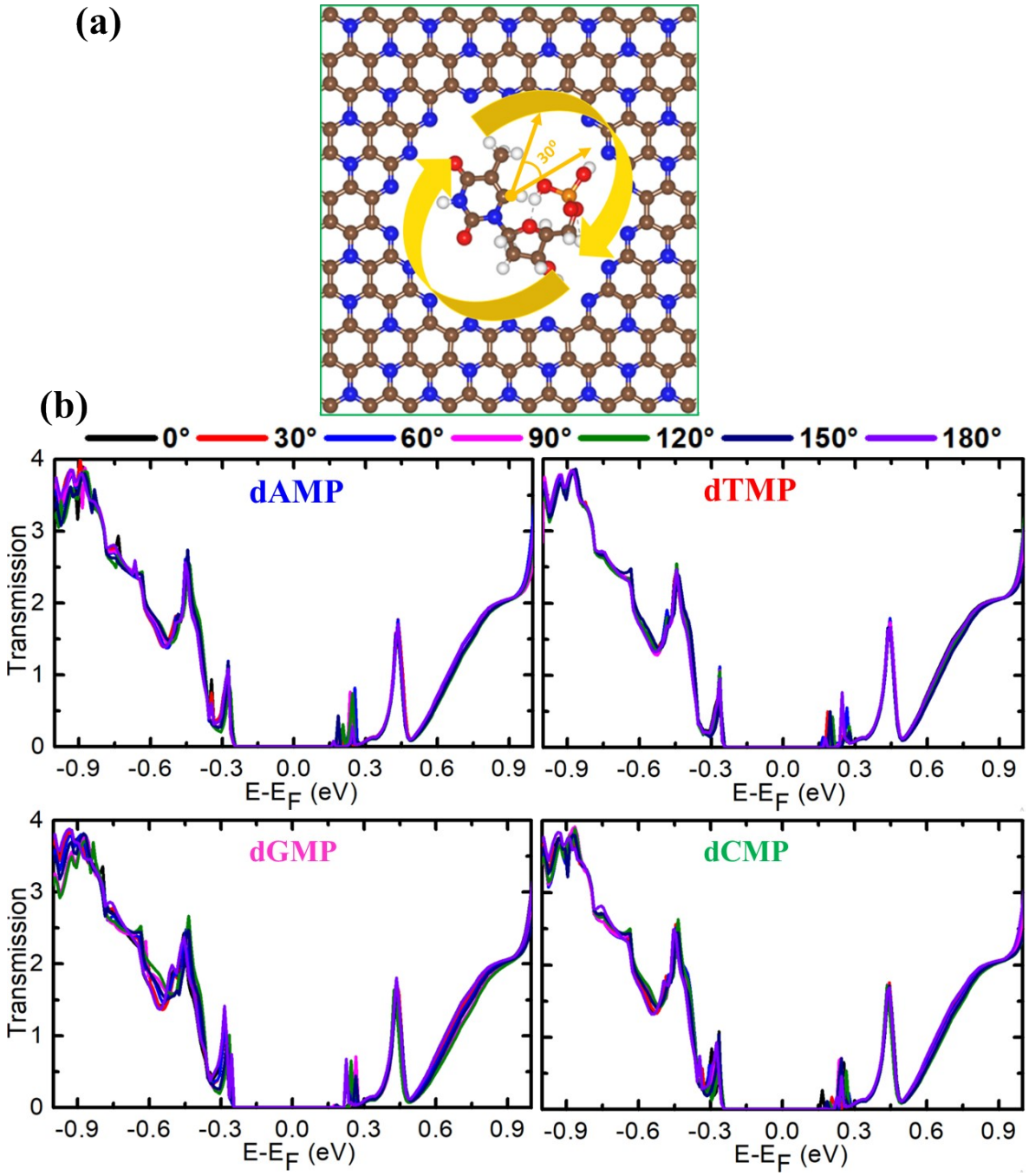


Figure S2. (a) Schematic illustration of rotational fluctuation of nucleotides while translocating through the C_3N nanopore. The change in nucleotide position affects the transmission function due to the change in the coupling strength between the nucleotide and nanopore and (b) changes in transmission spectra due to in-plane rotation (from 0° to 180° in steps of 30° along the x-axis in the yz plane) are shown for all four nucleotides.

4. Effect of Longitudinal Translation on Transmission Function

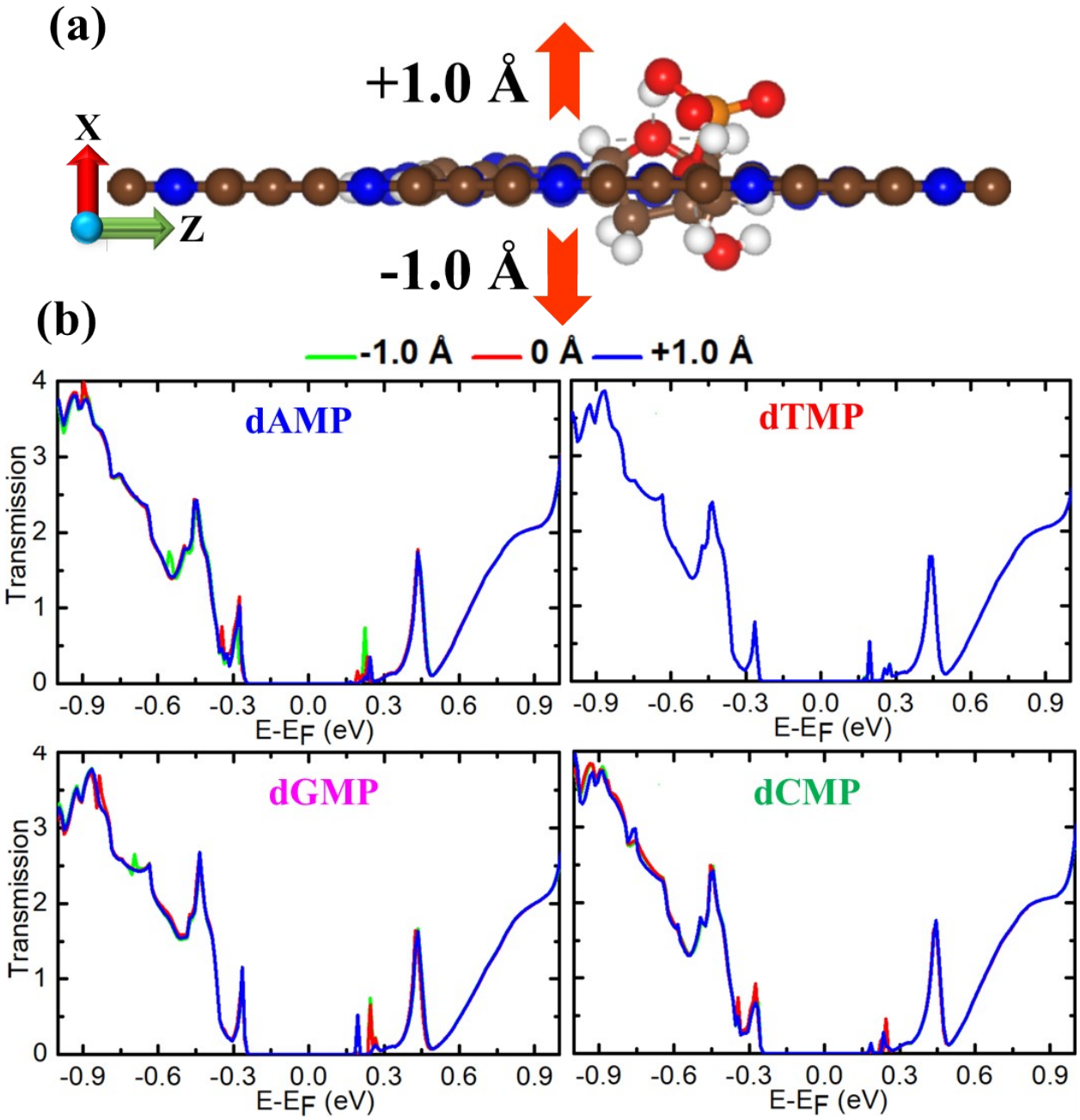


Figure S3. (a) Scheme of out-of-plane longitudinal translations in positive and negative directions by ± 1.0 Å for C_3N nanopore along the x-axis in the yz-plane and (b) the change in the transmission function for each targeted nucleotide due to out-of-plane translation.

5. Effect of Lateral Translation on Transmission Function

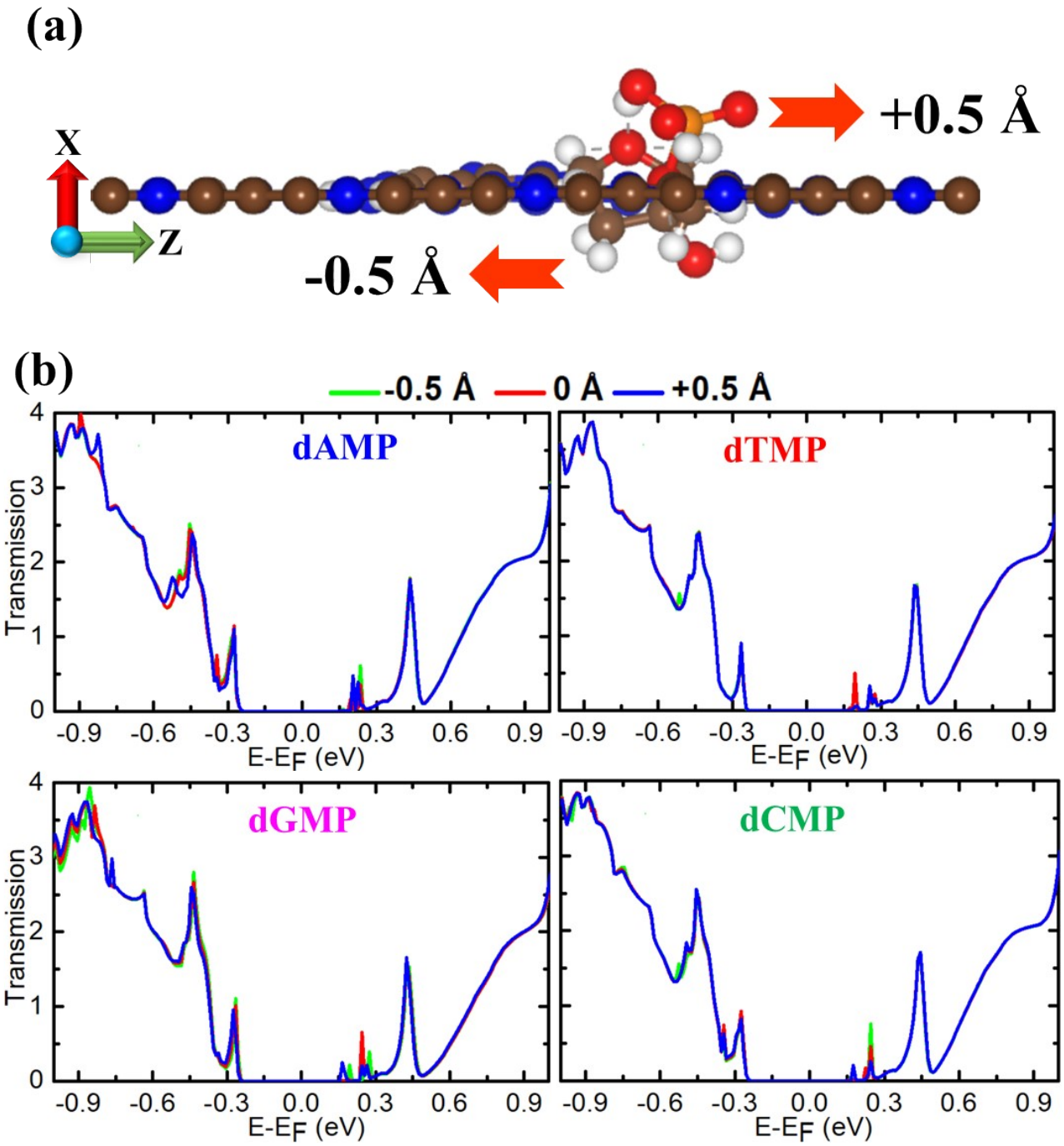


Figure S4. (a) Scheme of in-plane lateral translations in positive and negative directions by ± 0.5 Å for C_3N nanopore along the z-axis in the yz-plane and (b) the change in the transmission function for each targeted nucleotide due to in-plane translations.

6. Features Engineering

Table S2. The detailed description of the features along with their name.

Feature Name	Description	Symbol
F1	Mean valence electrons	$\bar{N}_v$
F2	Mean molecular weight	$\bar{M}_w$
F3	Mean electronegativity	$\bar{\chi}$
F4	Minimum distance between nucleotide H-atom and pore edge H-atom	d_{H-H}
F5	Minimum distance between nucleotide H-atom and pore edge C-atom	d_{H-C}
F6	Minimum distance between nucleotide H-atom and pore edge N-atom	d_{H-N}
F7	Minimum distance between nucleotide H-atom and pore edge O-atom	d_{H-O}
F8	Mean electron affinity	$\bar{E}_A$
F9	Mean van der Waals radii	$\bar{r}_{vdw}$
F10	Mean dipole polarizability	$\bar{Z}_e$
F11	Mean ionic radii	$\bar{r}_{ionic}$
F12	Mean covalent radii	$\bar{r}_{cov}$
F13	Mean ionization energy	$\bar{IE}$
F14	Mean effective nuclear charge	$\bar{Z}_{eff}$
F15	Energy range of transmission	E

7. Pearson's Feature Correlation Matrix Plots

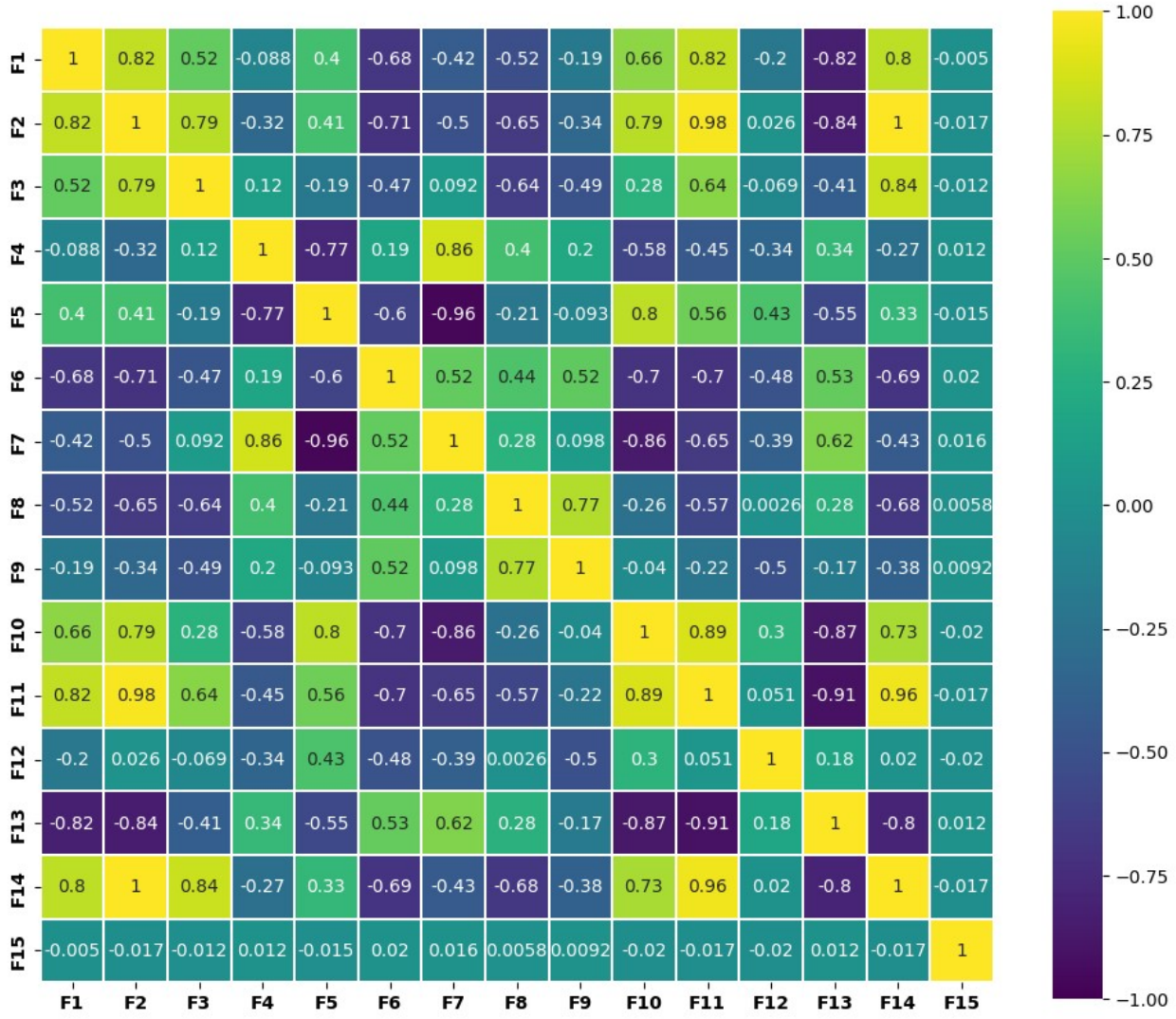


Figure S5. Feature-feature correlation matrix of Pearson's correlation coefficients (PCCs) for dAMP data set. The +1, -1, and 0 on the scale indicate the maximum positive, maximum negative, and no correlation, respectively.

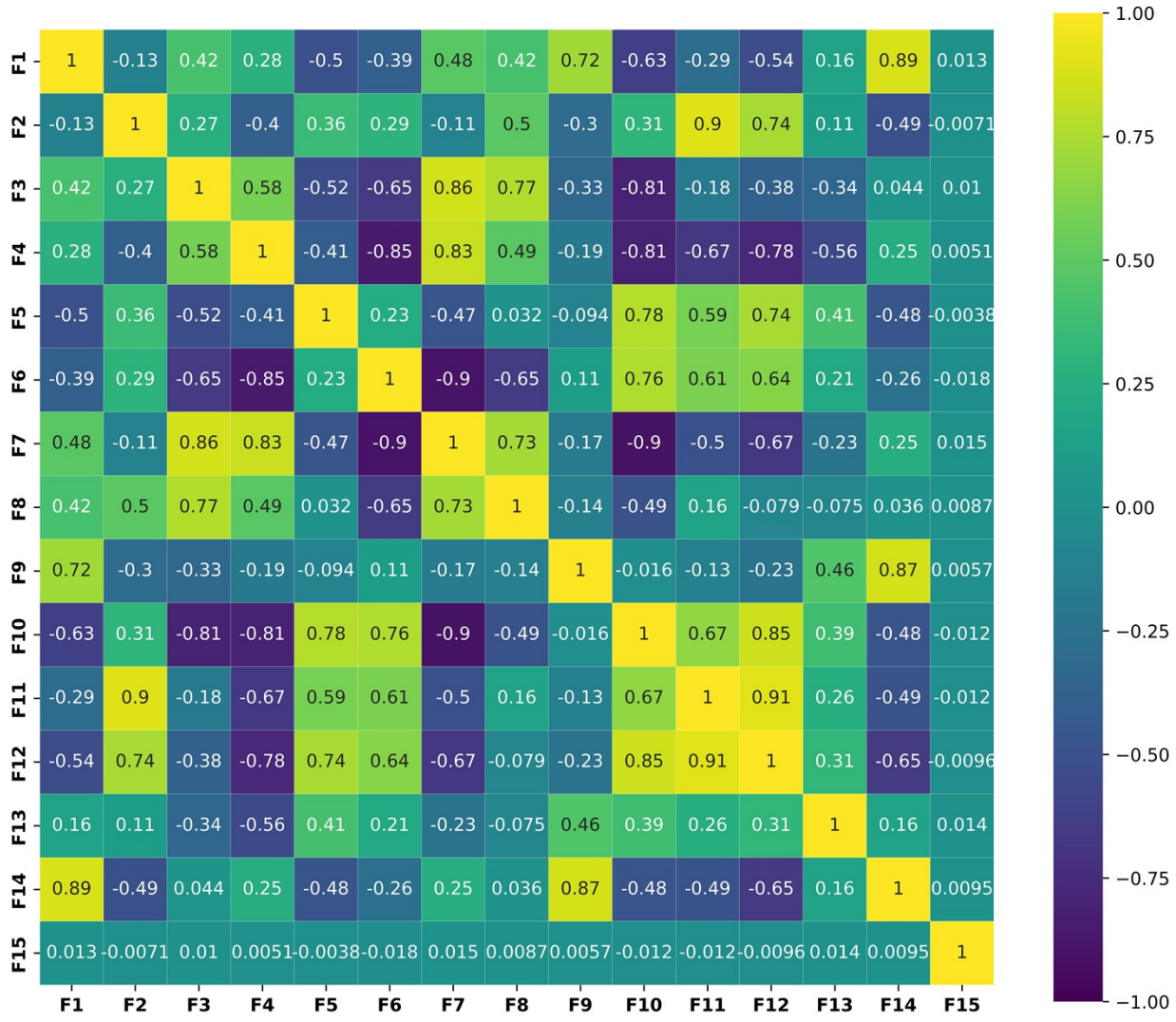


Figure S6. Feature-feature correlation matrix of Pearson's correlation coefficients (PCCs) for dTMP data set. The +1, -1, and 0 on the scale indicate the maximum positive, maximum negative, and no correlation, respectively.

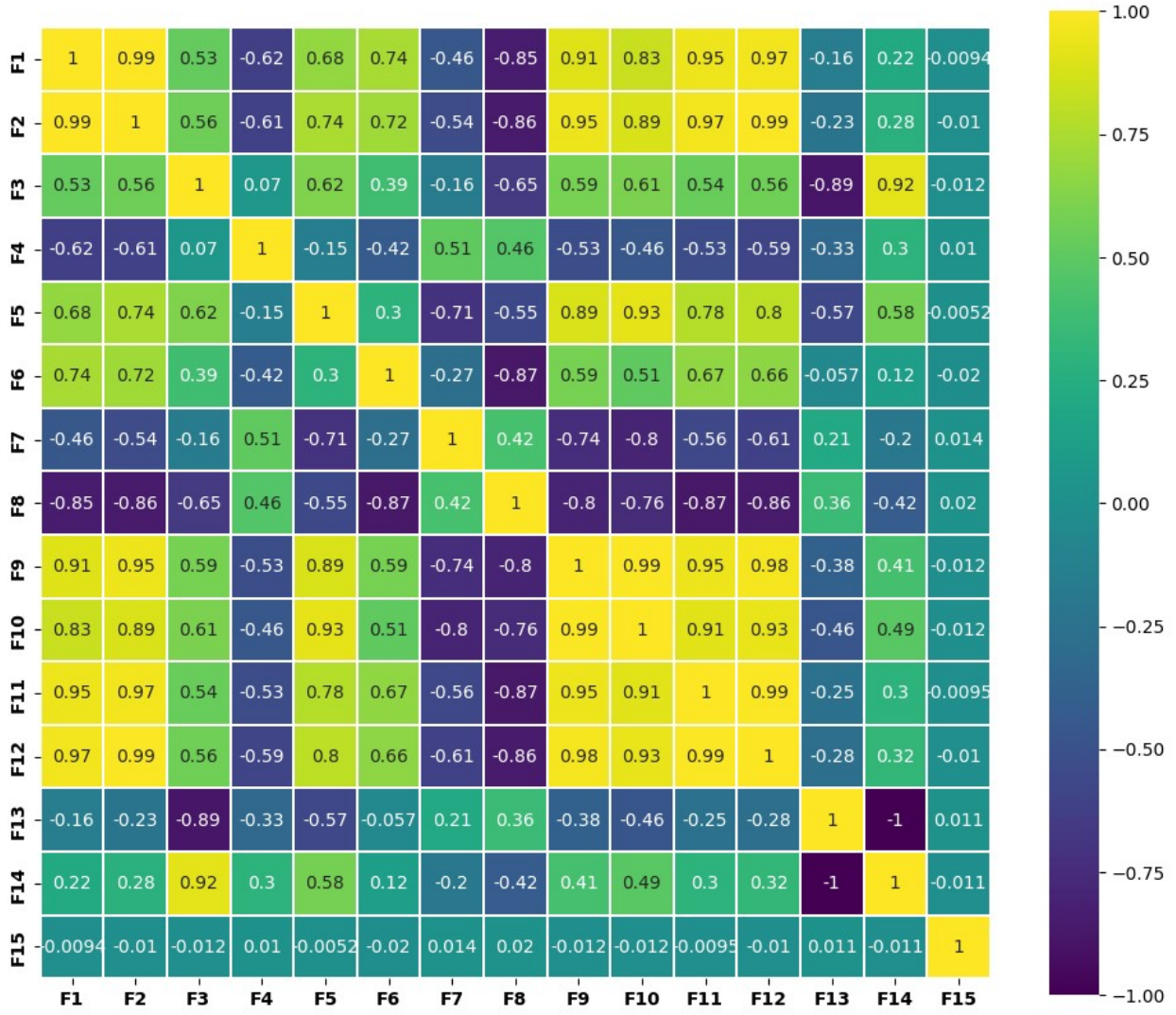


Figure S7. Feature-feature correlation matrix of Pearson's correlation coefficients (PCCs) for dGMP data set. The +1, -1, and 0 on the scale indicate the maximum positive, maximum negative, and no correlation, respectively.

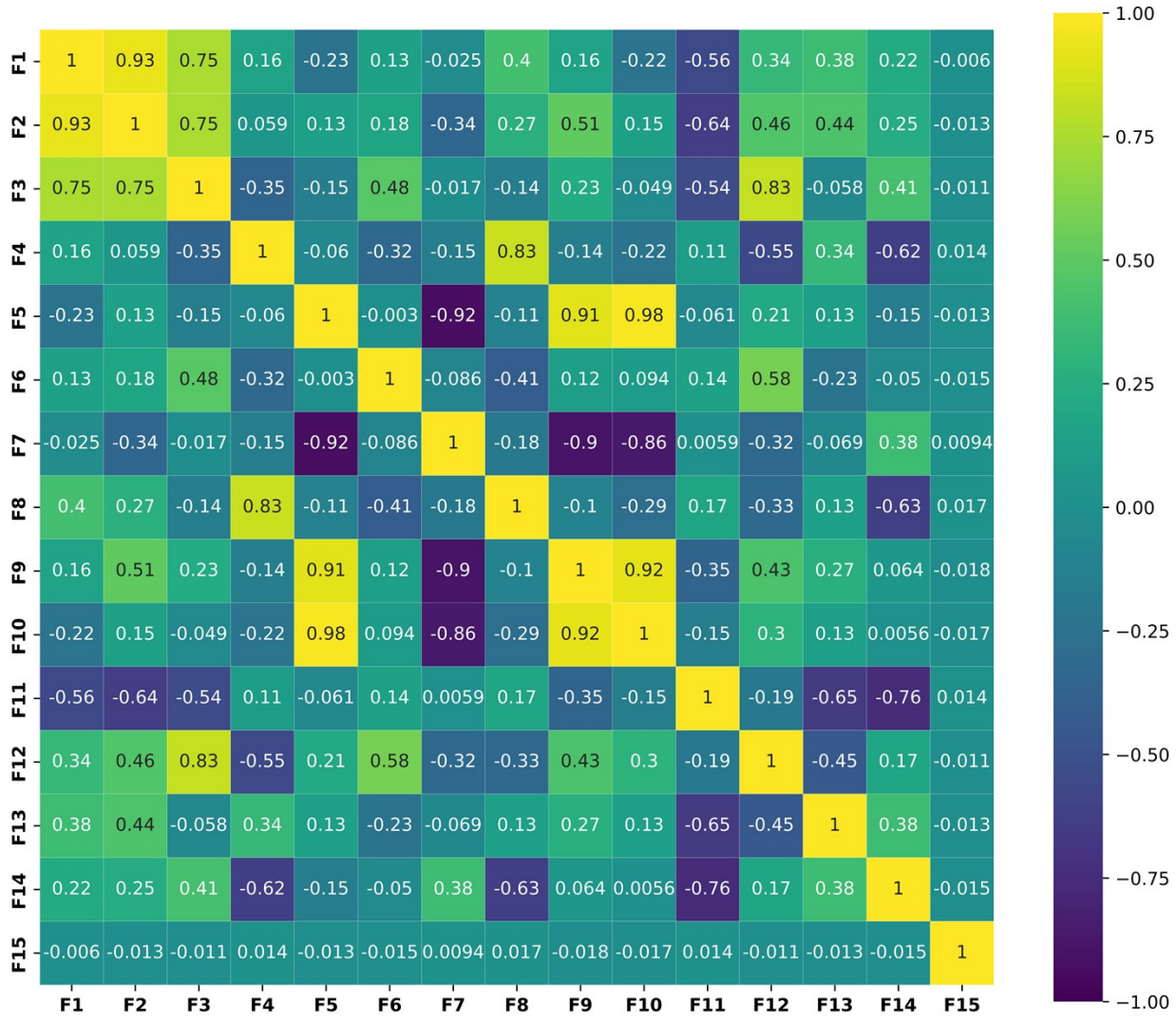


Figure S8. Feature-feature correlation matrix of Pearson's correlation coefficients (PCCs) for dCMP data set. The +1, -1, and 0 on the scale indicate the maximum positive, maximum negative, and no correlation, respectively.

8. Spearman's Rank Correlation Matrix Plots

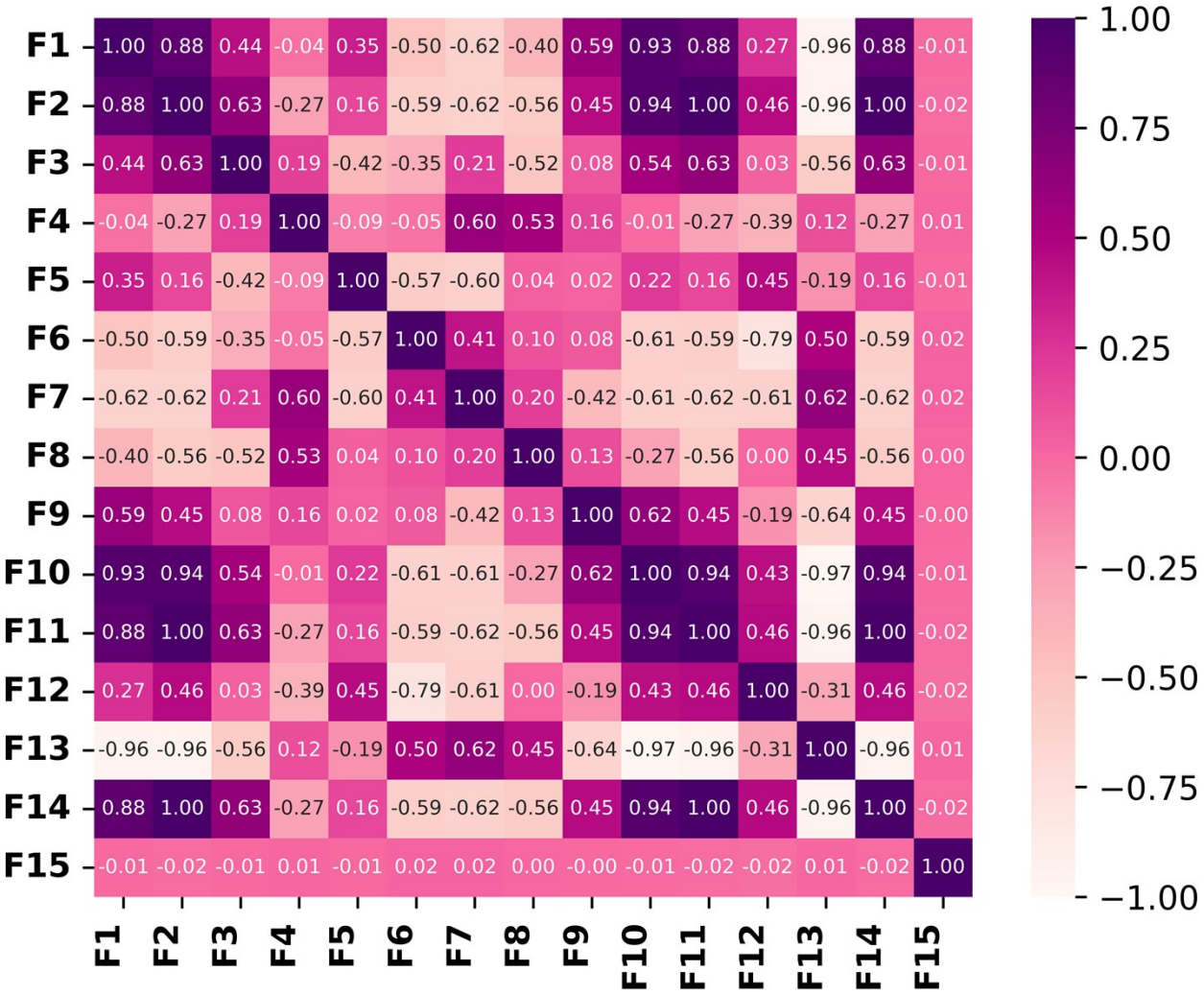


Figure S9. Spearman's rank correlation coefficients (ρ) for the dAMP data set. The +1, -1, and 0 on the scale indicate the maximum positive, maximum negative, and no correlation, respectively.

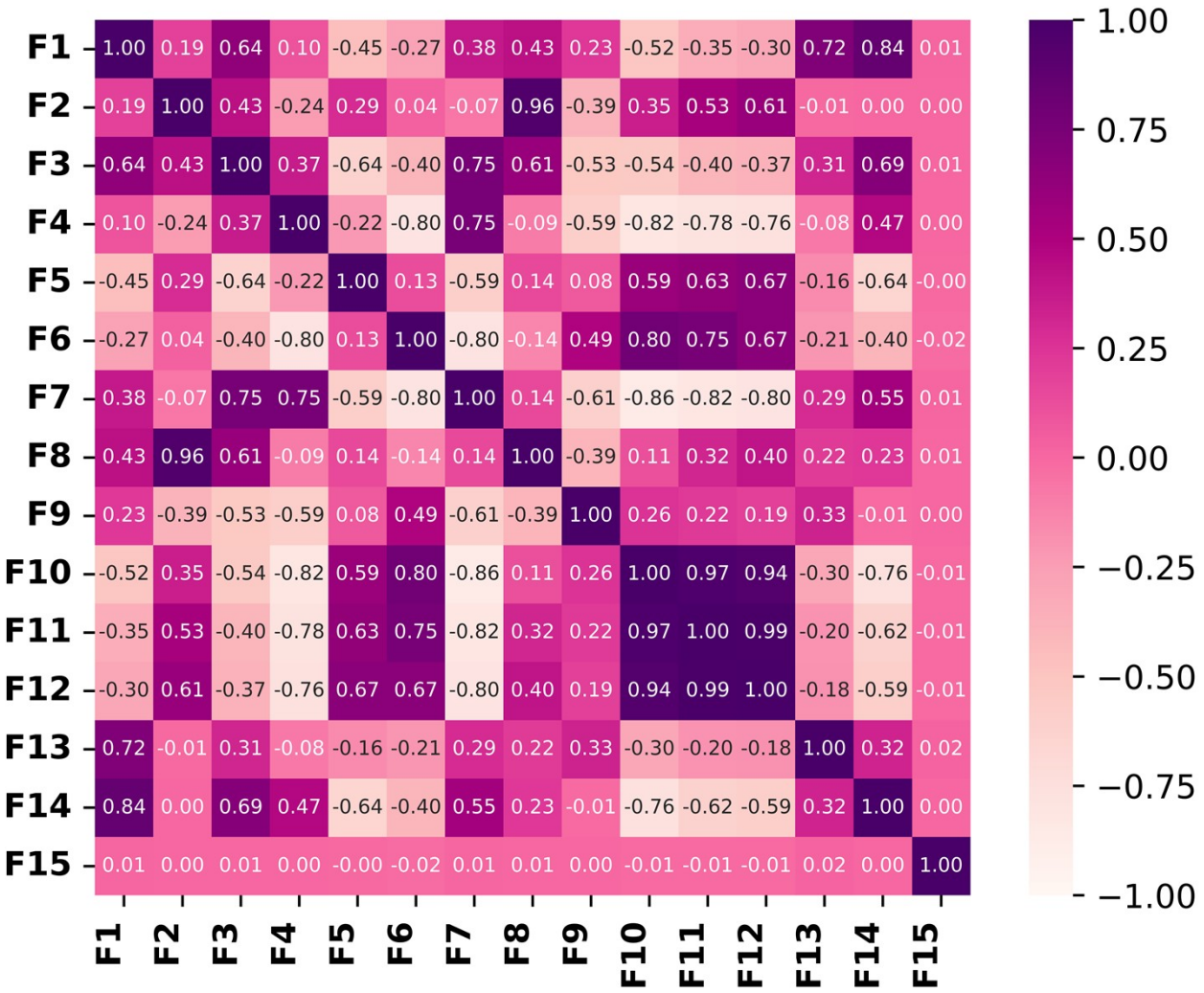


Figure S10. Spearman's rank correlation coefficients (ρ) for the dTMP data set. The +1, -1, and 0 on the scale indicate the maximum positive, maximum negative, and no correlation, respectively.

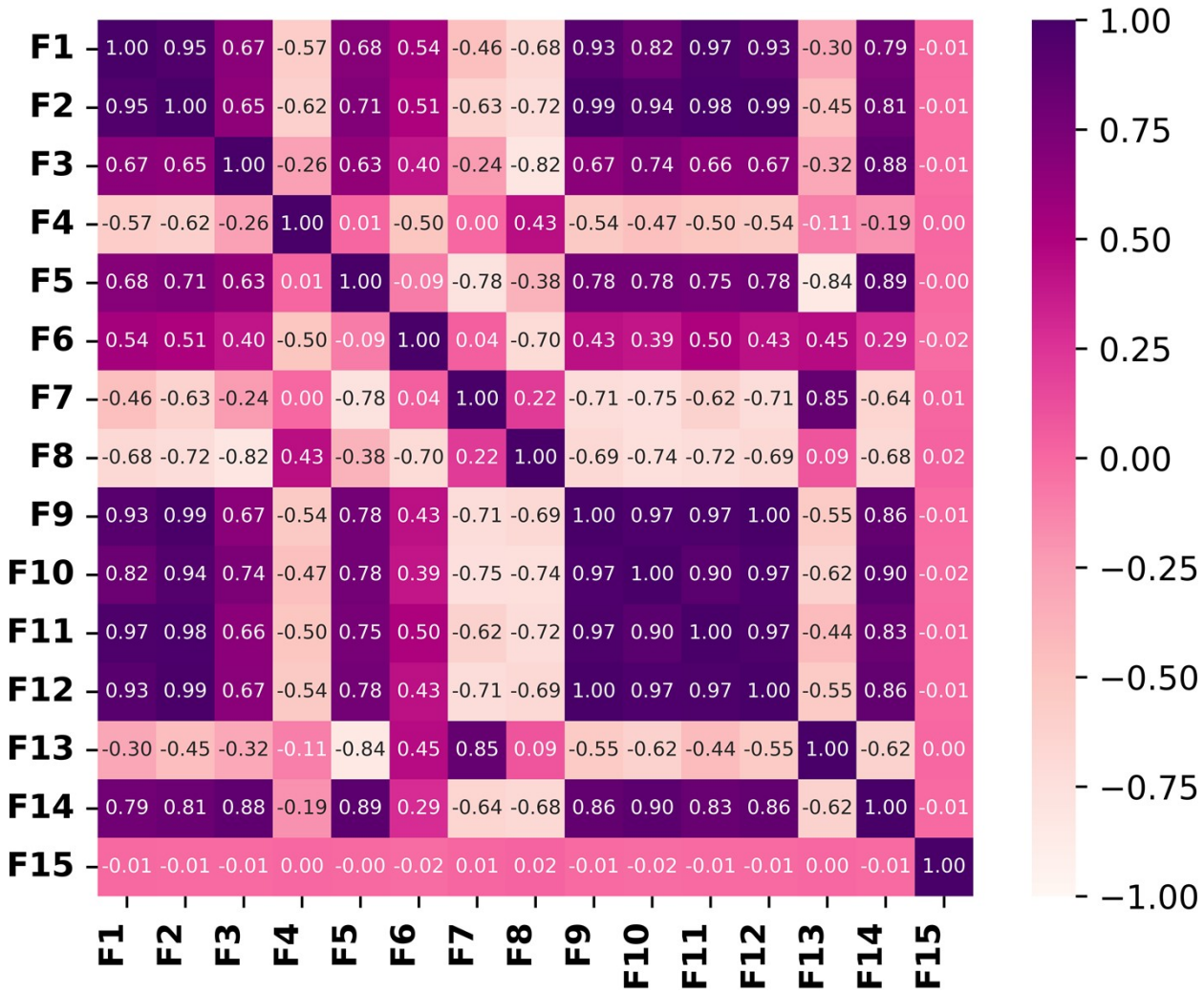


Figure S11. Spearman's rank correlation coefficients (ρ) for the dGMP data set. The +1, -1, and 0 on the scale indicate the maximum positive, maximum negative, and no correlation, respectively.

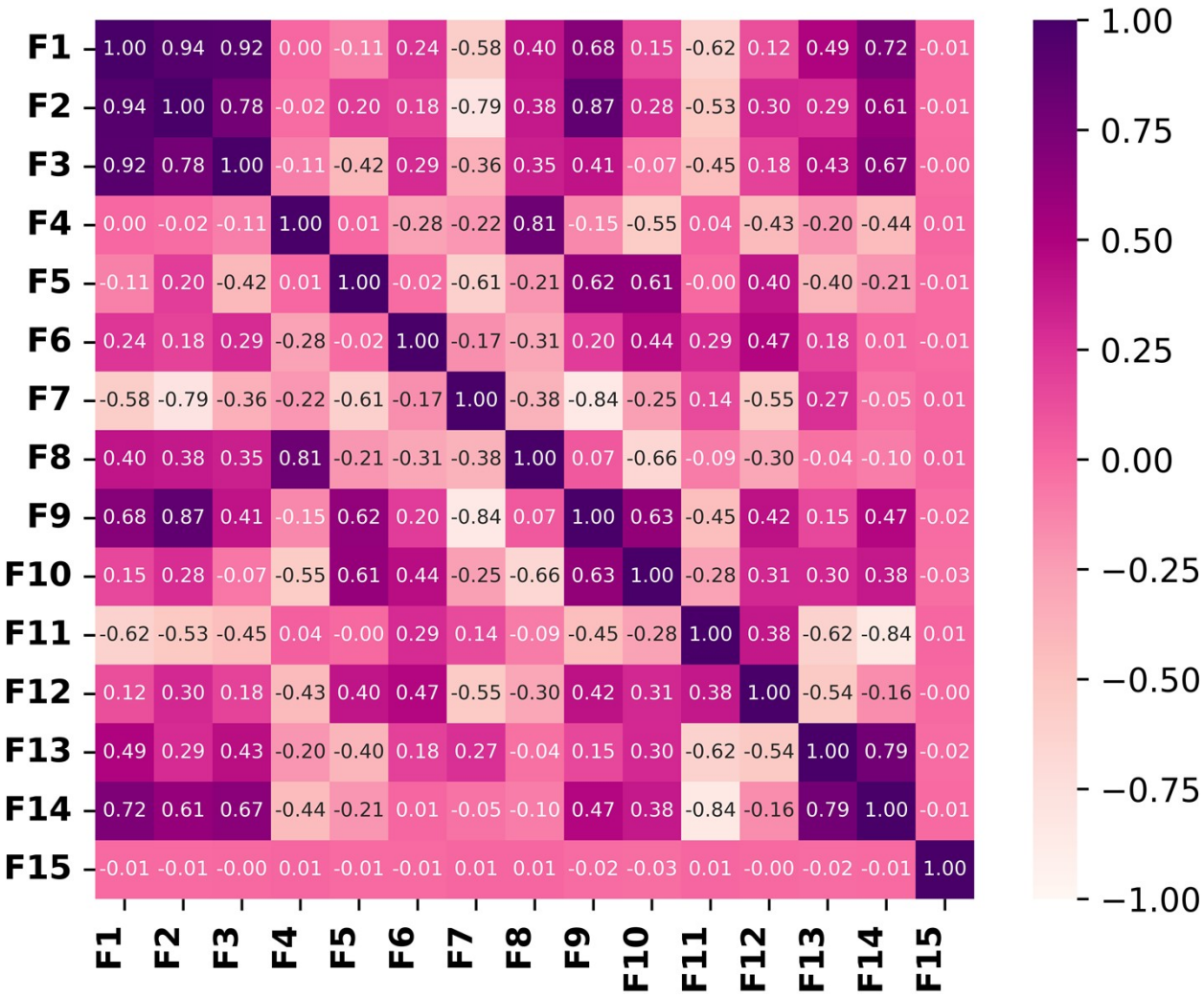


Figure S12. Spearman's rank correlation coefficients (ρ) for the dCMP data set. The +1, -1, and 0 on the scale indicate the maximum positive, maximum negative, and no correlation, respectively.

9. Optimized Hyperparameters of the ML Models

Table S3. Optimized Hyperparameters Values with their Corresponding RMSEs for the Seven Considered Regression Models with the dAMP Data Set.

Models	Hyperparameters	RMSE
LR	'positive': 'False', 'n_jobs': 50, 'fit_intercept': 'False', 'copy_X': 'True'	**1.2**
XGBR	'verbose': 10, 'n_estimators': 100, 'min_child_weight': 4, 'max_depth': 15, 'learning_rate': 0.2, 'cv': 1, 'booster': 'gbtree', 'base_score': 0.1	**0.08**
KRR	'kernel': 'laplacian', 'gamma': 0.4, 'degree': 5, 'coef0': 0, 'alpha': 0	**0.10**
RFR	'n_estimators': 50, 'min_samples_split': 20, 'min_samples_leaf': 9, 'max_features': 'auto', 'max_depth': 30, 'bootstrap': True	**0.11**
AdaBoost	'random_state': 52, 'n_estimators': 60, 'loss': 'linear', 'learning_rate': 1	**0.32**
ETR	'splitter': 'best', 'min_weight_fraction_leaf': 0.02, 'min_samples_leaf': 1, 'max_leaf_nodes': None, 'max_features': 'auto', 'max_depth': 14	**0.17**
GPR	'kernel__k2__noise_level': 2.0, 'kernel__k1__k2__length_scale': 10.0, 'kernel__k1__k1__constant_value': 2.0, 'alpha': 1e-05	**0.14**

Table S4. Optimized Hyperparameters Values with their Corresponding RMSEs for the Seven Considered Regression Models with the dTMP Data Set.

Models	Hyperparameters	RMSE
LR	'positive': 'True', 'n_jobs': 40, 'fit_intercept': 'True', 'copy_X': 'True'	**1.21**
XGBR	'verbose': 40, 'n_estimators': 2000, 'min_child_weight': 3, 'max_depth': 25, 'learning_rate': 0.06, 'cv': 15, 'booster': 'gbtree', 'base_score': 0	**0.07**
KRR	'kernel': 'laplacian', 'gamma': 1.0, 'degree': 3, 'coef0': 2, 'alpha': 0	**0.11**
RFR	'n_estimators': 600, 'min_samples_split': 2, 'min_samples_leaf': 1, 'max_features': 'auto', 'max_depth': 20, 'bootstrap': True	**0.08**
AdaBoost	'base_estimator': None, 'learning_rate': 1.0, 'loss': 'linear', 'n_estimators': 50, 'random_state': None	**0.27**
ETR	'splitter': 'best', 'min_weight_fraction_leaf': 0.02, 'min_samples_leaf': 1, 'max_leaf_nodes': None, 'max_features': 'auto', 'max_depth': 14	**0.16**
GPR	'kernel__k2__noise_level': 5.0, 'kernel__k1__k2__length_scale': 10.0, 'kernel__k1__k1__constant_value': 10.0, 'alpha': 1e-10	**0.13**

Table S5. Optimized Hyperparameters Values with their Corresponding RMSEs for the Seven Considered Regression Models with the dGMP Data Set.

Models	Hyperparameters	RMSE
LR	'positive': 'True', 'n_jobs': 20, 'fit_intercept': 'False', 'copy_X': 'True'	**1.19**
XGBR	'verbose': 20, 'n_estimators': 1100, 'min_child_weight': 5, 'max_depth': 10, 'learning_rate': 0.05, 'cv': 2, 'booster': 'gbtree', 'base_score': 0	**0.07**
KRR	'kernel': 'laplacian', 'gamma': 1.0, 'degree': 10, 'coef0': 1, 'alpha': 0	**0.04**
RFR	'n_estimators': 50, 'min_samples_split': 20, 'min_samples_leaf': 9, 'max_features': 'auto', 'max_depth': 30, 'bootstrap': True	**0.12**
AdaBoost	'random_state': 52, 'n_estimators': 60, 'loss': 'linear', 'learning_rate': 1	**0.29**
ETR	'splitter': 'best', 'min_weight_fraction_leaf': 0.02, 'min_samples_leaf': 1, 'max_leaf_nodes': None, 'max_features': 'auto', 'max_depth': 14	**0.18**
GPR	'kernel__k2__noise_level': 10.0, 'kernel__k1__k2__length_scale': 10.0, 'kernel__k1__k1__constant_value': 2.0, 'alpha': 1e-05	**0.14**

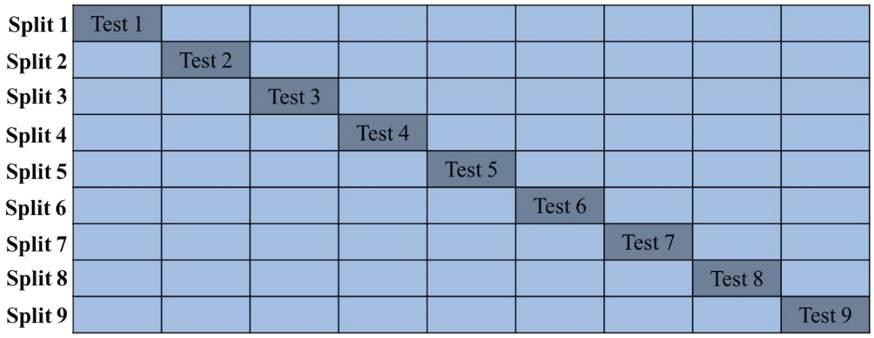
Table S6. Optimized Hyperparameters Values with their Corresponding RMSEs for the Seven Considered Regression Models with the dCMP Data Set.

Models	Hyperparameters	RMSE
LR	'positive': 'False', 'n_jobs': 30, 'fit_intercept': 'False', 'copy_X': 'True'	**1.20**
XGBOOST	'verbose': 10, 'n_estimators': 100, 'min_child_weight': 3, 'max_depth': 25, 'learning_rate': 0.1, 'cv': 5, 'booster': 'gbtree', 'base_score': -0.25	**0.08**
KRR	'kernel': 'laplacian', 'gamma': 1.0, 'degree': 3, 'coef0': 2, 'alpha': 0	**0.10**
RFR	'n_estimators': 80, 'min_samples_split': 20, 'min_samples_leaf': 1, 'max_features': 'auto', 'max_depth': 20, 'bootstrap': Fals	**0.10**
AdaBoost	'random_state': 52, 'n_estimators': 800, 'loss': 'linear', 'learning_rate': 2	**0.25**
ETR	'splitter': 'best', 'min_weight_fraction_leaf': 0.02, 'min_samples_leaf': 1, 'max_leaf_nodes': None, 'max_features': 'auto', 'max_depth': 14	**0.15**
GPR	'kernel__k2__noise_level': 1.0, 'kernel__k1__k2__length_scale': 5.0, 'kernel__k1__k1__constant_value': 5.0, 'alpha': 1e-05	**0.13**

10. K-fold Cross-Validation

Text S3:

Cross-validation is a common technique used in machine learning to assess the performance of a model. It involves splitting the data into multiple sets, with some being used for training the model and others for evaluating it. The idea is to simulate the model's performance on new, unseen data by using only a subset of the available data to train the model. This is done to avoid overfitting, where a model performs well on the training data but poorly on new data. Nine-fold cross-validation is a specific type of cross-validation that involves dividing the data into nine equal parts or folds. The model is trained on eight folds, and the remaining fold is used as a validation set to test the model's performance. This process is repeated nine times, with each fold used as the validation set once. The performance metrics are then averaged across all nine folds to obtain a final estimate of the model's performance.



11. Global Interpretability into the prediction of Transmission T(*E*) with XGBR:dAMP, XGBR:dTMP, and XGBR:dGMP Models, respectively

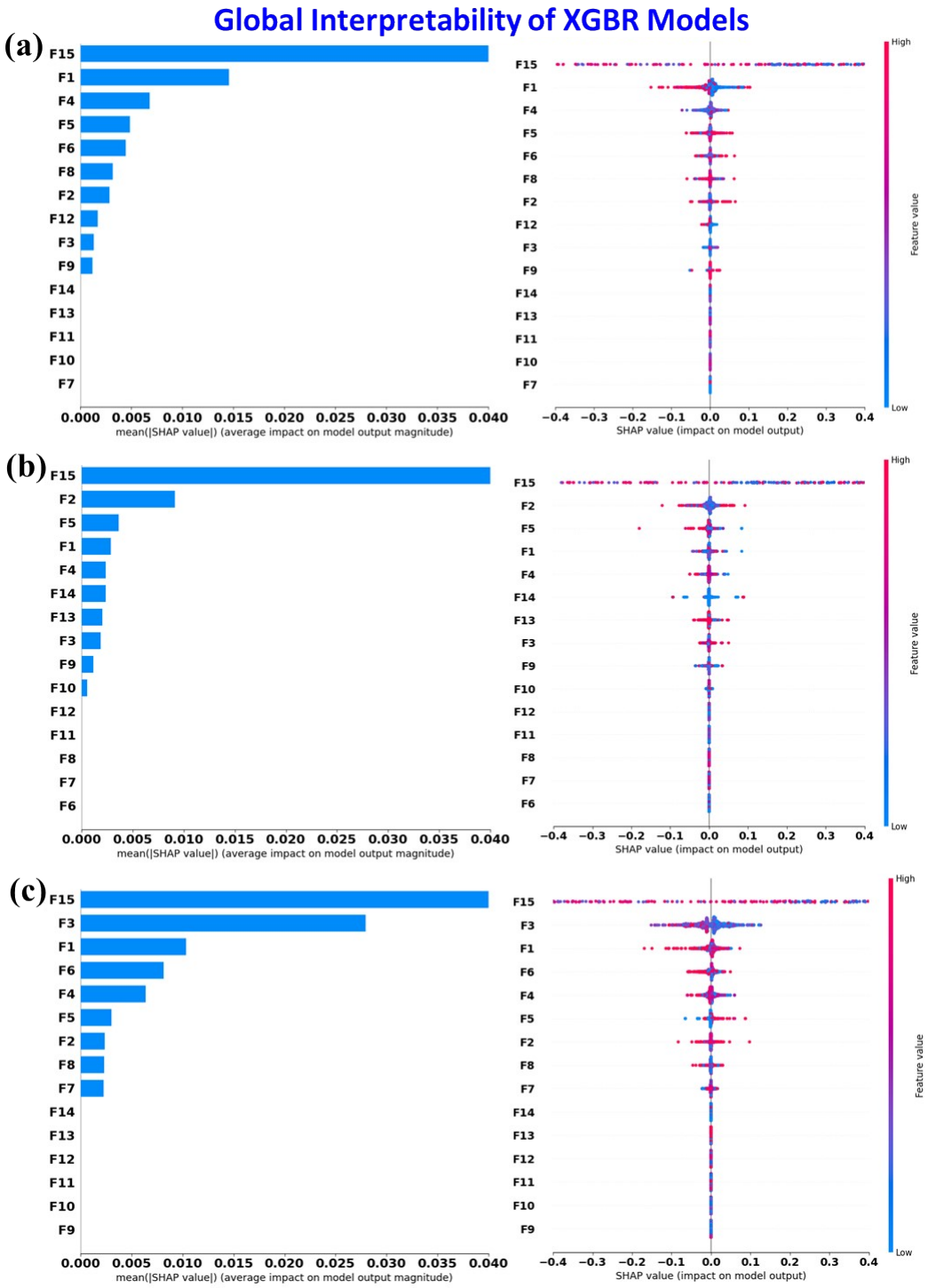


Figure S13. Global ML interpretability analysis with Shap bar and bee-swarm global feature importance plot for optimized (a) XGBR:dAMP, (b) XGBR:dTMP, and (c) XGBR:dGMP, respectively.

12. Local Interpretability into the prediction of Transmission T(*E*) with XGBR:dAMP, XGBR:dTMP, and XGBR:dGMP Models, respectively.

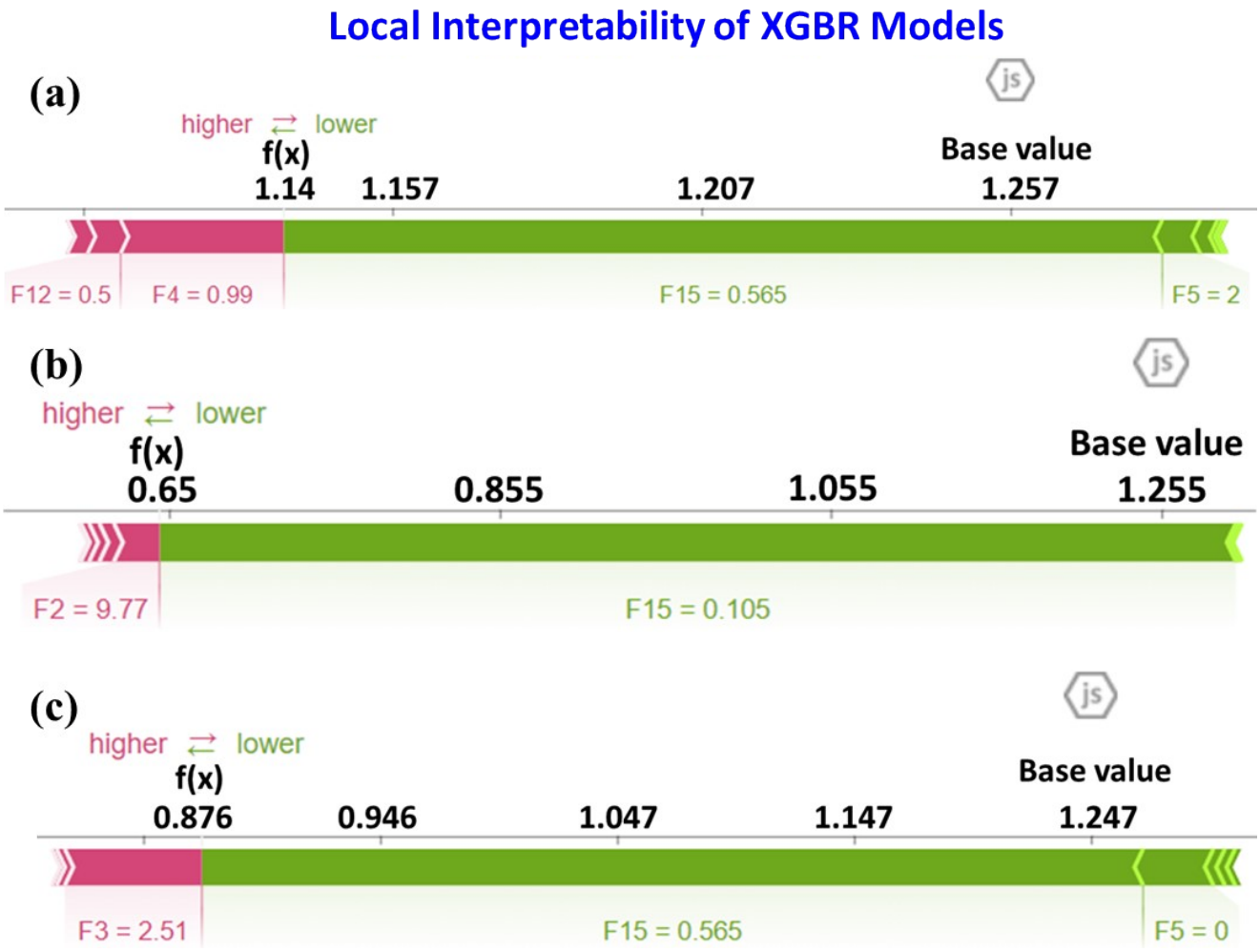


Figure S14. Local interpretability analysis with Shap force plot for optimized (a) XGBR:dAMP, (b) XGBR:dTMP, and XGBR:dGMP, respectively.

13. Rotation Dynamics Prediction of Nucleotides with Four Optimized Models

(a) Predicting capabilities of the optimized XGBR:dAMP model on the rotational dynamics of nucleotides

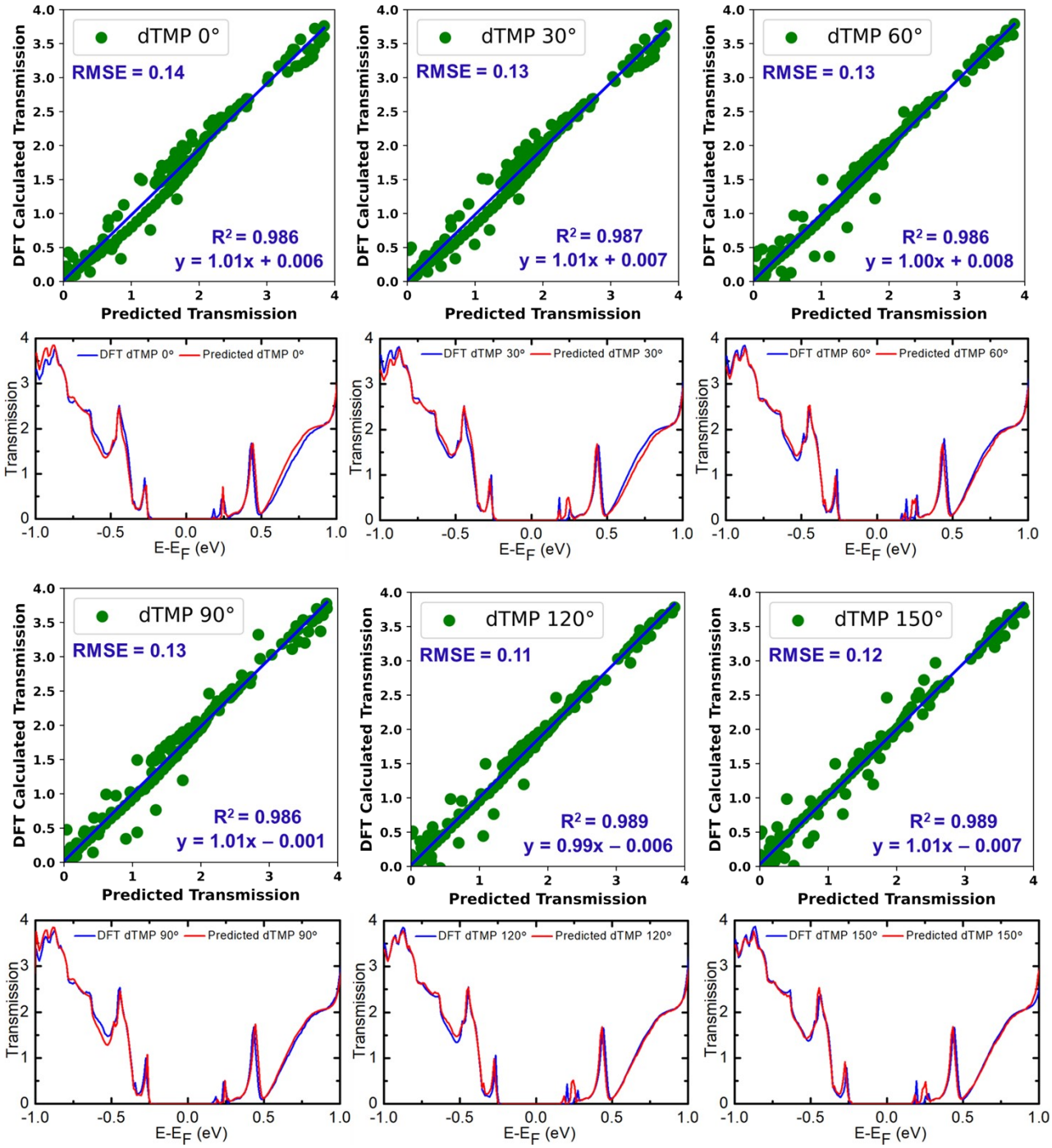


Figure S15a. The parity plots with calculated RMSE scores and R^2 values along with the predicted transmission versus DFT calculated transmission. A parallel comparison of DFT versus predicted transmission spectra is provided for predicted dynamic configurations of dTMP, dGMP, and dCMP nucleotides with the optimized **XGBR:dAMP** model.

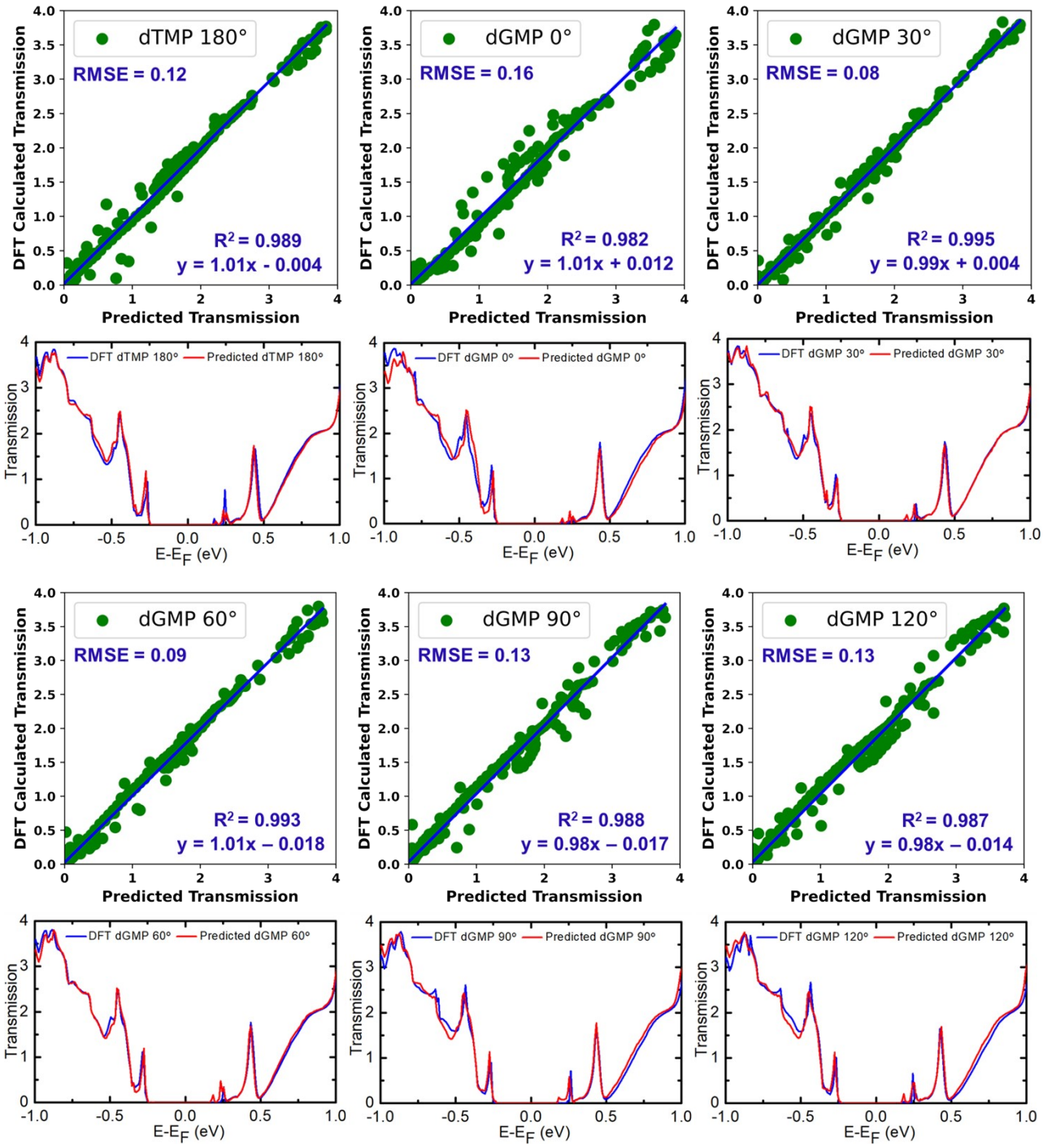


Figure S15b. The parity plots with calculated RMSE scores and R^2 values along with the predicted transmission versus DFT calculated transmission. A parallel comparison of DFT versus predicted transmission spectra is provided for predicted dynamic configurations of dTMP, dGMP, and dCMP nucleotides with the optimized **XGBR:dAMP** model.

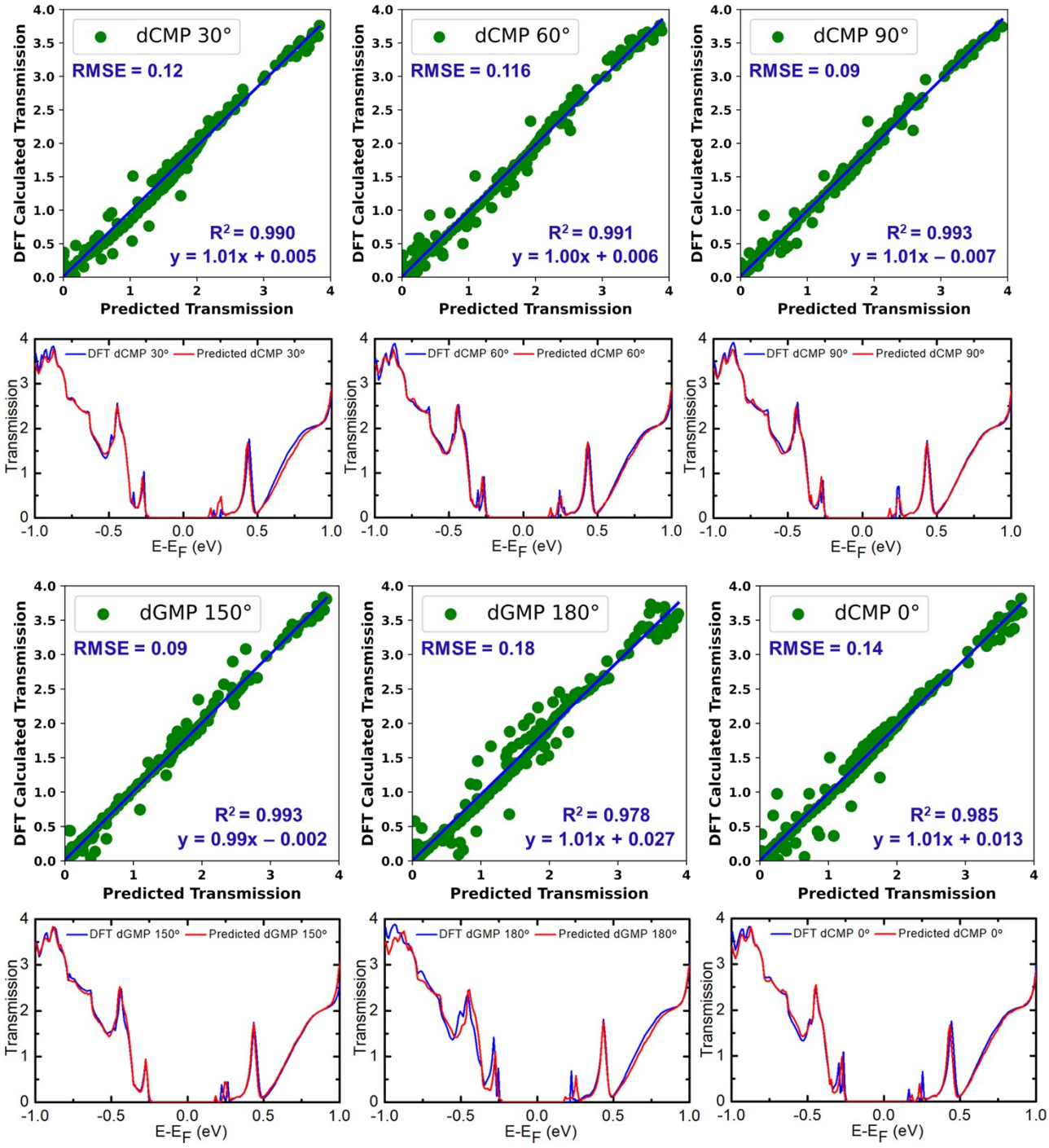


Figure S15c. The parity plots with calculated RMSE scores and R^2 values along with the predicted transmission versus DFT calculated transmission. A parallel comparison of DFT versus predicted transmission spectra is provided for predicted dynamic configurations of dTMP, dGMP, and dCMP nucleotides with the optimized **XGBR:dAMP** model.

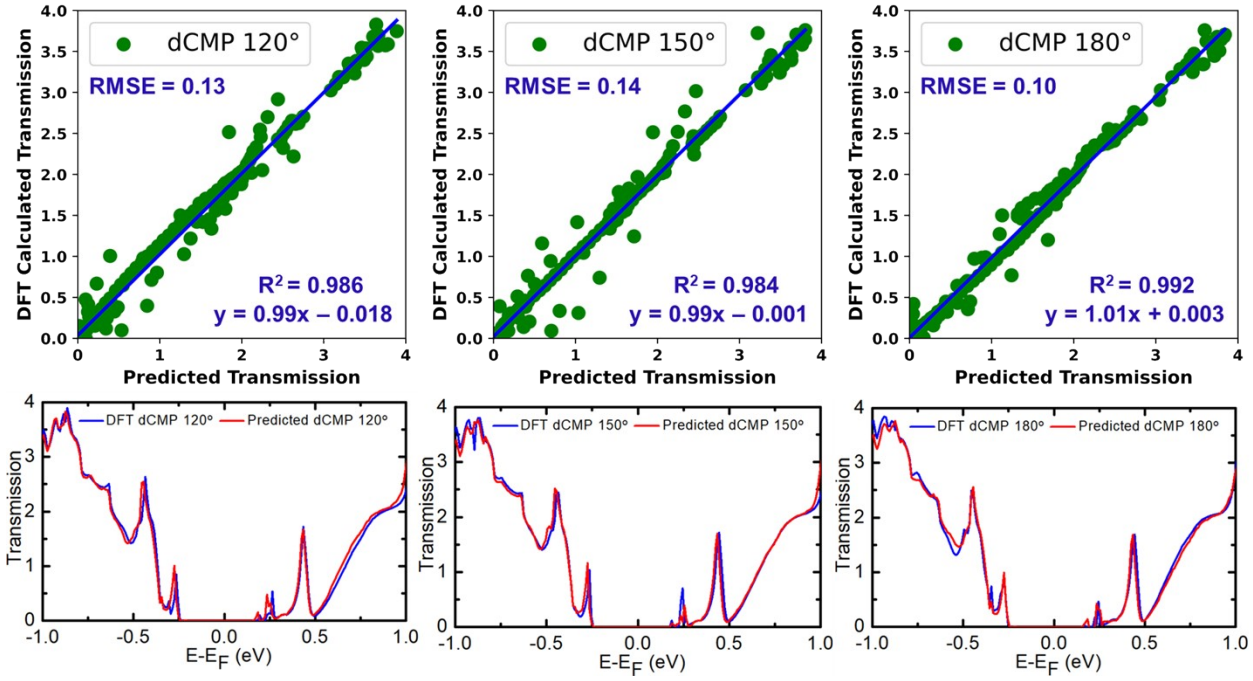


Figure S15d. The parity plots with calculated RMSE scores and R^2 values along with the predicted transmission versus DFT calculated transmission. A parallel comparison of DFT versus predicted transmission spectra is provided for predicted dynamic configurations of dTMP, dGMP, and dCMP nucleotides with the optimized **XGBR:dAMP** model.

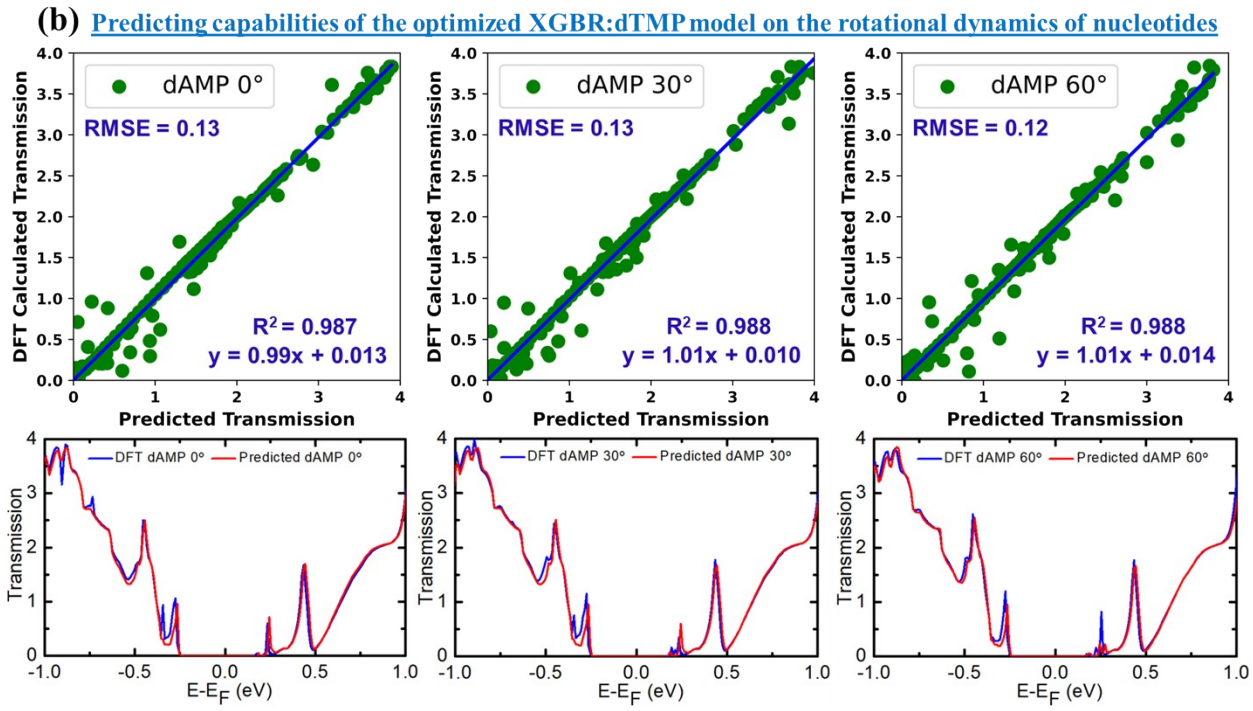


Figure S16a. The parity plots with calculated RMSE scores and R^2 values along with the predicted transmission versus DFT calculated transmission. A parallel comparison of DFT versus predicted transmission spectra is provided for predicted dynamic configurations of dAMP, dGMP, and dCMP nucleotides with the optimized **XGBR:dTMP** model.

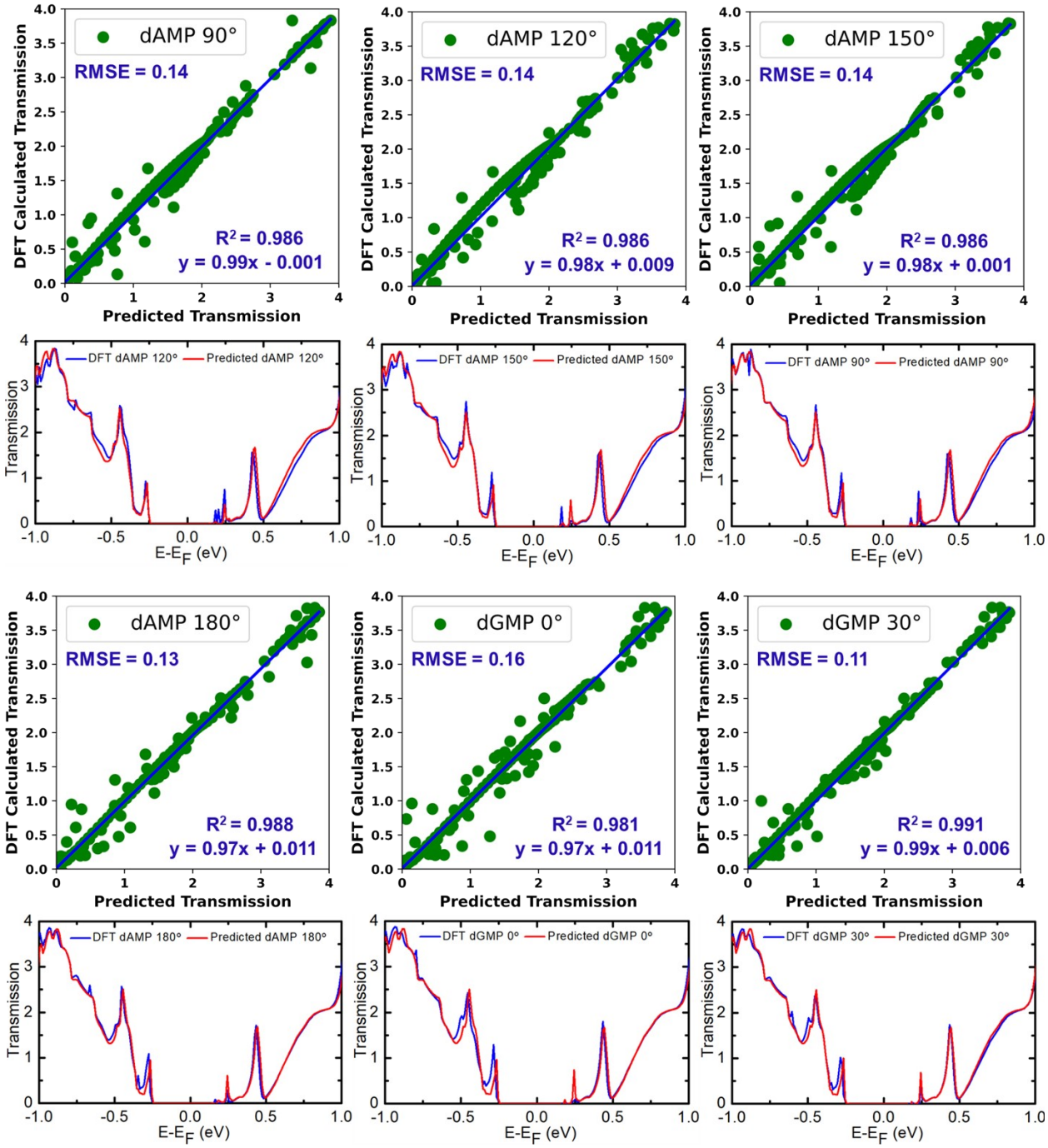


Figure S16b. The parity plots with calculated RMSE scores and R^2 values along with the predicted transmission versus DFT calculated transmission. A parallel comparison of DFT versus predicted transmission spectra is provided for predicted dynamic configurations of dAMP, dGMP, and dCMP nucleotides with the optimized **XGBR:dTMP** model.

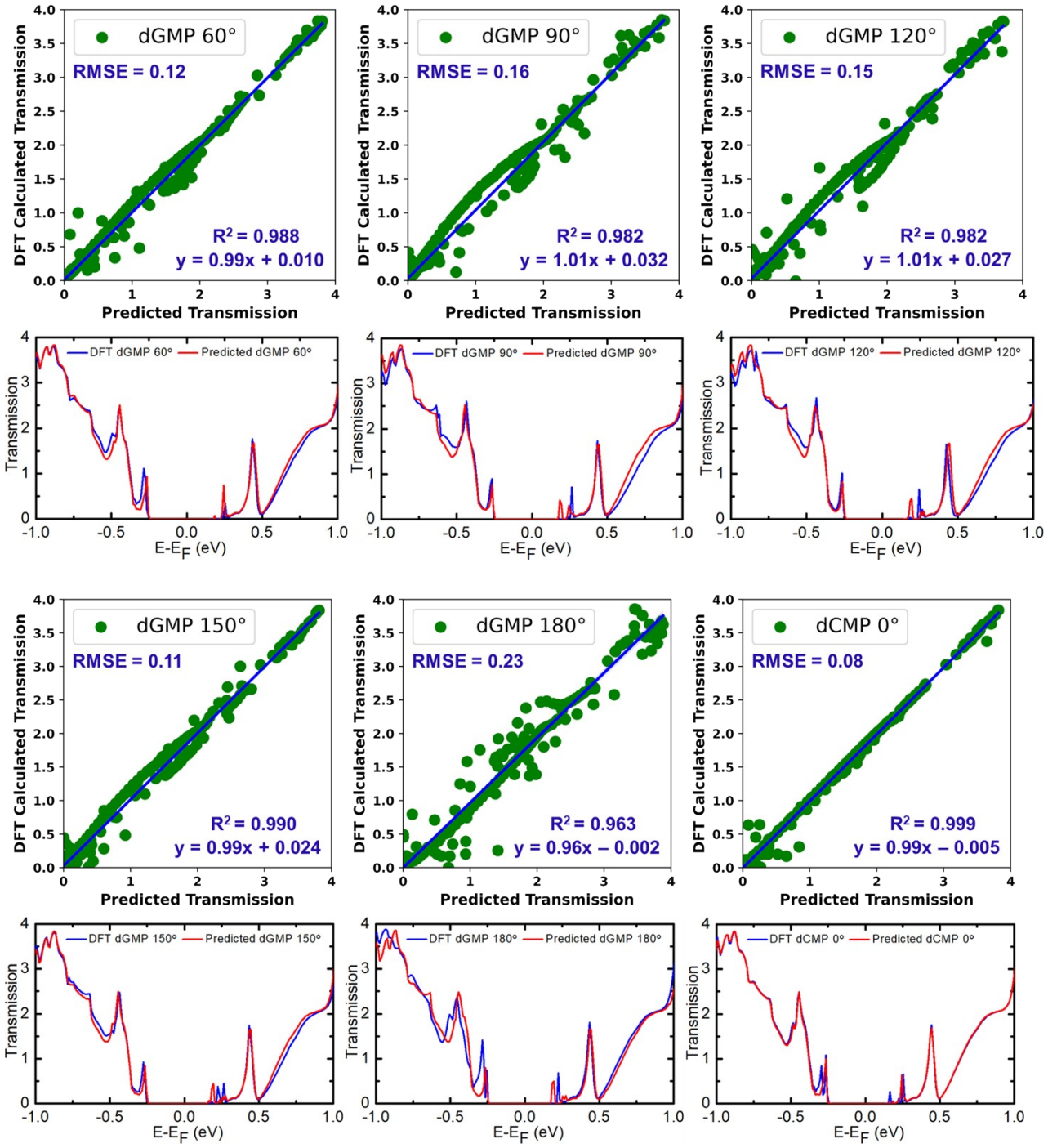


Figure S16c. The parity plots with calculated RMSE scores and R^2 values along with the predicted transmission versus DFT calculated transmission. A parallel comparison of DFT versus predicted transmission spectra is provided for predicted dynamic configurations of dAMP, dGMP, and dCMP nucleotides with the optimized **XGBR:dTMP** model.

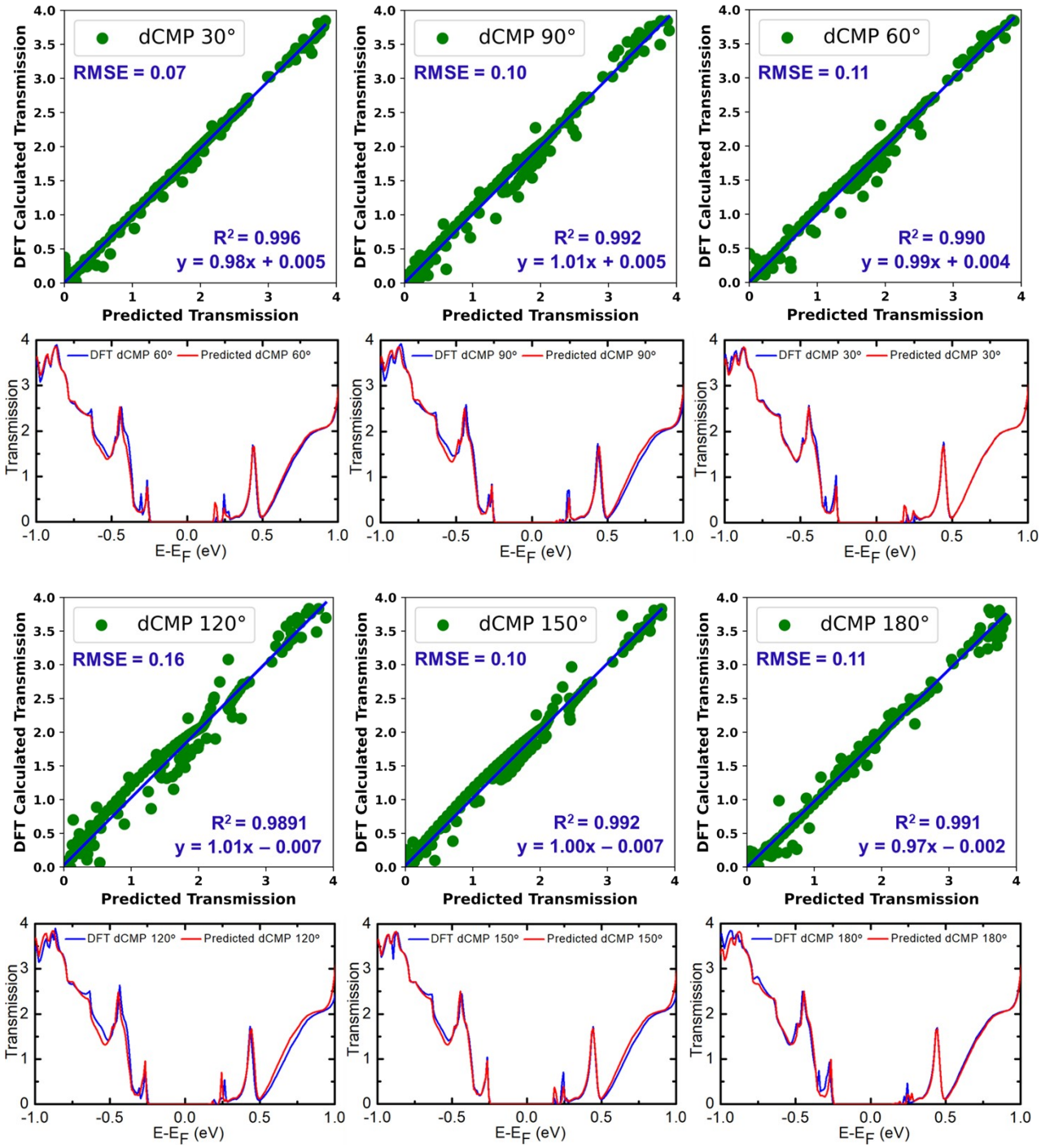


Figure S16d. The parity plots with calculated RMSE scores and R^2 values along with the predicted transmission versus DFT calculated transmission. A parallel comparison of DFT versus predicted transmission spectra is provided for predicted dynamic configurations of dAMP, dGMP, and dCMP nucleotides with the optimized **XGBR:dTMP** model.

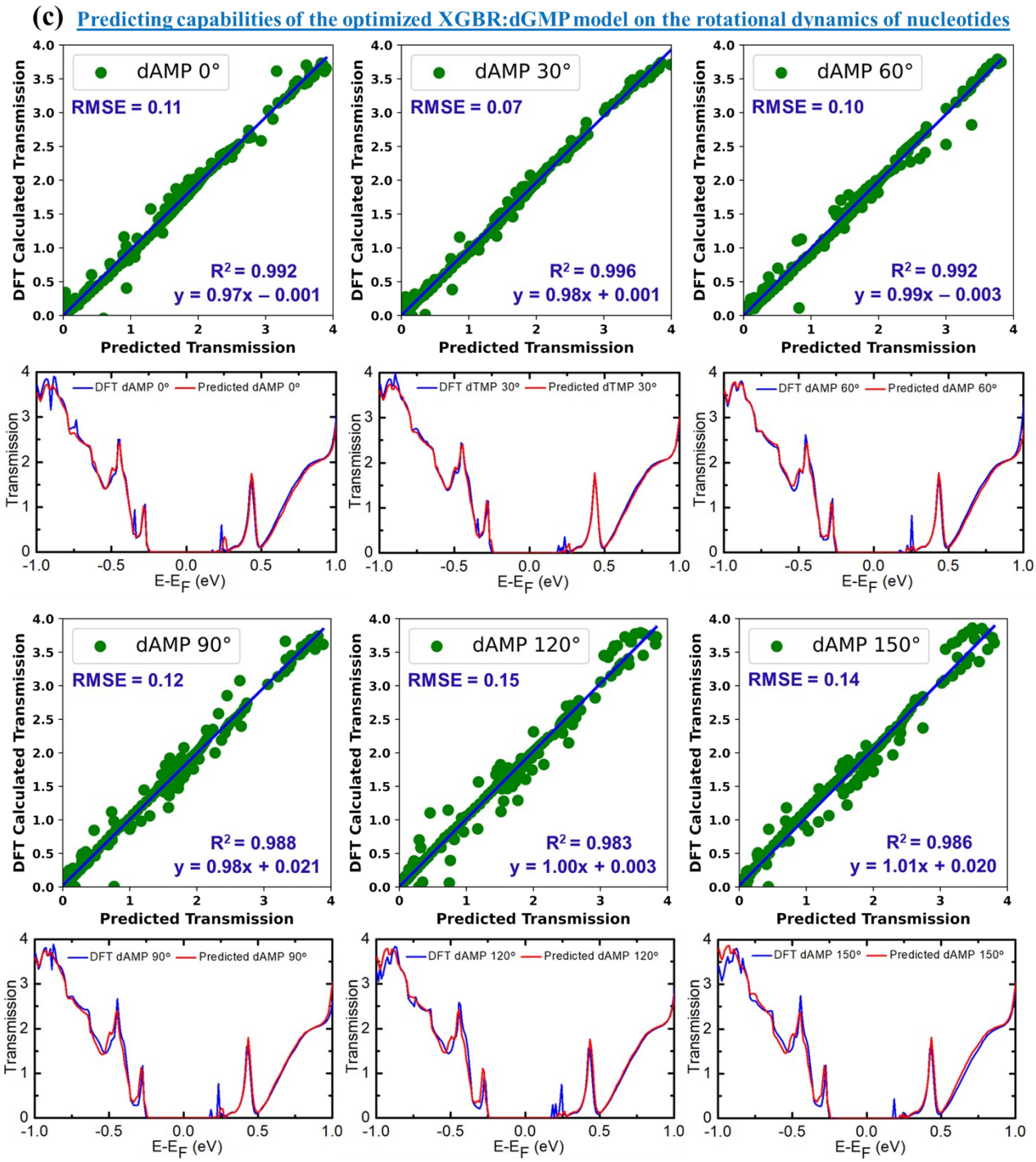


Figure S17a. The parity plots with calculated RMSE scores and R^2 values along with the predicted transmission versus DFT calculated transmission. A parallel comparison of DFT versus predicted transmission spectra is provided for predicted dynamic configurations of dAMP, dTMP, and dCMP nucleotides with the optimized **XGBR:dGMP** model.

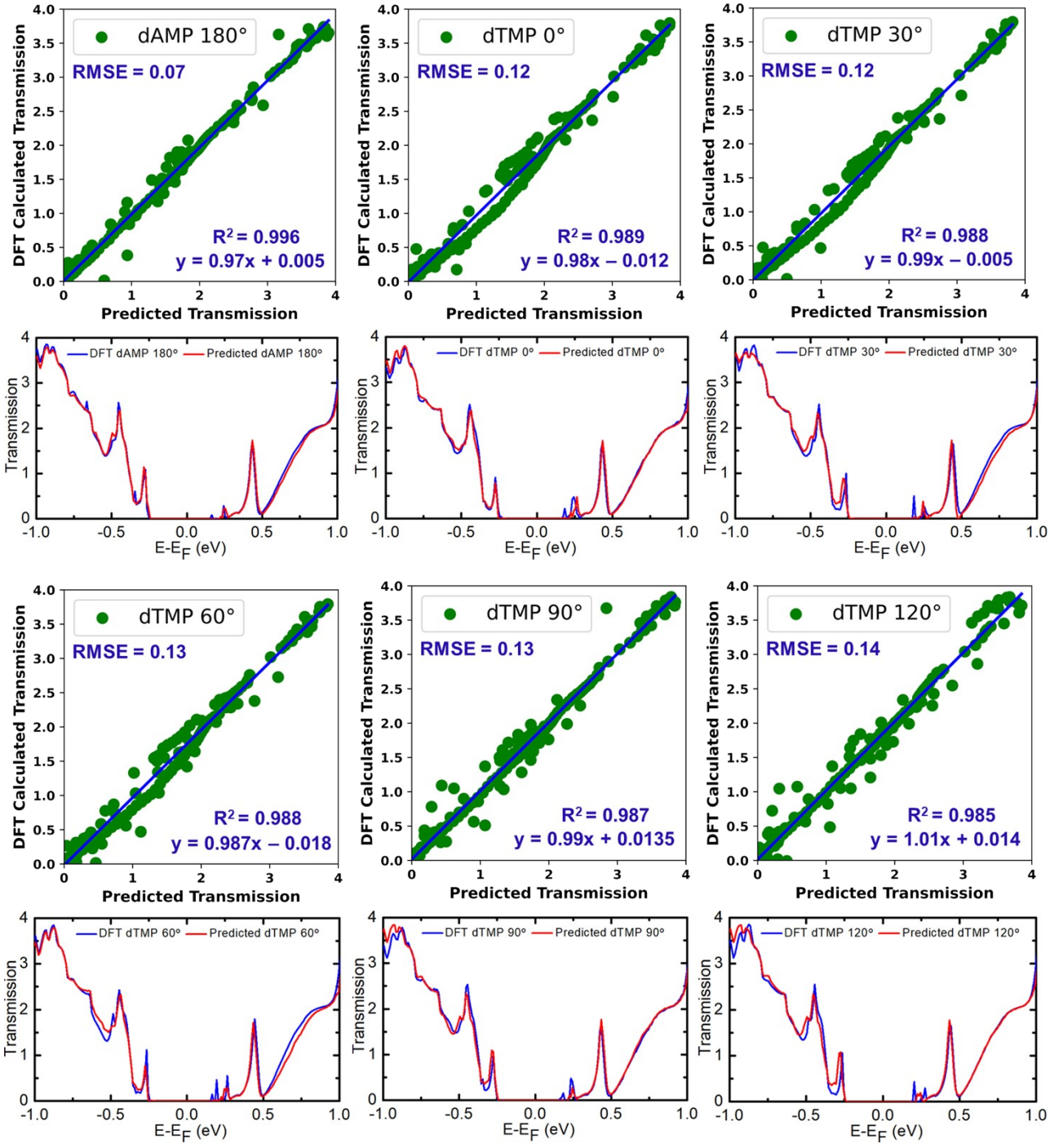


Figure S17b. The parity plots with calculated RMSE scores and R^2 values along with the predicted transmission versus DFT calculated transmission. A parallel comparison of DFT versus predicted transmission spectra is provided for predicted dynamic configurations of dAMP, dTMP, and dCMP nucleotides with the optimized **XGBR:dGMP** model.

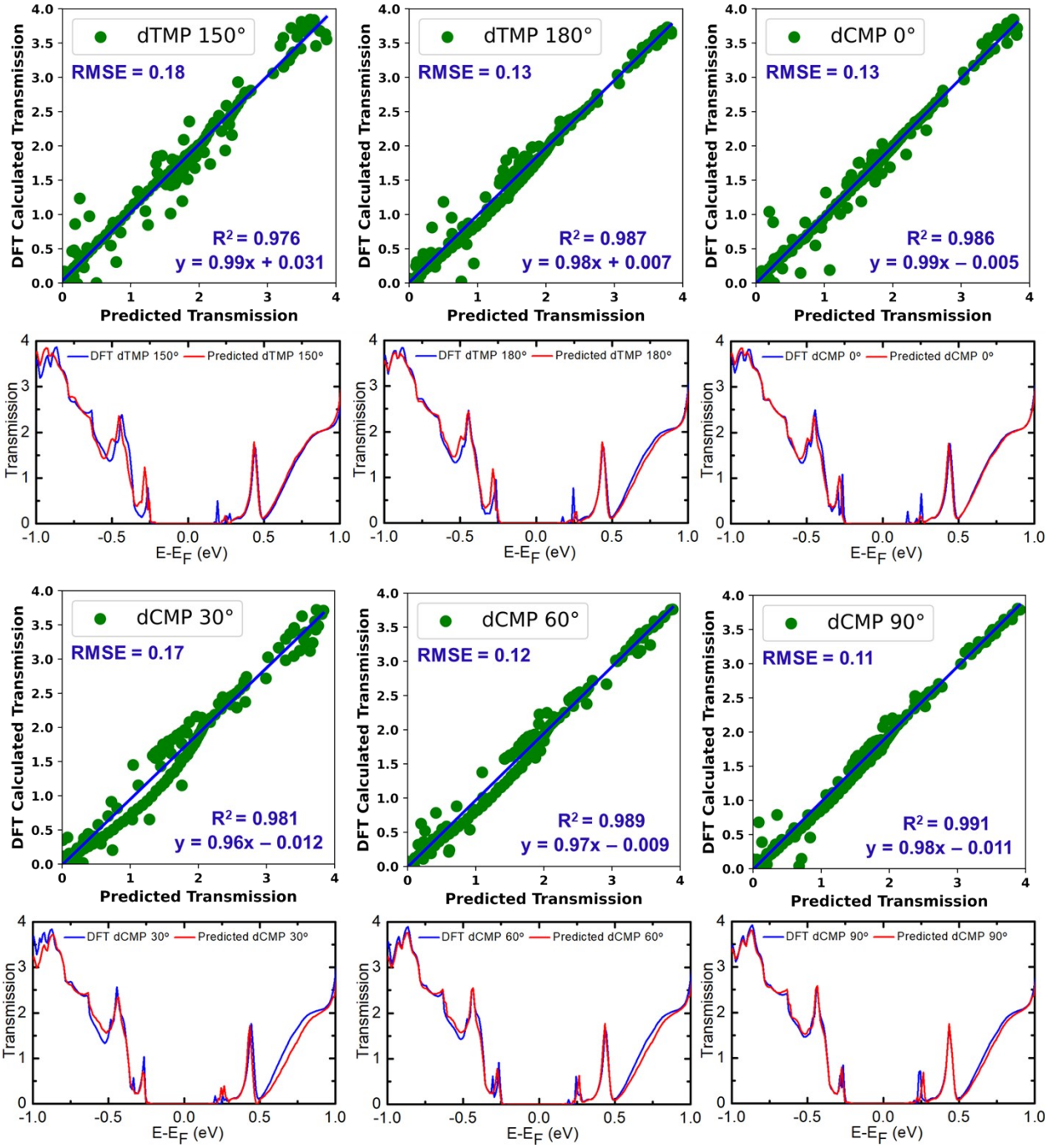


Figure S17c. The parity plots with calculated RMSE scores and R^2 values along with the predicted transmission versus DFT calculated transmission. A parallel comparison of DFT versus predicted transmission spectra is provided for predicted dynamic configurations of dAMP, dTMP, and dCMP nucleotides with the optimized **XGBR:dGMP** model.

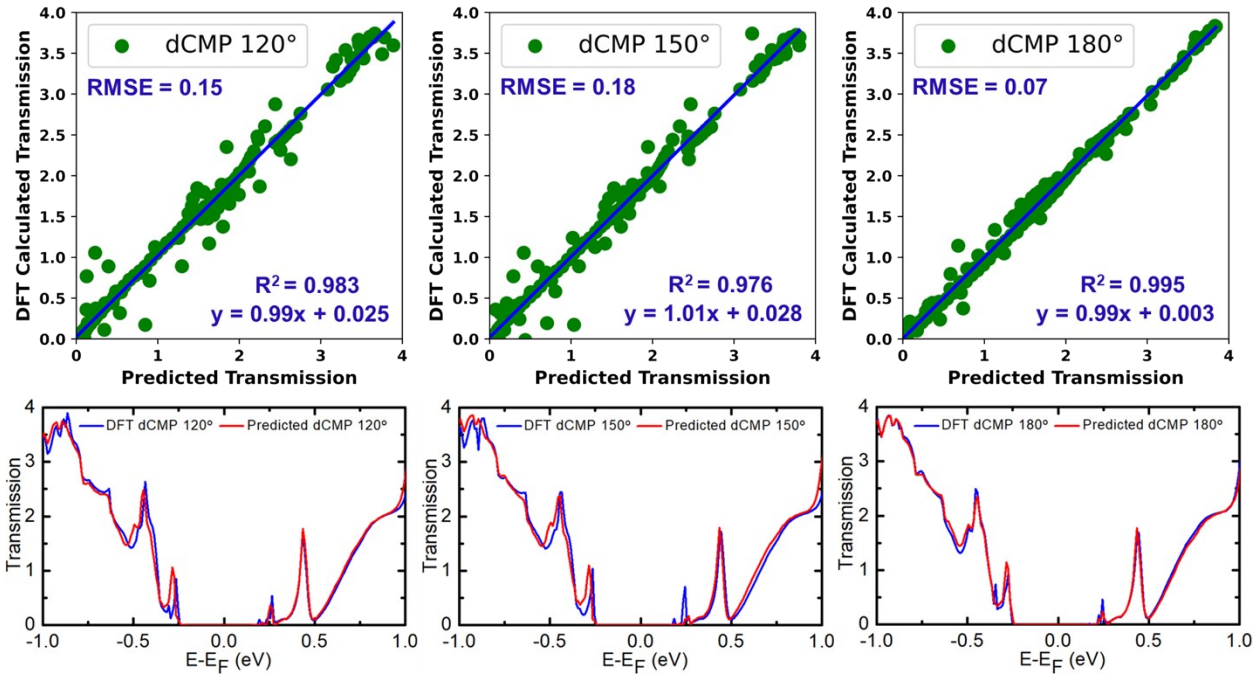


Figure S17d. The parity plots with calculated RMSE scores and R^2 values along with the predicted transmission versus DFT calculated transmission. A parallel comparison of DFT versus predicted transmission spectra is provided for predicted dynamic configurations of dAMP, dTMP, and dCMP nucleotides with the optimized **XGBR:dGMP** model.

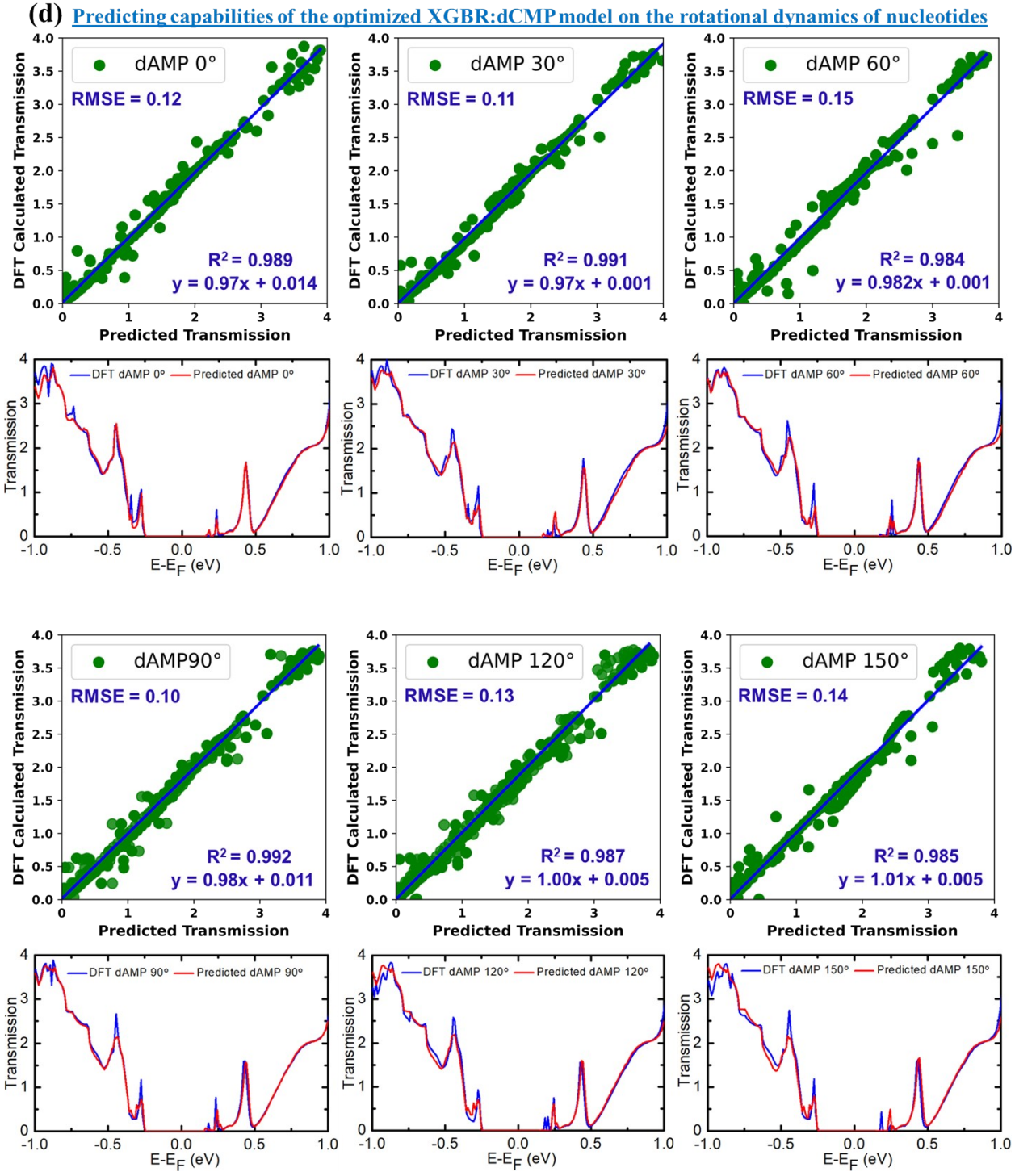


Figure S18a. The parity plots with calculated RMSE scores and R^2 values along with the predicted transmission versus DFT calculated transmission. A parallel comparison of DFT versus predicted transmission spectra is provided for predicted dynamic configurations of dAMP, dTMP, and dGMP nucleotides with the optimized **XGBR:dCMP** model.

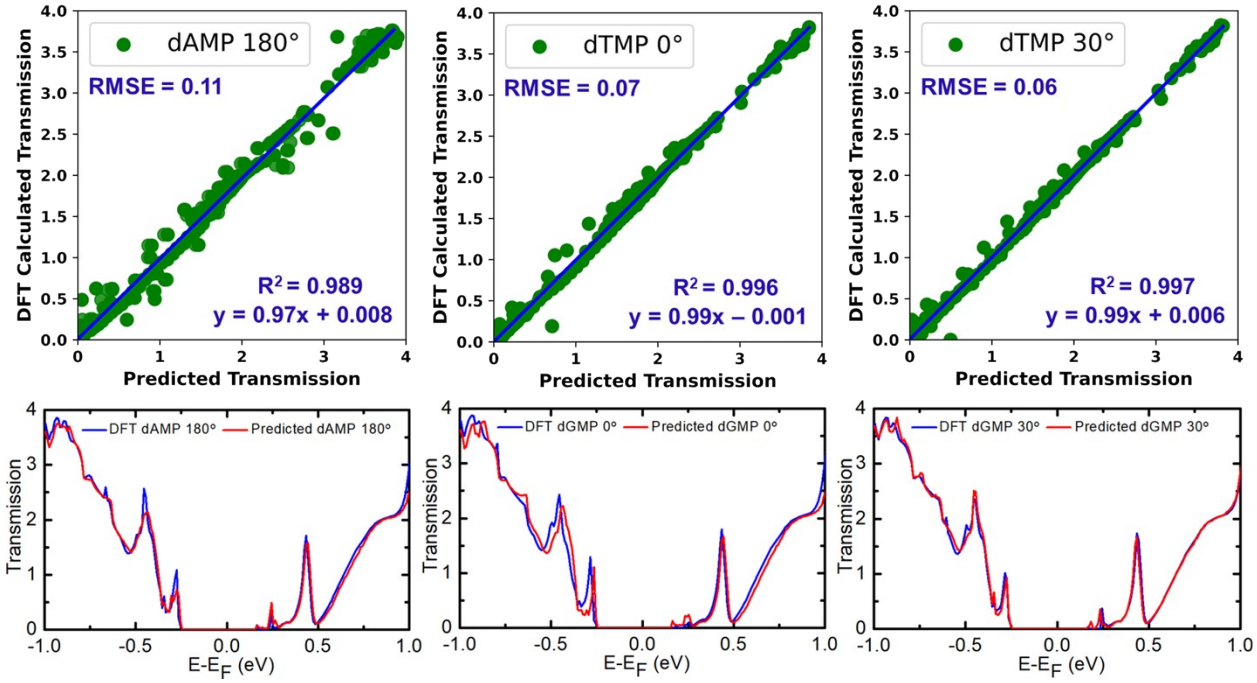


Figure S18b. The parity plots with calculated RMSE scores and R^2 values along with the predicted transmission versus DFT calculated transmission. A parallel comparison of DFT versus predicted transmission spectra is provided for predicted dynamic configurations of dAMP, dTMP, and dGMP nucleotides with the optimized **XGBR:dCMP** model.

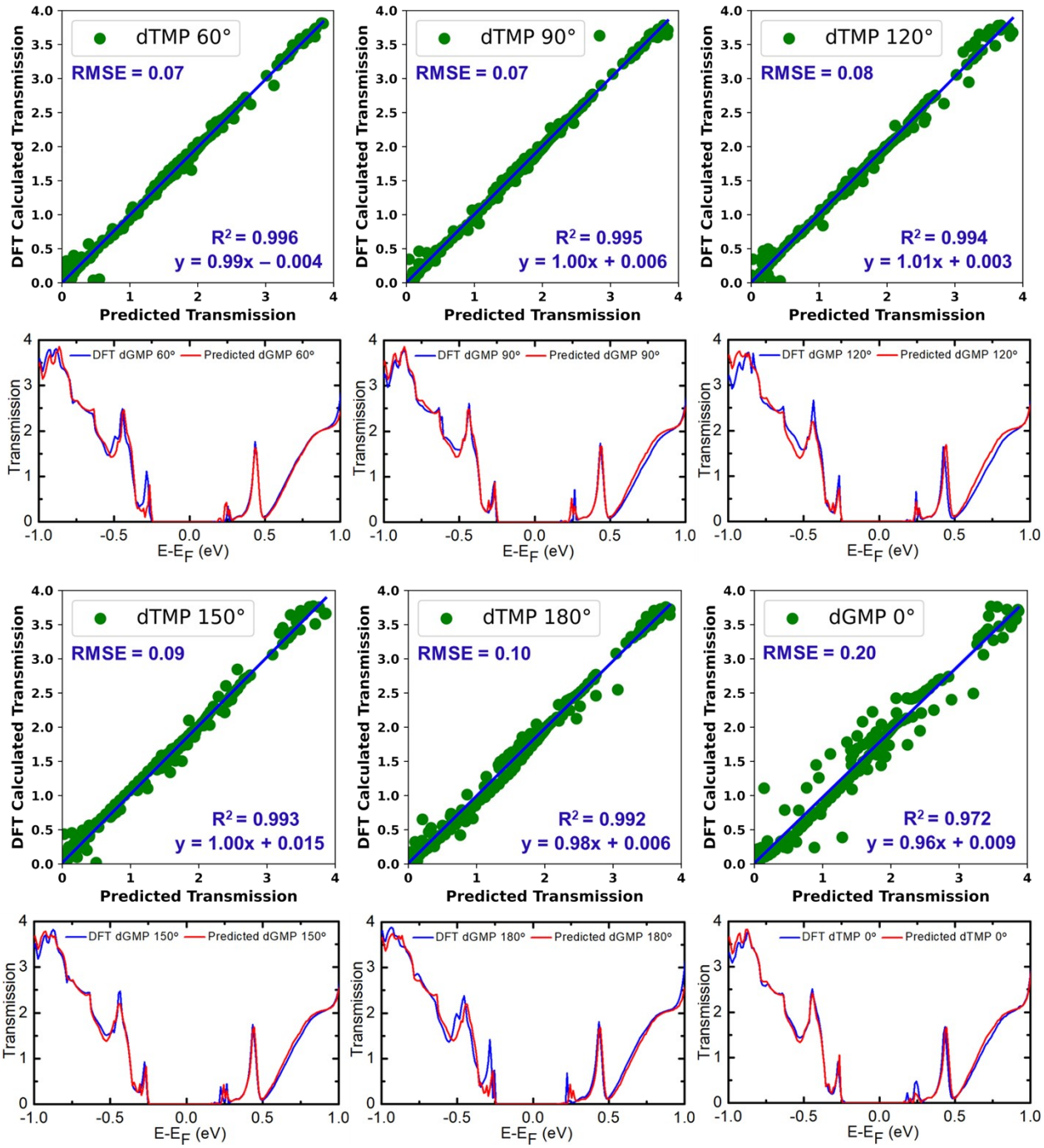


Figure S18c. The parity plots with calculated RMSE scores and R^2 values along with the predicted transmission versus DFT calculated transmission. A parallel comparison of DFT versus predicted transmission spectra is provided for predicted dynamic configurations of dAMP, dTMP, and dGMP nucleotides with the optimized **XGBR:dCMP** model.

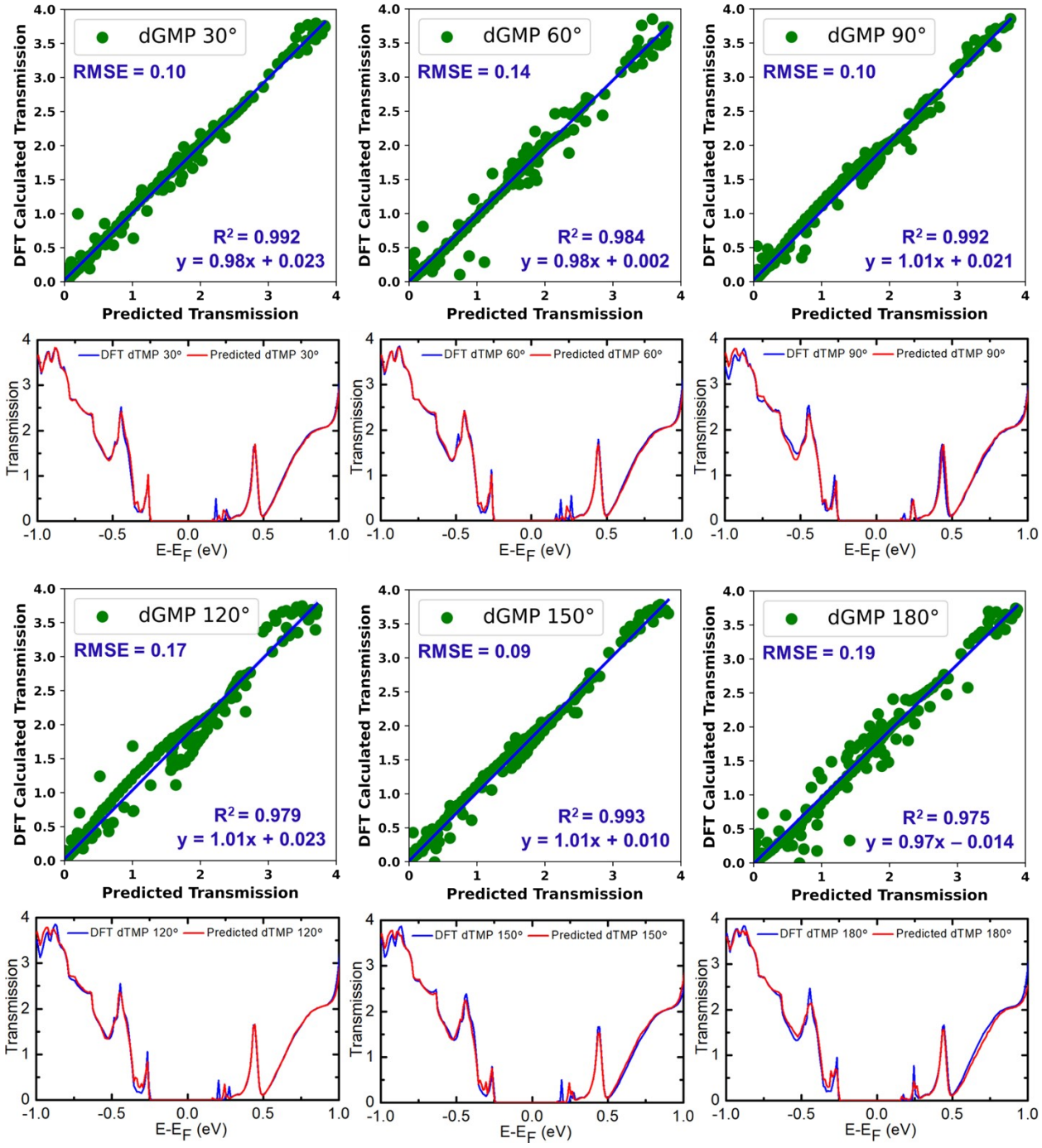


Figure S18d. The parity plots with calculated RMSE scores and R^2 values along with the predicted transmission versus DFT calculated transmission. A parallel comparison of DFT versus predicted transmission spectra is provided for predicted dynamic configurations of dAMP, dTMP, and dGMP nucleotides with the optimized **XGBR:dCMP** model.

14. Eigenchannel Wavefunctions for the Sharp Transmission Peaks

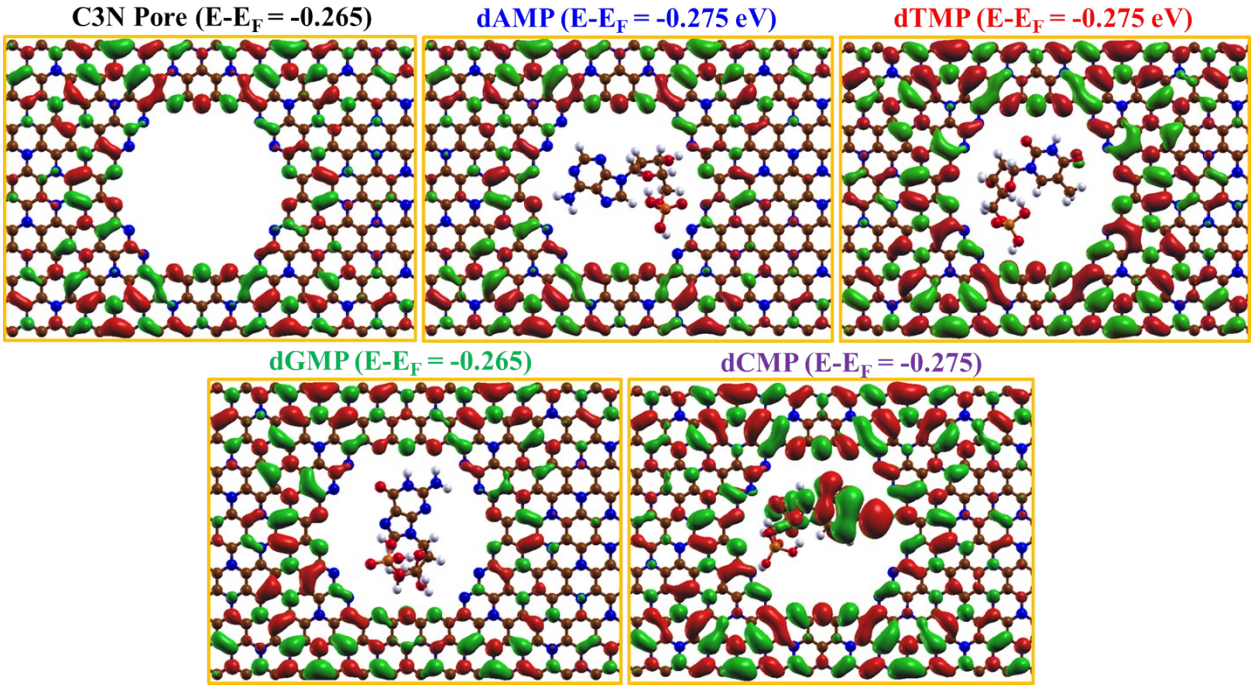


Figure S19. The wave function (WFs) studies show the frontier molecular orbitals which are responsible for the electron transport through the nanopore device.

15. ML Predicted Sensitivity of Nucleotides

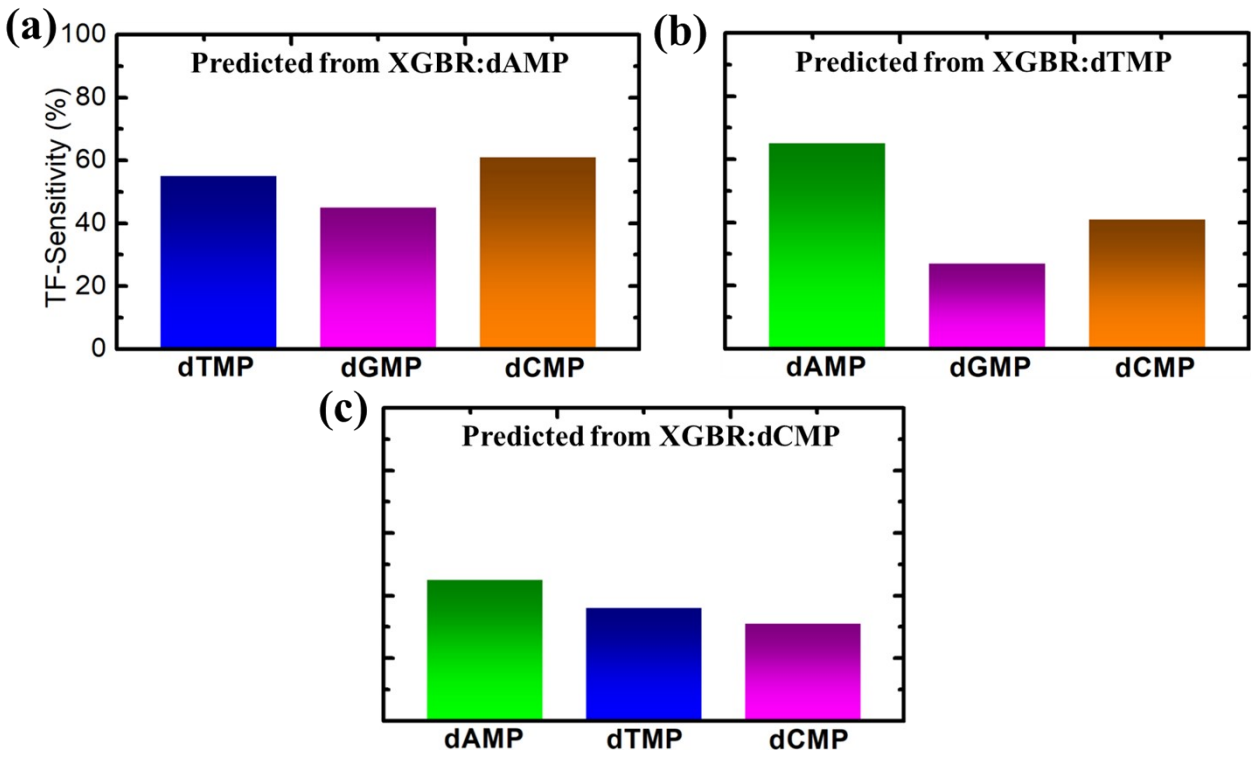


Figure S20. The ML sensitivity of the nucleotides calculated at the energy of $E - E_F = -0.265$ eV from the predicted transmission T(E) of XGBR:dGMP, and XGBR:dCMP model.

16. Ternary Classification Report

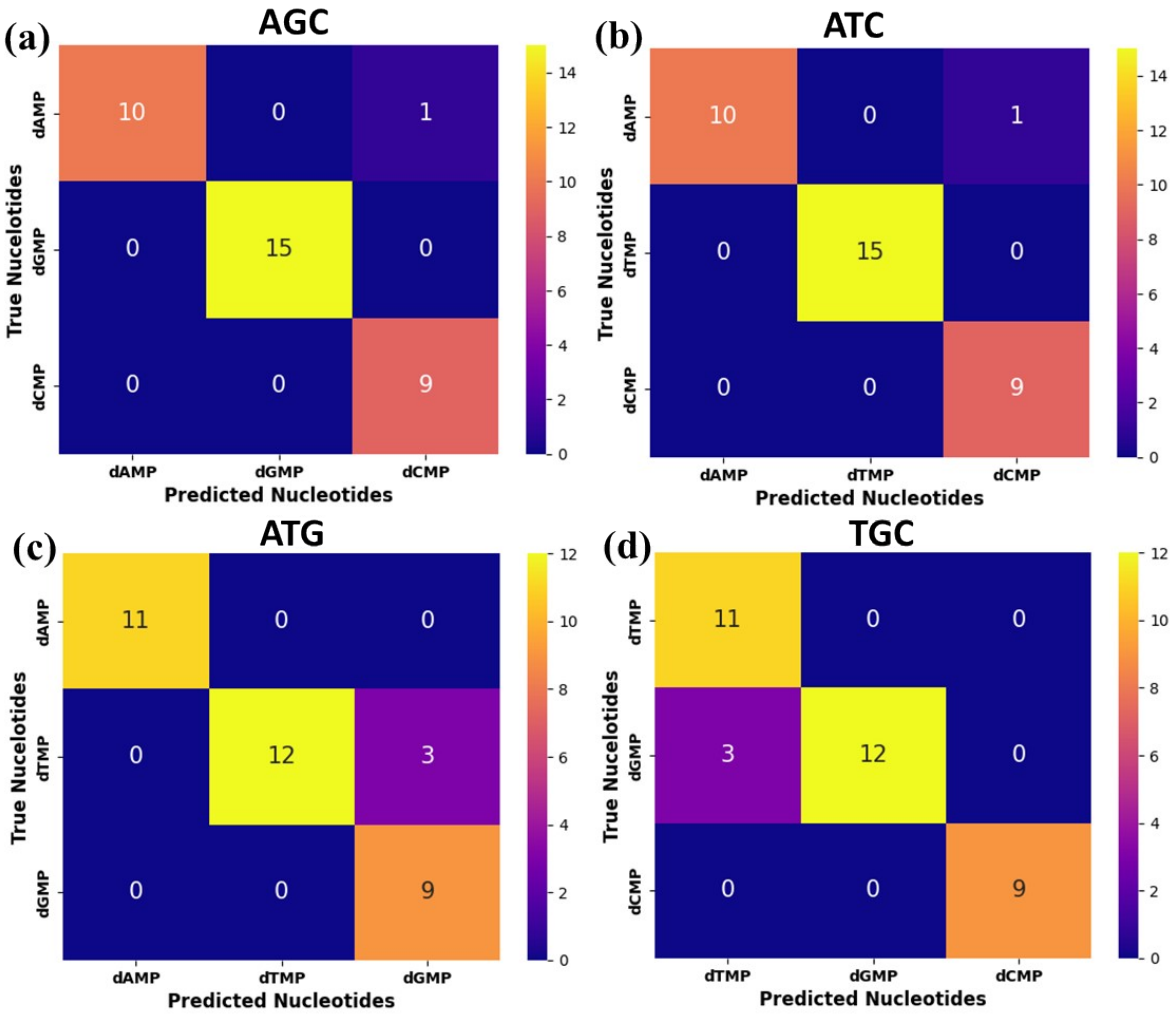


Figure S21. The ternary confusion matrix of classification with the true nucleotides and predicted nucleotides.

Nucleotide	Precision	Recall	F1 Score
A	1.00	0.91	0.95
T	1.00	1.00	1.00
C	0.90	1.00	0.95

Nucleotide	Precision	Recall	F1 Score
A	1.00	0.91	0.95
G	1.00	1.00	1.00
C	0.90	1.00	0.95

Nucleotide	Precision	Recall	F1 Score
A	1.00	1.00	1.00
T	1.00	0.80	0.89
G	0.75	1.00	0.86

Nucleotide	Precision	Recall	F1 Score
T	0.79	1.00	0.88
G	1.00	0.80	0.89
C	1.00	1.00	1.00

Figure S22. The machine learning classification report showing precision, recall, and F1 score is tabulated for all the considered ternary combinations.

17. Binary Classification Report

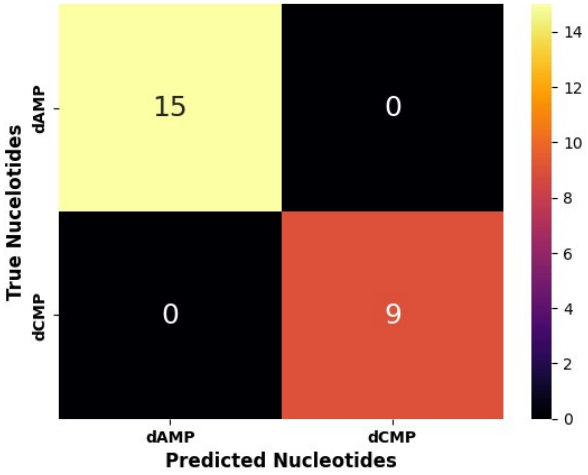


Figure S23. The confusion matrix of binary classification with the true nucleotides and predicted nucleotides.

Nucleotide	Precision	Recall	F1 Score
A	1.00	1.00	1.00
T	1.00	1.00	1.00

Nucleotide	Precision	Recall	F1 Score
A	1.00	1.00	1.00
G	1.00	1.00	1.00

Nucleotide	Precision	Recall	F1 Score
A	1.00	1.00	1.00
C	1.00	1.00	1.00

Nucleotide	Precision	Recall	F1 Score
T	1.00	1.00	1.00
G	1.00	1.00	1.00

Nucleotide	Precision	Recall	F1 Score
T	1.00	1.00	1.00
C	1.00	1.00	1.00

Nucleotide	Precision	Recall	F1 Score
G	1.00	1.00	1.00
C	1.00	1.00	1.00

Figure S24. The machine learning classification report showing precision, recall, and F1 score is tabulated for all six possible binary combinations.

Supplementary References:

(1) *Murphy, K. P. Machine Learning: A Probabilistic Perspective (MIT Press, 2012)*.

(2) Chen, T.; Guestrin, C. XGBoost: A Scalable Tree Boosting System. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*; KDD '16; Association for Computing Machinery: New York, NY, USA, 2016; pp 785–794.

(3) Friedman, J. H. Greedy Function Approximation: A Gradient Boosting Machine. *The Annals of Statistics* **2001**, *29* (5), 1189–1232.

(4) Breiman, L. Random Forests. *Machine Learning* **2001**, *45* (1), 5–32.

(5) Freund, Y.; Schapire, R. E. A Decision-Theoretic Generalization of On-Line Learning and an Application to Boosting. *Journal of Computer and System Sciences* **1997**, *55* (1), 119–139.

(6) Sanger, F. Determination of Nucleotide Sequences in DNA. *Science* **1981**.

(7) Frisch, M. J.; Trucks, G. W.; Schlegel, H. B.; Scuseria, G. E.; Robb, M. A.; Cheeseman, J. R.; Scalmani, G.; Barone, V.; Mennucci, B.; Petersson, G. A.; Nakatsuji, H. Gaussian 09; revision A.1; Gaussian, Inc.: *Wallingford, CT*, **2009**.

(8) Soler, J. M.; Artacho, E.; Gale, J. D.; García, A.; Junquera, J.; Ordejón, P.; Sánchez-Portal, D. The SIESTA method for ab initio order-N materials simulation. J. Phys.: Condens. Matter **2002**, *14* (11), 2745–2779.

(9) Ordejón, P.; Artacho, E.; Soler, J. M. Self-Consistent Order-N Density-Functional Calculations for Very Large Systems. *Phys. Rev. B* **1996**, *53* (16), R10441–R10444.

(10) Rocha, A. R.; García-Suárez, V. M.; Bailey, S.; Lambert, C.; Ferrer, J.; Sanvito, S. Spin and Molecular Electronics in Atomically Generated Orbital Landscapes. *Physical Review B - Condensed Matter and Materials Physics* **2006**, *73* (8).

(11) Prasongkit, J.; Grigoriev, A.; Pathak, B.; Ahuja, R.; Scheicher, R. H. Transverse Conductance of DNA Nucleotides in a Graphene Nanogap from First Principles. *Nano Lett.* **2011**, *11* (5), 1941–1945.

(12) Amorim, R. G.; Scheicher, R. H. Silicene as a New Potential DNA Sequencing Device. *Nanotechnology* **2015**, *26* (15), 154002.

(13) Hedström, S.; Chaudhuri, S.; Negre, C. F. A.; Ding, W.; Adam, J.; Batista, V. S. Non - Equilibrium Green's Function Calculations with TranSIESTA — a Tutorial. **2016**, 1–12.

(14) Papior, N. R.; Brandbyge, M. Multiple Electrode (Ne > 1) Support in the DFT+NEGF Code TranSIESTA. *Psi-K* **2015**, 1–2.

(15) Imry, Y.; Landauer, R. Conductance Viewed as Transmission. *Rev. Mod. Phys.* **1999**, *71* (2), S306–S312.

(16) Landauer, R. Electrical Transport in Open and Closed Systems. *Z. Physik B - Condensed Matter* **1987**, *68* (2), 217–228.

(17) Pathak, B.; Löf\aas, H.; Prasongkit, J.; Grigoriev, A.; Ahuja, R.; Scheicher, R. H. Double-Functionalized Nanopore-Embedded Gold Electrodes for Rapid DNA Sequencing. *Applied Physics Letters* **2012**, *100* (2), 137–140.